John Lee · Jow-Ran Chang · Lie-Jane Kao · Cheng-Few Lee

Essentials of Excel VBA, Python, and R

Volume II: Financial Derivatives, Risk Management and Machine Learning

Second Edition

Springer

Essentials of Excel VBA, Python, and R

John Lee • Jow-Ran Chang •
Lie-Jane Kao • Cheng-Few Lee

Essentials of Excel VBA, Python, and R

Volume II: Financial Derivatives, Risk Management and Machine Learning

Second Edition

Springer

John Lee
Center for PBBEF Research
Morris Plains, NJ, USA

Lie-Jane Kao
College of Finance
Takming University of Science
and Technology
Taipei City, Taiwan

Jow-Ran Chang
Dept of Quantitative Finance
National Tsing Hua University
Hsinchu, Taiwan

Cheng-Few Lee
Rutgers School of Business
The State University of New Jersey
North Brunswick, NJ, USA

ISBN 978-3-031-14282-6 ISBN 978-3-031-14283-3 (eBook)
https://doi.org/10.1007/978-3-031-14283-3

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

In the new edition of this book, there are 49 chapters, and they are divided into two volumes. Volume I, entitled "Microsoft Excel VBA, Python, and R For Financial Statistics and Portfolio Analysis," contains 26 chapters. Volume II entitled, "Microsoft Excel VBA, Python, and R For Financial Derivatives, Financial Management, and Machine Learning," contains 23 chapters. Volume I is divided into two parts. Part I Financial Statistics contains 21 chapters. Part II Portfolio Analysis contains five chapters. Volume II is divided into five parts. Part I Excel VBA contains three chapters. Part II Financial Derivatives contains six chapters. Part III Applications of Python, Machine Learning for Financial Derivatives, and Risk Management contains six chapters. Part IV Financial Management contains four chapters, and Part V Applications of R Programs for Financial Analysis and Derivatives contains three chapters.

Part I of this volume discusses advanced applications of Microsoft Excel Programs. Chapter 2 introduces Excel programming, Chap. 3 introduces VBA programming, and Chap. 4 discusses professional techniques used in Excel and Excel VBA techniques. There are six chapters in Part II. Chapter 5 discusses the decision tree approach for the binomial option pricing model, Chap. 6 discusses the Microsoft Excel approach to estimating alternative option pricing models, Chap. 7 discusses how to use Excel to estimate implied variance, Chap. 8 discusses Greek letters and portfolio insurance, Chap. 9 discusses portfolio analysis and option strategies, and Chap. 10 discusses simulation and its application.

There are six chapters in Part III, which describe applications of Python, machine learning for financial analysis, and risk management. These six chapters are Linear Models for Regression (Chap. 11), Kernel Linear Model (Chap. 12), Neural Networks and Deep Learning (Chap. 13), Applications of Alternative Machine Learning Methods for Credit Card Default Forecasting (Chap. 14), An Application of Deep Neural Networks for Predicting Credit Card Delinquencies (Chap. 15), and Binomial/Trinomial Tree Option Pricing Using Python (Chap. 16).

Part IV shows how Excel can be used to perform financial management. Chapter 17 shows how Excel can be used to perform financial ratio analysis, Chap. 18 shows how Excel can be used to perform time value money analysis, Chap. 19 shows how Excel can be used to perform capital budgeting under certainty and uncertainty, and Chap. 20 shows how Excel can be used for financial planning and forecasting. Finally, Part V discusses applications of R programs for financial analysis and derivatives. Chapter 21 discusses the theory and application of hedge ratios. In this chapter, we show how the R program can be used for hedge ratios in terms of three econometric methods. Chapter 22 discusses applications of a simultaneous equation in finance research in terms of the R program. Finally, Chap. 23 discusses how to use the R program to estimate the binomial option pricing model and the Black and Scholes option pricing model.

In this volume, Chap. 14 was contributed by Huei-Wen Teng and Michael Lee. Chapter 15 was contributed by Ting Sun, and Chap. 22 was contributed by Fu-Lai Lin.

There are two possible applications of this volume:

A. to supplement financial derivative and risk management courses.

B. to teach students how to use Excel VBA, Python, and R to analyze financial derivatives and perform risk management.

In sum, this book can be used by academic courses and for practitioners in the financial industry. Finally, we appreciate the extensive help of our assistants Xiaoyi Huang and Natalie Krawczyk.

Morris Plains, USA — John Lee
Hsinchu, Taiwan — Jow-Ran Chang
Taipei City, Taiwan — Lie-Jane Kao
North Brunswick, USA — Cheng-Few Lee
2021

Contents

1 Introduction . . . 1
- 1.1 Introduction . . . 1
- 1.2 Brief Description of Chap. 1 of Volume 1 . . . 1
- 1.3 Structure of This Volume . . . 1
  - 1.3.1 Excel VBA . . . 1
  - 1.3.2 Financial Derivatives . . . 2
  - 1.3.3 Applications of Python, Machine Learning for Financial Derivatives, and Risk Management . . . 2
  - 1.3.4 Financial Management . . . 2
  - 1.3.5 Applications of R Programs for Financial Analysis and Derivatives . . . 3
- 1.4 Summary . . . 3

Part I Excel VBA

2 Introduction to Excel Programming and Excel 365 Only Features . . . 7
- 2.1 Introduction . . . 7
- 2.2 Excel's Macro Recorder . . . 7
- 2.3 Excel's Visual Basic Editor . . . 13
- 2.4 Running an Excel Macro . . . 14
- 2.5 Adding Macro Code to a Workbook . . . 16
- 2.6 Macro Button . . . 18
- 2.7 Sub Procedures . . . 21
- 2.8 Message Box and Programming Help . . . 21
- 2.9 Excel 365 Only Features . . . 26
  - 2.9.1 Dynamic Arrays . . . 26
  - 2.9.2 Rich Data Types . . . 31
  - 2.9.3 STOCKHISTORY Function . . . 35
- 2.10 Summary . . . 37

References . . . 37

3 Introduction to VBA Programming . . . 39
- 3.1 Introduction . . . 39
- 3.2 Excel's Object Model . . . 39
- 3.3 Intellisense Menu . . . 42
- 3.4 Object Browser . . . 43
- 3.5 Variables . . . 50
- 3.6 Option Explicit . . . 54
- 3.7 Object Variables . . . 55
- 3.8 Functions . . . 56
- 3.9 Adding a Function Description . . . 58
- 3.10 Specifying a Function Category . . . 60

3.11 Conditional Programming with the IF Statement . . . 61
3.12 For Loop . . . 63
3.13 While Loop . . . 65
3.14 Arrays . . . 68
3.15 Option Base 1 . . . 72
3.16 Collections . . . 72
3.17 Summary . . . 74
References . . . 74

4 Professional Techniques Used in Excel and VBA . . . 75
4.1 Introduction . . . 75
4.2 Finding the Range of a Table: CurrentRegion Property . . . 75
4.3 Offset Property of the Range Object . . . 77
4.4 Resize Property of the Range Object . . . 78
4.5 UsedRange Property of the Range Object . . . 79
4.6 Go to Special Dialog Box of Excel . . . 81
4.7 Importing Column Data into Arrays . . . 84
4.8 Importing Row Data into an Array . . . 93
4.9 Transferring Data from an Array to a Range . . . 94
4.10 Workbook Names . . . 96
4.11 Dynamic Range Names . . . 98
4.12 Global Versus Local Workbook Names . . . 102
4.13 List of All Files in a Directory . . . 108
4.14 Summary . . . 111
References . . . 111

Part II Financial Derivatives

5 Binomial Option Pricing Model Decision Tree Approach . . . 115
5.1 Introduction . . . 115
5.2 Call and Put Options . . . 115
5.3 Option Pricing—One Period . . . 117
5.4 Put Option Pricing—One Period . . . 118
5.5 Option Pricing—Two Period . . . 119
5.6 Option Pricing—Four Period . . . 120
5.7 Using Microsoft Excel to Create the Binomial Option Call Trees . . . 121
5.8 American Options . . . 124
5.9 Alternative Tree Methods . . . 125
5.9.1 Cox, Ross, and Rubinstein . . . 125
5.9.2 Trinomial Tree . . . 127
5.9.3 Compare the Option Price Efficiency . . . 129
5.10 Retrieving Option Prices from Yahoo Finance . . . 130
5.11 Summary . . . 130
Appendix 5.1: EXCEL CODE—Binomial Option Pricing Model . . . 131
References . . . 135

6 Microsoft Excel Approach to Estimating Alternative Option Pricing Models . . . 137
6.1 Introduction . . . 137
6.2 Option Pricing Model for Individual Stock . . . 137
6.3 Option Pricing Model for Stock Indices . . . 138
6.4 Option Pricing Model for Currencies . . . 139
6.5 Futures Options . . . 140

6.6 Using Bivariate Normal Distribution Approach to Calculate American Call Options . . . 142
6.7 Black's Approximation Method for American Option with One Dividend Payment . . . 148
6.8 American Call Option When Dividend Yield is Known . . . 149
6.8.1 Theory and Method . . . 149
6.8.2 VBA Program for Calculating American Option When Dividend Yield is Known . . . 150
6.9 Summary . . . 153
Appendix 6.1: Bivariate Normal Distribution . . . 153
Appendix 6.2: Excel Program to Calculate the American Call Option When Dividend Payments are Known . . . 153
References . . . 156

7 Alternative Methods to Estimate Implied Variance . . . 157
7.1 Introduction . . . 157
7.2 Excel Program to Estimate Implied Variance with Black–Scholes Option Pricing Model . . . 157
7.2.1 Black, Scholes, and Merton Model . . . 157
7.2.2 Approximating Linear Function for Implied Volatility . . . 158
7.2.3 Nonlinear Method for Implied Volatility . . . 160
7.3 Volatility Smile . . . 167
7.4 Excel Program to Estimate Implied Variance with CEV Model . . . 169
7.5 WEBSERVICE Function . . . 174
7.6 Retrieving a Stock Price for a Specific Date . . . 176
7.7 Calculated Holiday List . . . 177
7.8 Calculating Historical Volatility . . . 178
7.9 Summary . . . 180
Appendix 7.1: Application of CEV Model to Forecasting Implied Volatilities for Options on Index Futures . . . 180
References . . . 189

8 Greek Letters and Portfolio Insurance . . . 191
8.1 Introduction . . . 191
8.2 Delta . . . 191
8.2.1 Formula of Delta for Different Kinds of Stock Options . . . 191
8.2.2 Excel Function of Delta for European Call Options . . . 192
8.2.3 Application of Delta . . . 193
8.3 Theta . . . 194
8.3.1 Formula of Theta for Different Kinds of Stock Options . . . 194
8.3.2 Excel Function of Theta of the European Call Option . . . 194
8.3.3 Application of Theta . . . 195
8.4 Gamma . . . 195
8.4.1 Formula of Gamma for Different Kinds of Stock Options . . . 196
8.4.2 Excel Function of Gamma for European Call Options . . . 196
8.4.3 Application of Gamma . . . 197
8.5 Vega . . . 198
8.5.1 Formula of Vega for Different Kinds of Stock Options . . . 198
8.5.2 Excel Function of Vega for European Call Options . . . 198
8.5.3 Application of Vega . . . 199

8.6 Rho . . . 200
8.6.1 Formula of Rho for Different Kinds of Stock Options . . . 200
8.6.2 Excel Function of Rho for European Call Options . . . 201
8.6.3 Application of Rho . . . 201
8.7 Formula of Sensitivity for Stock Options with Respect to Exercise Price . . . 202
8.8 Relationship Between Delta, Theta, and Gamma . . . 202
8.9 Portfolio Insurance . . . 202
8.10 Summary . . . 203
References . . . 203

9 Portfolio Analysis and Option Strategies . . . 205
9.1 Introduction . . . 205
9.2 Three Alternative Methods to Solve the Simultaneous Equation . . . 205
9.2.1 Substitution Method (Reference: Wikipedia) . . . 205
9.2.2 Cramer's Rule . . . 205
9.2.3 Matrix Method . . . 206
9.2.4 Excel Matrix Inversion and Multiplication . . . 207
9.3 Markowitz Model for Portfolio Selection . . . 207
9.4 Option Strategies . . . 210
9.4.1 Long Straddle . . . 210
9.4.2 Short Straddle . . . 211
9.4.3 Long Vertical Spread . . . 213
9.4.4 Short Vertical Spread . . . 213
9.4.5 Protective Put . . . 213
9.4.6 Covered Call . . . 216
9.4.7 Collar . . . 219
9.5 Summary . . . 222
Appendix 9.1: Monthly Rates of Returns for S&P500, IBM, and MSFT . . . 223
Appendix 9.2: Options Data for IBM (Stock Price = 141.34) on July 23, 2021 . . . 224
References . . . 225

10 Simulation and Its Application . . . 227
10.1 Introduction . . . 227
10.2 Monte Carlo Simulation . . . 227
10.3 Antithetic Variables . . . 231
10.4 Quasi-Monte Carlo Simulation . . . 233
10.5 Application . . . 237
10.6 Summary . . . 244
Appendix 10.1: EXCEL CODE—Share Price Paths . . . 245
References . . . 246
On the Web . . . 246

Part III Applications of Python, Machine Learning for Financial Derivatives and Risk Management

11 Linear Models for Regression . . . 249
11.1 Introduction . . . 249
11.2 Loss Functions and Least Squares . . . 249
11.3 Regularized Least Squares—Ridge and Lasso Regression . . . 250
11.4 Logistic Regression for Classification: A Discriminative Model . . . 250
11.5 *K*-fold Cross-Validation . . . 251
11.6 Types of Basis Function . . . 251

11.7 Accuracy Measures in Classification . . . 252
11.8 Python Programming Example . . . 252
Questions and Problems for Coding . . . 253
References . . . 259

12 Kernel Linear Model . . . 261
12.1 Introduction . . . 261
12.2 Constructing Kernels . . . 261
12.3 Kernel Regression (Nadaraya–Watson Model) . . . 261
12.4 Relevance Vector Machines . . . 262
12.5 Gaussian Process for Regression . . . 263
12.6 Support Vector Machines . . . 263
12.7 Python Programming . . . 264
12.8 Kernel Linear Model and Support Vector Machines . . . 265
References . . . 277

13 Neural Networks and Deep Learning Algorithm . . . 279
13.1 Introduction . . . 279
13.2 Feedforward Network Functions . . . 279
13.3 Network Training: Error Backpropagation . . . 280
13.4 Gradient Descent Optimization . . . 282
13.5 Regularization in Neural Networks and Early Stopping . . . 282
13.6 Deep Feedforward Network Versus Deep Convolutional Neural Networks . . . 283
13.7 Python Programing . . . 284
References . . . 284

14 Alternative Machine Learning Methods for Credit Card Default Forecasting* . . . 285
14.1 Introduction . . . 285
14.2 Literature Review . . . 285
14.3 Description of the Data . . . 287
14.4 Alternative Machine Learning Methods . . . 287
14.4.1 *k*-Nearest Neighbors . . . 287
14.4.2 Decision Trees . . . 288
14.4.3 Boosting . . . 290
14.4.4 Support Vector Machines . . . 290
14.4.5 Neural Networks . . . 291
14.5 Study Plan . . . 292
14.5.1 Data Preprocessing and Python Programming . . . 292
14.5.2 Tuning Optimal Parameters . . . 292
14.5.3 Learning Curves . . . 294
14.6 Summary and Concluding Remarks . . . 295
Appendix 14.1: Python Codes . . . 295
References . . . 297

15 Deep Learning and Its Application to Credit Card Delinquency Forecasting . . . 299
15.1 Introduction . . . 299
15.2 Literature Review . . . 299
15.3 The Methodology . . . 300
15.3.1 Deep Learning in a Nutshell . . . 300

15.3.2 Deep Learning Versus Conventional Machine Learning Approaches . . . 300
15.3.3 The Structure of a DNN and the Hyper-Parameters . . . 301
15.4 Data . . . 303
15.5 Experimental Analysis . . . 304
15.5.1 Splitting the Data . . . 304
15.5.2 Tuning the Hyper-Parameters . . . 305
15.5.3 Techniques of Handling Data Imbalance . . . 306
15.6 Results . . . 306
15.6.1 The Predictor Importance . . . 306
15.6.2 The Predictive Result for Cross-Validation Sets . . . 307
15.6.3 Prediction on Test Set . . . 308
15.7 Conclusion . . . 309
Appendix 15.1: Variable Definition . . . 310
References . . . 311

16 Binomial/Trinomial Tree Option Pricing Using Python . . . 313
16.1 Introduction . . . 313
16.2 European Option Pricing Using Binomial Tree Model . . . 313
16.2.1 European Option Pricing—Two Period . . . 315
16.2.2 European Option Pricing—*N* Periods . . . 317
16.3 American Option Pricing Using Binomial Tree Model . . . 318
16.4 Alternative Tree Models . . . 320
16.4.1 Cox, Ross, and Rubinstein Model . . . 320
16.4.2 Trinomial Tree . . . 321
16.5 Summary . . . 321
Appendix 16.1: Python Programming Code for Binomial Tree Option Pricing . . . 323
Appendix 16.2: Python Programming Code for Trinomial Tree Option Pricing . . . 330
References . . . 334

Part IV Financial Management

17 Financial Ratio Analysis and Its Applications . . . 337
17.1 Introduction . . . 337
17.2 Financial Statements: A Brief Review . . . 337
17.2.1 Balance Sheet . . . 337
17.2.2 Statement of Earnings . . . 339
17.2.3 Statement of Equity . . . 340
17.2.4 Statement of Cash Flows . . . 340
17.2.5 Interrelationship Among Four Financial Statements . . . 343
17.2.6 Annual Versus Quarterly Financial Data . . . 344
17.3 Static Ratio Analysis . . . 344
17.3.1 Static Determination of Financial Ratios . . . 344
17.4 Two Possible Methods to Estimate the Sustainable Growth Rate . . . 348
17.5 DFL, DOL, and DCL . . . 349
17.5.1 Degree of Financial Leverage . . . 349
17.5.2 Operating Leverage and the Combined Effect . . . 350
17.6 Summary . . . 354

Appendix 17.1: Calculate 26 Financial Ratios with Excel . . . 354
Appendix 17.2: Using Excel to Calculate Sustainable Growth Rate . . . 363
Appendix 17.3: How to Compute DOL, DFL, and DCL with Excel . . . 364
References . . . 368

18 Time Value of Money Determinations and Their Applications . . . 369
18.1 Introduction . . . 369
18.2 Basic Concepts of Present Values . . . 369
18.3 Foundation of Net Present Value Rules . . . 370
18.4 Compounding and Discounting Processes . . . 371
18.4.1 Single Payment Case—Future Values . . . 371
18.4.2 Continuous Compounding . . . 371
18.4.3 Single Payment Case—Present Values . . . 372
18.4.4 Annuity Case—Present Values . . . 373
18.4.5 Annuity Case—Future Values . . . 373
18.4.6 Annual Percentage Rate . . . 373
18.5 Present and Future Value Tables . . . 374
18.5.1 Future Value of a Dollar at the End of t Periods . . . 374
18.5.2 Future Value of a Dollar Continuously Compounded . . . 375
18.5.3 Present Value of a Dollar Received t Periods in the Future . . . 376
18.5.4 Present Value of an Annuity of a Dollar Per Period . . . 377
18.6 Why Present Values Are Basic Tools for Financial Management Decisions . . . 377
18.6.1 Managing in the Stockholders' Interest . . . 378
18.6.2 Productive Investments . . . 379
18.7 Net Present Value and Internal Rate of Return . . . 381
18.8 Summary . . . 382
Appendix 18A . . . 382
Appendix 18B . . . 384
Appendix 18C . . . 384
Appendix 18D: Applications of Excel for Calculating Time Value of Money . . . 386
Appendix 18E: Tables of Time Value of Money . . . 390
References . . . 401

19 Capital Budgeting Method Under Certainty and Uncertainty . . . 403
19.1 Introduction . . . 403
19.2 The Capital Budgeting Process . . . 403
19.2.1 Identification Phase . . . 404
19.2.2 Development Phase . . . 405
19.2.3 Selection Phase . . . 405
19.2.4 Control Phase . . . 406
19.3 Cash-Flow Evaluation of Alternative Investment Projects . . . 407
19.4 Alternative Capital-Budgeting Methods . . . 409
19.4.1 Accounting Rate-of-Return . . . 409
19.4.2 Internal Rate-of-Return Method . . . 410
19.4.3 Payback Method . . . 410
19.4.4 Net Present Value Method . . . 410
19.4.5 Profitability Index . . . 411
19.5 Capital-Rationing Decision . . . 411
19.5.1 Basic Concepts of Linear Programming . . . 412
19.5.2 Capital Rationing . . . 412

19.6 The Statistical Distribution Method . . . 413
19.6.1 Statistical Distribution of Cash Flow . . . 414
19.7 Simulation Methods . . . 416
19.7.1 Simulation Analysis and Capital Budgeting . . . 418
19.8 Summary . . . 421
Appendix 19.1: Solving the Linear Program Model for Capital Rationing . . . 422
Appendix 19.2: Decision Tree Method for Investment Decisions . . . 429
Appendix 19.3: Hillier's Statistical Distribution Method for Capital Budgeting Under Uncertainty . . . 430
References . . . 431

20 Financial Analysis, Planning, and Forecasting . . . 433
20.1 Introduction . . . 433
20.2 Procedures for Financial Planning and Analysis . . . 433
20.3 The Algebraic Simultaneous Equations Approach to Financial Planning and Analysis . . . 435
20.4 The Linear Programming Approach to Financial Planning and Analysis . . . 441
20.4.1 Profit Maximization . . . 442
20.4.2 Linear Programming and Capital Rationing . . . 443
20.4.3 Linear Programming Approach to Financial Planning . . . 444
20.5 The Econometric Approach to Financial Planning and Analysis . . . 446
20.5.1 A Dynamic Adjustment of the Capital Budgeting Model . . . 446
20.5.2 Simplified Spies Model . . . 447
20.6 Sensitivity Analysis . . . 447
20.7 Summary . . . 449
Appendix 20.1: The Simplex Algorithm for Capital Rationing . . . 449
Appendix 20.2: Description of Parameter Inputs Used to Forecast Johnson & Johnson's Financial Statements and Share Price . . . 450
Appendix 20.3: Procedure of Using Excel to Implement the FinPlan Program . . . 451
References . . . 455

Part V Applications of R Programs for Financial Analysis and Derivatives

21 Hedge Ratio Estimation Methods and Their Applications . . . 459
21.1 Introduction . . . 459
21.2 Alternative Theories for Deriving the Optimal Hedge Ratio . . . 460
21.2.1 Static Case . . . 461
21.2.2 Dynamic Case . . . 464
21.2.3 Case with Production and Alternative Investment Opportunities . . . 464
21.3 Alternative Methods for Estimating the Optimal Hedge Ratio . . . 465
21.3.1 Estimation of the Minimum-Variance (MV) Hedge Ratio . . . 465
21.3.2 Estimation of the Optimum Mean–Variance and Sharpe Hedge Ratios . . . 467
21.3.3 Estimation of the Maximum Expected Utility Hedge Ratio . . . 467
21.3.4 Estimation of Mean Extended-Gini (MEG) Coefficient Based Hedge Ratios . . . 468
21.3.5 Estimation of Generalized Semivariance (GSV) Based Hedge Ratios . . . 468
21.4 Applications of OLS, GARCH, and CECM Models to Estimate Optimal Hedge Ratio . . . 468

21.5 Hedging Horizon, Maturity of Futures Contract, Data Frequency, and Hedging Effectiveness . . . 470
21.6 Summary and Conclusions . . . 471
Appendix 21.1: Theoretical Models . . . 473
Appendix 21.2: Empirical Models . . . 475
Appendix 21.3: Monthly Data of S&P500 Index and Its Futures (January 2005–August 2020) . . . 483
Appendix 21.4: Applications of R Language in Estimating the Optimal Hedge Ratio . . . 487
References . . . 488

22 Application of Simultaneous Equation in Finance Research: Methods and Empirical Results . . . 491
22.1 Introduction . . . 491
22.2 Literature Review . . . 491
22.3 Methodology . . . 492
22.3.1 Application of GMM Estimation in the Linear Regression Model . . . 493
22.3.2 Applications of GMM Estimation in the Simultaneous Equations Model . . . 494
22.3.3 Weak Instruments . . . 496
22.4 Applications in Investment, Financing, and Dividend Policy . . . 497
22.4.1 Model and Data . . . 497
22.4.2 Results of Weak Instruments . . . 497
22.4.3 Empirical Results . . . 498
22.5 Conclusion . . . 505
Appendix 22.1: Data for Johnson & Johnson and IBM . . . 505
Appendix 22.2: Applications of R Language in Estimating the Parameters of a System of Simultaneous Equations . . . 507
References . . . 509

23 Three Alternative Programs to Estimate Binomial Option Pricing Model and Black and Scholes Option Pricing Model . . . 511
23.1 Introduction . . . 511
23.2 Microsoft Excel Program for the Binomial Tree Option Pricing Model . . . 511
23.3 Black and Scholes Option Pricing Model for Individual Stock . . . 512
23.4 Black and Scholes Option Pricing Model for Stock Indices . . . 514
23.5 Black and Scholes Option Pricing Model for Currencies . . . 514
23.6 R Codes to Implement the Binomial Trees Option Pricing Model . . . 514
23.7 R Codes to Compute Option Prices by Black and Scholes Model . . . 519
23.8 Summary . . . 519
Appendix 23.1: SAS Programming to Implement the Binomial Option Trees . . . 519
Appendix 23.2: SAS Programming to ComputeOption Prices Using Black and Scholes Model . . . 521
References . . . 523

Introduction

1

1.1 Introduction

In Volume I of this book, we have shown how Excel VBA, Python, and R can be used in financial statistics analysis and portfolio analysis. In this volume, we will further demonstrate how these tools can be used to perform financial derivatives, machine learning, risk management, financial management, and financial analysis. In Sect. 1.2, we briefly describe the contents of Chap. 1 of Volume 1. In Sect. 1.3, we will discuss the structure of this volume. Finally, in Sect. 1.4, we will summarize this chapter.

1.2 Brief Description of Chap. 1 of Volume 1

In Volume I of this book, there are 26 chapters. The introduction chapter of this volume discusses (a) the statistical environment of Microsoft Excel 365; (b) Python programming language; (c) R programming language; (d) web scraping for market and financial data; (e) case study and Google study and active study approach; and (f) structure of the book. Items a, b, c, d, and e need to be read before reading Volume II. Part A includes 20 chapters, which discuss different statistical methods and their application in finance, economics, accounting, and other business applications. In this part, Microsoft Excel VBA, Python, and R are used to investigate financial statistics. In Part B, there are six chapters, which discuss how Microsoft Excel VBA can be used to analyze portfolio analysis and portfolio management.

1.3 Structure of This Volume

There are 23 chapters in Volume II of this book. Besides the introduction chapter, Volume II is divided into five parts. Part A includes three chapters, which discuss Microsoft Excel VBA. Part B includes six chapters which discuss how Excel VBA can be used in financial derivatives. In Part C, there are six chapters that discuss applications of Python, machine learning for financial derivatives, and risk management. Part D includes four chapters which discuss how Excel VBA can be used for financial management, and Part E includes three chapters which discuss applications of R programs for financial analysis and derivatives.

1.3.1 Excel VBA

In Part B of this volume, there are three chapters which describe how Excel VBA can be used for beta analysis. In Chap. 2 of this part, we discuss the introduction of Excel programming in detail. We go over how to use many of Excel's features including Excel's macro recorder; Visual Basic Editor; how to run an Excel macro; how to add macro code to a workbook; how to push a button to apply an Excel program; subprocedures; and message box and programming help.

In Chap. 3, we discuss the introduction to VBA programming. We talk about Excel's object model; auto list members; the object browser; variables; option explicit; object variables; functions; how to add a function description; specifying a function category; conditional programming with the IF statement; a for loop; a while loop; arrays; option base 1; collections; and looping.

In Chap. 4, we discuss professional techniques used in Excel and Excel VBA techniques. We talk about finding the range of a table; the offset property of the range object; the resize property of the range object; the used range property of the range object; the special dialog box in Excel; how to import column data into an array; how to import row data into an array; how to transfer data from an array to a range; workbook names; dynamic ranges; global versus local workbook names; dynamic charting; and how to search all the files in a directory.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_1

1.3.2 Financial Derivatives

In the financial derivatives part, which contains six chapters, we try to show how to use Excel to evaluate the option pricing model in terms of the decision tree method and the Black and Scholes model. In addition, we show how implied variance can be estimated in terms of both the Black and Scholes model and the CEV model. How to use Excel to perform simulation is also discussed.

In Chap. 5 of this part, we discuss the decision tree approach for the binomial option pricing model. We talk about the call and put options; option pricing: one period; put option pricing: one period; option pricing: two periods; option pricing: four periods; how to use Excel to create binomial option call trees; American options; alternative tree methods, which include binomial and trinomial option pricing model; and how to retrieve option prices from Yahoo Finance. Overall, this chapter extensively shows how Excel VBA can be used to estimate binomial and trinomial European option pricing model. In addition, how to apply the binomial option pricing model to American options is also demonstrated.

In Chap. 6, we discuss the Microsoft Excel approach to estimating alternative option pricing models. We talk about the option pricing model for individual stock; option pricing model for stock indices; option pricing model for currencies; future option; how to use the bivariate normal distribution approach to calculate American call options; Black's approximation method for American options with one dividend payment; and American call option when the dividend yield is known.

In Chap. 7, we discuss alternative methods to estimate implied variances. We talk about how to use Excel to estimate implied variance with Black–Scholes OPM, volatility smile, how Excel can be used to estimate implied variance with the CEV model, the WEBSERVICE Excel function, how to retrieve a stock price for a specific date, calculated holiday list, and how to calculate historical volatility.

In Chap. 8, we discuss Greek letters and portfolio insurance. We specifically discuss delta, theta, gamma, vega, rho, the formula of sensitivity for stock options with respect to exercise price, the relationship between delta, theta, and gamma, and portfolio insurance.

In Chap. 9, we discuss portfolio analysis and option strategies. We talk about the three alternative methods to solve a simultaneous equation and how the Markowitz model can be used for portfolio selection. Alternative option strategies for option investment decision are also discussed in detail.

In Chap. 10, we discuss alternative simulation methods and their applications. We talk about the Monte Carlo simulation; antithetic variables; Quasi-Monte Carlo simulation; and their applications.

1.3.3 Applications of Python, Machine Learning for Financial Derivatives, and Risk Management

In Chap. 11 of this part, we discuss linear models for regression. We talk about loss functions and least squares; regularized least squares—Ridge and Lasso regression; logistic regression for classification: a discriminative model; K-fold cross-validation; types of basis function; accuracy measures in classification; and a Python programming example.

In Chap. 12, we discuss the Kernel linear model. We talk about constructing kernels, kernel regression—Nadaraya–Watson model, relevance vector machines, and Gaussian process for regression; support vector machines; and Python programming.

In Chap. 13, we discuss neural networks and deep learning. We talk about feedforward network functions, network training, gradient descent optimization, error back-propagation, regularization in neural networks, early stopping, tangent propagation, deep neural network, recurrent neural networks, training with transformed data—convolutional neural networks, and Python programming.

In Chap. 14, we discuss the applications of five alternative machine learning methods for credit card default forecasting. We talk about a description of data, machine learning, and a study plan.

An application of deep neural networks for predicting credit card delinquencies is discussed in Chap. 15. We review the literature, and the methodology of artificial neural networks, and look at data and experimental analysis.

In Chap. 16, binomial and trinomial tree option pricing using Python is discussed. In this chapter, we first reproduce the content of Chap. 6 using Excel. Then in Appendix 16.1, we present the Python programming code for binomial tree option pricing, and in Appendix 16.2 we show the Python programming code for binomial tree option pricing.

1.3.4 Financial Management

In Chap. 17 of this part, financial ratios and their applications are discussed. We talk about financial statements; how to calculate static financial ratios with Excel; how to calculate DOL, DSL, and DCL with Excel; and the application of financial ratios in the investment decision is discussed in detail.

In Chap. 18, the time value of money analysis is discussed. We talk about the basic concepts of present values; the foundation of net present value rules; compounding and discounting processes; the applications of Excel in calculating the time value of money; and the application of the

time value of money in mortgage payment in an investment decision.

We discuss capital budgeting under certainty and uncertainty in Chap. 19. More specifically, we discuss the capital budgeting process; the cash-flow evaluation of alternative investment projects; NPV and IRR methods; capital-rationing decision with Excel; the statistical distribution method with Excel; the decision tree method for investment decisions with Excel; and simulation methods with Excel.

Financial planning and forecasting are discussed in Chap. 20. We talk about procedures for financial planning and analysis; the algebraic simultaneous equations approach to financial planning and analysis; and the procedure of using Excel for financial planning and forecasting.

1.3.5 Applications of R Programs for Financial Analysis and Derivatives

Lastly, Part E contains three chapters, which show how R programming can be useful for financial analysis and derivatives. In Chap. 21 of this part, we discuss theories and applications of hedge ratios. We talk about alternative theories for deriving the optimal hedge ratio; alternative methods for estimating the optimal hedge ratio; using OLS, GARCH, and CECM models to estimate the optimal hedge ratio; and hedging horizon, maturity of futures contract, data frequency, and hedging effectiveness.

In Chap. 22, we first discuss the simultaneous equation model for investment, financing, and dividend decision. Then we show how the R program can be used to estimate the empirical results of investment, financing, and dividend decision in terms of two-stage least squares, three-stage least squares, and generalized method of moments.

In Chap. 23, we review binomial, trinomial, and American option pricing models, which were previously discussed in Chaps. 5 and 6. We then show how the R program can be used to estimate the binomial option pricing model and the Black–Scholes option pricing model.

1.4 Summary

In this volume, we have shown how Excel VBA can be used to evaluate binomial, trinomial, and American option models. In addition, we also showed how implied variance in terms of the Black–Scholes and CEV models can be estimated. Option strategy and portfolio analysis are also explored in some detail. We have also shown how Excel can be used to perform different simulation models.

We also showed how Python can be used for regression analysis and credit analysis in this volume. In addition, the application of Python in estimating binomial and trinomial option pricing models is also discussed in some detail.

The application of the R language to estimate hedge ratios and investigate the relationship among investment, financing, and dividend policy is also discussed in this volume. We also show how the R language can be used to estimate the binomial option trees. Finally, in Part E we also show how the R language can be used to estimate option pricing for individual stock, stock indices, and currency options.

Part I
Excel VBA

Check for updates

Introduction to Excel Programming and Excel 365 Only Features

2

2.1 Introduction

A lot of the work done by an Excel user is repetitive and time-consuming. Fortunately for an Excel user, Excel offers a powerful and professional programming language and a powerful and professional programming environment to automate their work. This book will illustrate some of the things that can be accomplished by Excel's programming language called Visual Basic for Applications or more commonly known as VBA.

We will also look at some of the features only available in Excel 365.

This chapter will be broken down into the following sections. In section 2.2, we will discuss Excel's macro reader, and in section 2.3 we will discuss Excel's Visual Basic Editor. In section 2.4, we look at how to run an Excel macro. Section 2.5 discusses how to add macro code to a workbook. Section 2.6 discusses the macro button, and section 2.7 discusses subprocedures. In section 2.8, we look at the message box and programming help. In section 2.9, we discuss Excel 365 only features. Finally, in section 2.10 we summarize the chapter.

2.2 Excel's Macro Recorder

There is one common question that both a novice and an experienced Excel VBA programmer will ask about Excel VBA programming: "How do I program this in Excel VBA?" The reason that the novice VBA programmer will ask this question is because of his or her lack of experience. To understand why the experienced VBA programmer will ask this question, we need to realize that Excel has an enormous amount of features. It is virtually impossible for anybody to remember how to program every feature of Excel. Interestingly, the answer to the question is the same for both the novice and the experienced programmer. The answer is Excel's macro recorder. Excel's macro recorder will record any action done by the user. The recorded result is the Excel VBA code. The resulting VBA code is important because both the novice and the experienced VBA programmer can study the resulting Excel VBA code.

Suppose that we have a great need to do the following to the cell that we selected:

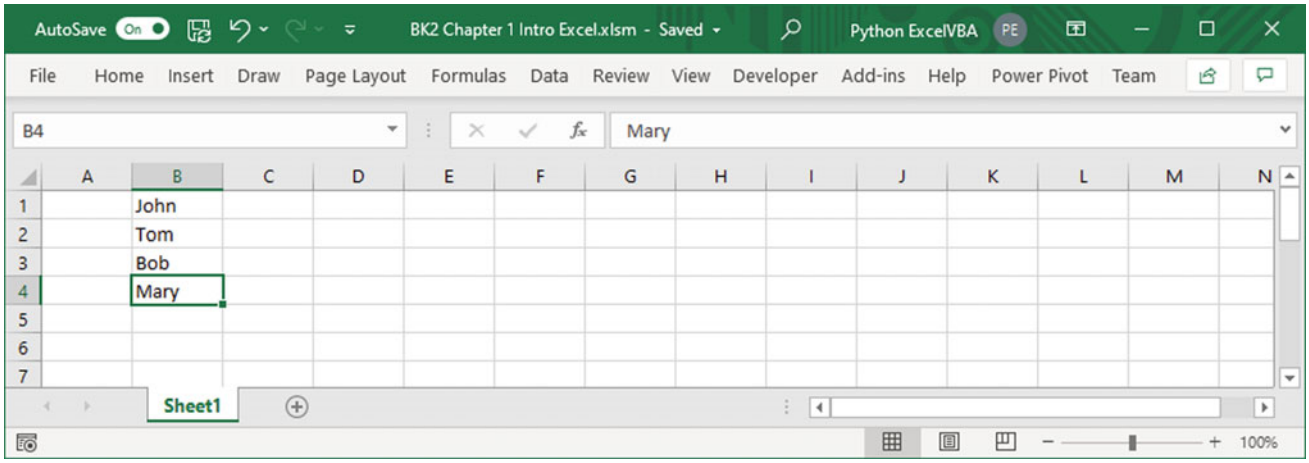
1. Bolden the words in the cells that we selected.
2. Italicize the words in the cells that we selected.
3. Underline the words in the cells that we selected.
4. Center the words in the cells that we selected.

What is the Excel VBA code to accomplish the above list? The thing for both the novice and the experienced VBA programmer to do is to use Excel's macro recorder to record manually the actions required to get the desired results. This process is shown below.

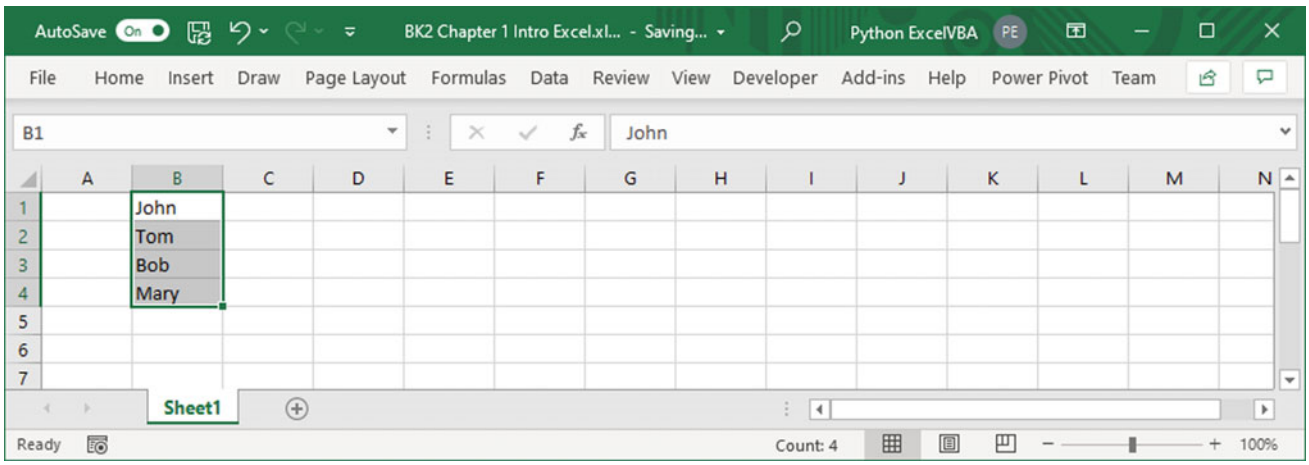
Before we do anything, let's type in the words as shown in worksheet "Sheet1" shown below.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

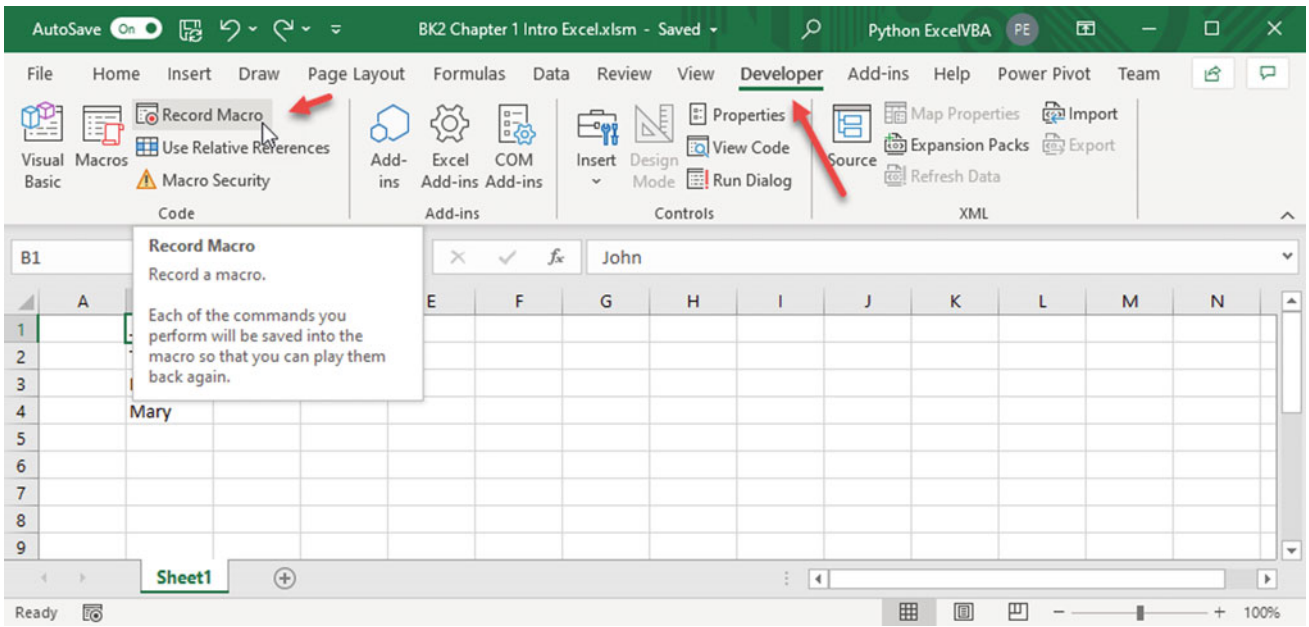
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_2



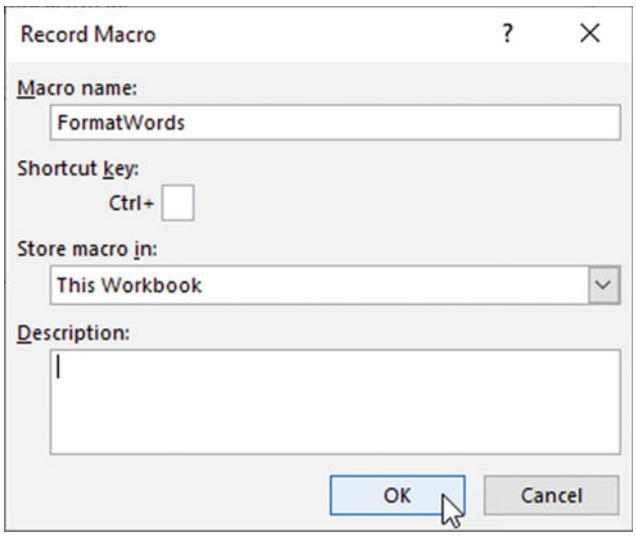
Next, highlight the words above before we start using Excel's macro recorder to generate the VBA code. To highlight the list, first select the word "John," then press the *Shift* key on the keyboard, and while pressing the *Shift* key, press the ↓ key on the keyboard three times. The result is shown below.



Now let's turn on Excel's macro recorder. To do this, we would choose Developer → Record Macro. The steps to do this are shown below.

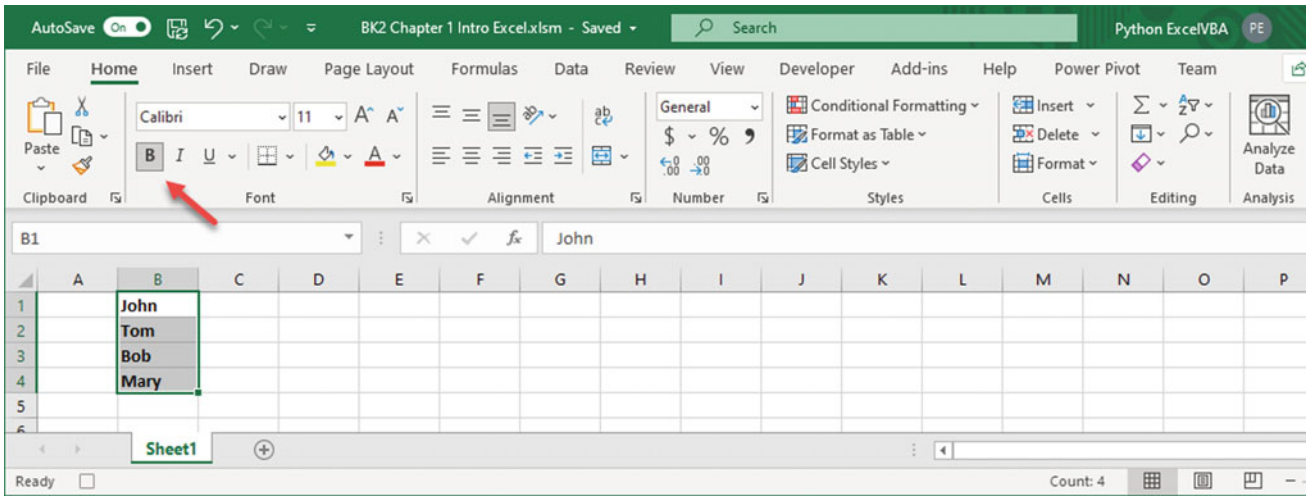


Choosing the *Record Macro* menu item would result in the *Record Macro* dialog box shown below.

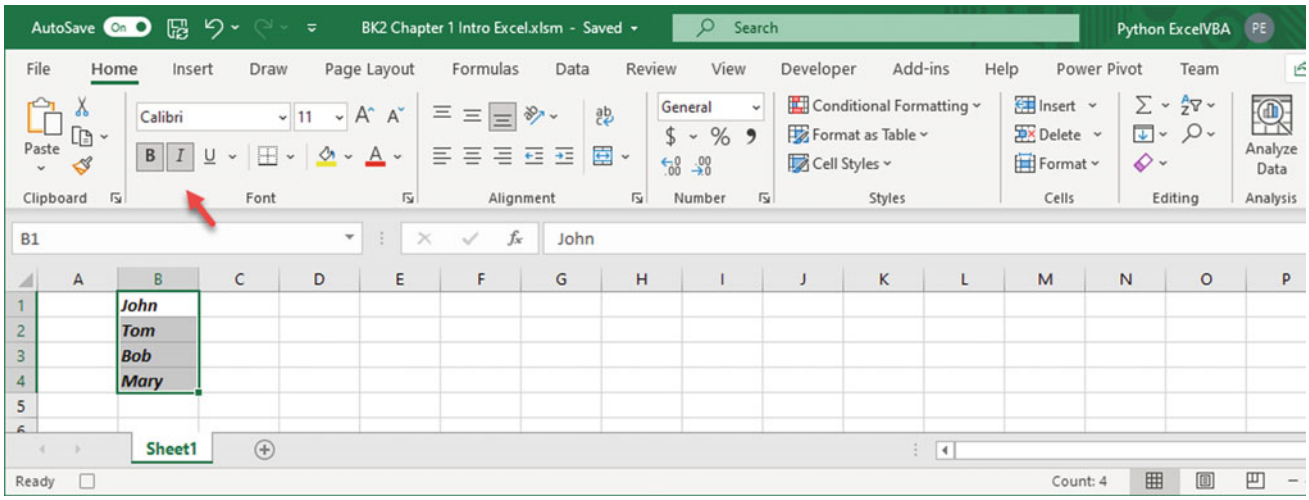


Next, type "FormatWords" in the *Macro name*: Option to indicate the name of our macro. After doing this, press the *OK* button.

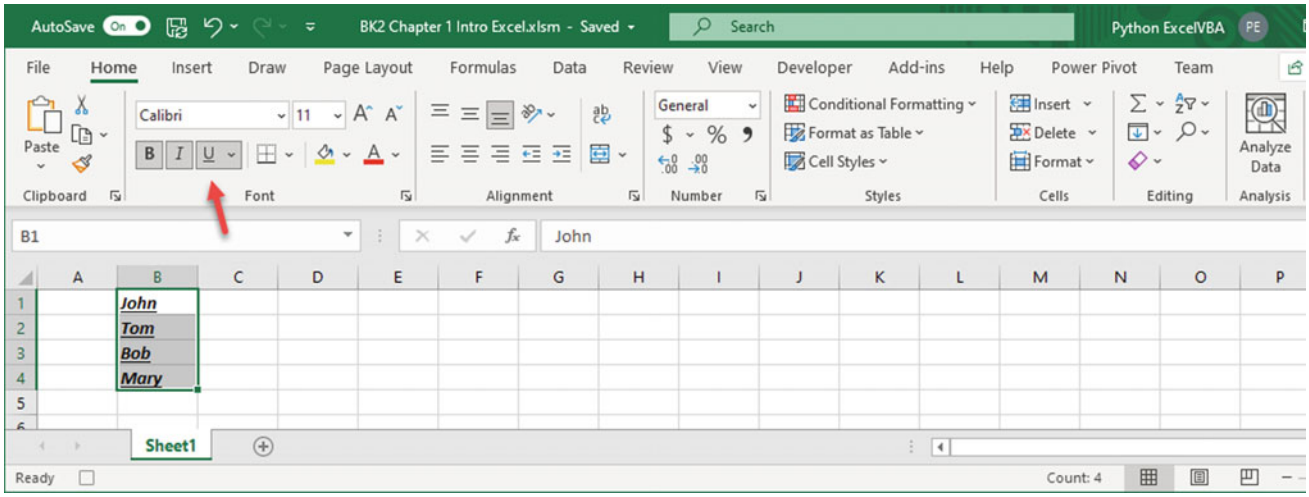
Let's first bolden the words by pressing *Ctrl* + *B* key combination on the keyboard or press the *B* button under the *Home* tab. The result of this action is shown below.



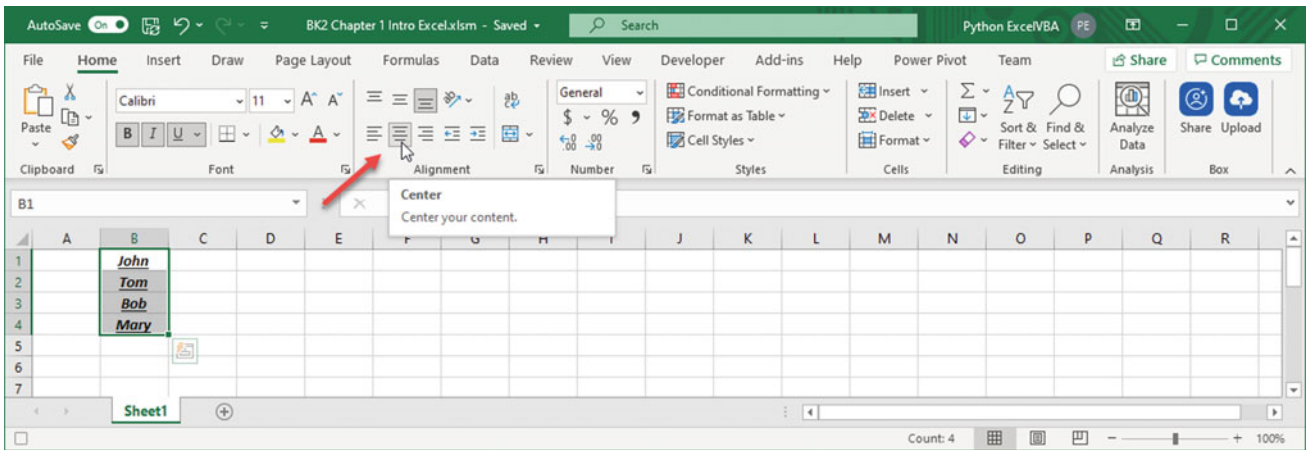
Next, italicize the words by pressing *Ctrl* + *I* key combination on the keyboard or press the *I* button under the *Home* tab. The result of this action is shown below.



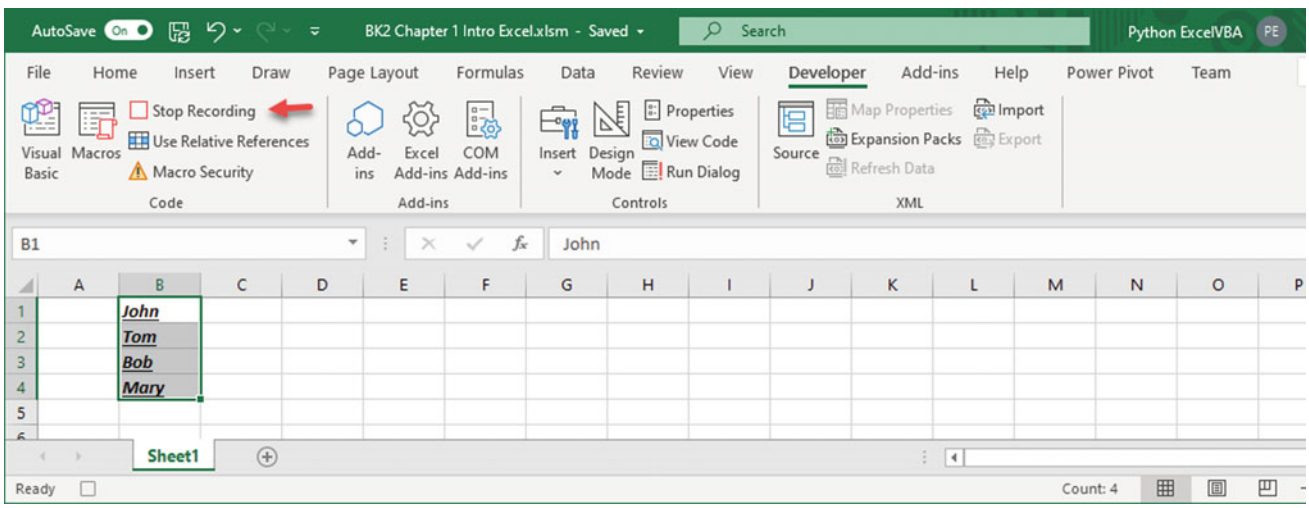
Next, underline the words by pressing *Ctrl* + *U* key combination on the keyboard or press the *U* button under the *Home* tab. The result of this action is shown below.



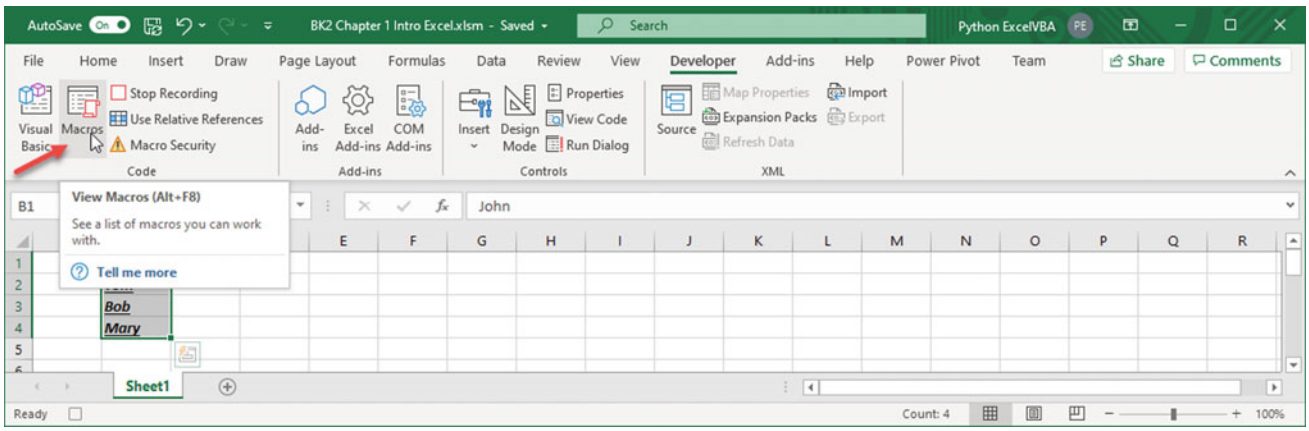
Next, center the words by pressing the *Center* button under the *Home* tab. The result of this action is shown below.



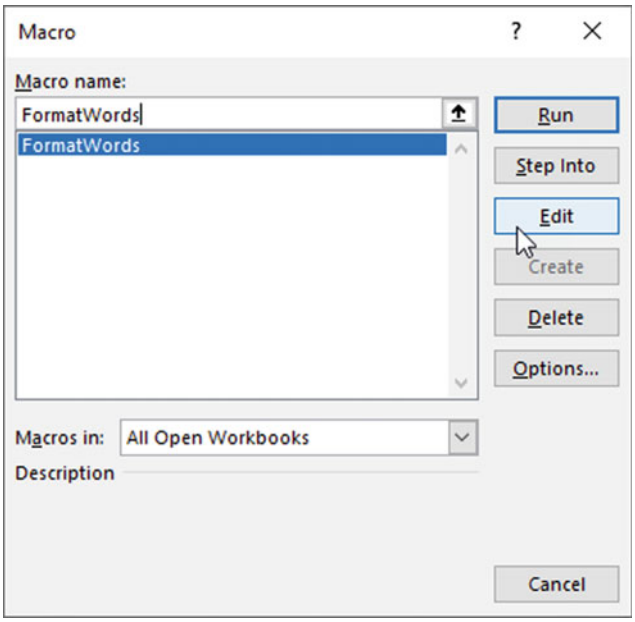
The next thing to do is stop Excel's macro recorder by clicking on the *Stop Recorder* button under the *Developer* tab. The result of this action is shown below.



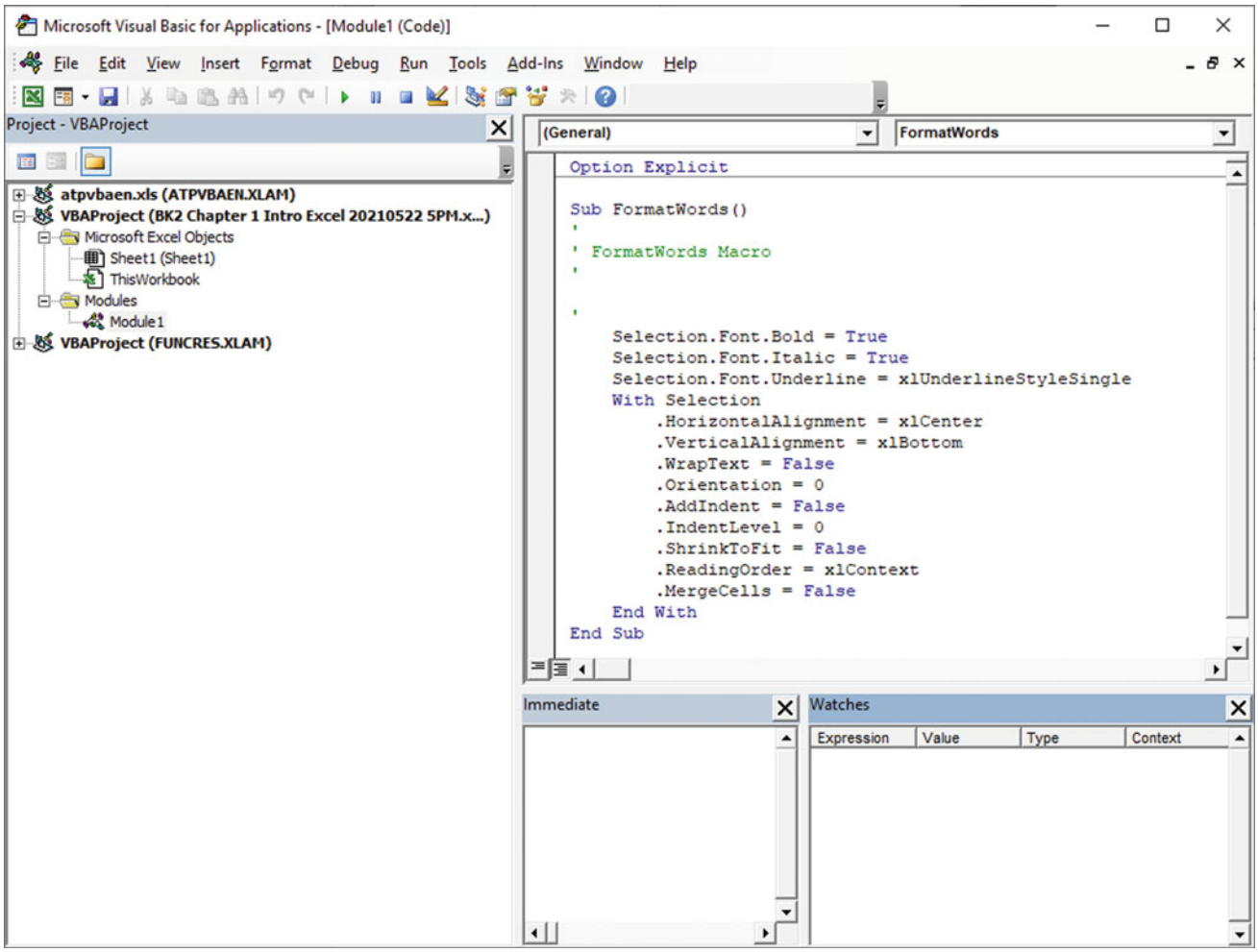
Let's look at the resulting VBA code that Excel created by pressing the *Alt + F8* key combination on the keyboard or clicking on the *Macro* button on the *Developer* tab.



Clicking on the *Macro* button will result in the *Macro* dialog box shown below.



The *Macro* dialog box shows all the available macros in a workbook. The *Macro* dialog box shows one macro, the macro that we created. Let's now look at the "FormatWords" macro that we created. To look at this macro, highlight the macro name and then press the *Edit* button on the *Macro* dialog box. Pushing the *Edit* button would result in the *Microsoft Visual Basic Editor* (VBE). The below shows the VBA code created by Excel's macro recorder.



2.3 Excel's Visual Basic Editor

The *Visual Basic Editor* (VBE) is Excel's programming environment. This programming environment is very similar to Visual Basic's programming environment. Visual Basic is the language used by professional programmers. At the top left corner of the VBE environment is the *project window*. The project window shows all workbooks and add-ins that are open in Excel. In the VBE environment, the workbooks and add-ins are called *projects*.

The *module* component is where our "FormatWords" macro resides.

The VBE environment is presented to the user in a different window than Excel. To go to the Excel window from the VBE window, press the *Alt* key and the *F11* key on the keyboard. Pressing *Alt* + *F11* keys will also navigate the user from the Excel window to the VBE window.

It should be noted that Excel writes bad VBA code. But even though Excel writes bad VBA code, it is valuable to the experienced VBA programmer. As noted above, we should realize that Excel is a feature-rich application. It is almost impossible for even an expert VBA programmer to remember how to program every feature in VBA. The above-recorded macro would be valuable to an experienced programmer that never has or has forgotten how to program the "Bold" or "Italic" or "Underline" or "Center" feature of Excel. This is where Excel's macro recorder comes to play. The end result helps guide the experienced and expert VBA programmer in how to program an Excel feature in VBA. The way that an experienced VBA programmer would write the macro "FormatWords" is shown below. We name it "FormatWords2" to distinguish it from the recorded macro.

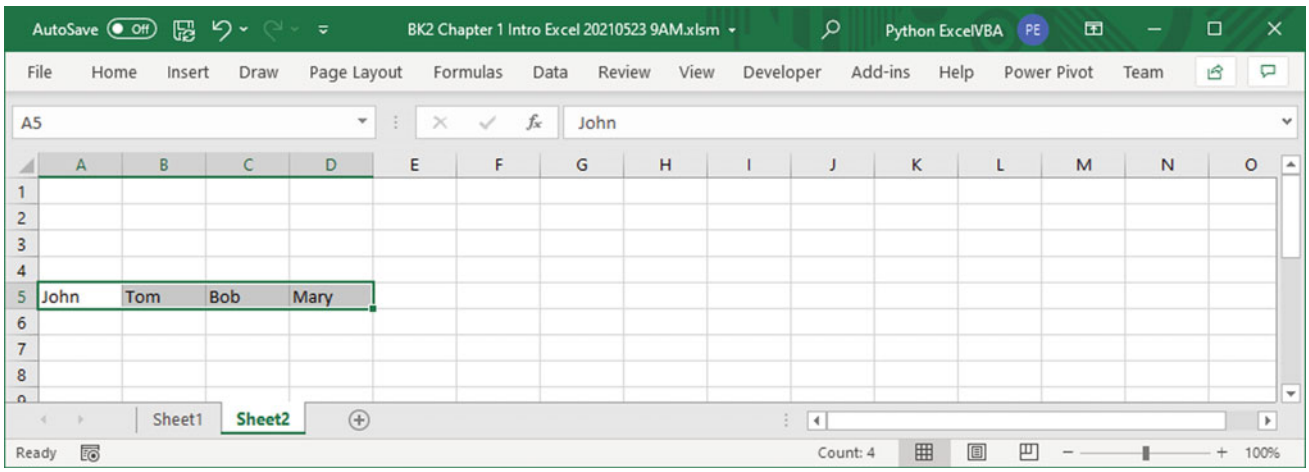
```
Sub FormatWords2()

    With Selection
        With .Font
            .Bold = True
            .Italic = True
            .Underline = xlUnderlineStyleSingle
        End With
        .HorizontalAlignment = xlCenter
    End With
End Sub
```

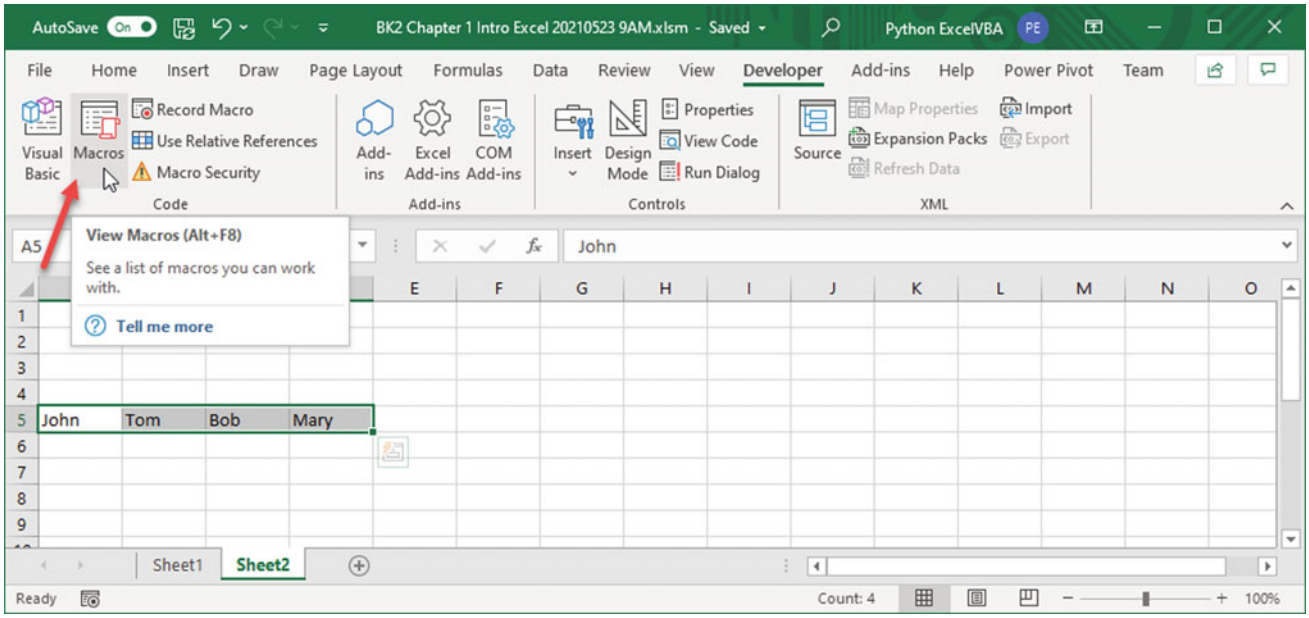
Note how much more efficient "FormatWords2" is compared to "FormatWords."

2.4 Running an Excel Macro

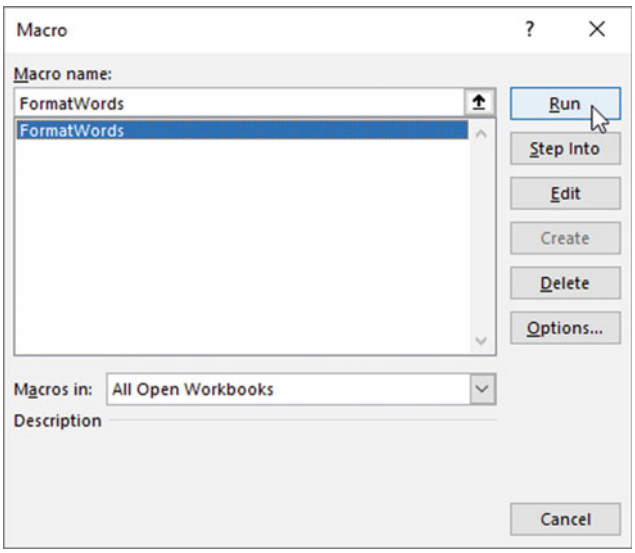
The previous section recorded the macro "FormatWords." This section will show how to run that macro. Before we do this, we will need to set up the worksheet "Sheet2." The "Sheet2" format is shown below.



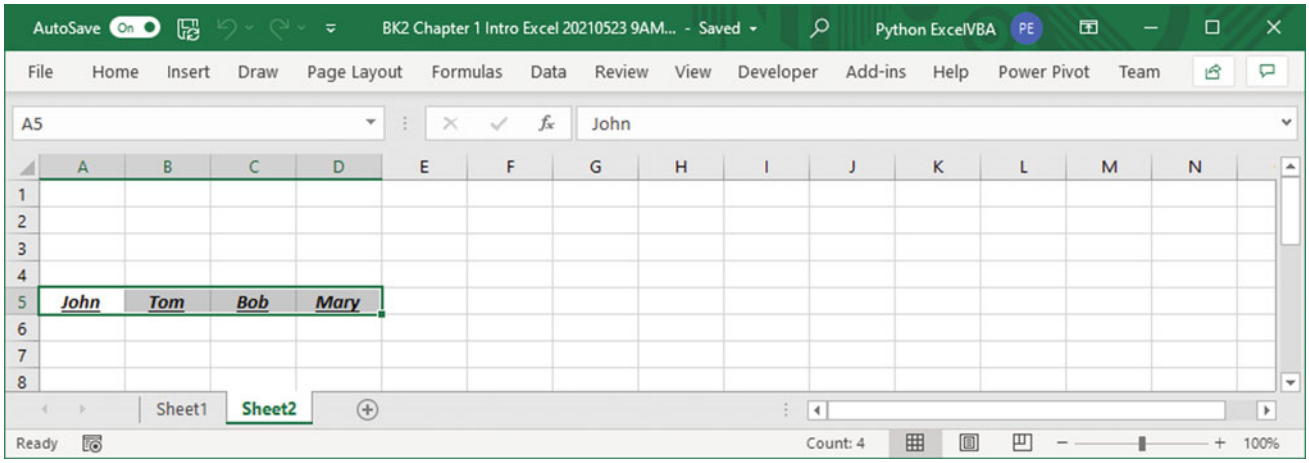
We will use the "FormatWords" macro to format the names in worksheet "Sheet2." To do this, we will need to select the names as shown above and then choose Developer → Macros or press the *Alt* + *F8* key combination.



Choosing the *Macros* menu item will display the *Macro* dialog box shown below.

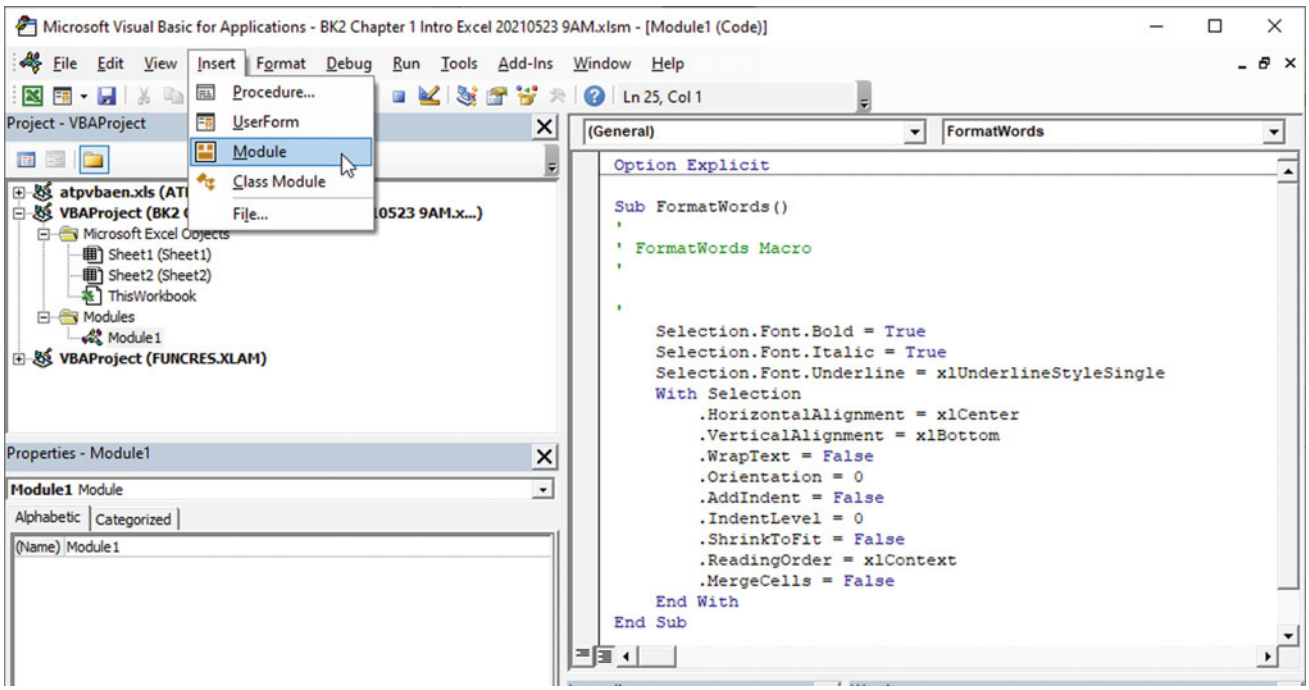


The *Macro* dialog box shows all the macros available for use. Currently, the *Macro* dialog box shows only the macro that we created. To run the macro that we created, select the macro and then press the Run button as shown above. The below shows the end result after pressing the *Run* button.

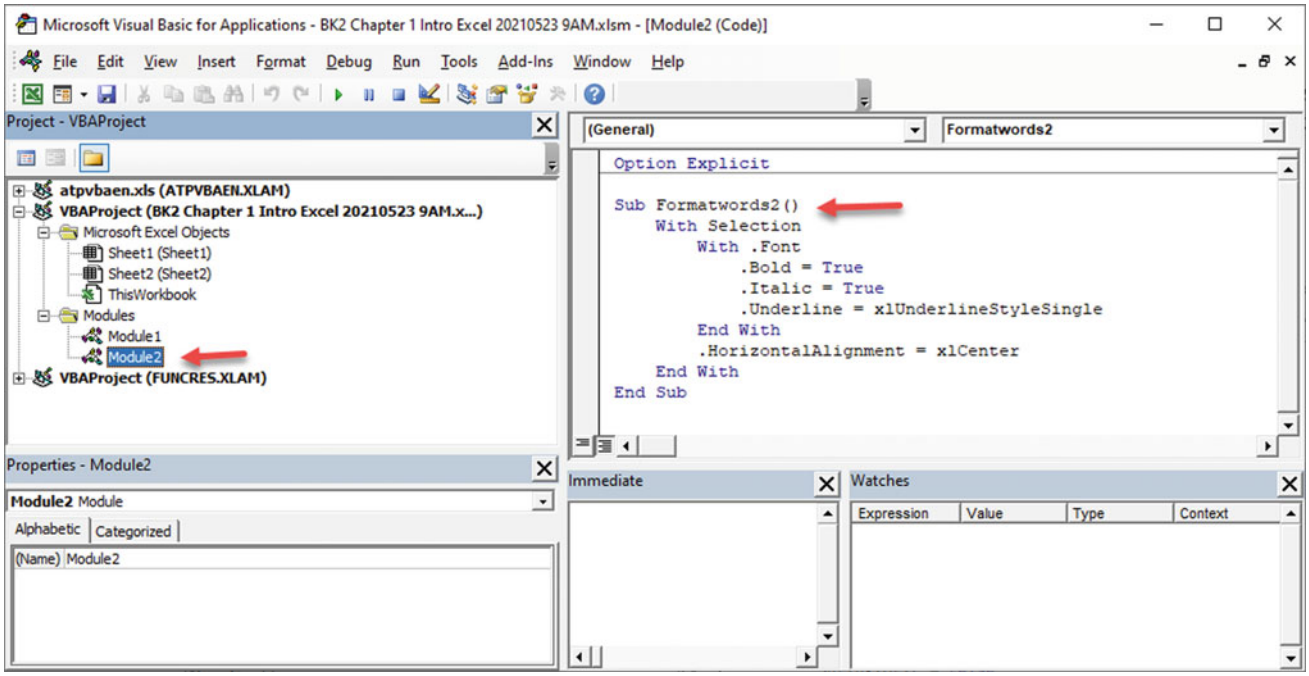


2.5 Adding Macro Code to a Workbook

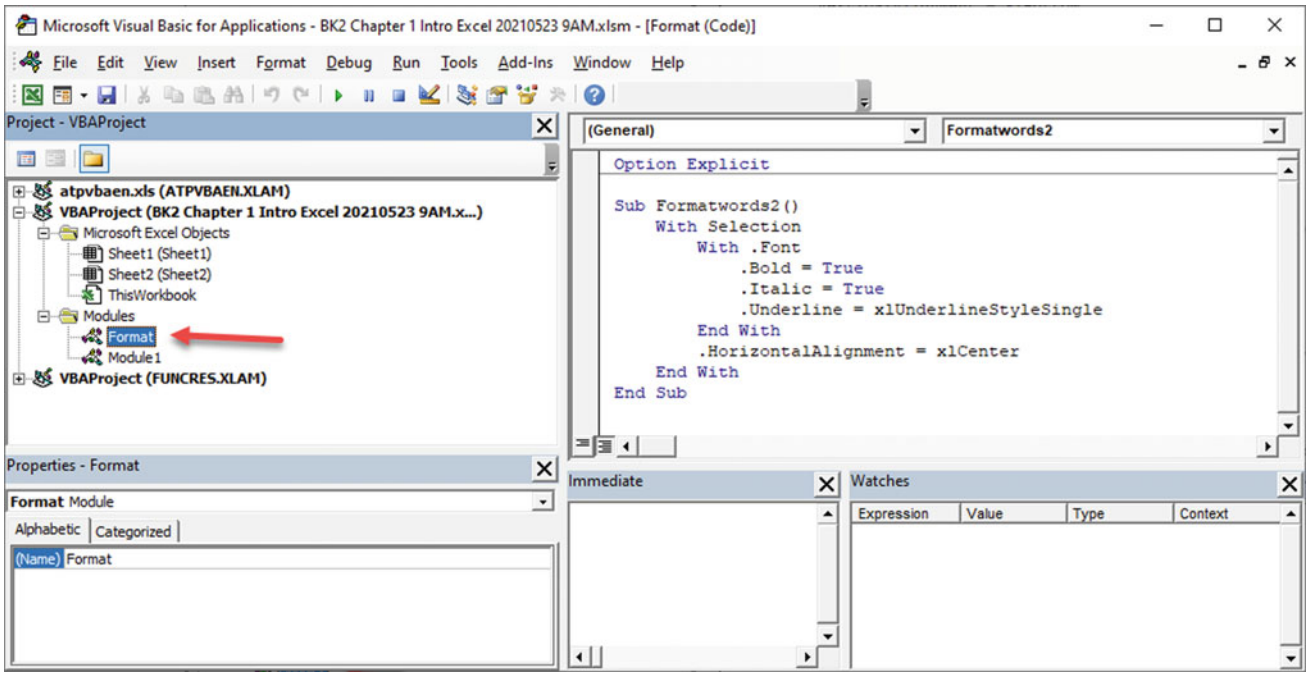
Let's now add another macro called "FormatWords2" to the workbook shown above. The first thing that we need to do is to go to the VBE editor by pressing the key combination *Alt + F11*. Let's put this macro in another module. Click on the menu item *Module* in the menu *Insert*.



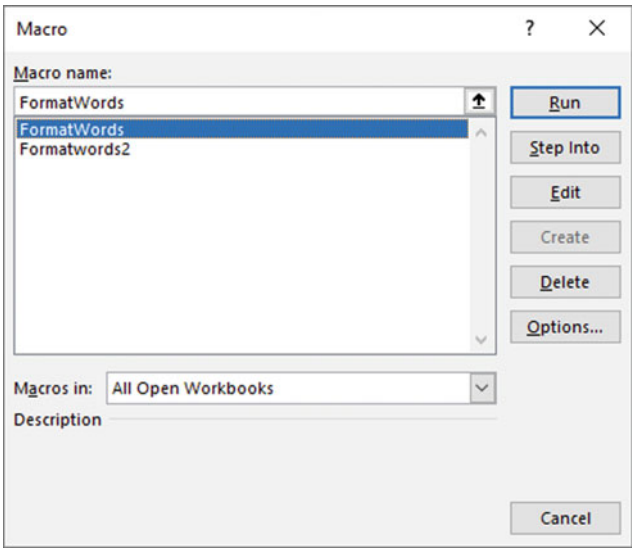
In "Module2," type in the macro "FormatWords2." The above shows the two modules and the macro "FormatWords2" in the VBE. The below also indicates that "Module2" is the active component in the project.



When the VBA program gets larger, it might make sense to name the modules to a more meaningful name. In the bottom left of the VBE window, there is a properties window for "Module2." Shown in the properties window (left bottom corner) is the name property for "Module2." Let's change the name to "Format." The below shows the end result. Notice in the project window that it now shows a "Format" module.



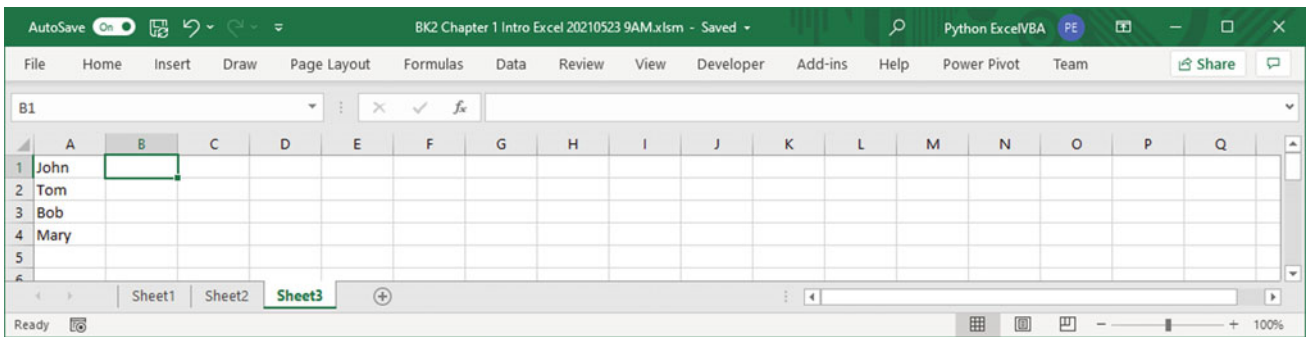
Now let's go back and look at the *Macro* dialog box. The below shows the *Macro* dialog box after typing in the macro "FormatWords2" into the VBE editor.



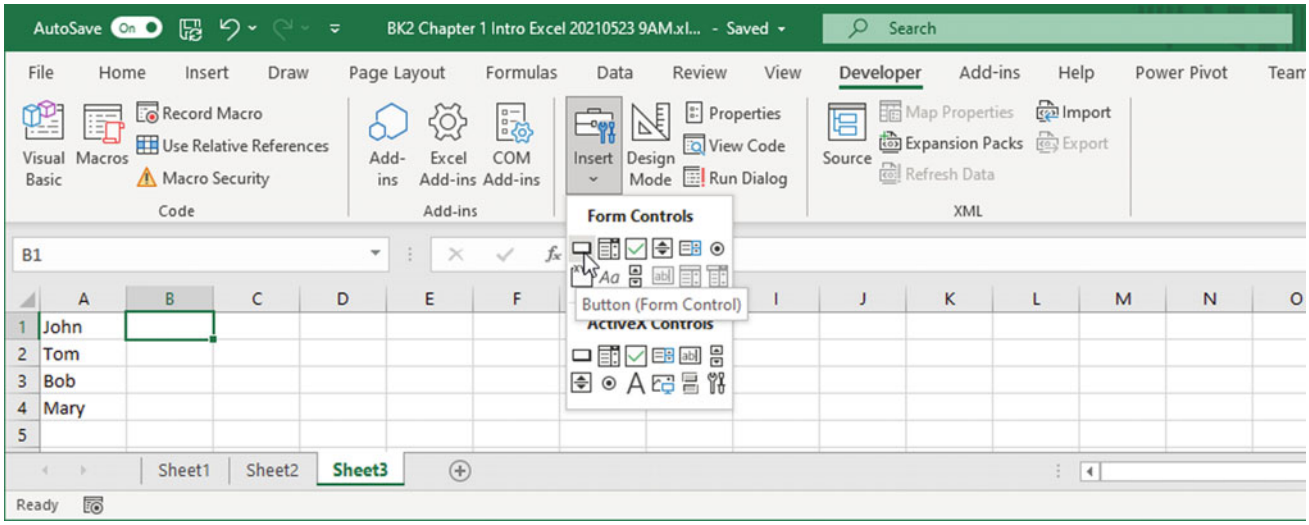
The *Macro* dialog box now shows the two macros that were created.

2.6 Macro Button

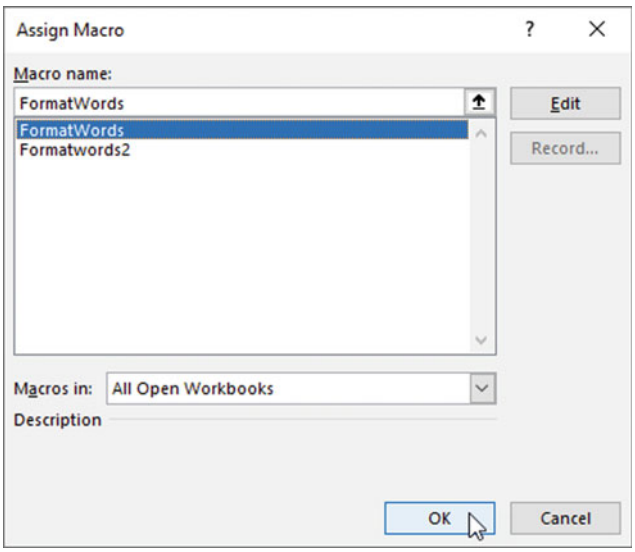
In the sections above, we used menu items to run macros. In this section, we will use *macro* buttons to execute a specific macro. *Macro* buttons are used when a specific macro is used frequently. Before we illustrate *macro* buttons, let's set up the worksheet "Sheet3," as shown below.



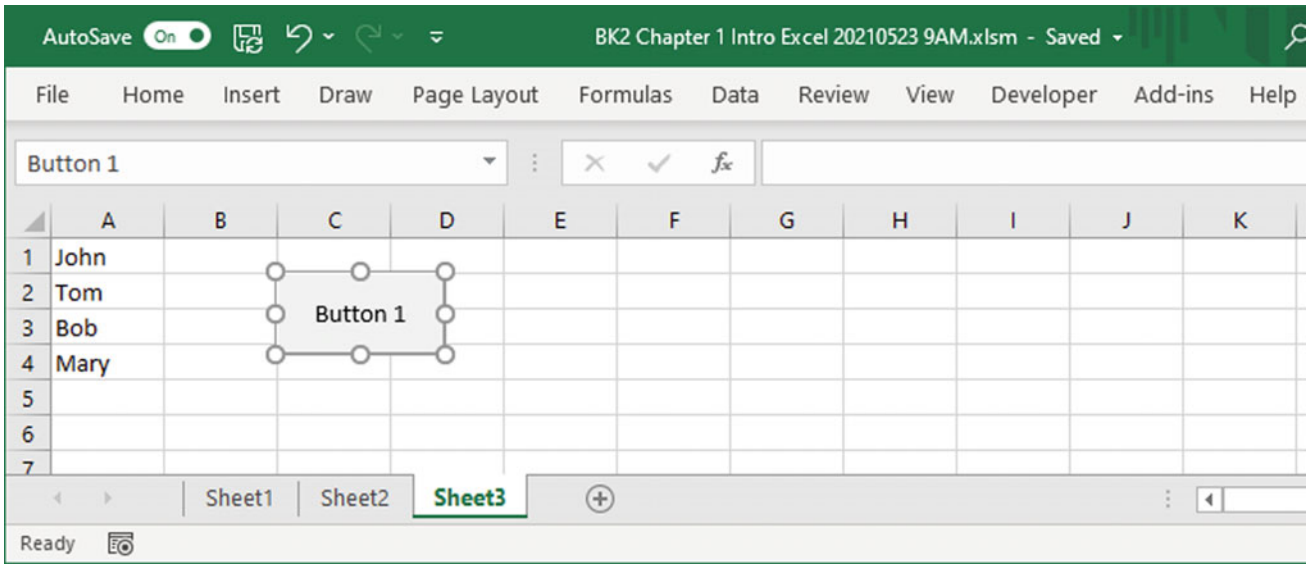
To create a macro button, go to the *Developer* tab and click on the *Form Controls* button in the *Insert* menu item, as shown below.



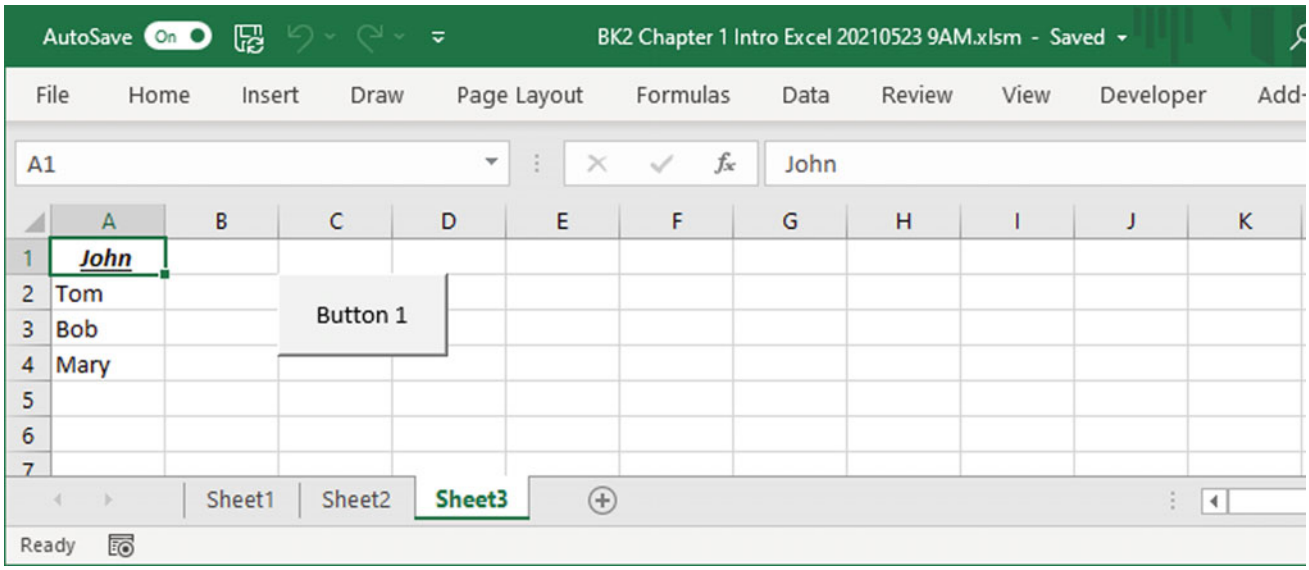
After that, click on the cell where we want the button to be located, and the *Assign Macro* dialog box will be displayed.



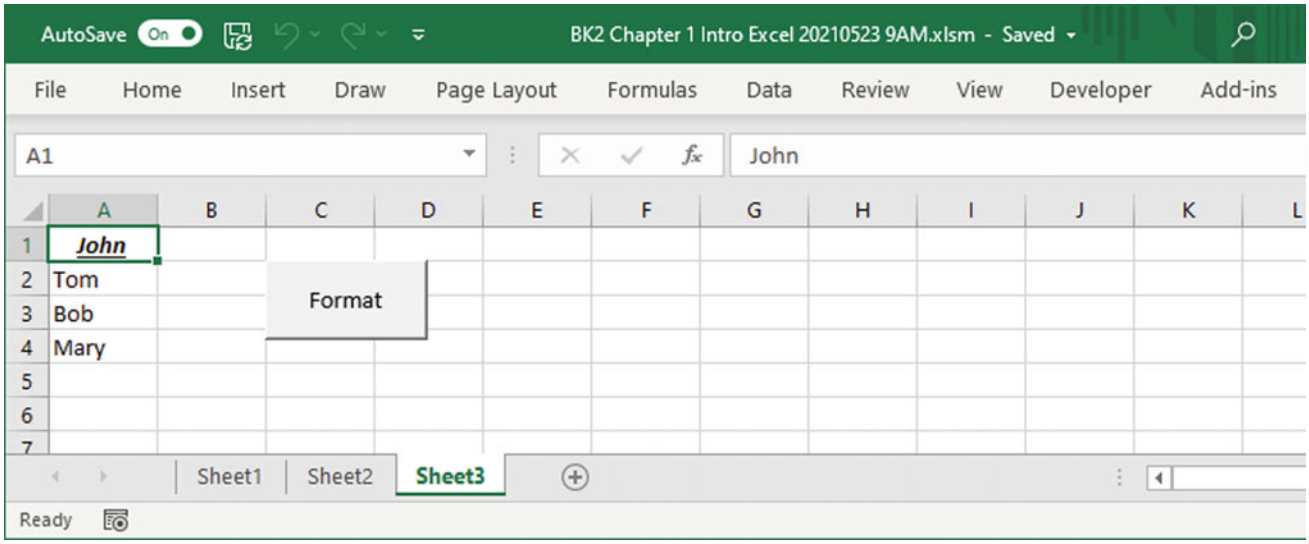
The *Assign Macro* dialog box shows all the available macros to be assigned to the button. Choose the macro "FormatWord2" as shown above and press the *OK* button. Pressing the *OK* button will assign the macro "FormatWord2" to the button. The end result is shown below.



Next, select cell A1 and move the mouse cursor over the button "Button 1" and click on the left mouse button. This action will result in cell A1 to be formatted. The end result is shown below.



The name "Button 1" for the button is probably not a good name. To change the name, move the mouse pointer over the button. After doing this, click on the right mouse button to display a shortcut menu for the button. Select *Edit Text* from the shortcut menu. Change the name to "Format." The end result is shown below.



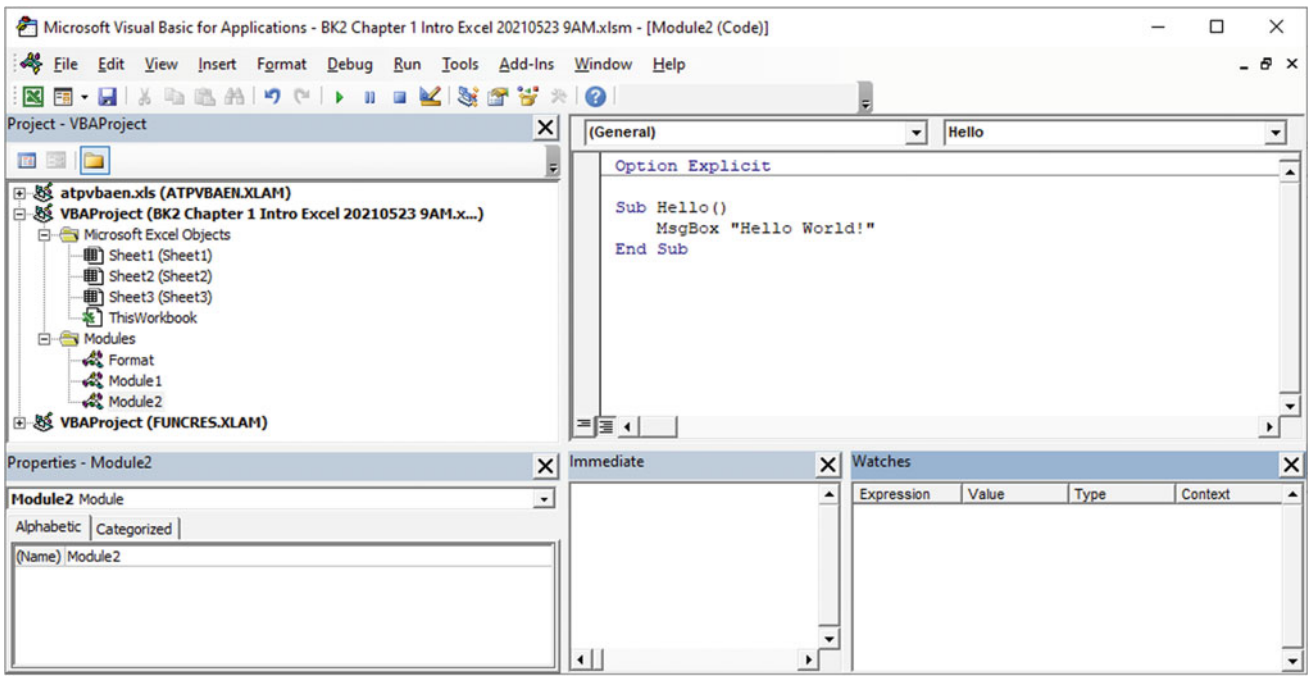
2.7 Sub Procedures

In the previous sections, we dealt with two logic groups of Excel VBA code. One group was called "FormatWords," and the other group of VBA code was called "FormatWords2." In both groups, the word *sub* was used to indicate the beginning of the group of VBA code and the words *end sub* to indicate the end of a group VBA code. Both *sub* and *end sub* are called *keywords*. Keywords are words that are part of the VBA programming language. In a basic sense, a program is an accumulation of groups of VBA codes.

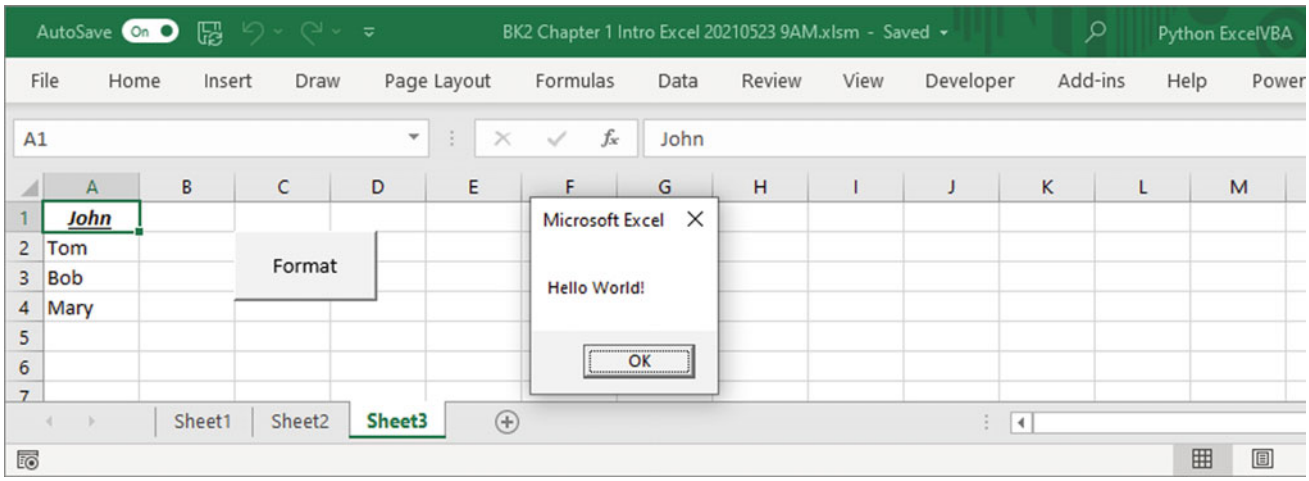
We saw in the previous sections that subprocedures in modules are all listed in the *macro* dialog box. Modules are not the only place where subprocedures are. Subprocedures can all be put in class modules and forms. These subprocedures will not be displayed in the *macro* dialog box.

2.8 Message Box and Programming Help

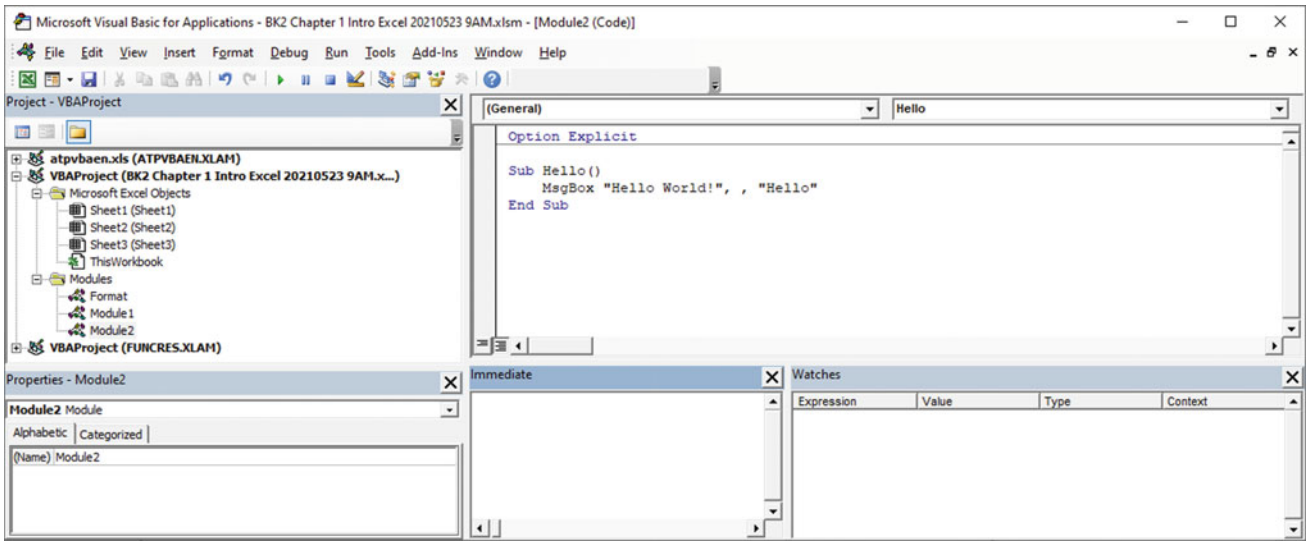
In Excel programming, it is usually necessary to communicate with the user. A simple but very popular VBA command to communicate with the user is the *msgbox* command. This command is used to display a message to the user. The below shows the very popular "Hello World" subprocedures in VBA.



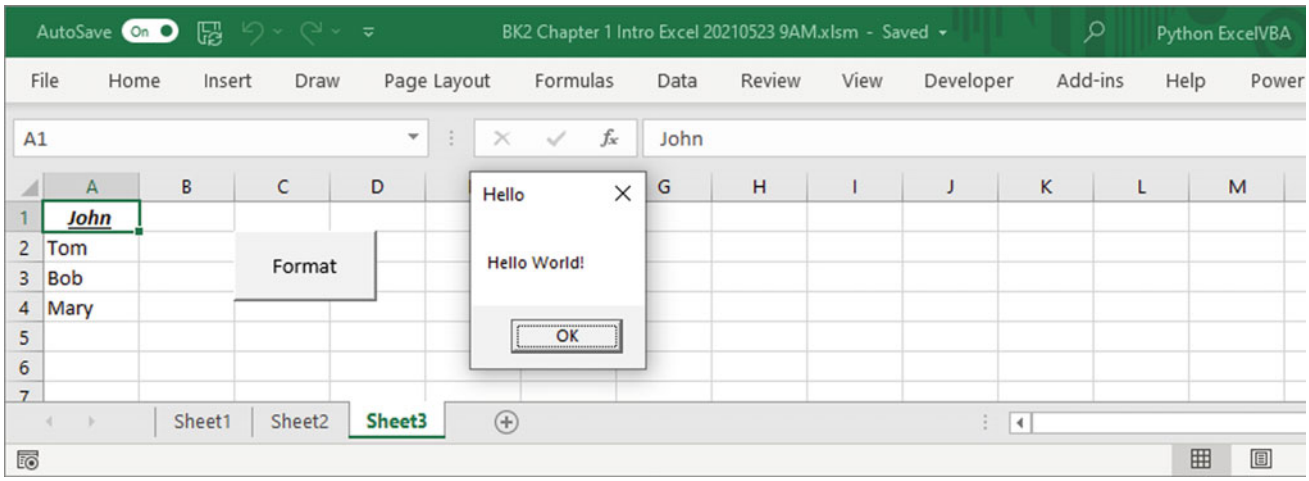
It is not necessary, as indicated in the previous section, to go to the *Macro* dialog box to run the "Hello" subprocedure shown above. To run this macro, place the cursor inside the procedure and press the *F5* key on the keyboard. Pressing the *F5* key will result in the following.



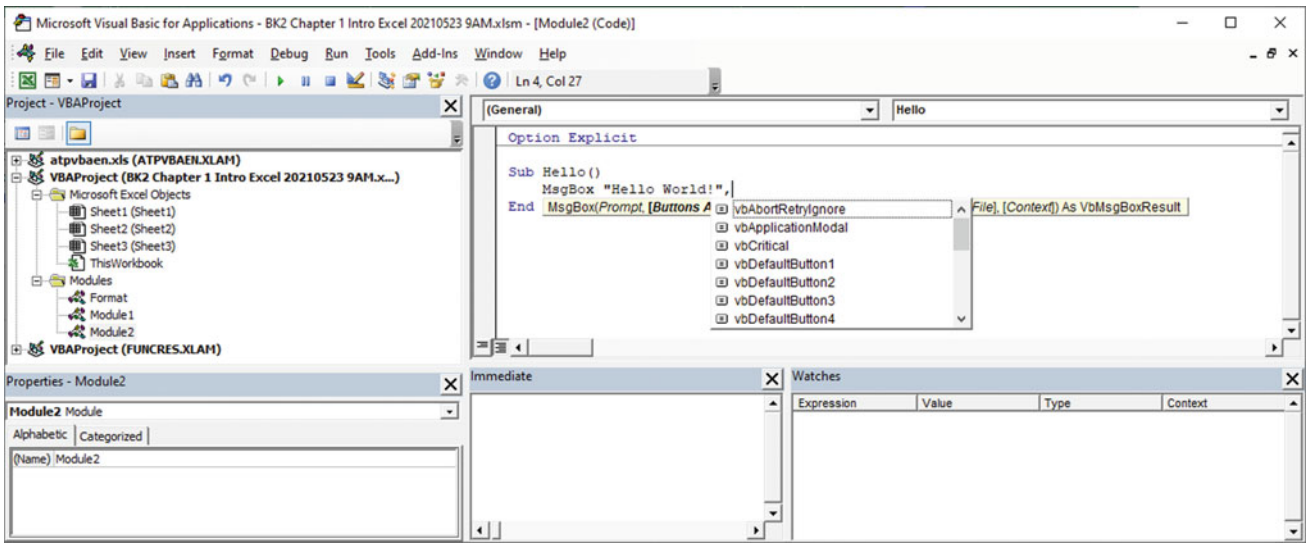
Notice that in the message box above, the title of the message box is "Microsoft Excel." Suppose we want the title of the message box to be "Hello." The below shows the VBA code to accomplish this.



The below shows the result of running the above code. Notice that the title of the message box is "Hello."

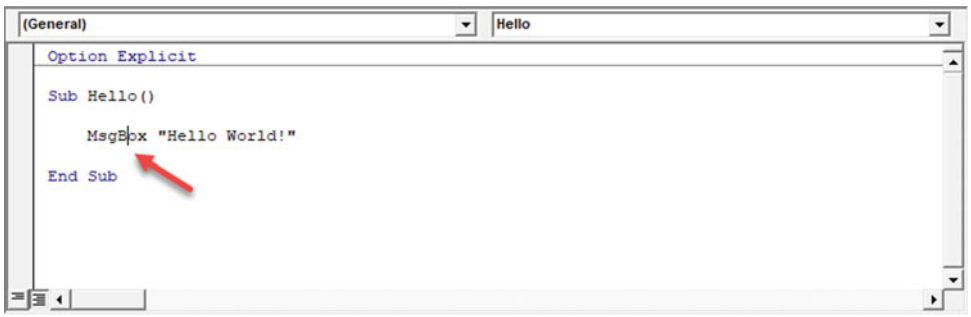


The *msgbox* command can do a lot of things. But one problem is remembering how to program all the features. The VBE editor is very good at dealing with this specific issue. Notice in the above code that commas separate the arguments to set the *msgbox* command. This then brings up the question: How many arguments does the VBA *msgbox* have? The below shows how the VBE editor assists the programmer in programming the *msgbox* command.

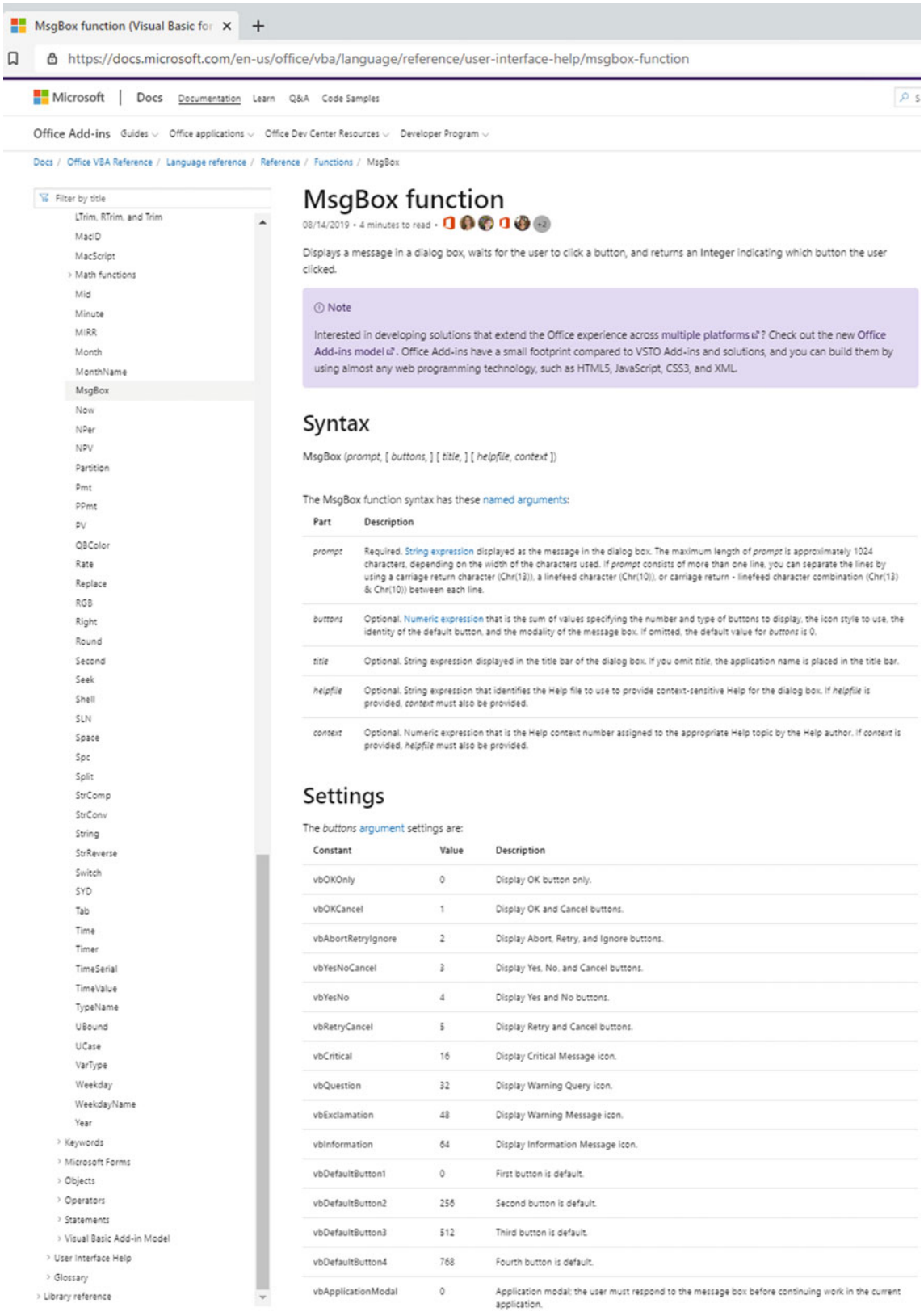


We see that after typing the first comma, the VBE editor shows two things. The first thing is a horizontal list that shows and names all the arguments of the *msgbox* command. In that list, it boldens the argument that is being updated. The second thing that the VBE editor shows is a vertical list that lists all the possible values of the arguments that we are currently working on. A list is only shown when an argument has a set of predefined values.

If the above two features are insufficient in aiding in how to program the *msgbox* command, we can place the cursor on the *msgbox* command as shown below and press the *F1* key on the keyboard.



The *F1* key launches the web browsers and navigates to the URL https://docs.microsoft.com/en-us/office/vba/language/reference/user-interface-help/msgbox-function



2.9 Excel 365 Only Features

2.9.1 Dynamic Arrays

Dynamic array is a powerful new feature that is only available in Excel 365. Dynamic arrays return array values to neighboring cells. The URL https://www.ablebits.com/office-addins-blog/2020/07/08/excel-dynamic-arrays-functions-formulas/ defines dynamic arrays as.

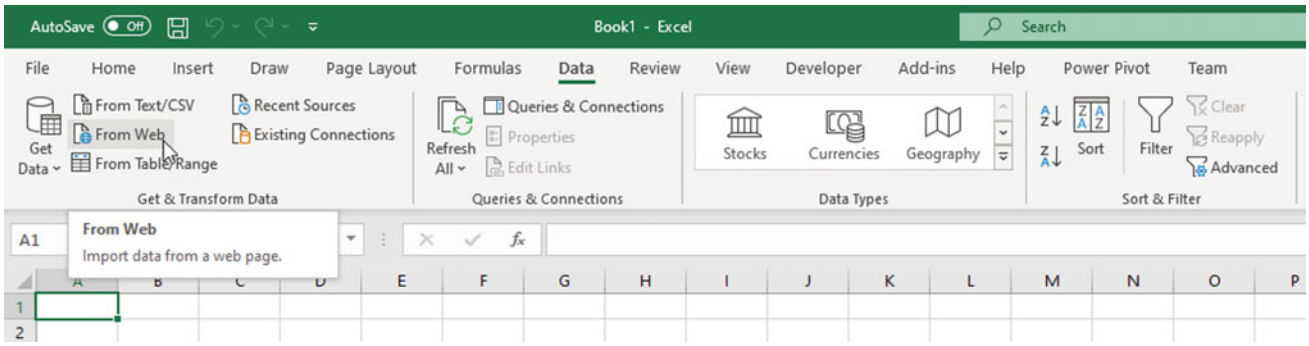
> resizable arrays that calculate automatically and return values into multiple cells based on a formula entered in a single cell.

We will demonstrate dynamic arrays on a table that shows the component performance of every component of the S&P 500. We will first demonstrate how to retrieve every component performance of the S&P 500.

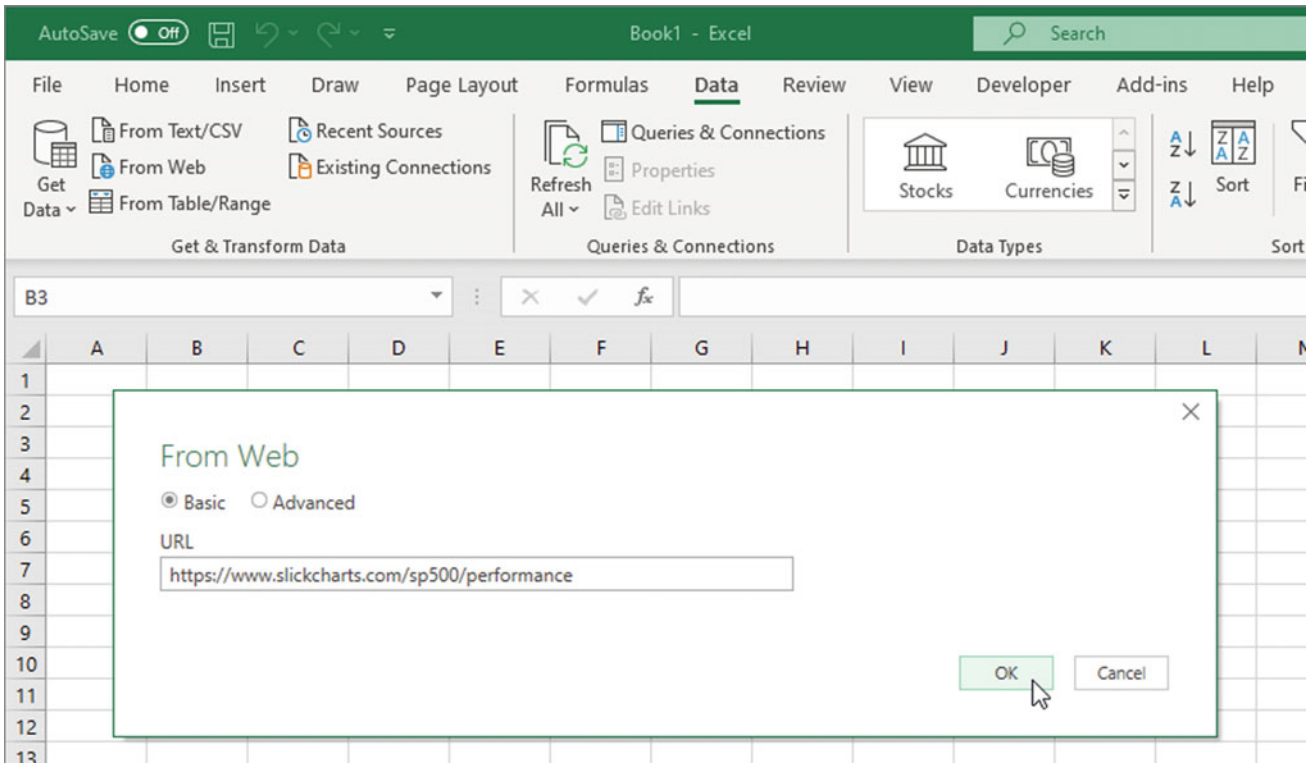
2.9.1.1 Year to Date Performance of S&P 500 Components

We will use *Power Query* to retrieve from the URL https://www.slickcharts.com/sp500/performance the year to date performance of every component of the S&P 500.

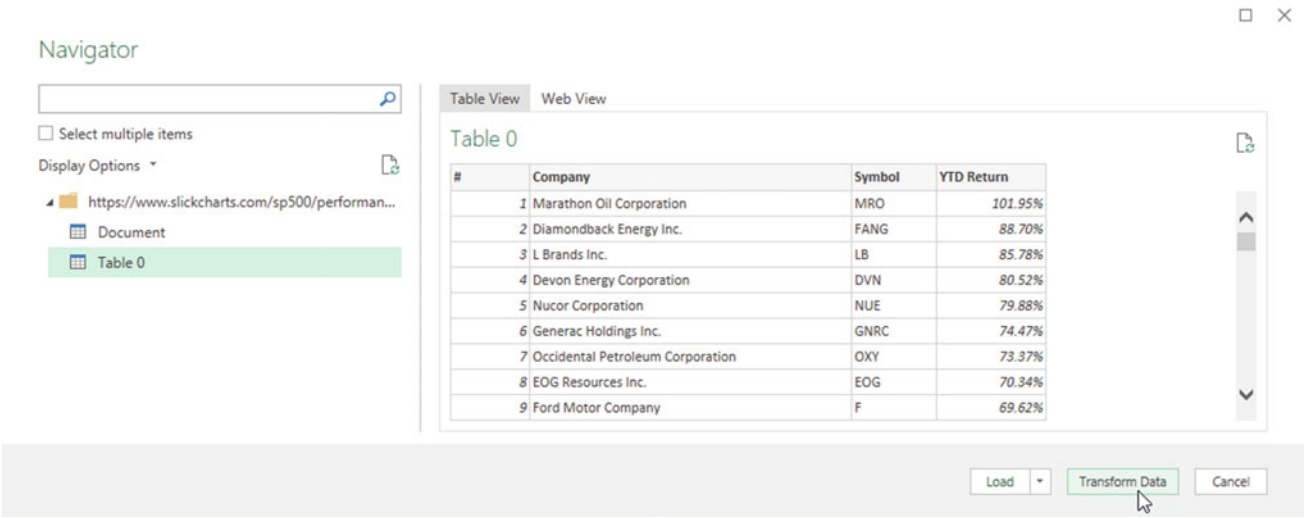
Step 1 is to click on the *From Web* button from the *Data* tab.



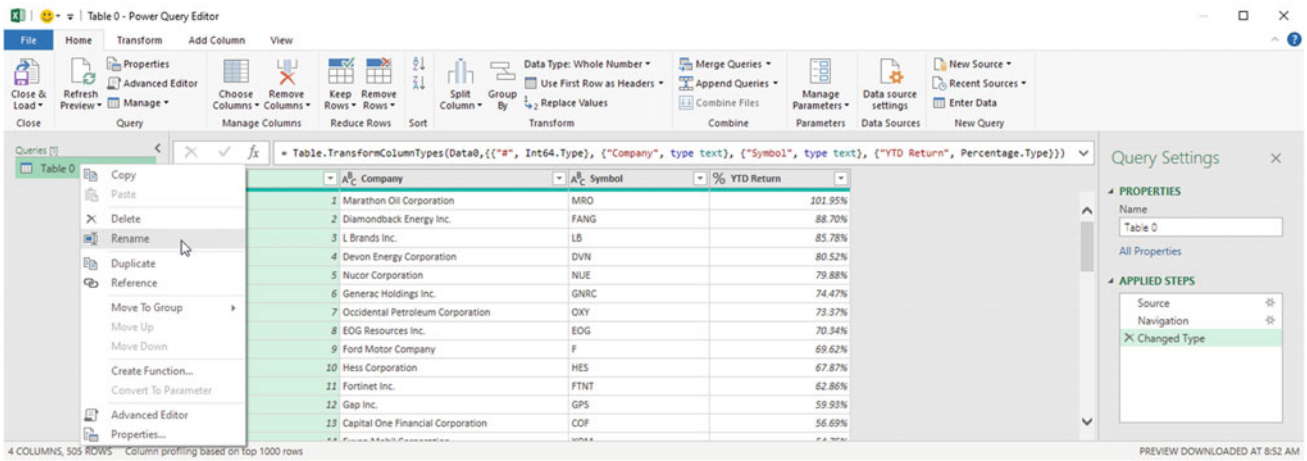
Step 2 is to enter the URL https://www.slickcharts.com/sp500/performance and then press the *OK* button.



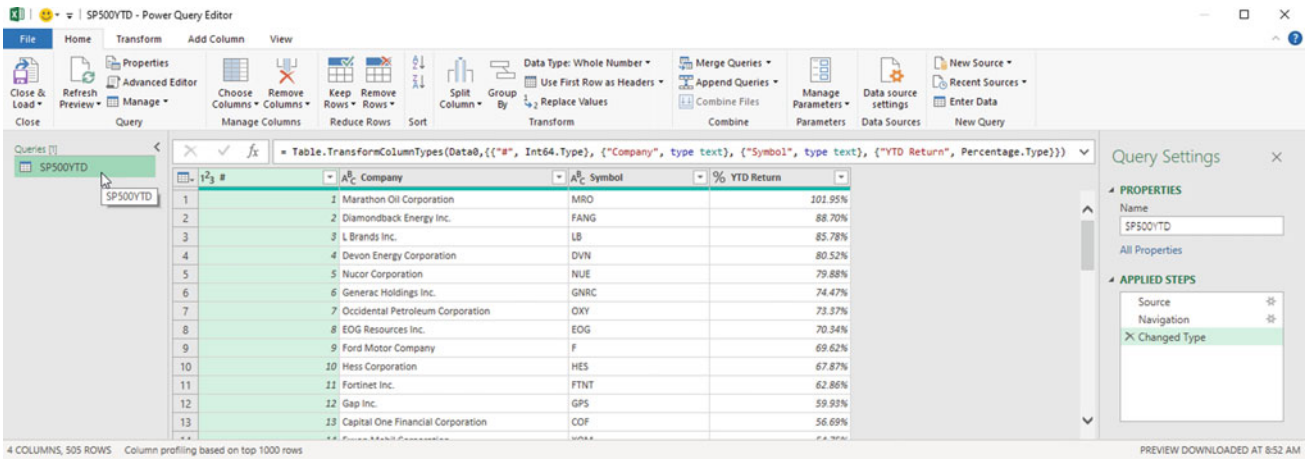
Step 3 is to click on *Table 0* and then click on the *Transform Data* button.



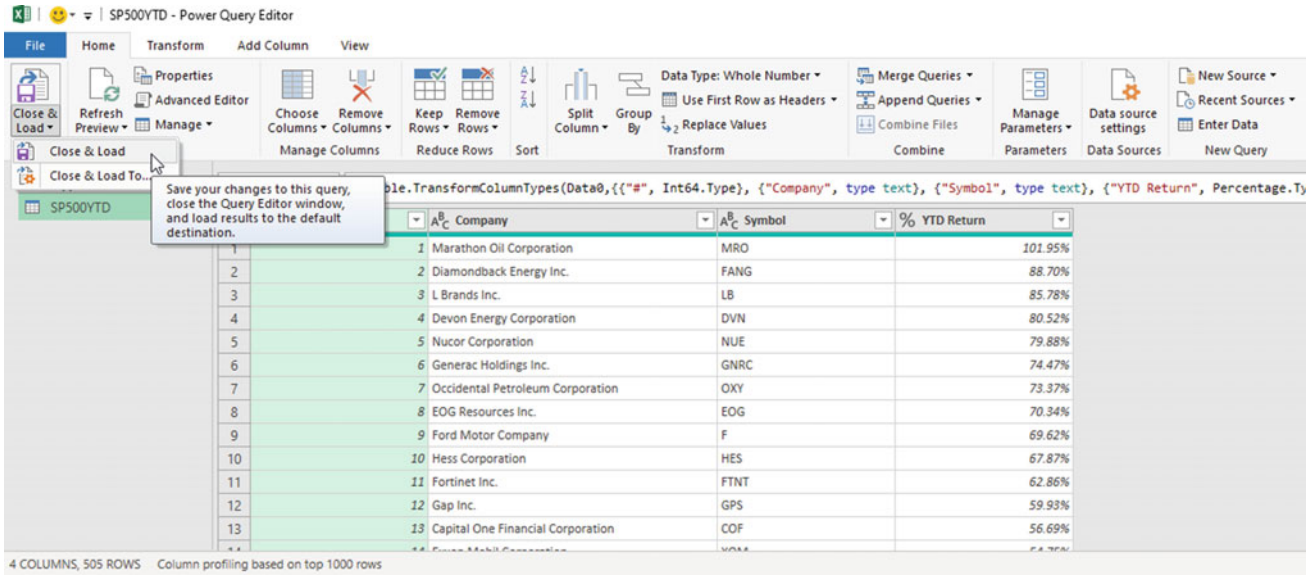
Step 4 is to right-mouse click on *Table 0,* and click on the *Rename* menu item.



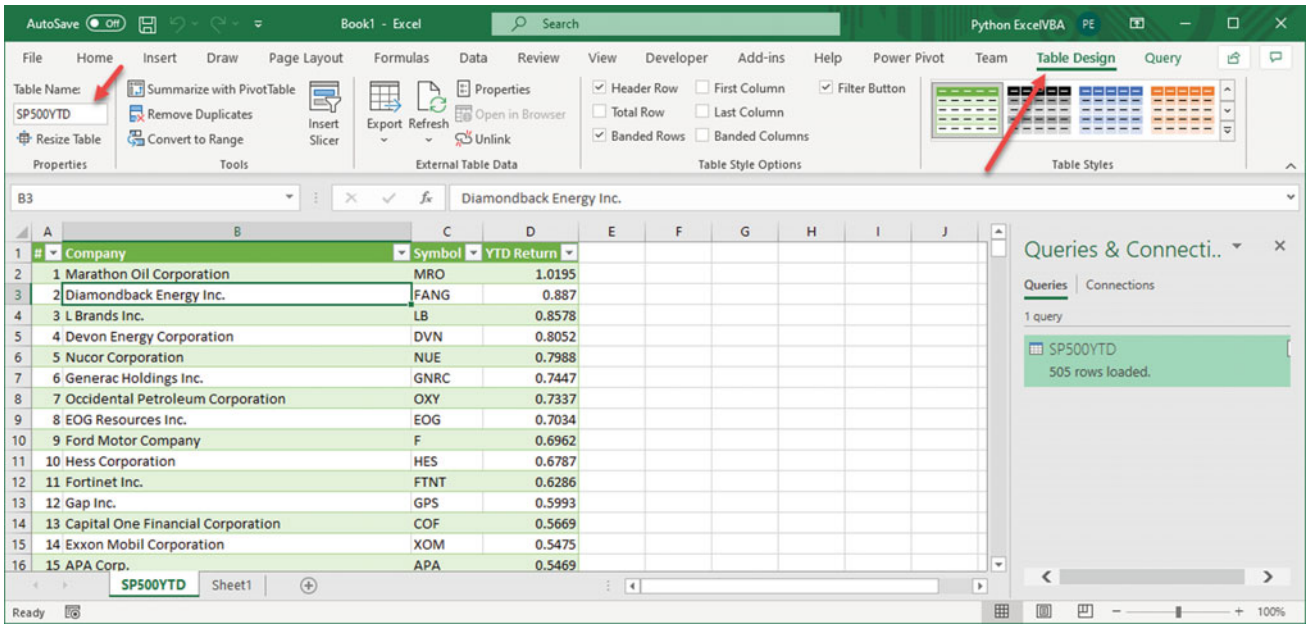
Step 4 is to rename the query *Table 0* to SP500YTD.



Step 5 is to click on *Close & Load* to load the S&P 500 YTD returns to Microsoft Excel.



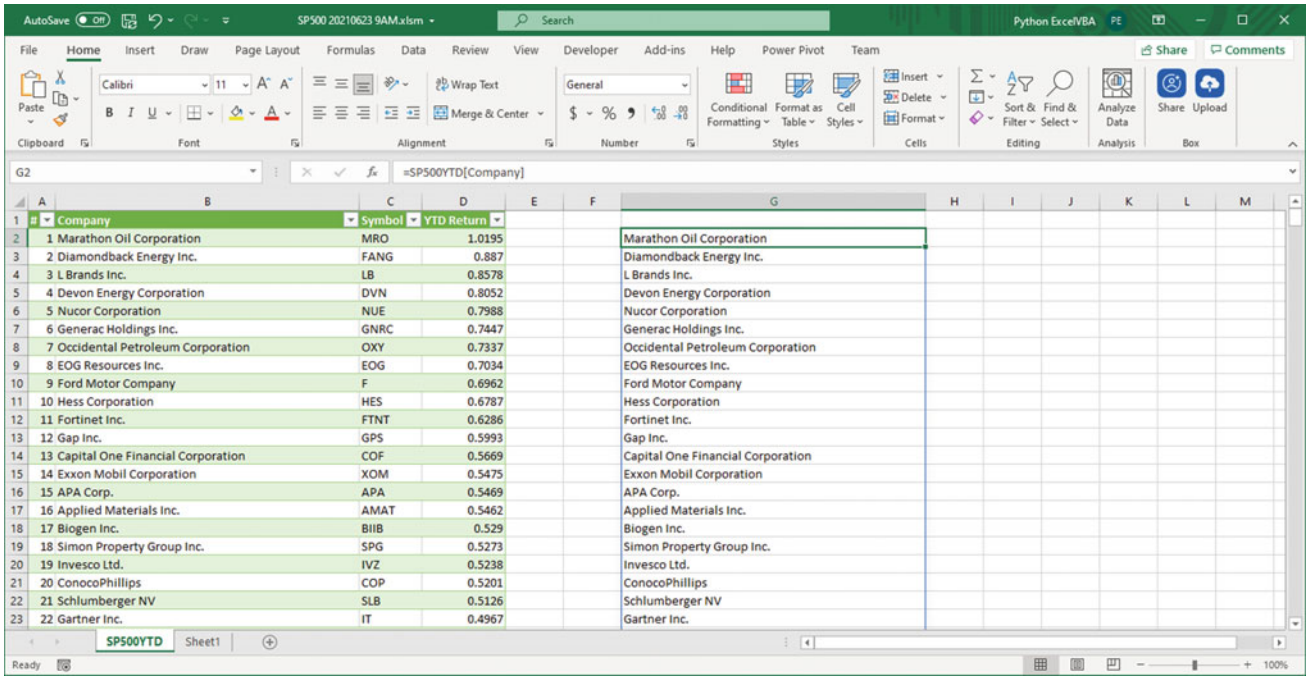
The *Power Query* result is saved in an Excel table, and the Excel table has the same name as the query *SP500YTD*. When a cell is inside an Excel table, the *Table Design* menu appears.



2.9.1.2 SORT Function

The *SORT* function is a new Excel 365 function to handle and sort dynamic arrays.

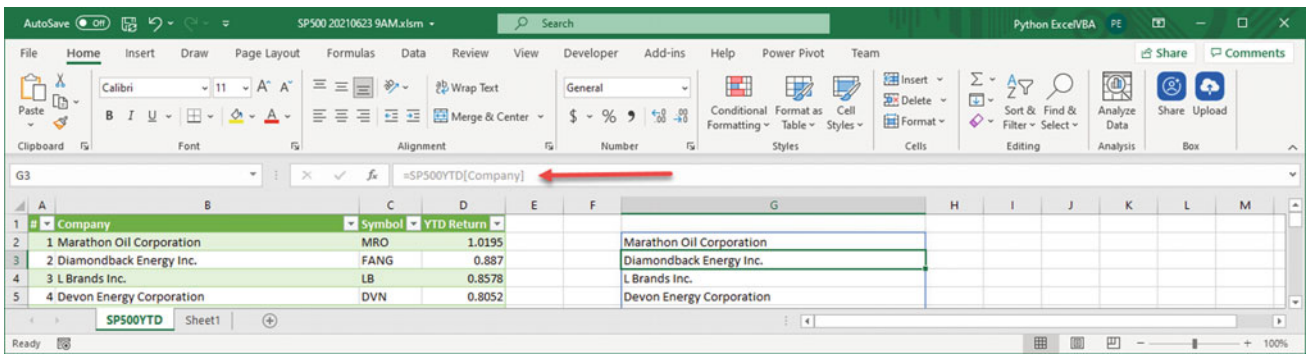
The following dynamic array returns the "Company" column in the SP500YTD table.



#	Company	Symbol	YTD Return	G
1	Marathon Oil Corporation	MRO	1.0195	Marathon Oil Corporation
2	Diamondback Energy Inc.	FANG	0.887	Diamondback Energy Inc.
3	L Brands Inc.	LB	0.8578	L Brands Inc.
4	Devon Energy Corporation	DVN	0.8052	Devon Energy Corporation
5	Nucor Corporation	NUE	0.7988	Nucor Corporation
6	Generac Holdings Inc.	GNRC	0.7447	Generac Holdings Inc.
7	Occidental Petroleum Corporation	OXY	0.7337	Occidental Petroleum Corporation
8	EOG Resources Inc.	EOG	0.7034	EOG Resources Inc.
9	Ford Motor Company	F	0.6962	Ford Motor Company
10	Hess Corporation	HES	0.6787	Hess Corporation
11	Fortinet Inc.	FTNT	0.6286	Fortinet Inc.
12	Gap Inc.	GPS	0.5993	Gap Inc.
13	Capital One Financial Corporation	COF	0.5669	Capital One Financial Corporation
14	Exxon Mobil Corporation	XOM	0.5475	Exxon Mobil Corporation
15	APA Corp.	APA	0.5469	APA Corp.
16	Applied Materials Inc.	AMAT	0.5462	Applied Materials Inc.
17	Biogen Inc.	BIIB	0.529	Biogen Inc.
18	Simon Property Group Inc.	SPG	0.5273	Simon Property Group Inc.
19	Invesco Ltd.	IVZ	0.5238	Invesco Ltd.
20	ConocoPhillips	COP	0.5201	ConocoPhillips
21	Schlumberger NV	SLB	0.5126	Schlumberger NV
22	Gartner Inc.	IT	0.4967	Gartner Inc.

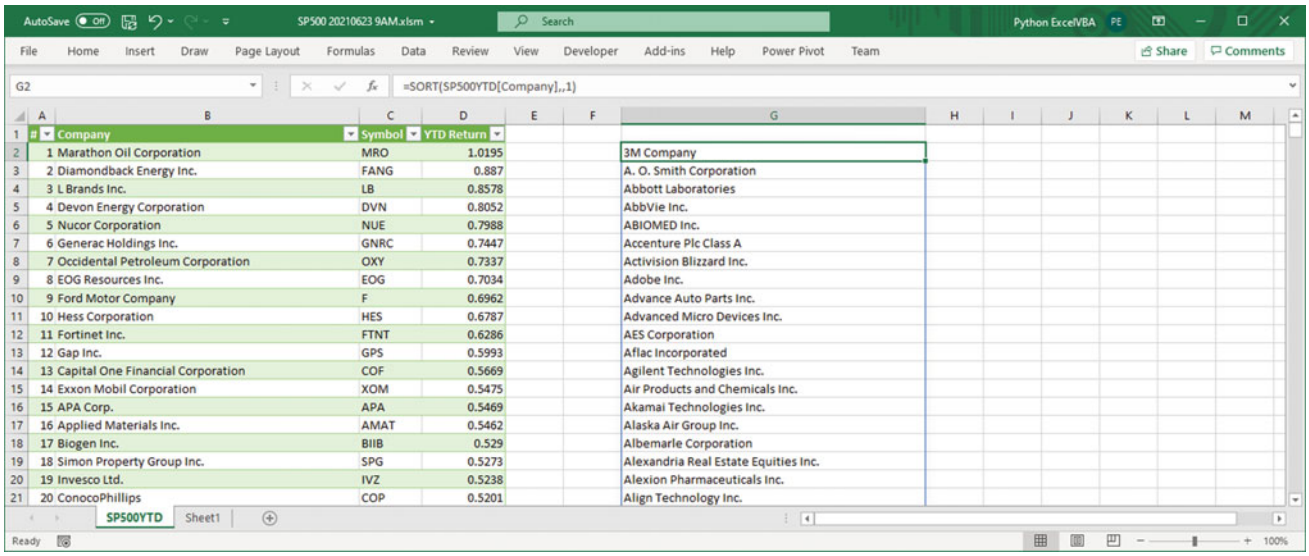
The outline in column G indicates the formula in cell G2. Dynamic arrays return array values to neighboring cells—the formula in cell G2 returns values to cells below it.

The cells below G2 contain the same formula as G2, but the formula is dimmed in the formula bar.



#	Company	Symbol	YTD Return	G
1	Marathon Oil Corporation	MRO	1.0195	Marathon Oil Corporation
2	Diamondback Energy Inc.	FANG	0.887	Diamondback Energy Inc.
3	L Brands Inc.	LB	0.8578	L Brands Inc.
4	Devon Energy Corporation	DVN	0.8052	Devon Energy Corporation

Below is the *SORT* function sorting the "Company" names.



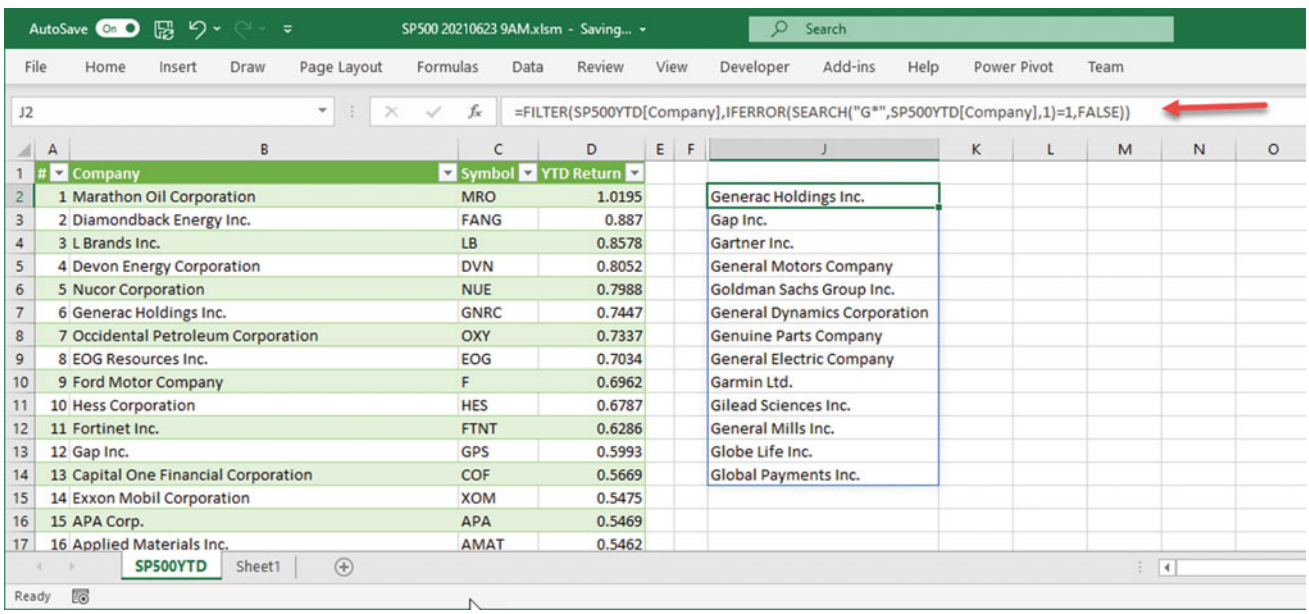
#	Company	Symbol	YTD Return
1	Marathon Oil Corporation	MRO	1.0195
2	Diamondback Energy Inc.	FANG	0.887
3	L Brands Inc.	LB	0.8578
4	Devon Energy Corporation	DVN	0.8052
5	Nucor Corporation	NUE	0.7988
6	Generac Holdings Inc.	GNRC	0.7447
7	Occidental Petroleum Corporation	OXY	0.7337
8	EOG Resources Inc.	EOG	0.7034
9	Ford Motor Company	F	0.6962
10	Hess Corporation	HES	0.6787
11	Fortinet Inc.	FTNT	0.6286
12	Gap Inc.	GPS	0.5993
13	Capital One Financial Corporation	COF	0.5669
14	Exxon Mobil Corporation	XOM	0.5475
15	APA Corp.	APA	0.5469
16	Applied Materials Inc.	AMAT	0.5462
17	Biogen Inc.	BIIB	0.529
18	Simon Property Group Inc.	SPG	0.5273
19	Invesco Ltd.	IVZ	0.5238
20	ConocoPhillips	COP	0.5201

G
3M Company
A. O. Smith Corporation
Abbott Laboratories
AbbVie Inc.
ABIOMED Inc.
Accenture Plc Class A
Activision Blizzard Inc.
Adobe Inc.
Advance Auto Parts Inc.
Advanced Micro Devices Inc.
AES Corporation
Aflac Incorporated
Agilent Technologies Inc.
Air Products and Chemicals Inc.
Akamai Technologies Inc.
Alaska Air Group Inc.
Albemarle Corporation
Alexandria Real Estate Equities Inc.
Alexion Pharmaceuticals Inc.
Align Technology Inc.

2.9.1.3 FILTER Function

The *FILTER* function is a new Excel 365 function to handle and filter dynamic arrays.

The following *FILTER* function shows all S&P 500 companies that start with the letter "G."



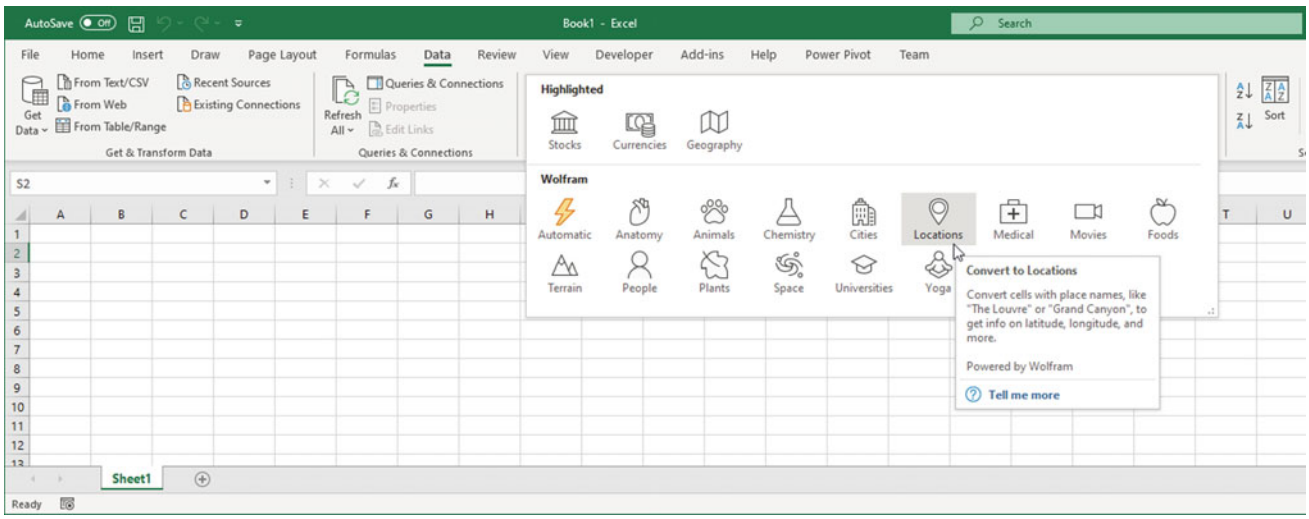
#	Company	Symbol	YTD Return
1	Marathon Oil Corporation	MRO	1.0195
2	Diamondback Energy Inc.	FANG	0.887
3	L Brands Inc.	LB	0.8578
4	Devon Energy Corporation	DVN	0.8052
5	Nucor Corporation	NUE	0.7988
6	Generac Holdings Inc.	GNRC	0.7447
7	Occidental Petroleum Corporation	OXY	0.7337
8	EOG Resources Inc.	EOG	0.7034
9	Ford Motor Company	F	0.6962
10	Hess Corporation	HES	0.6787
11	Fortinet Inc.	FTNT	0.6286
12	Gap Inc.	GPS	0.5993
13	Capital One Financial Corporation	COF	0.5669
14	Exxon Mobil Corporation	XOM	0.5475
15	APA Corp.	APA	0.5469
16	Applied Materials Inc.	AMAT	0.5462

J
Generac Holdings Inc.
Gap Inc.
Gartner Inc.
General Motors Company
Goldman Sachs Group Inc.
General Dynamics Corporation
Genuine Parts Company
General Electric Company
Garmin Ltd.
Gilead Sciences Inc.
General Mills Inc.
Globe Life Inc.
Global Payments Inc.

2.9.2 Rich Data Types

Rich Data Type connects to a data source outside of Microsoft Excel. The data from *Rich Data Types* can be refreshed.

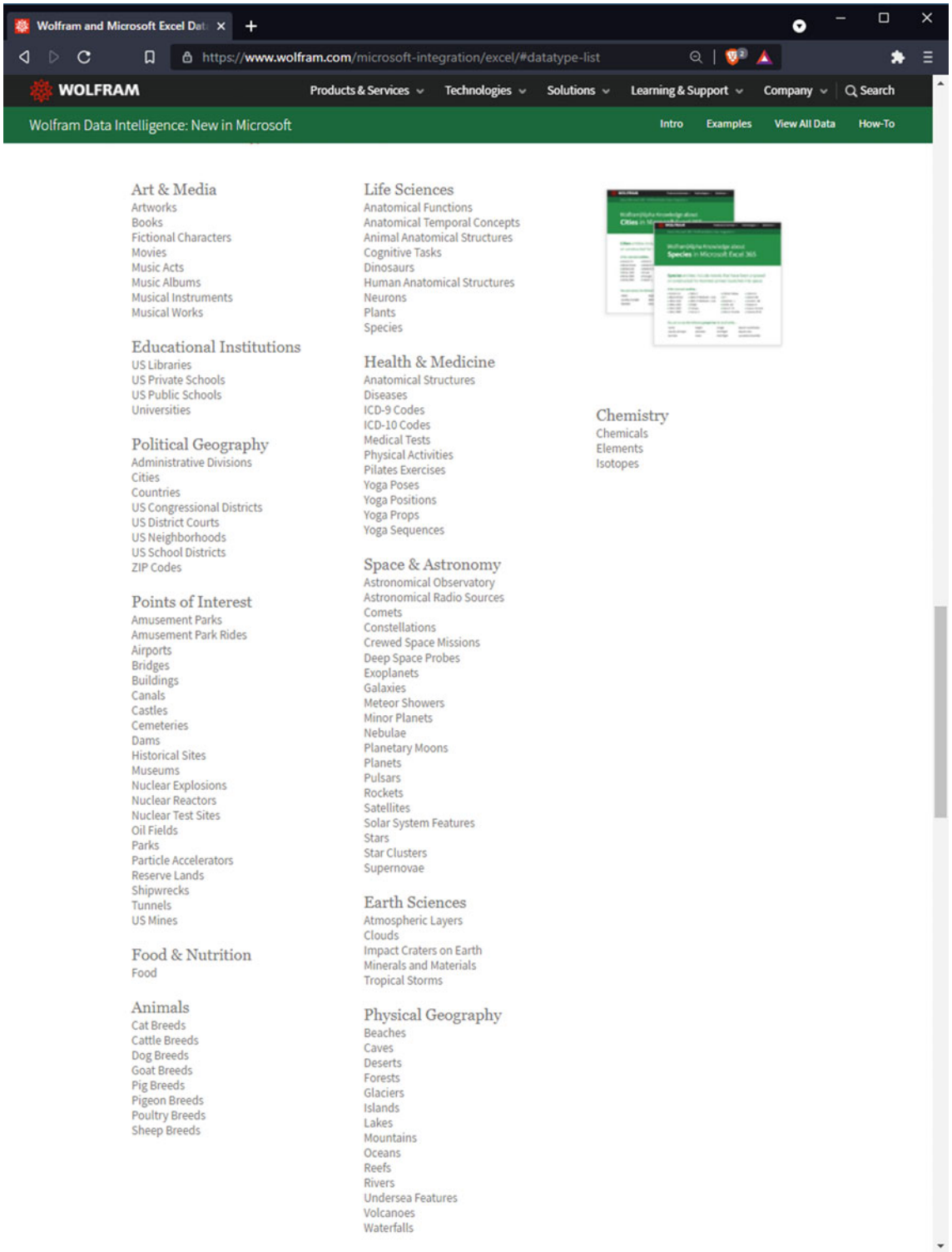
Rich Data Types are located in the *Data* tab.



Refinitiv, https://www.refinitiv.com/en, is the data source for the *Stock Data* and *Currencies* type.

Wolfram, https://www.wolfram.com/, is the data source for more than 100 data types. Use the *Automatic* data type and let Excel detect which data type to use.

The URL https://www.wolfram.com/microsoft-integration/excel/#datatype-list lists the available data types from Wolfram.

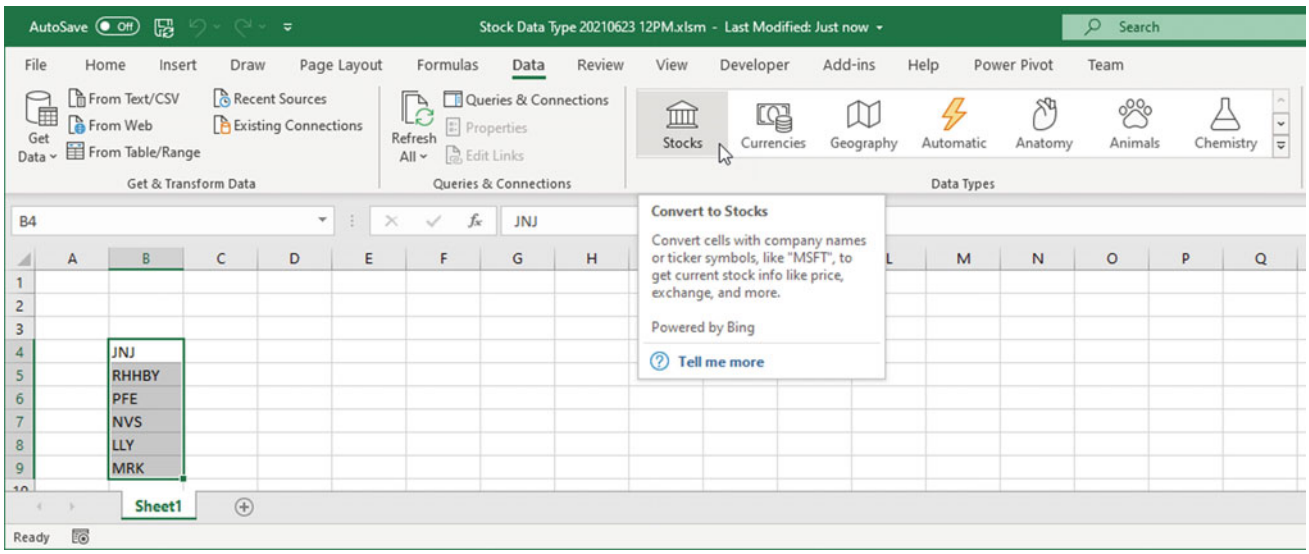

Wolfram and Microsoft Excel Dat
https://www.wolfram.com/microsoft-integration/excel/#datatype-list
WOLFRAM
Products & Services
Technologies
Solutions
Learning & Support
Company
Search
Wolfram Data Intelligence: New in Microsoft
Intro
Examples
View All Data
How-To
Art & Media
Artworks
Books
Fictional Characters
Movies
Music Acts
Music Albums
Musical Instruments
Musical Works
Educational Institutions
US Libraries
US Private Schools
US Public Schools
Universities
Political Geography
Administrative Divisions
Cities
Countries
US Congressional Districts
US District Courts
US Neighborhoods
US School Districts
ZIP Codes
Points of Interest
Amusement Parks
Amusement Park Rides
Airports
Bridges
Buildings
Canals
Castles
Cemeteries
Dams
Historical Sites
Museums
Nuclear Explosions
Nuclear Reactors
Nuclear Test Sites
Oil Fields
Parks
Particle Accelerators
Reserve Lands
Shipwrecks
Tunnels
US Mines
Food & Nutrition
Food
Animals
Cat Breeds
Cattle Breeds
Dog Breeds
Goat Breeds
Pig Breeds
Pigeon Breeds
Poultry Breeds
Sheep Breeds
Life Sciences
Anatomical Functions
Anatomical Temporal Concepts
Animal Anatomical Structures
Cognitive Tasks
Dinosaurs
Human Anatomical Structures
Neurons
Plants
Species
Health & Medicine
Anatomical Structures
Diseases
ICD-9 Codes
ICD-10 Codes
Medical Tests
Physical Activities
Pilates Exercises
Yoga Poses
Yoga Positions
Yoga Props
Yoga Sequences
Space & Astronomy
Astronomical Observatory
Astronomical Radio Sources
Comets
Constellations
Crewed Space Missions
Deep Space Probes
Exoplanets
Galaxies
Meteor Showers
Minor Planets
Nebulae
Planetary Moons
Planets
Pulsars
Rockets
Satellites
Solar System Features
Stars
Star Clusters
Supernovae
Earth Sciences
Atmospheric Layers
Clouds
Impact Craters on Earth
Minerals and Materials
Tropical Storms
Physical Geography
Beaches
Caves
Deserts
Forests
Glaciers
Islands
Lakes
Mountains
Oceans
Reefs
Rivers
Undersea Features
Volcanoes
Waterfalls
Chemistry
Chemicals
Elements
Isotopes

2.9.2.1 Stocks Data Type

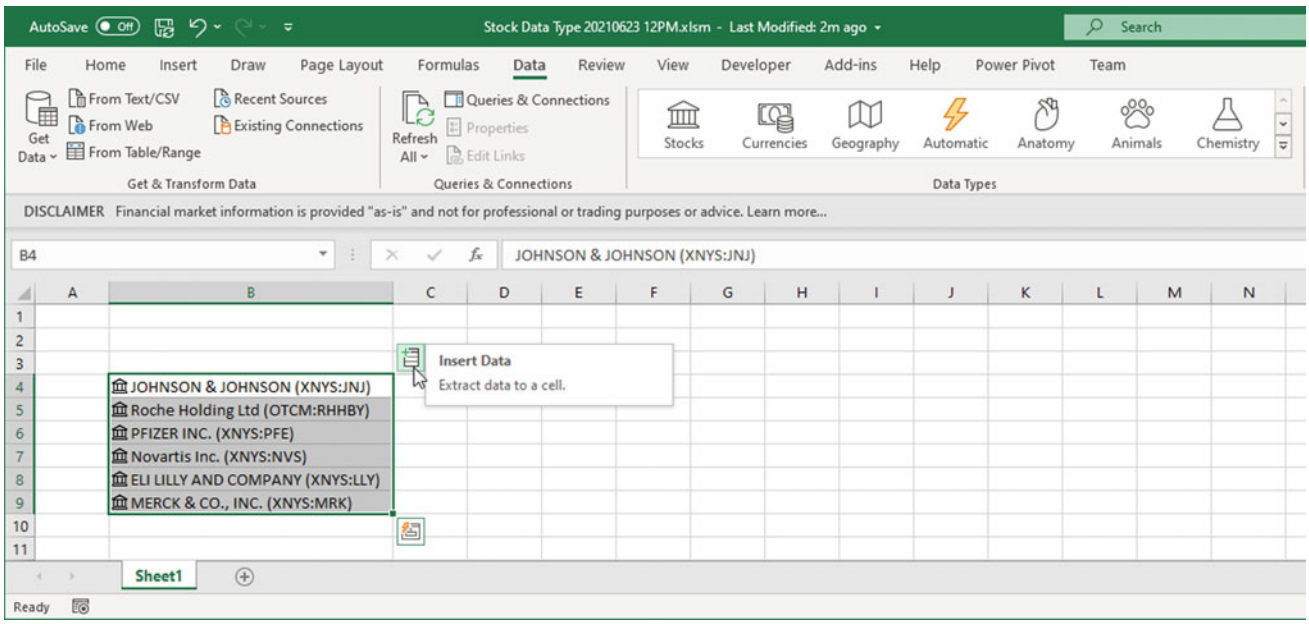
2.9.2.1.1 Stock

The below steps demonstrate the retrieval of stock attributes.

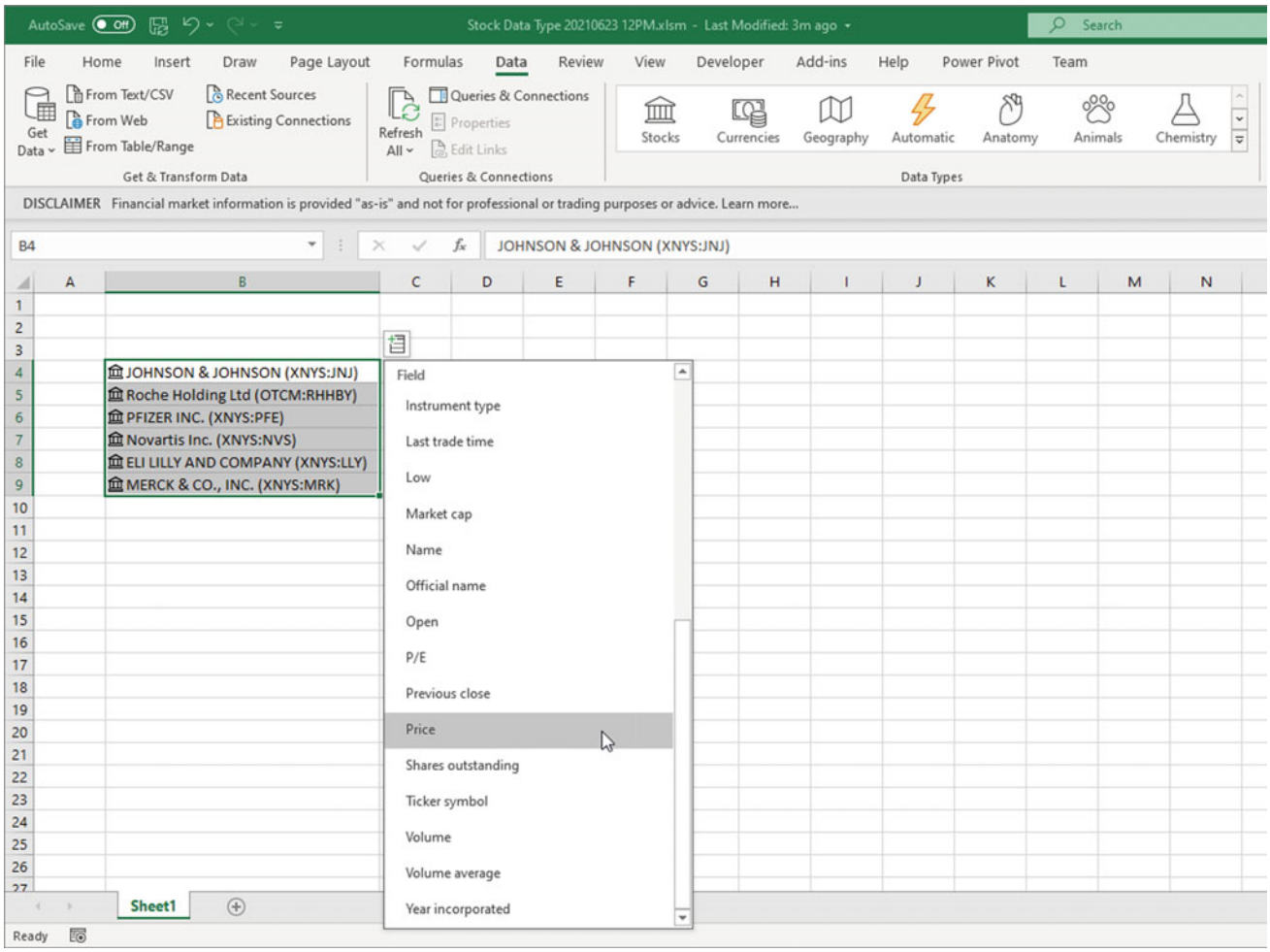
Step 1. Select tickers and then click on the *Stocks* button.



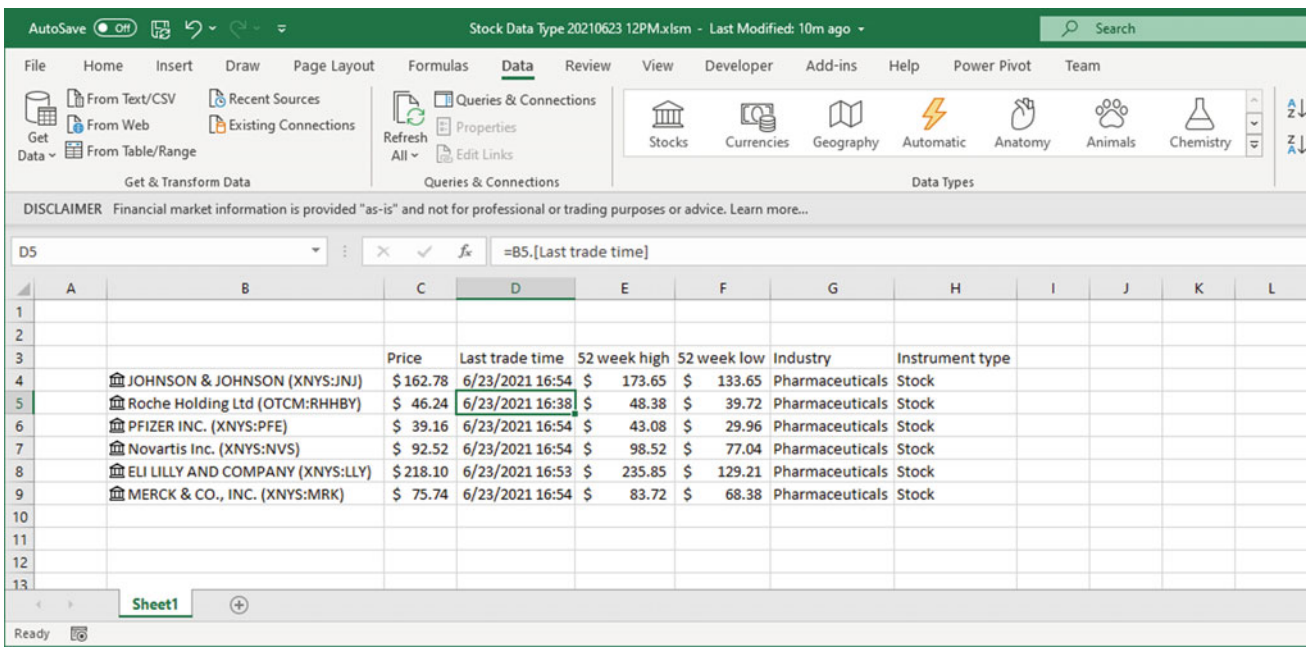
Step 2. Click on the *Insert Data* icon to add ticker attributes.



Step 3. Select the attributes of interest from the list.

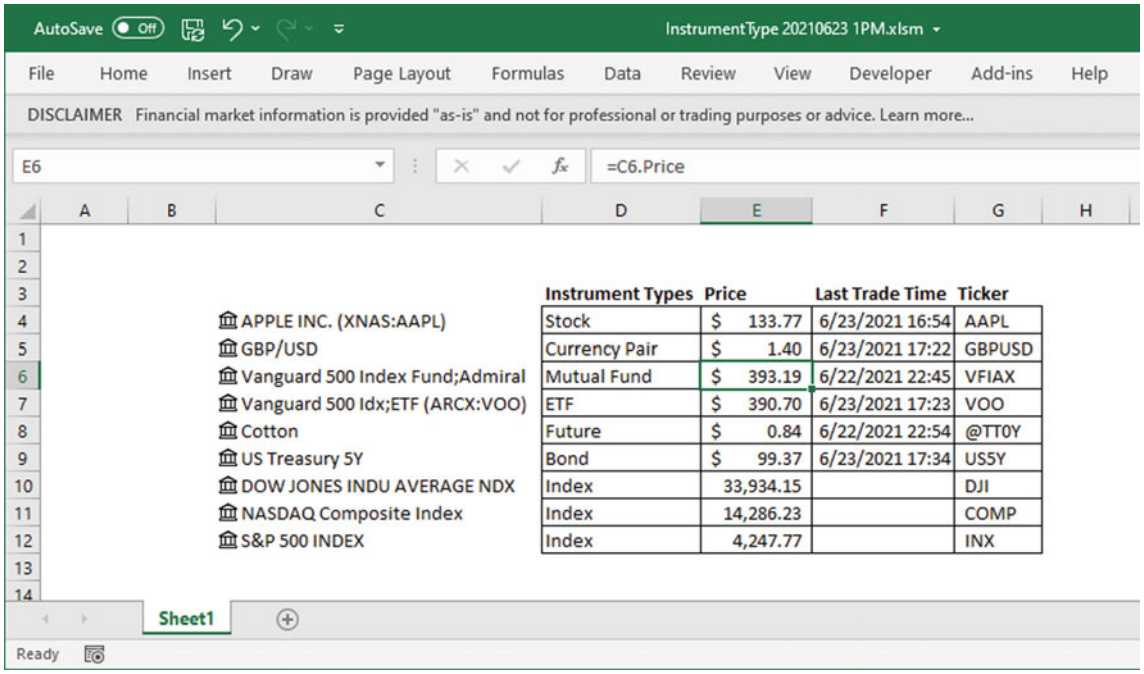


The below shows some of the attributes available for the *Stock* data type.



2.9.2.1.2 Instrument Types

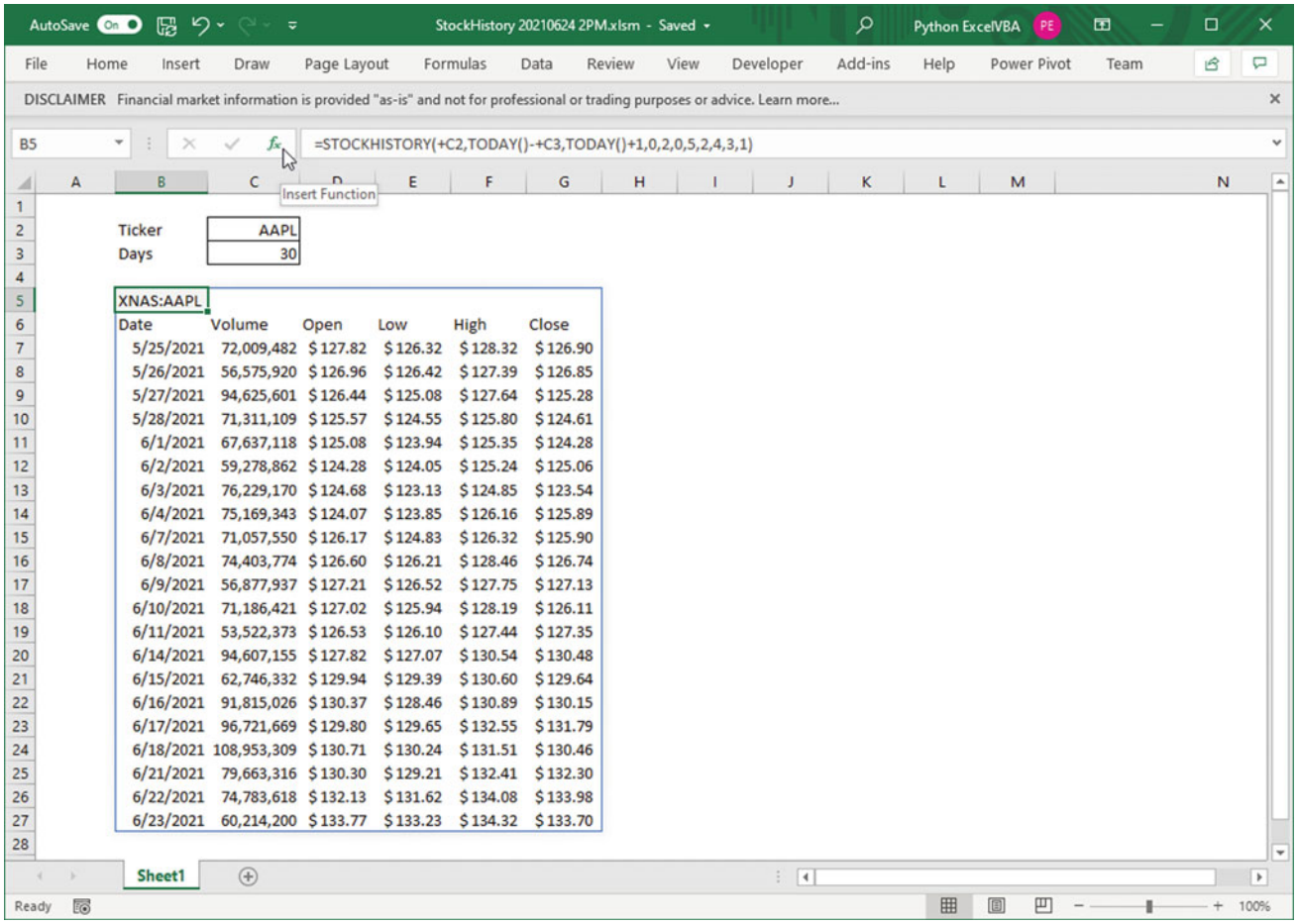
Below are the types of *Instrument Types* available for the *Stocks* data type.



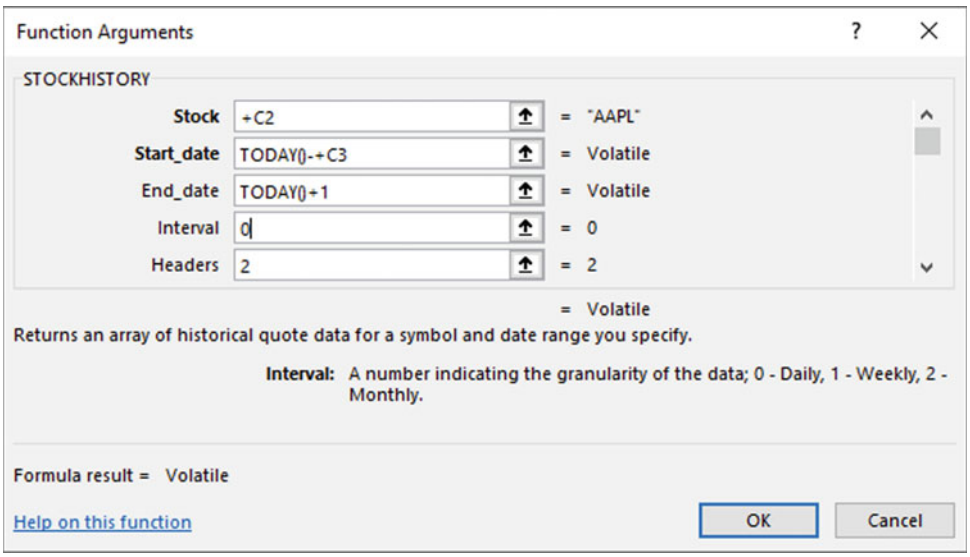
	Instrument Types	Price	Last Trade Time	Ticker
APPLE INC. (XNAS:AAPL)	Stock	$ 133.77	6/23/2021 16:54	AAPL
GBP/USD	Currency Pair	$ 1.40	6/23/2021 17:22	GBPUSD
Vanguard 500 Index Fund;Admiral	Mutual Fund	$ 393.19	6/22/2021 22:45	VFIAX
Vanguard 500 Idx;ETF (ARCX:VOO)	ETF	$ 390.70	6/23/2021 17:23	VOO
Cotton	Future	$ 0.84	6/22/2021 22:54	@TT0Y
US Treasury 5Y	Bond	$ 99.37	6/23/2021 17:34	US5Y
DOW JONES INDU AVERAGE NDX	Index	33,934.15		DJI
NASDAQ Composite Index	Index	14,286.23		COMP
S&P 500 INDEX	Index	4,247.77		INX

2.9.3 STOCKHISTORY Function

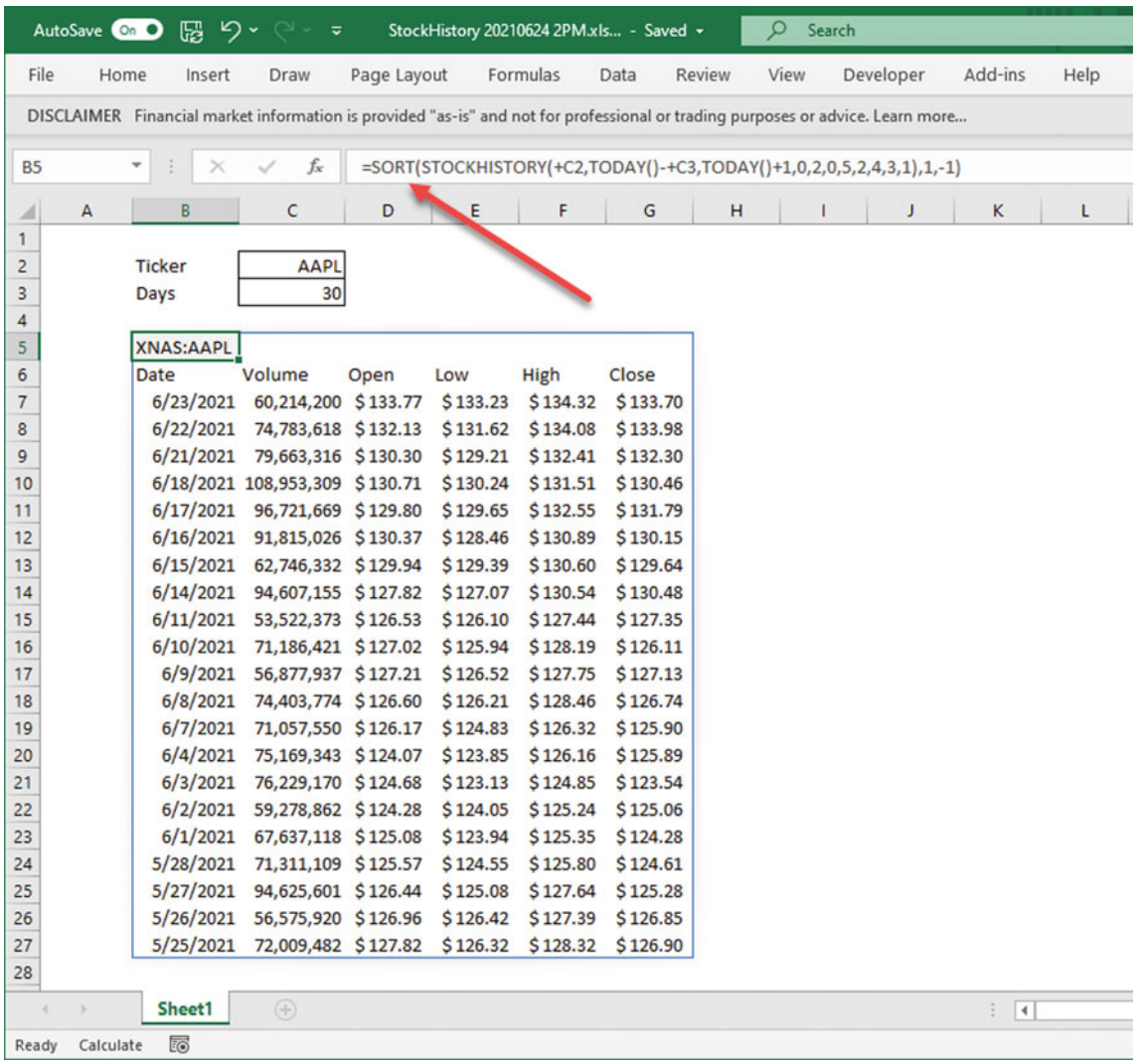
The *Stocks* data type returns only the current price of an instrument. Use the *STOCKHISTORY* function to return a range of prices for an instrument. Historical data is returned as a dynamic array. This is indicated by the blue border around the historical data.



To know more about the *STOCKHISTORY* function, click on the *Insert Function* icon to get the *Function Arguments* dialog box.



By default, the historical data shown by the *STOCKHISTORY* function is shown in date ascending order. Often it is shown in date descending order. To accomplish this, use the *SORT* function to show the historical data in date descending order.



2.10 Summary

In this chapter, we have discussed Excel's marco reader and Excel's Visual Basic Editor. We looked at how to run an Excel macro and discussed how to add macro code to a workbook. We discussed the macro button and subprocedures. We also looked at the message box and programming help, and finally we discussed features only found in Excel 365. In this section, we discussed dynamic arrays, rich data types, and STOCKHISTORY function.

References

https://www.ablebits.com/office-addins-blog/2020/07/08/excel-dynamic-arrays-functions-formulas/
https://exceljet.net/dynamic-array-formulas-in-excel
https://support.microsoft.com/en-us/office/dynamic-array-formulas-and-spilled-array-behavior-205c6b06-03ba-4151-89a1-87a7eb36e531
https://exceljet.net/formula/filter-text-contains
https://www.howtoexcel.org/general/data-types/
https://theexcelclub.com/rich-data-types-in-excel/
https://sfmagazine.com/post-entry/september-2020-excel-historical-weather-data-arrives-in-excel/
https://www.wolfram.com/microsoft-integration/excel/#datatype-list



Introduction to VBA Programming

3

3.1 Introduction

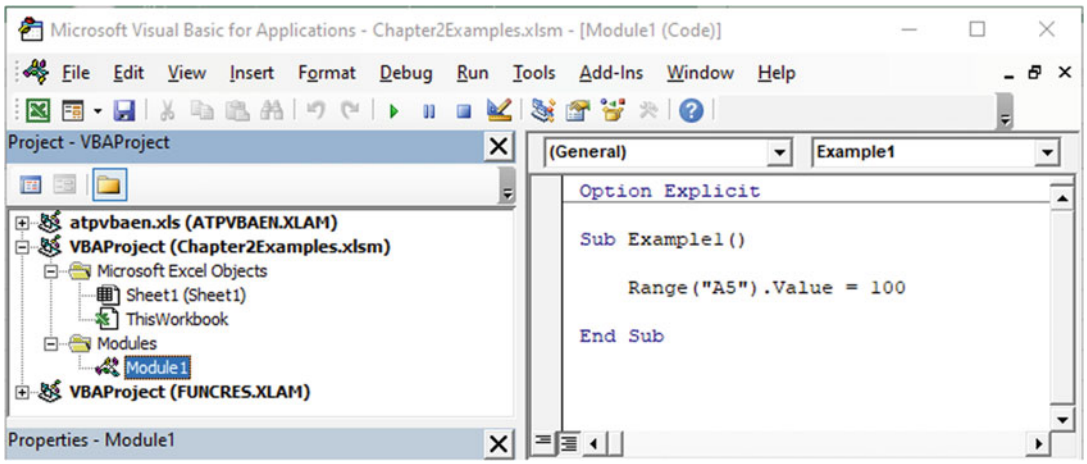
In the previous chapter, we mentioned that VBA was Excel's programming language. It turns out that VBA is the programming language for all Microsoft Office applications. In this chapter, we will study VBA and specific Excel VBA issues.

This chapter is broken down into the following sections. Section 3.2 discusses Excel's object model, Sect. 3.3 discusses the Intellisense menu, and Sect. 3.4 discusses the object browser. In Sect. 3.5, we look at variables, and in Sect. 3.6 we talk about option explicit. Section 3.7 discusses object variables, and Sect. 3.8 talks about functions. In Sect. 3.9, we add a function description, and in Sect. 3.10 we specify a function category. Section 3.11 discusses conditional programming with the IF statement, and Sect. 3.12 discusses for a loop. Section 3.13 discusses the while loop, and Sect. 3.14 discusses arrays. In Sect. 3.15, we talk about option base 1, and in Sect. 3.16 we discuss collections. Finally, in Sect. 3.17 we summarize the chapter.

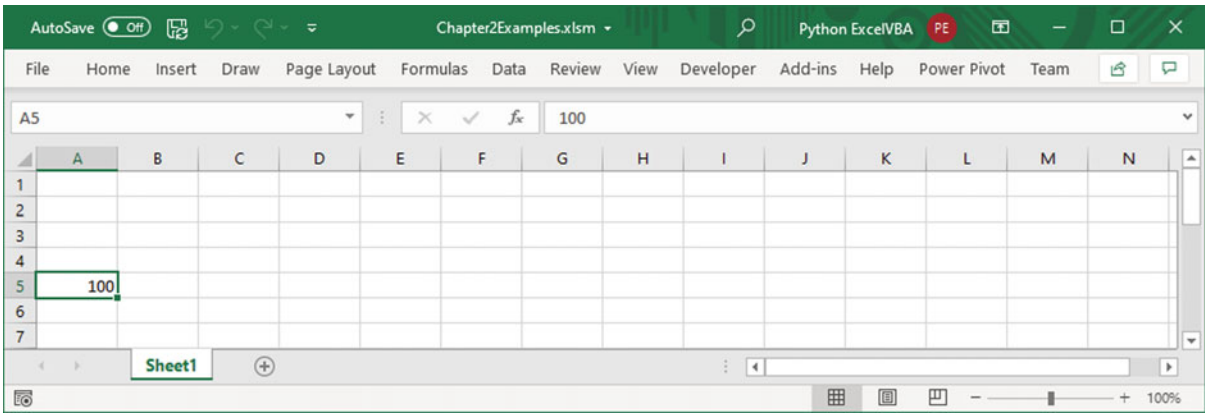
3.2 Excel's Object Model

There is one thing that is frequently done with an Excel VBA program; it sets a value to a cell or a range of cells. For example, suppose we are interested in setting the cell A5 in worksheet "Sheet1" to the value of 100. Below is a common way that a novice would program a VBA program to set the cell A5 to 100.

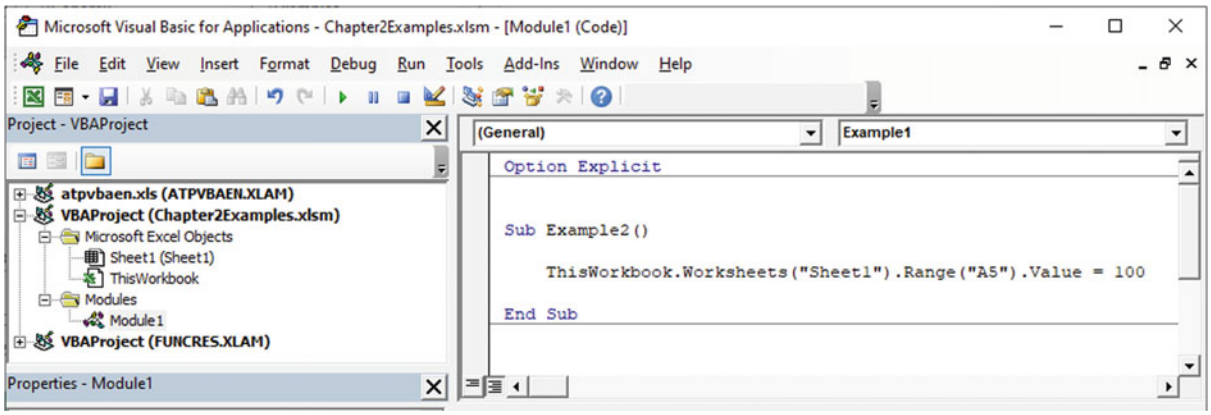
© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_3



The *range* command above is used to reference specific cells of a worksheet. So, if the worksheet "Sheet1" is the active worksheet, cell A5 of worksheet "Sheet1" will be populated with the value of 100. This is shown below.



"Sheet1" has the value of 100 and not cell A5 in the other worksheets of the workbook. But if we run the above macro when worksheet "Sheet2" is active, cell A5 in worksheet "Sheet2" will be populated with the value of 100. To solve this issue, experienced programmers will rewrite the above VBA procedure as shown below.

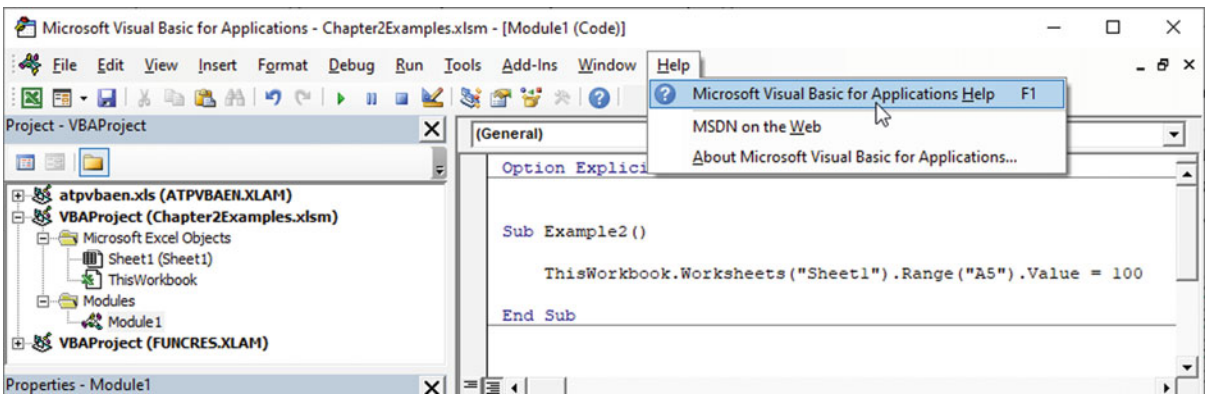


Notice that the VBA code line is longer in the procedure "Example2" than in the procedure "Example1." To understand why, we will need to look at Excel's object model. We can think of Excel's object as an upside-down tree. A lot of Excel VBA programming is basically traversing the tree. In VBA programming, moving from one level of a tree to another level is indicated by a period. The VBA code in the procedure "Example2" traverses Excel's object model through three levels.

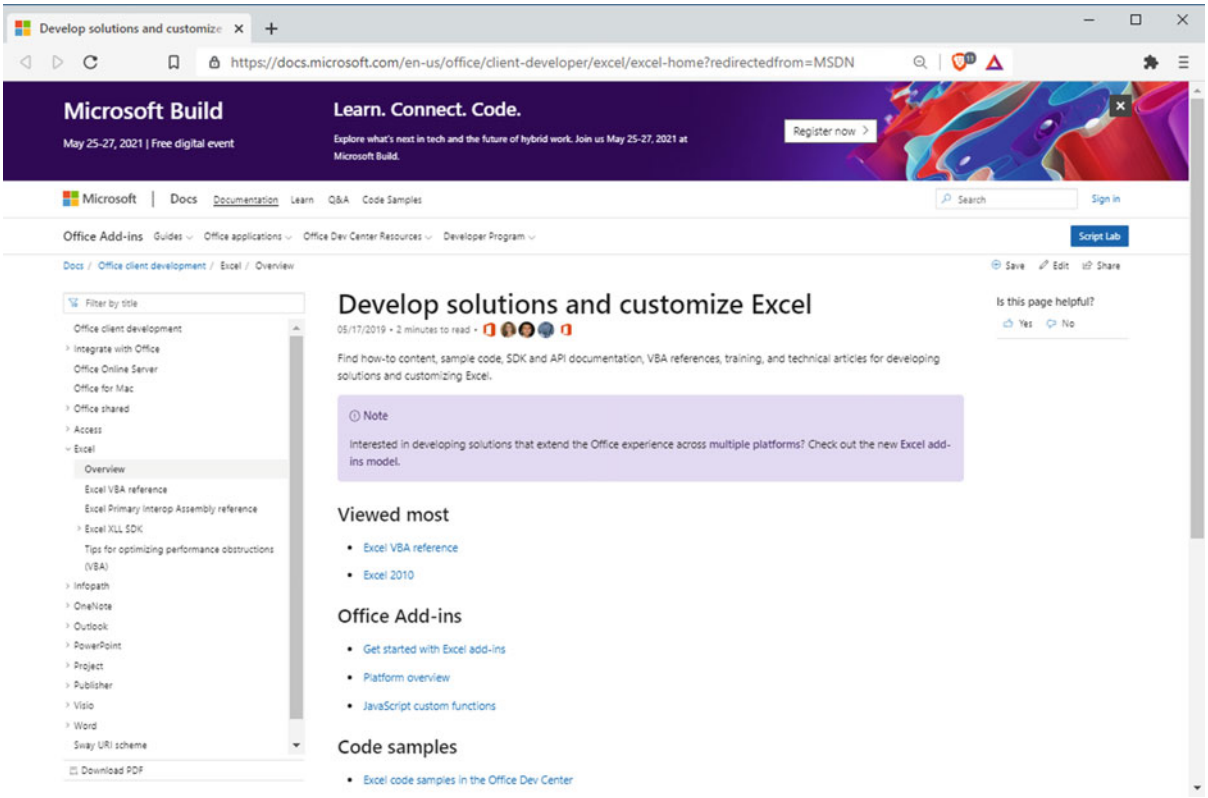
Among all Microsoft Office products, Excel has the most detailed object model. When we talk about object models, we are talking about concepts that a professional programmer would talk about. When we are talking about object models, there are three words that even a novice must know. Those three words are *objects*, *properties*, and *methods*. These words can take up chapters or even books to explain. A very crude but somewhat effective way to think about what these words mean is to think about English grammar. We can crudely equate *objects* as a noun. We can crudely equate *properties* as an adjective. We can crudely equate *methods* as an adverb. In Excel, some examples of objects are worksheets, workbooks, and charts. These objects have properties that describe them or have methods that act on them.

In the Excel object model, there is a parent and child relationship between objects. The topmost object is the Excel object. A frequently used object and a child of the Excel object is the *workbook* object. Another frequently used object and a child of the *workbook* object is the *worksheet*. Another frequently used object and a child of the *worksheet* object is the *range* object. If we look at the Excel object model, we will be able to see the relationship between the Excel object, the workbook object, the worksheet object, and the range object.

We can use the help in the VB Editor (VBE) to look at the Excel object model. To do this, we would need to choose Help → Microsoft Visual Basic for Application Help.

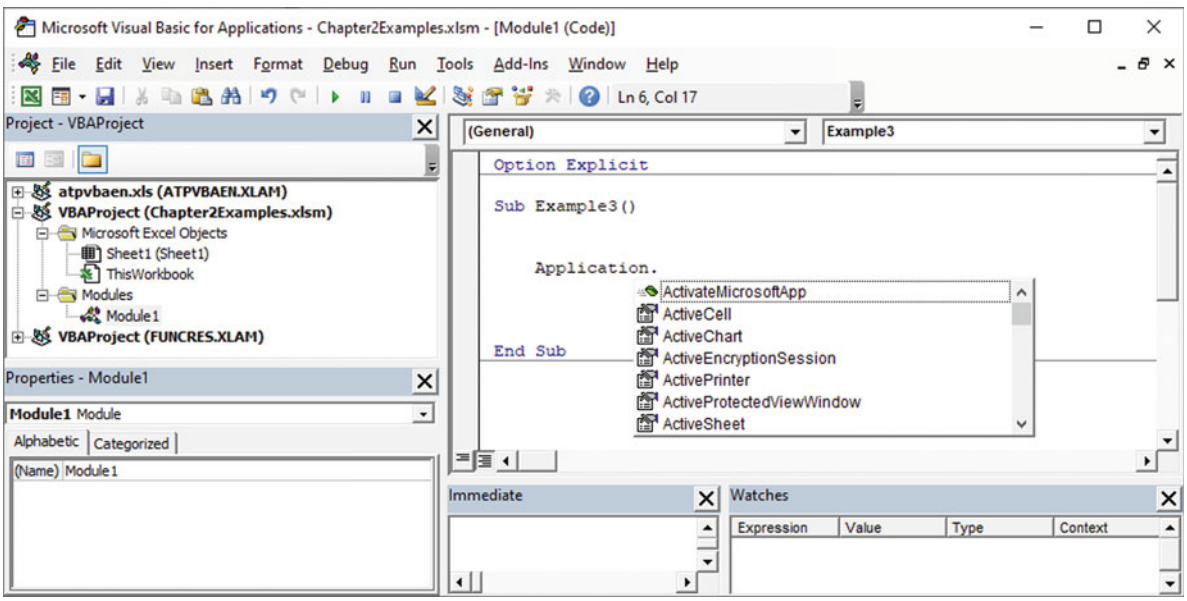


In Excel, there is no offline help. The online help is located at https://docs.microsoft.com/en-us/office/client-developer/excel/excel-home.



3.3 Intellisense Menu

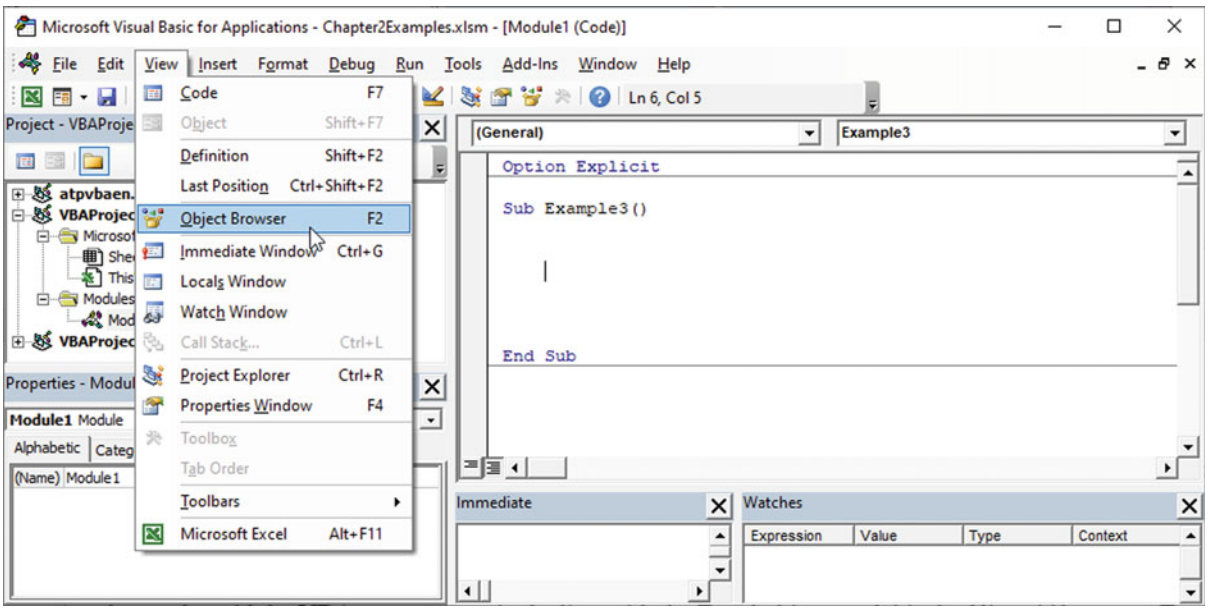
The Excel VBA programmer should always be thinking about the Excel object model. Because of the importance of the Excel object model, the VBE has tools to aid the VBA programmer in dealing with Excel's object model. The first tool is the *Intellisense* menu of the Visual Basic Editor. This feature will display for an object a list that contains information that would logically complete the statement at the current insertion point. For example, the below shows the list that would complete the *Application* object. This list contains the properties, methods, and child objects of the *Application* object.



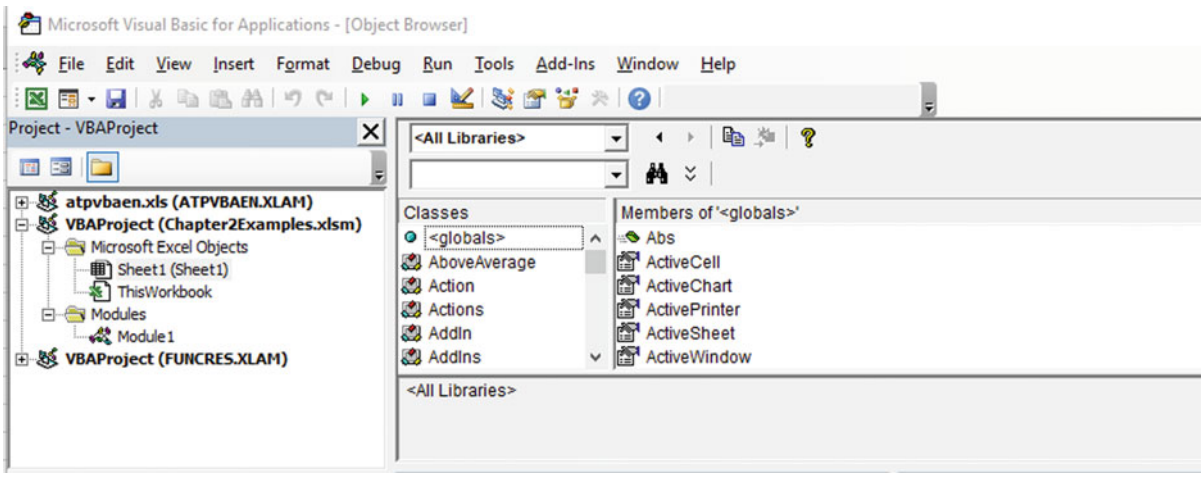
Intellisense is a great aid in helping the VBA programmer in dealing with the methods, properties, and child objects of each object.

3.4 Object Browser

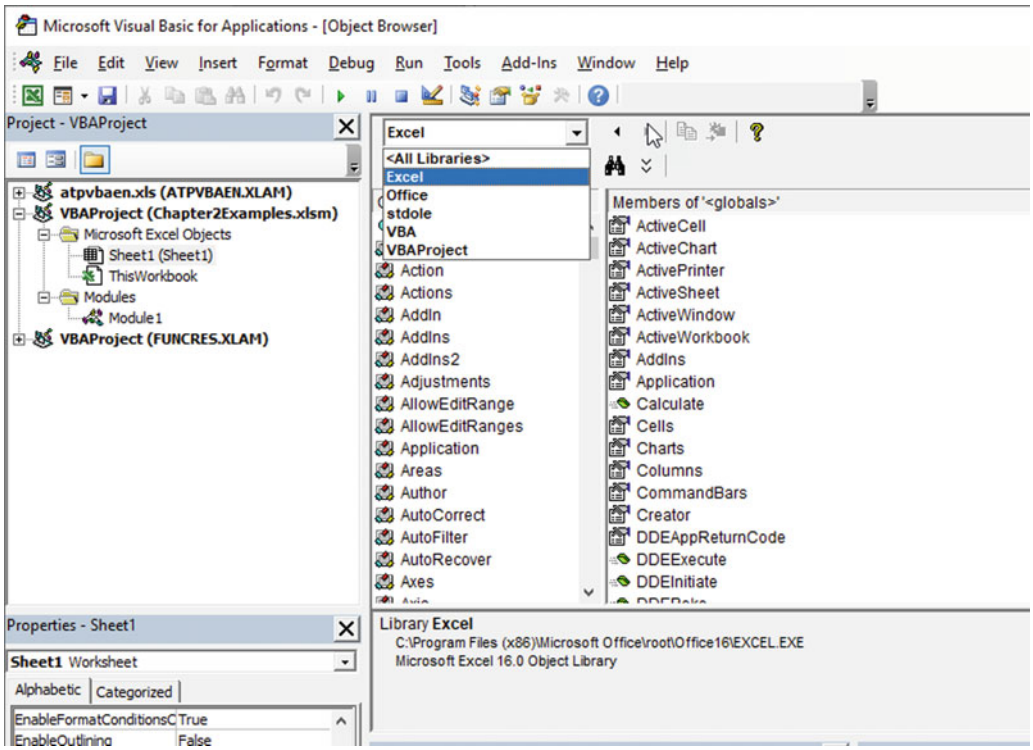
Another tool to aid the VBA programmer in dealing with the Excel object model is the *Object Browser*. To view the *Object Browser*, choose View → Object Browser. This is shown below.



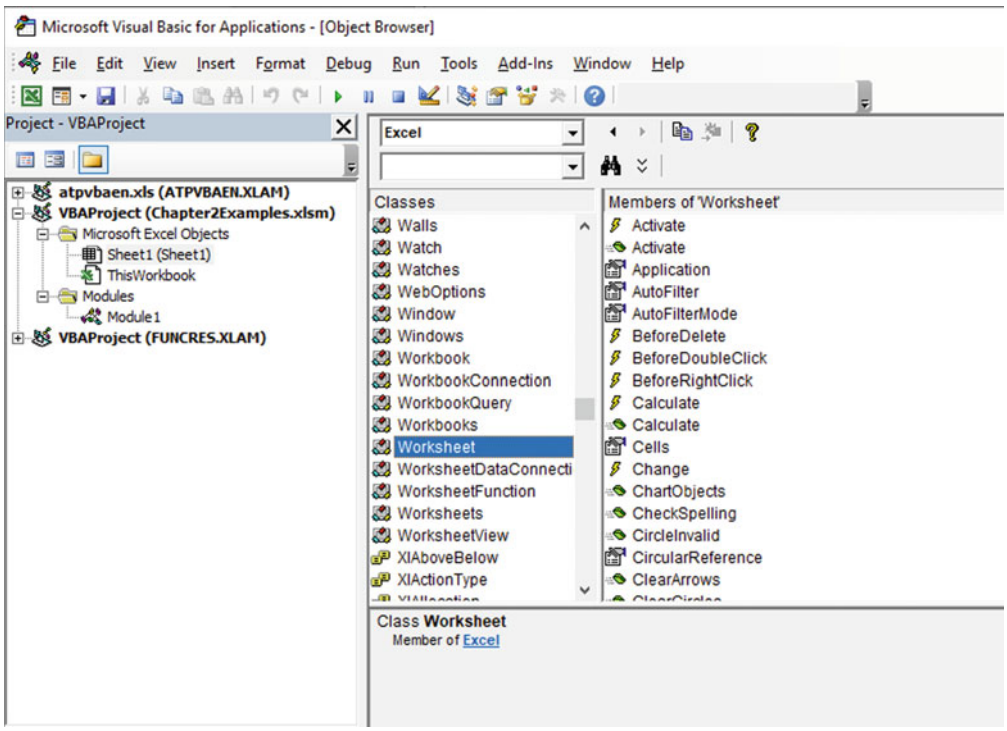
The default display for the *Object Browser* is shown below.



The below shows how to view the Excel object model from the *Object Browser*.

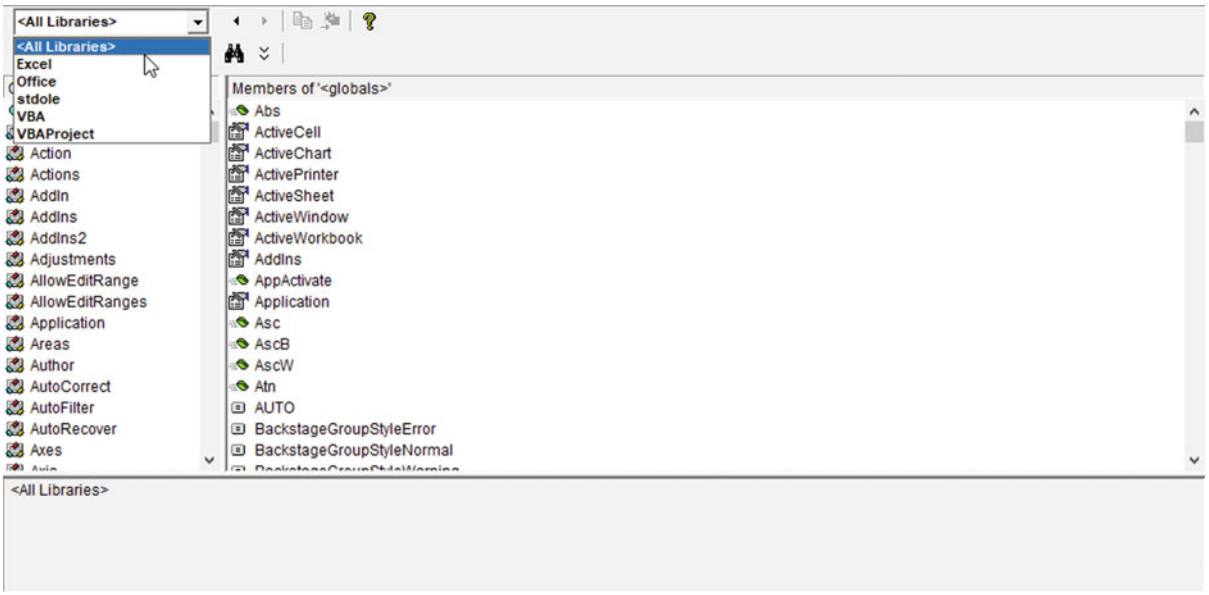


The below shows the objects, properties, and methods for the *Worksheet* object.

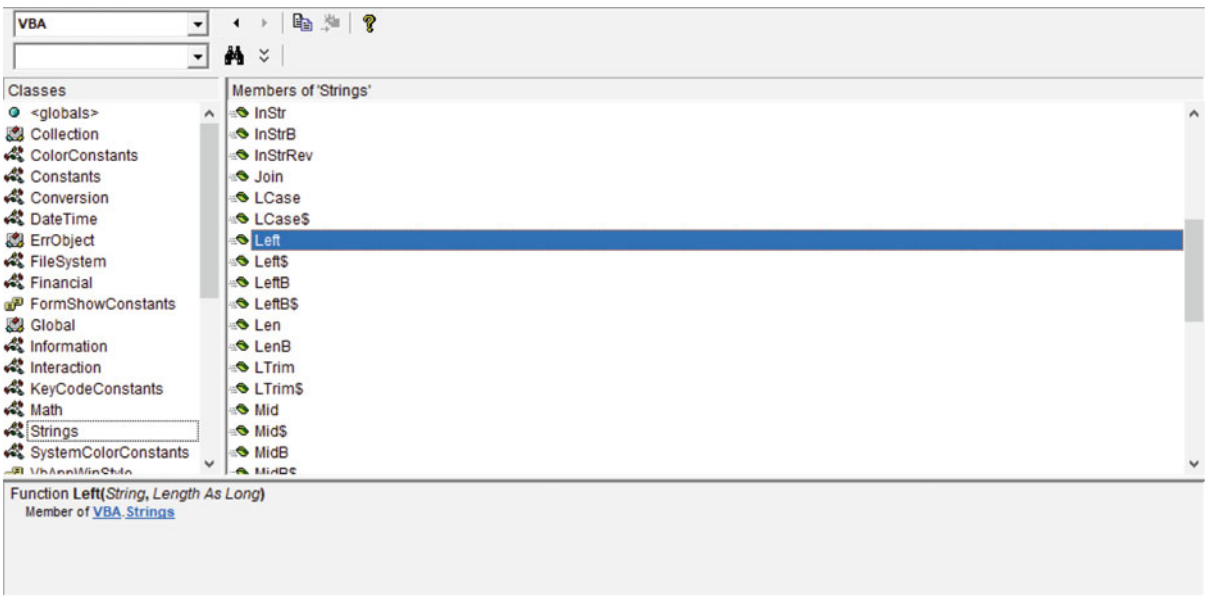


In the *Object Browser* above, the object *worksheet* is chosen on the left side of the object browser, and on the right side, all the properties, methods, and child objects of the *worksheet* object are shown.

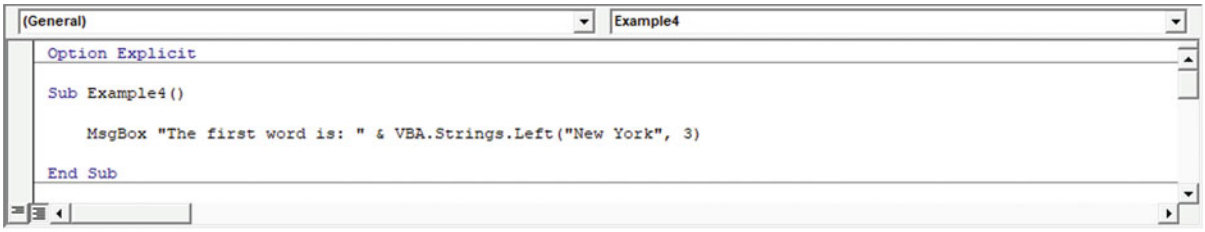
It is important to note that the Excel object model is not the only object model that the VBE handles. This issue was alluded to above. The default display for the *Object Browser* shows " < All Libraries > ". This suggests that other object models were available. Above, we also saw the following list in the object browser:



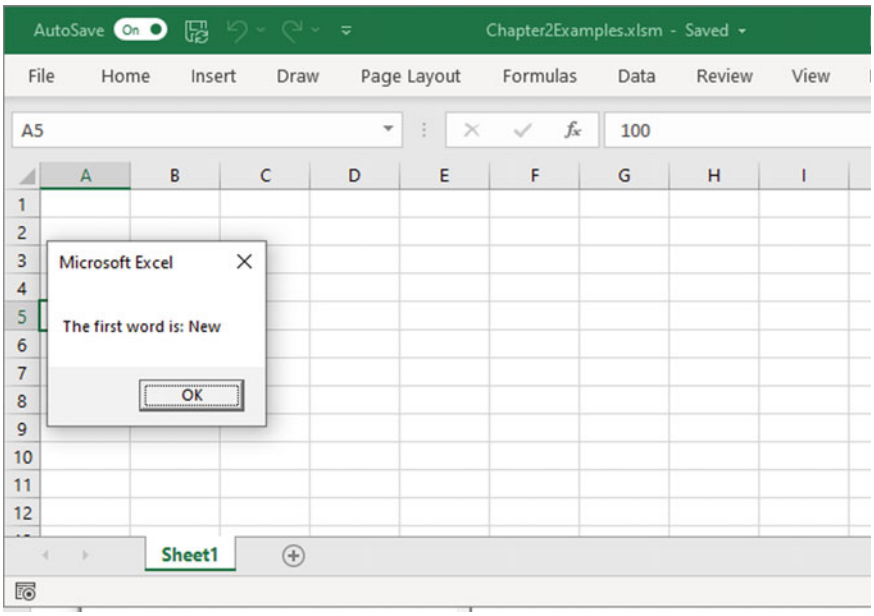
The list above indicates the object models used by the Visual Basic Editor. Of all the object models shown above, the VBA object model is used most after the Excel object model. The below shows the VBA object model in the *object browser*.



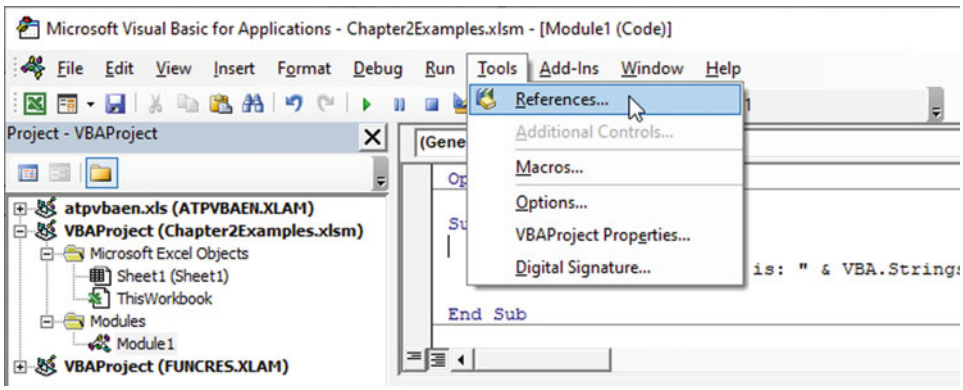
The main reason that an Excel VBA programmer uses the VBA object model is that the VBA object model provides a lot of useful functions. Professional programmers will say that the functions of an object model are properties of an object model. For example, for the *Left* function shown above, we can say that the *Left* function is a property of the VBA object model. The below shows an example of using the property *Left* of the VBA object model.



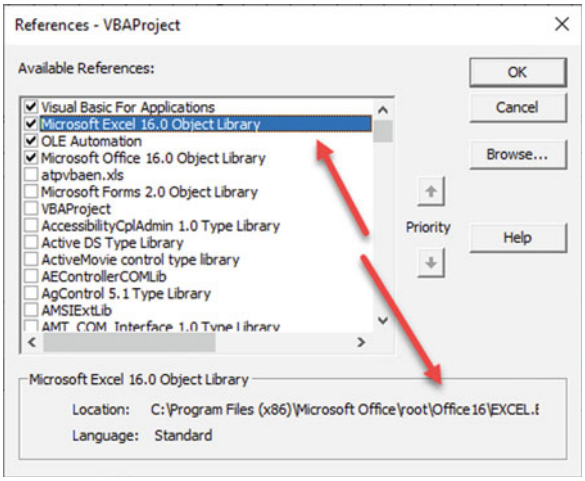
The below shows the result of executing the "Example4" macro.



Many times, an Excel VBA programmer will write macros that use both Microsoft Excel and Microsoft Access. To do this, we would need to set up the VBE so that it can also use Access's object model. To do this, we would first have to choose Tools → Reference in the VBE. This is shown below.

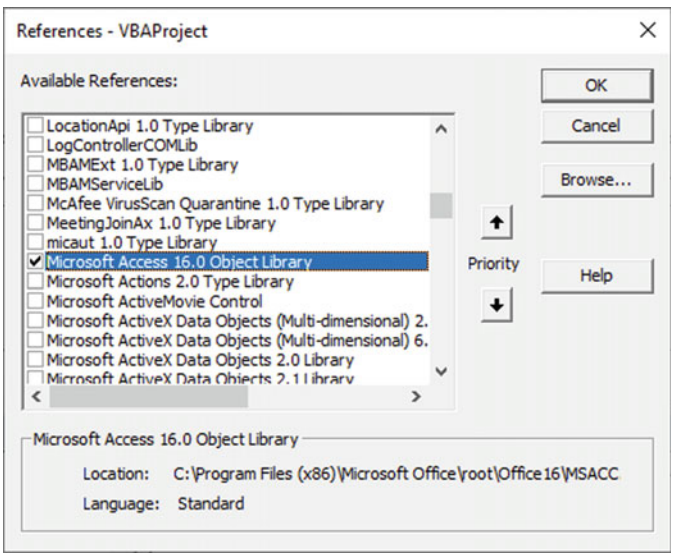


The resulting *Reference* dialog box is shown below.

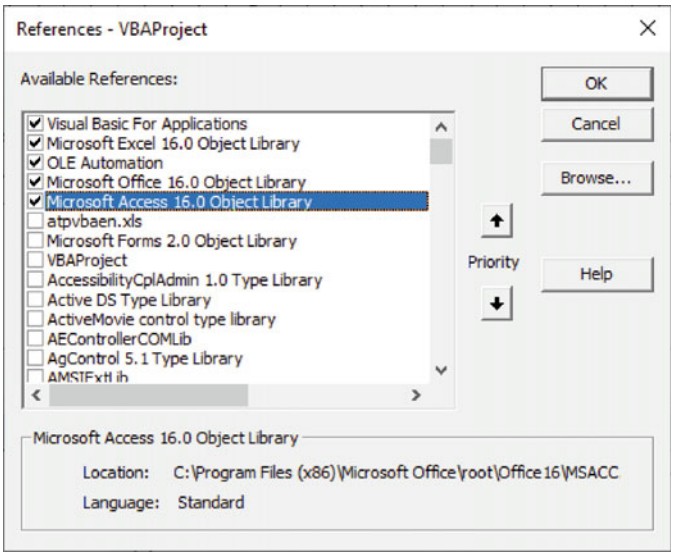


In the above *References* dialog box, the Excel object model is selected. The bottom of the dialog box shows the location of the file that contains Excel's object model. The file that contains an object model is called a *type library*.

To program Microsoft Access while programming Excel, we will need to find the type library for Microsoft Access. The below shows the Microsoft Access object model being selected.



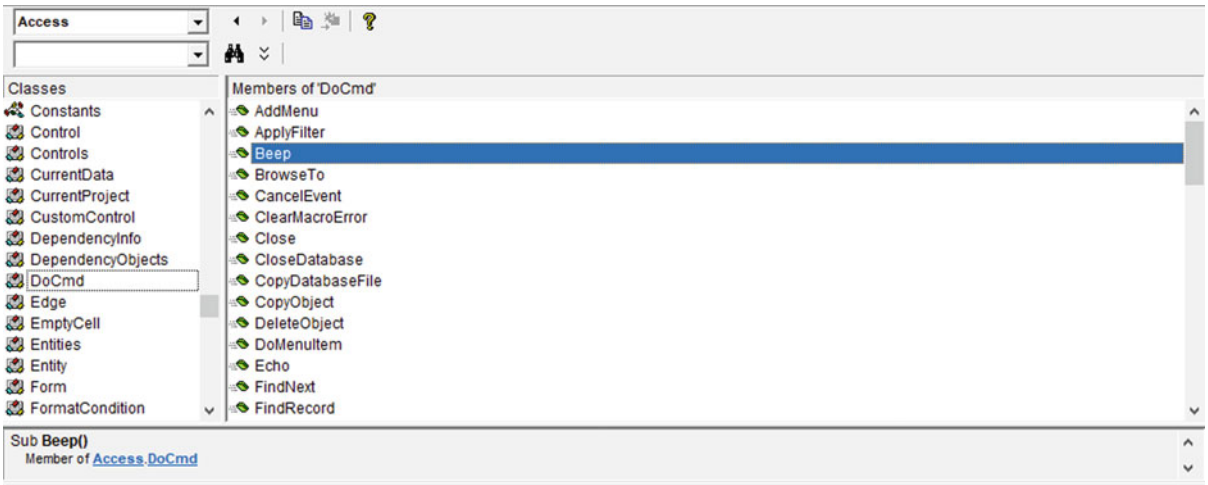
If we press the *OK* button and go back to the *References* dialog box, we will see the following.



Notice that the *References* dialog box now shows all the selected object libraries on the top. We now should be able to see Microsoft Access's object model in the object browser. The below shows that Microsoft Access's object model is included in the object browser's list.



The below shows Microsoft Access's object model in the object browser.



The Excel object model does not have a method to make the PC make a beep sound. Fortunately, it turns out that the Access object does have a method to make the PC make a beep sound. The below is a macro that will make the PC make a beep sound. The *Access* keyword indicates that we are using the Access object model. The keyword *Docmd* is a child object of the *Access* object. The keyword *Beep* is a method of the *DoCmd* object.

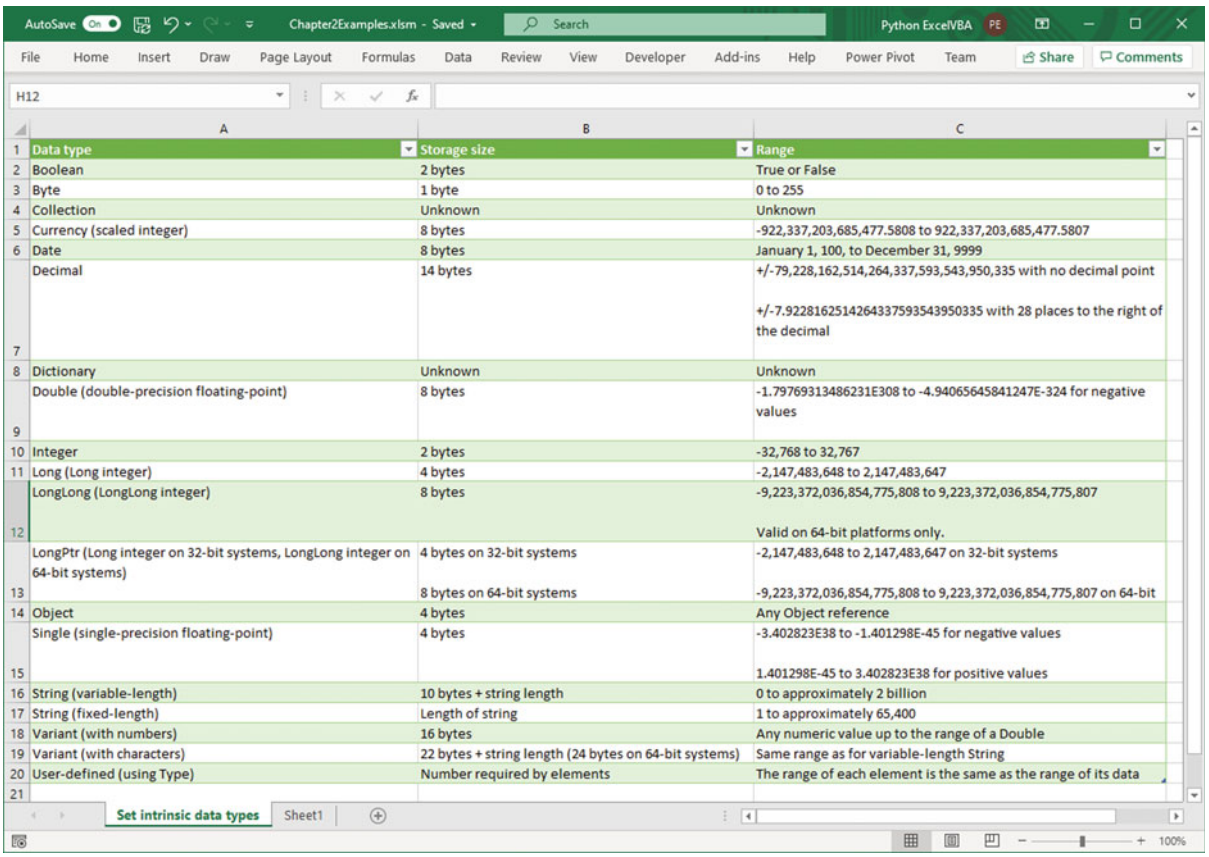


It turns out that in the VBA object model, there is also a *beep* method. The below shows a macro using the VBA object model to make the PC make a beep sound.



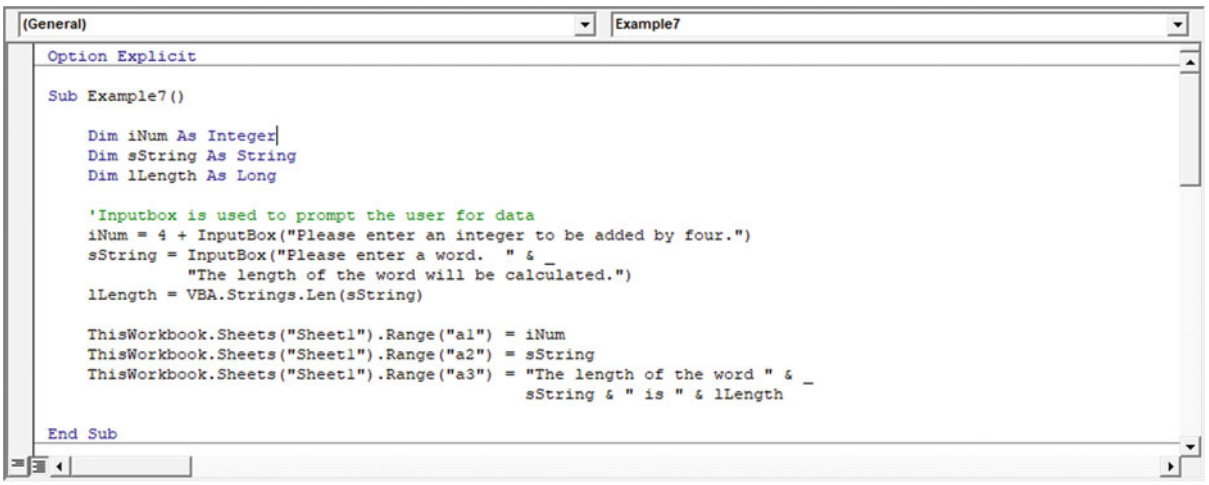
3.5 Variables

In VBA, programming *variables* are used to store and manipulate data during macro execution. When dealing with data, it is often useful when processing data to only deal with a specific type of data. In VBA, it is possible to define a specific type for specific variables. Below is a summary of the different types available in VBA. This list was obtained from the URL https://docs.microsoft.com/en-us/office/vba/language/reference/user-interface-help/data-type-summary.



Data type	Storage size	Range
Boolean	2 bytes	True or False
Byte	1 byte	0 to 255
Collection	Unknown	Unknown
Currency (scaled integer)	8 bytes	-922,337,203,685,477.5808 to 922,337,203,685,477.5807
Date	8 bytes	January 1, 100, to December 31, 9999
Decimal	14 bytes	+/-79,228,162,514,264,337,593,543,950,335 with no decimal point +/-7.9228162514264337593543950335 with 28 places to the right of the decimal
Dictionary	Unknown	Unknown
Double (double-precision floating-point)	8 bytes	-1.79769313486231E308 to -4.94065645841247E-324 for negative values
Integer	2 bytes	-32,768 to 32,767
Long (Long integer)	4 bytes	-2,147,483,648 to 2,147,483,647
LongLong (LongLong integer)	8 bytes	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Valid on 64-bit platforms only.
LongPtr (Long integer on 32-bit systems, LongLong integer on 64-bit systems)	4 bytes on 32-bit systems 8 bytes on 64-bit systems	-2,147,483,648 to 2,147,483,647 on 32-bit systems -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 on 64-bit
Object	4 bytes	Any Object reference
Single (single-precision floating-point)	4 bytes	-3.402823E38 to -1.401298E-45 for negative values 1.401298E-45 to 3.402823E38 for positive values
String (variable-length)	10 bytes + string length	0 to approximately 2 billion
String (fixed-length)	Length of string	1 to approximately 65,400
Variant (with numbers)	16 bytes	Any numeric value up to the range of a Double
Variant (with characters)	22 bytes + string length (24 bytes on 64-bit systems)	Same range as for variable-length String
User-defined (using Type)	Number required by elements	The range of each element is the same as the range of its data

The below shows how to define and use variables in VBA.



```
(General)                                   Example7

Option Explicit

Sub Example7()

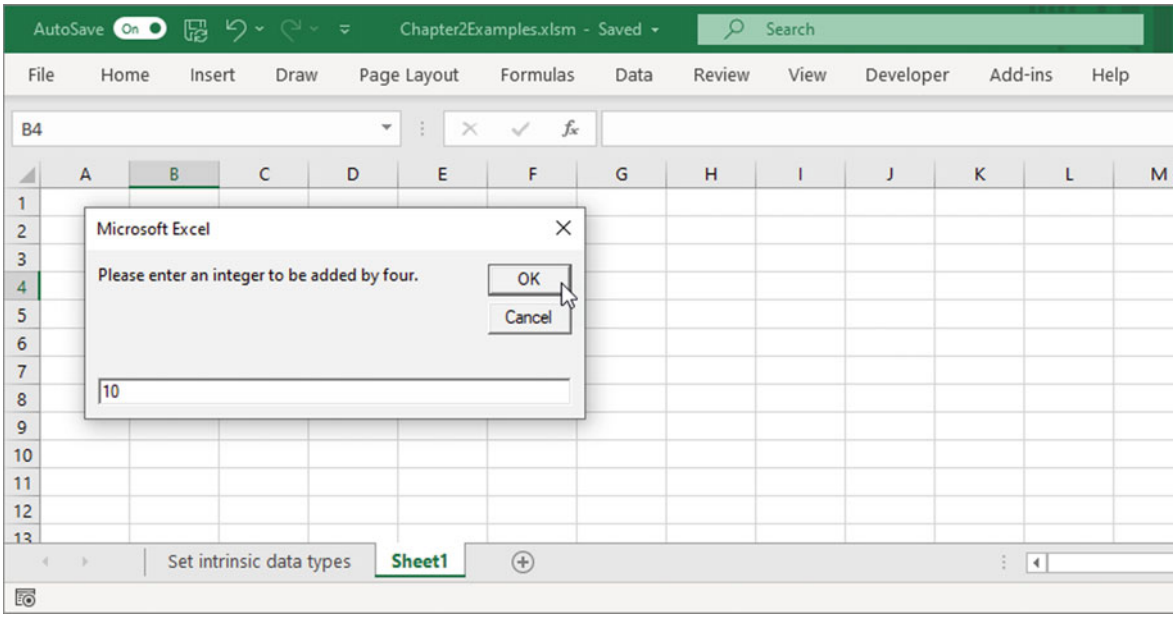
    Dim iNum As Integer
    Dim sString As String
    Dim lLength As Long

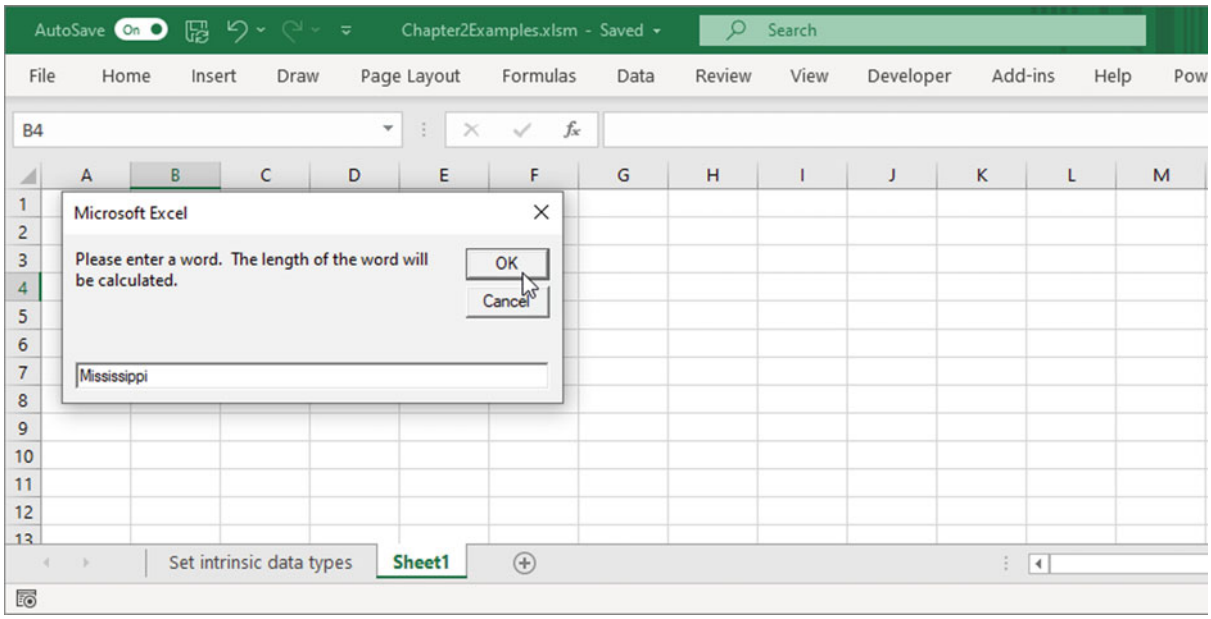
    'Inputbox is used to prompt the user for data
    iNum = 4 + InputBox("Please enter an integer to be added by four.")
    sString = InputBox("Please enter a word.  " & _
              "The length of the word will be calculated.")
    lLength = VBA.Strings.Len(sString)

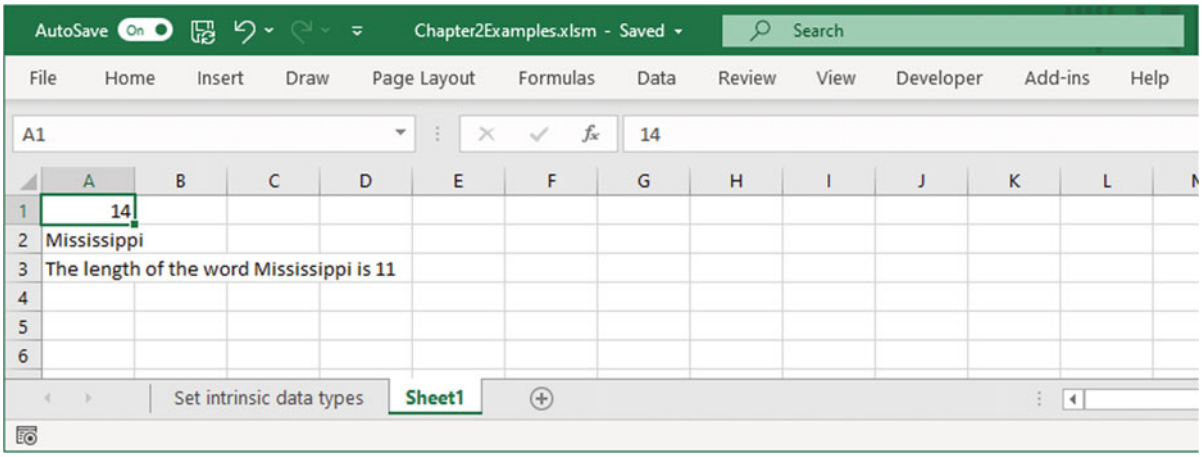
    ThisWorkbook.Sheets("Sheet1").Range("a1") = iNum
    ThisWorkbook.Sheets("Sheet1").Range("a2") = sString
    ThisWorkbook.Sheets("Sheet1").Range("a3") = "The length of the word " & _
                                                sString & " is " & lLength

End Sub
```

Running the above will result in the following.



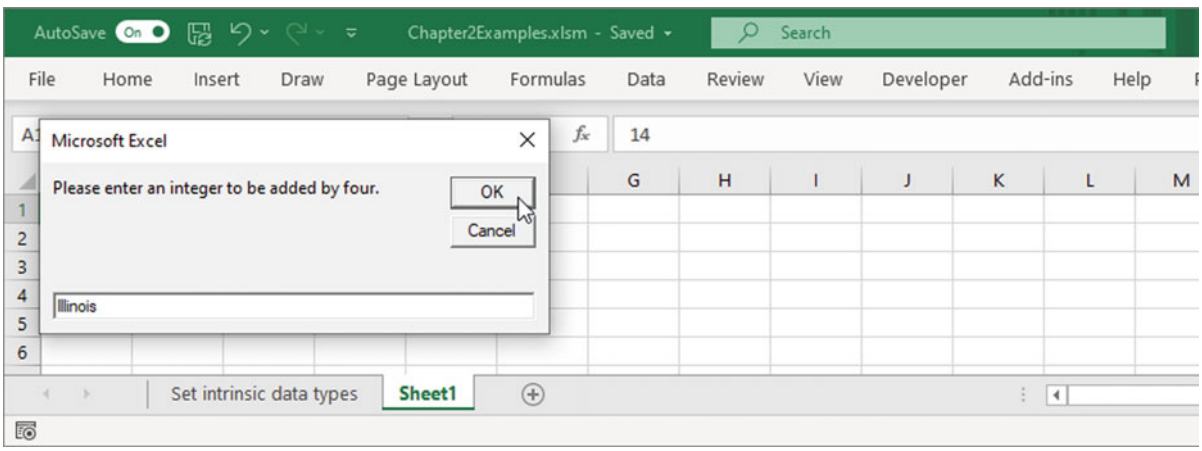


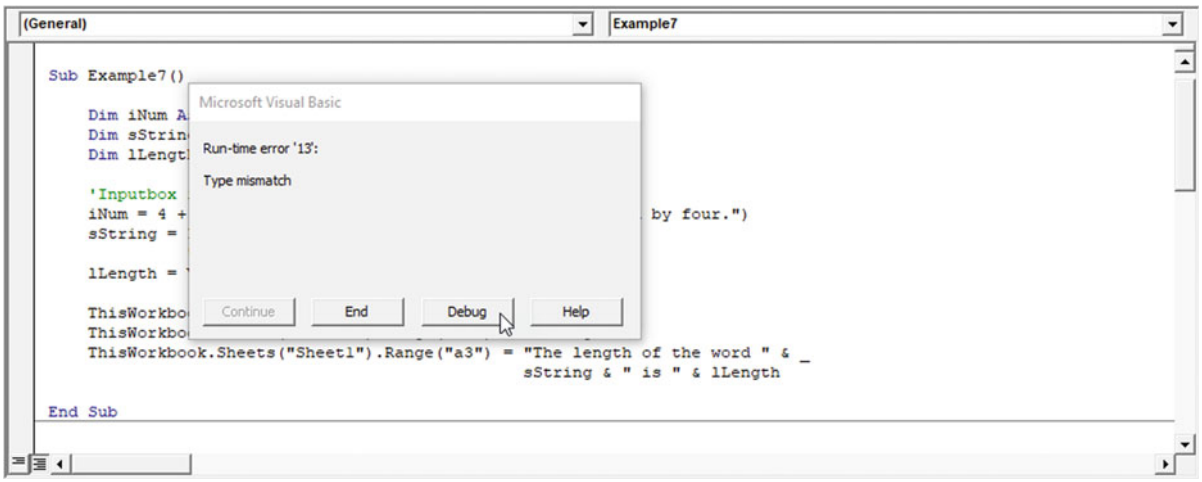


There are a lot of things happening in the macro "Example7":

1. In this macro, we used the keyword *Dim* to define one variable to hold an *integer* data type and one variable to hold a *string* data type, and one variable to hold a long data type.
2. In this macro, we used the keyword *inputbox* to prompt the user for data.
3. We used the single apostrophe to tell the VBE to ignore everything to the right. Programmers use the single apostrophe to comment about the VBA code.
4. Double quotes are used to hold string values.
5. "&" is used to put together two strings.
6. The character "_" is used to indicate that the VBA command line is continued in the next line.
7. We calculated the data we received and put the calculated result in ranges A1–A3.

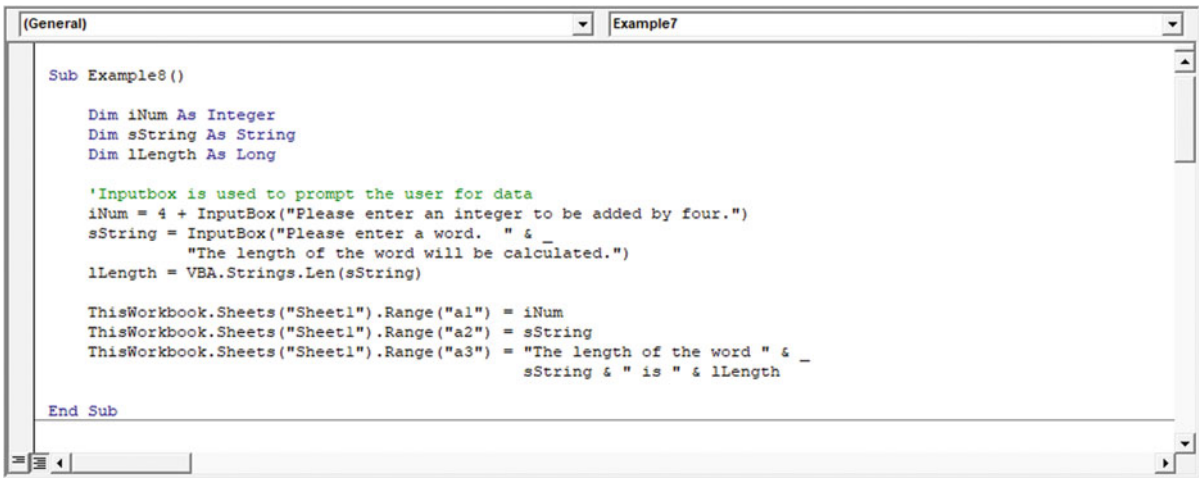
We will now show why data-typing a variable is important. The first input box requested an integer. The number four will be added to the inputted number. Suppose that by accident, we enter a word instead. The below shows what happens when we do this.





The above shows that the VBE will complain about having the wrong data type for the variable "iNum." There are VBA techniques to handle this type of situation so the user will not have to see the above VBA error message.

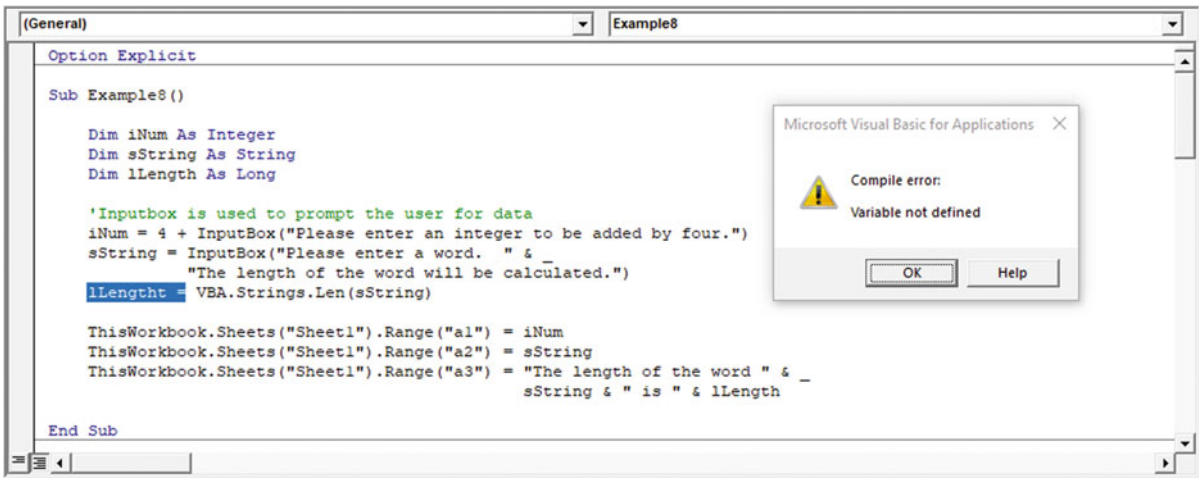
From the data type list, it is important to note that the *variant* data type is a data type that can be any type. The type of a variable is determined during run time (when the macro is running). The macro "Example7" can be rewritten as follows.



Experienced VBA programmers prefer macro "Example7" over macro "Example8."

3.6 Option Explicit

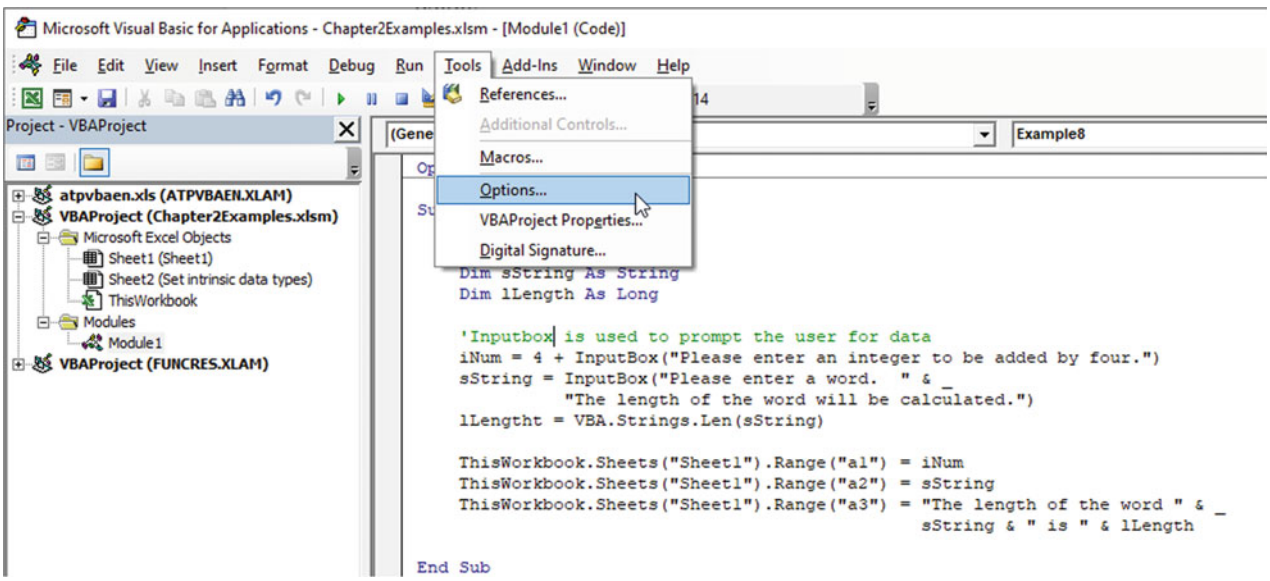
In VBA programming, it is actually possible to use variables without first being defined, but good programming practice dictates that every variable should be defined. Excel VBA has the two keywords *Option Explicit* to indicate that every variable must be declared. The below shows what happens when *Option Explicit* is used and when a variable is not defined when trying to run a macro.



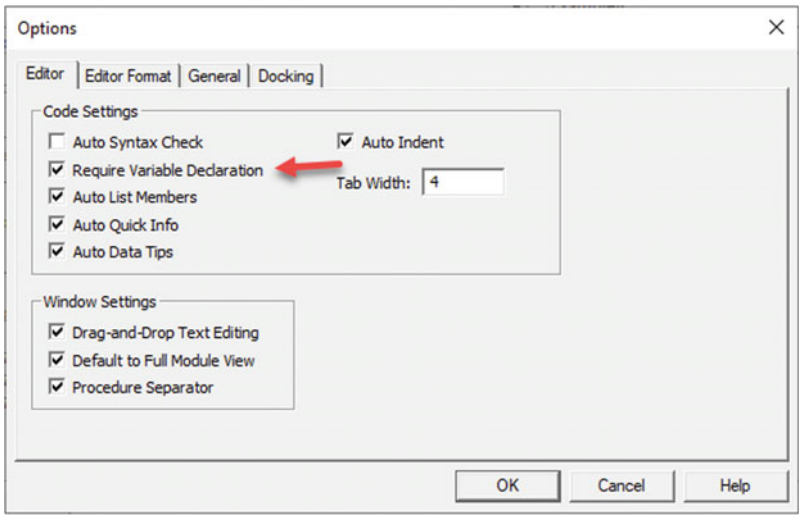
Notice that using the *Option Explicit* keywords results in the following:

1. The variable that is not defined is highlighted.
2. A message indicating that a variable is not defined is displayed.

When a new module is inserted into a project, the keywords *Option Explicit* by default are not inserted into the new module. This can cause problems, especially in bigger macros. The VBE has a feature where the keywords *Option Explicit* are automatically included in a new module. To do this, choose Tools → Options. This is shown below.



This will result in the following *Options* dialog box.



Choose the *Required Variable Declaration* option in the *Editor* tab of the *Options* dialog box to set it so the keywords *Options Explicit* are included with every new module. It is important to note that by default the *Required Variable Declaration* option is not selected.

3.7 Object Variables

The data type *Object* is used to define a variable to "point" to objects in the Excel object model. Like the data type *Variant*, the specific object data type for the data type *Object* is determined at run time. The macro below will set the cell A5 in the worksheet "Sheet2" to the value "VBA Programming." This macro is not sensitive to which worksheet is active.

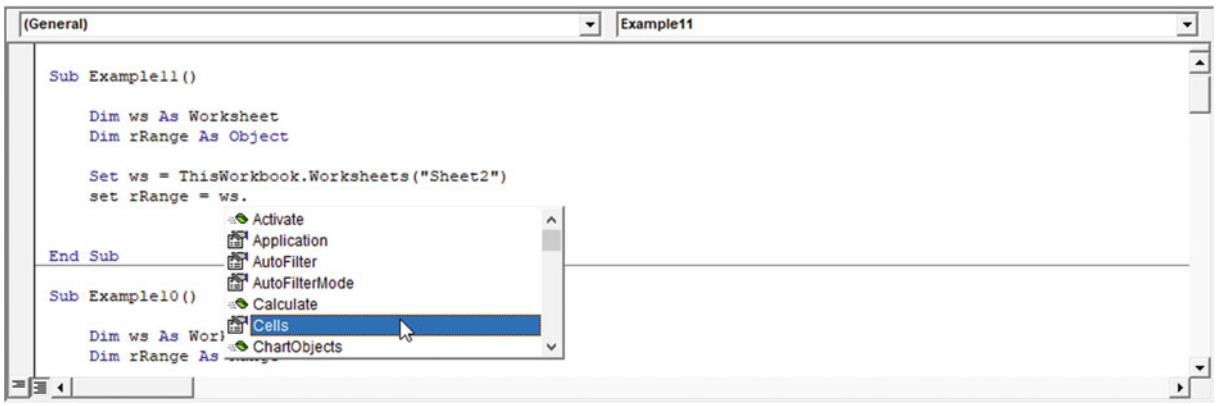


The below rewrites the macro "Example9" by defining the variable "ws" as a *worksheet* data type and the variable "rRange" as a *range* data type.



Experienced VBA programmers prefer the macro "Example10" over the macro "Example9."

One reason to use specific data object types over the generic *object* data type is that the auto list member feature will not work with variables that are defined as an *object* data type. The auto list member feature will work with variables that are defined as specifically defined data types. This is shown below.



3.8 Functions

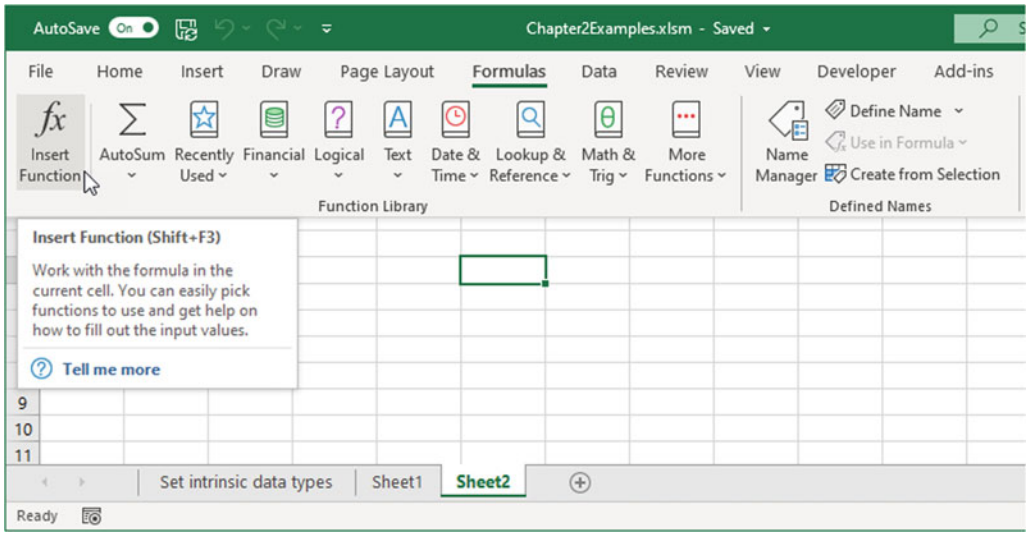
Functions in VBA act very much like functions in math. For example, below is a function that multiplies every number by 0.10.

$$f(x) = x * .1$$

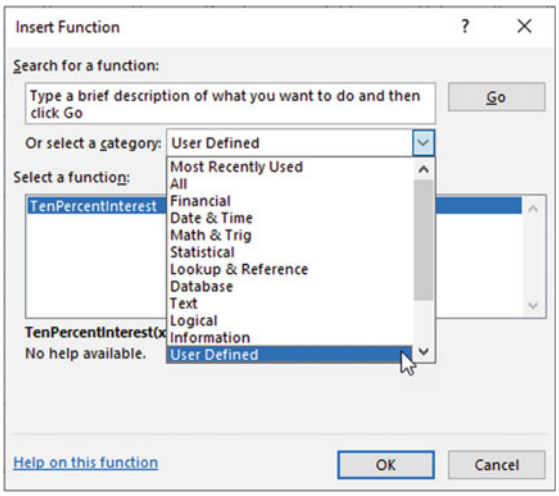
So in the above function, if x is 1,000, then $f(x)$ is 100. The above function can be used in a bank that has a certificate of deposit or CD that pays 10%. So if a customer opens a $1,000 CD, a banker can use the above function to calculate the interest. The function indicates that the interest is $100. Below is a VBA function that creates the above mathematical function.



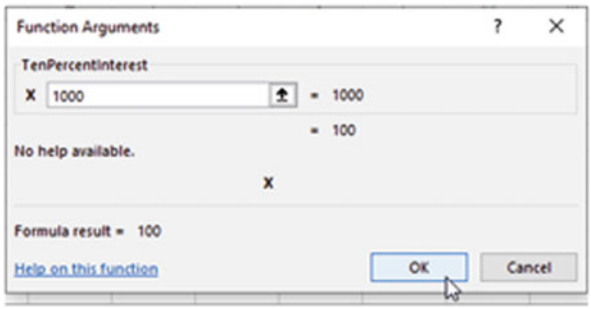
Functions created in Excel VBA can be used in the workbook that contains the function. To demonstrate this, go to the *Formula* tab and click on *Insert Function*.



Next, in the *Insert Function* dialog box, select *User Defined* in the *category* drop-down box.

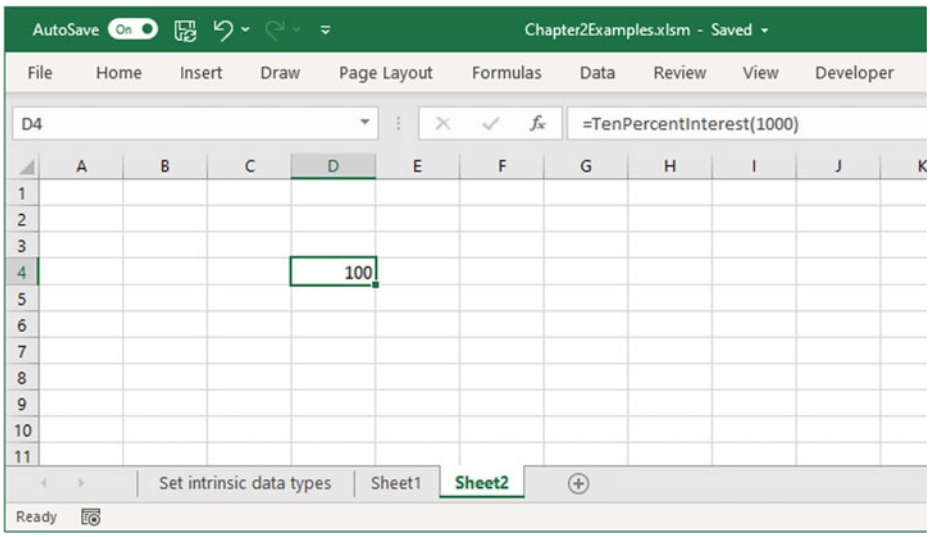


Notice that the function *TenPercentInterest* is listed in the *Insert Function* dialog box. To use the function we created, highlight the function that we created as shown above and then press the *OK* button. Pressing the *OK* button will result in the following.



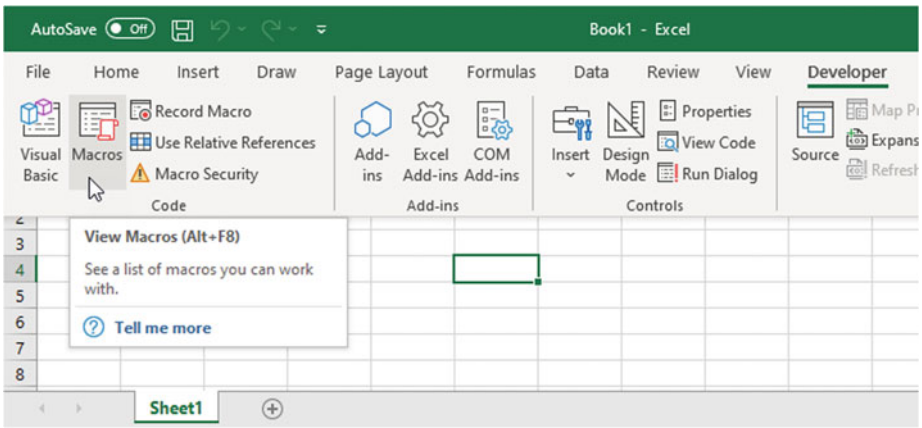
Notice that the above dialog box displays the parameter of the function. The above dialog box shows that entering the value 1000 for the parameter *x* will result in a value of 100. In functions that come with Excel, this dialog box will describe the function of interest. We can also do this for our *TenPercentInterest* function.

The following is the result after pressing the *OK* button in the above dialog box.

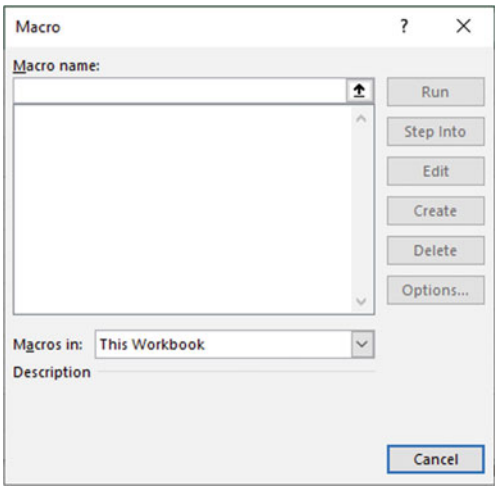


3.9 Adding a Function Description

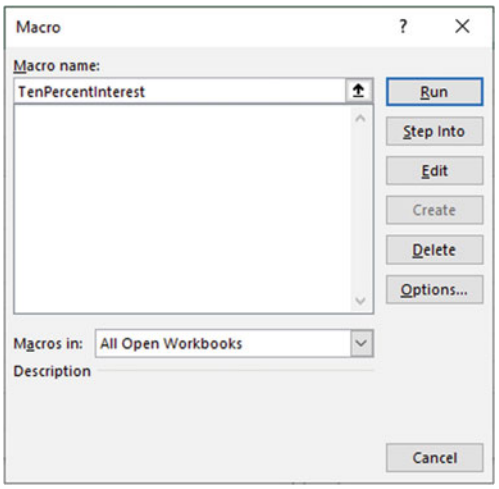
We will now show how to make it so there is a description for our *TenPercentInterest* function in the *Insert Function* dialog box. The first thing that we will need to do is to choose Developer → Macro as shown below



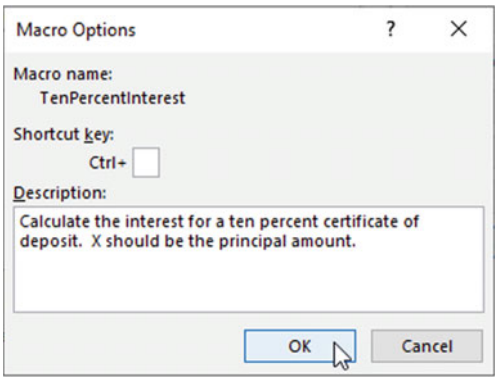
The resulting *Macro* dialog box is shown below.



Notice that in the above *Macro* dialog box no macro name is displayed and the only button active is the *Cancel* button. The reason for this is that the *Macro* dialog box only shows subprocedures. We did not include any subprocedures in our workbook. To write a description for a function, we would type in our function name in the *Macro name:* option of the *Macro* dialog box as shown below.

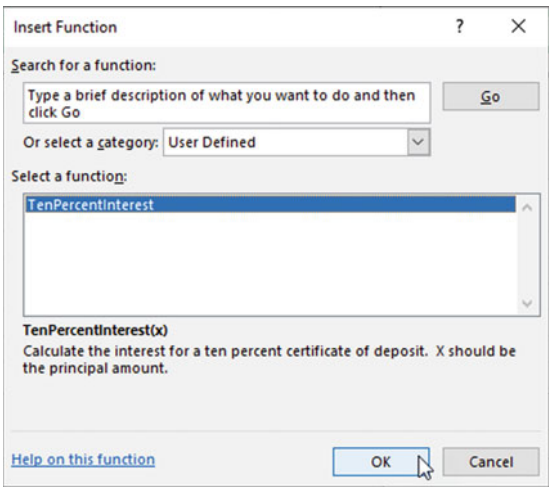


The next thing to do would be to press the *Options* button of the *Macro* dialog box to get the *Macro Options* dialog box shown below.



The next thing to do is to type the description for the function in the *Description* option of the *Macro Options* dialog box. After you finish typing in the description, press the *OK* button.

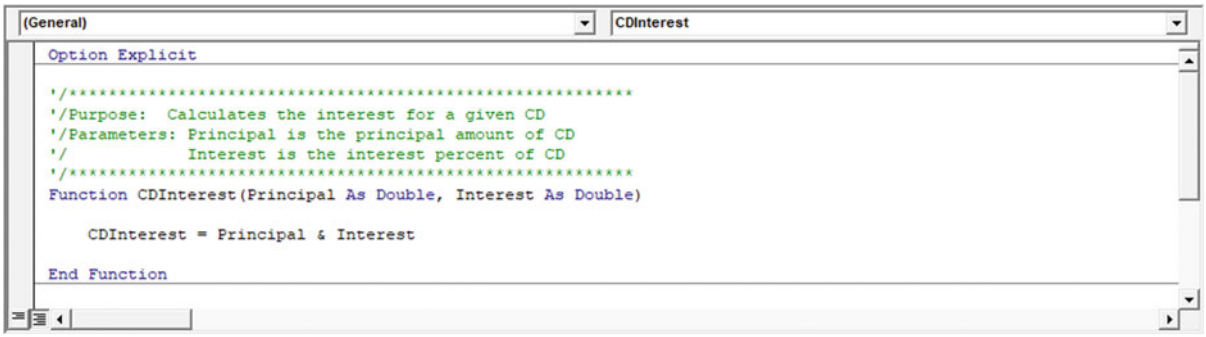
If we now go back to the *Insert Function* dialog box, we should now see the description that we typed in for our function. This is shown below.



There are a few limitations with the function *TenPercentInterest*. The limitations are

1. This function is only good for CDs that have a 10% interest rate.
2. The parameter x is not very descriptive.

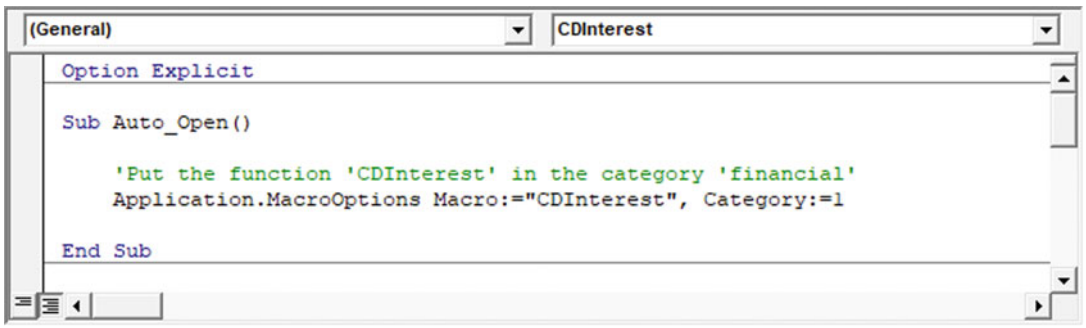
The function *CDInterest* addresses these issues.



3.10 Specifying a Function Category

When you create a custom function in VBA, Excel, by default, puts the function in the *User Defined* category of the *Insert Function* dialog box. In this section, we will show how through VBA to set it so that the function CDInterest shows up in the "financial" category of the *Insert Function* dialog box.

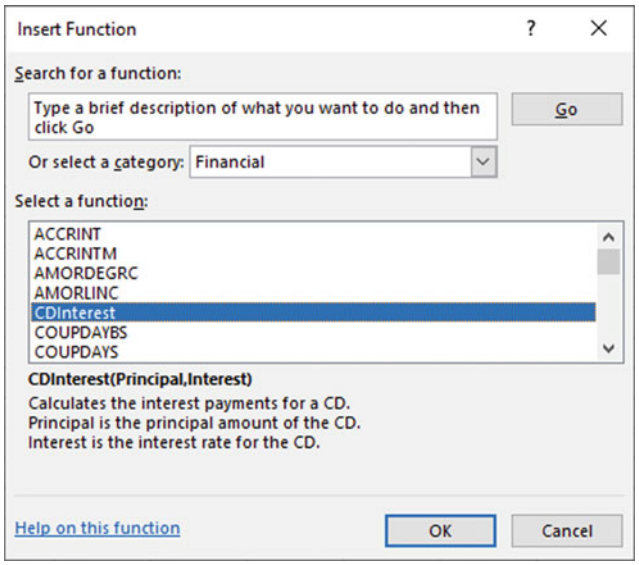
Below is the VBA procedure to set it so that the *CDInterest* function will be categorized in the "financial" category.



The *MacroOptions* method of the *Application* object puts the function *CDInterest* in the "finance" category of the *Insert Function* dialog box. The *MacroOptions* method must be executed every time when we open the workbook that contains the function *CDInterest*.

This task is done by the procedure *Auto_Open* because VBA will execute the procedure called "Auto_Open" when a workbook is opened.

The below shows the function *CDInterest* in the "Financial" category in the *Insert Function* dialog box.



Below is a table showing the category number for the other categories of the *Insert Function* dialog box.

Category number	Category name
0	All
1	Financial
2	Date and time
3	Math and trig
4	Statistical
5	Lookup and reference
6	Database
7	Text
8	Logical
9	Information
14	User defined
15	Engineering

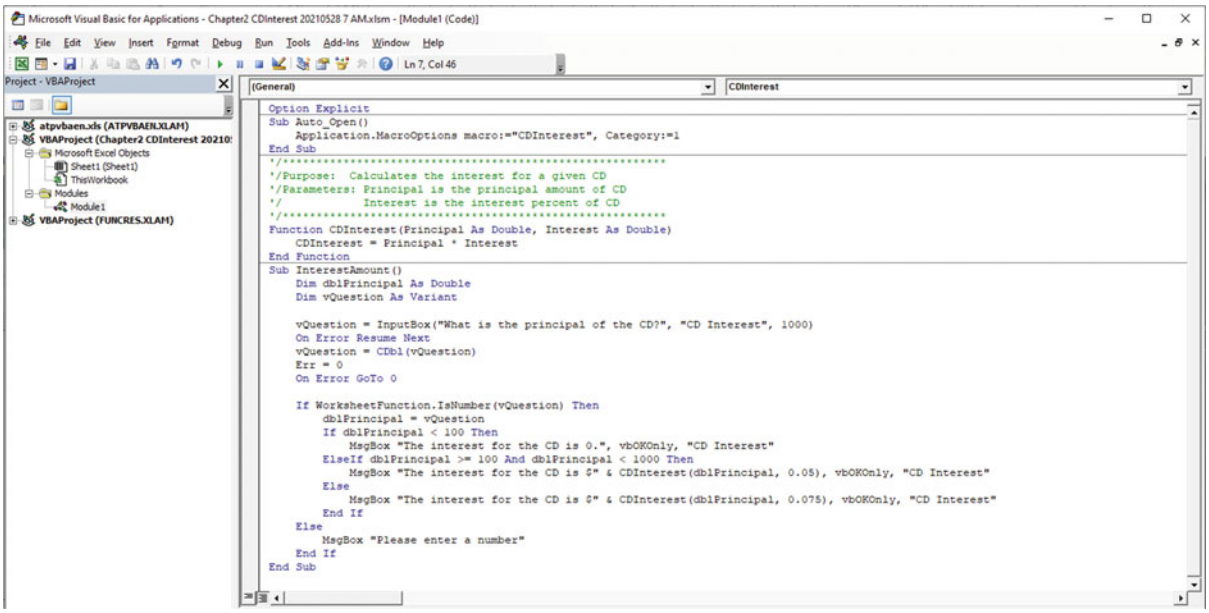
3.11 Conditional Programming with the IF Statement

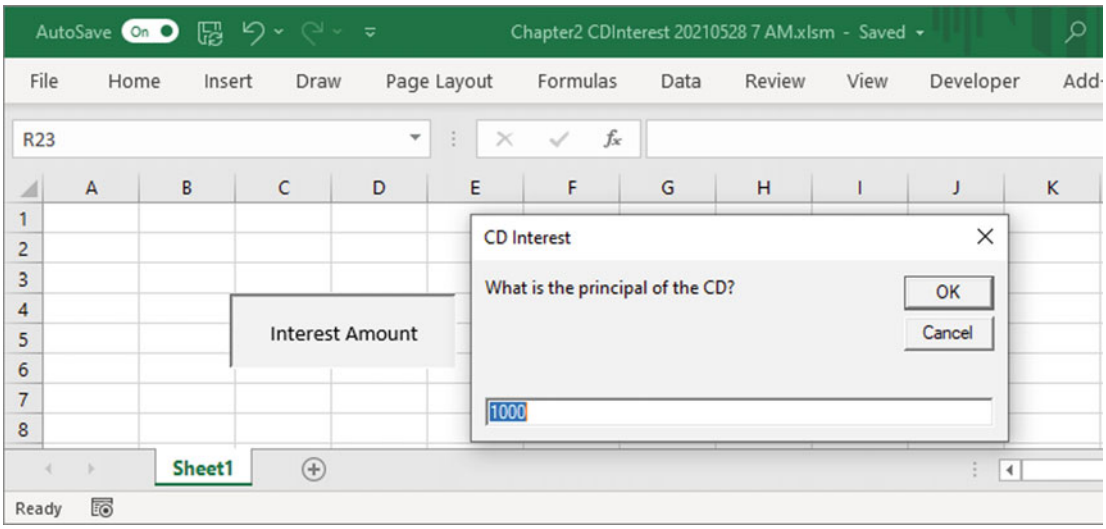
The VBA *If* statement is used to do conditional programming. The below shows the procedure "InterestAmount" This procedure will assign an interest rate based on the amount of the CD balance and then give the interest for the CD. The procedure "InterestAmount" uses the function "CDInterest" that we created in the previous section to calculate the interest amount.

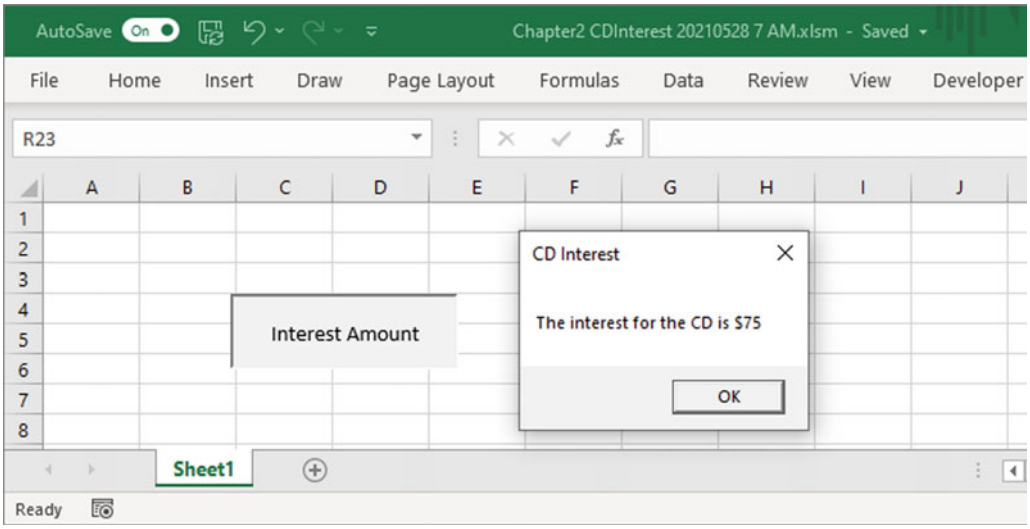
It is possible to use most of the built-in worksheet functions in VBA programming. The procedure "CDInterest" uses the worksheet function "Isnumber" to check if the principal amount entered is a number or not. Worksheet functions belong to the *worksheetfunction* object of the Excel object model.

We can say that module "module1" in the workbook is a program. It is a program because "module1" has two procedures and one function. A VBA program is basically a grouping of procedures and functions.

The below demonstrates the procedure "InterestAmount".

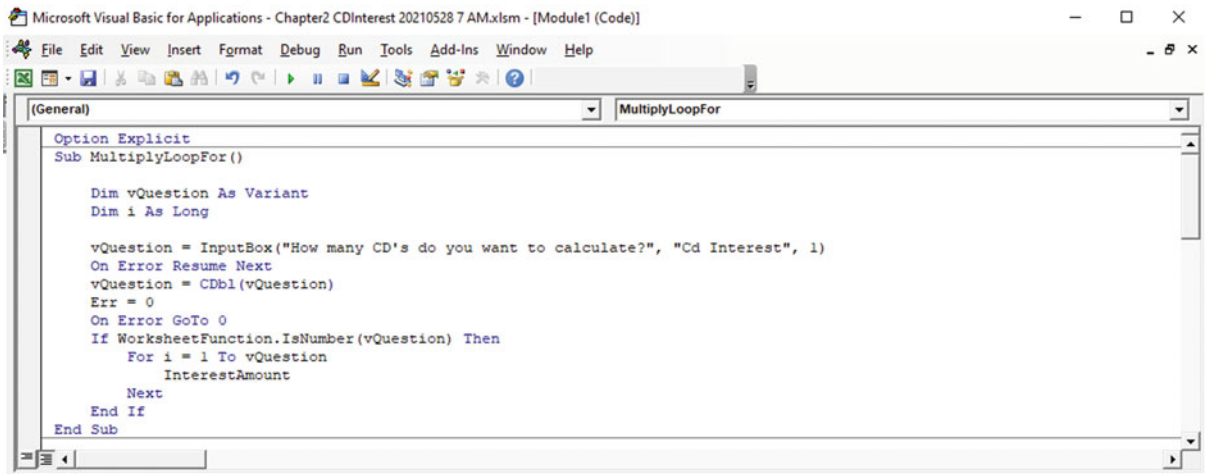

Microsoft Visual Basic for Applications - Chapter2 CDInterest 20210528 7 AM.xlsm - [Module1 (Code)]
File Edit View Insert Format Debug Run Tools Add-Ins Window Help
Ln 7, Col 46
Project - VBAProject
atpvbaen.xls (ATPVBAEN.XLAM)
VBAProject (Chapter2 CDInterest 20210
Microsoft Excel Objects
Sheet1 (Sheet1)
ThisWorkbook
Modules
Module1
VBAProject (FUNCRES.XLAM)
(General)
CDInterest
Option Explicit
Sub Auto_Open()
Application.MacroOptions macro:="CDInterest", Category:=1
End Sub
'/**********
'/Purpose: Calculates the interest for a given CD
'/Parameters: Principal is the principal amount of CD
'/ Interest is the interest percent of CD
'/**********
Function CDInterest(Principal As Double, Interest As Double)
CDInterest = Principal * Interest
End Function
Sub InterestAmount()
Dim dblPrincipal As Double
Dim vQuestion As Variant
vQuestion = InputBox("What is the principal of the CD?", "CD Interest", 1000)
On Error Resume Next
vQuestion = CDbl(vQuestion)
Err = 0
On Error GoTo 0
If WorksheetFunction.IsNumber(vQuestion) Then
dblPrincipal = vQuestion
If dblPrincipal < 100 Then
MsgBox "The interest for the CD is 0.", vbOKOnly, "CD Interest"
ElseIf dblPrincipal >= 100 And dblPrincipal < 1000 Then
MsgBox "The interest for the CD is $" & CDInterest(dblPrincipal, 0.05), vbOKOnly, "CD Interest"
Else
MsgBox "The interest for the CD is $" & CDInterest(dblPrincipal, 0.075), vbOKOnly, "CD Interest"
End If
Else
MsgBox "Please enter a number"
End If
End Sub


AutoSave On
Chapter2 CDInterest 20210528 7 AM.xlsm - Saved
File Home Insert Draw Page Layout Formulas Data Review View Developer Add
R23
A B C D E F G H I J K
1 2 3 4 5 6 7 8
Interest Amount
CD Interest
What is the principal of the CD?
OK
Cancel
1000
Sheet1
Ready

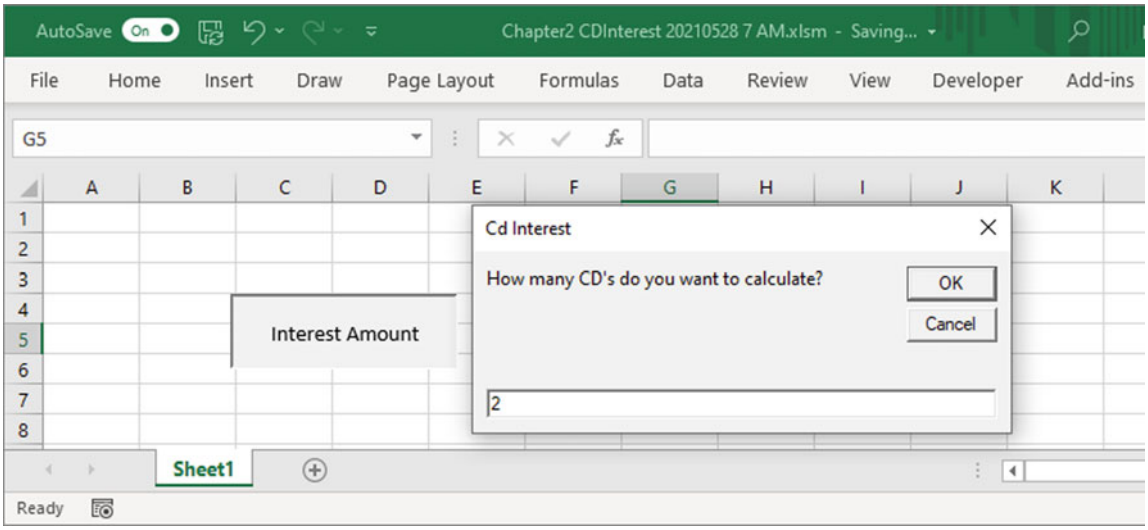


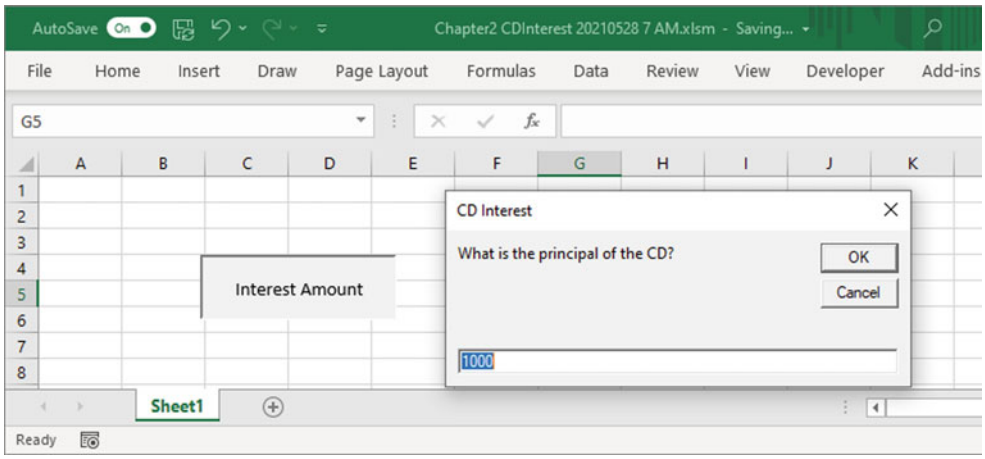
3.12 For Loop

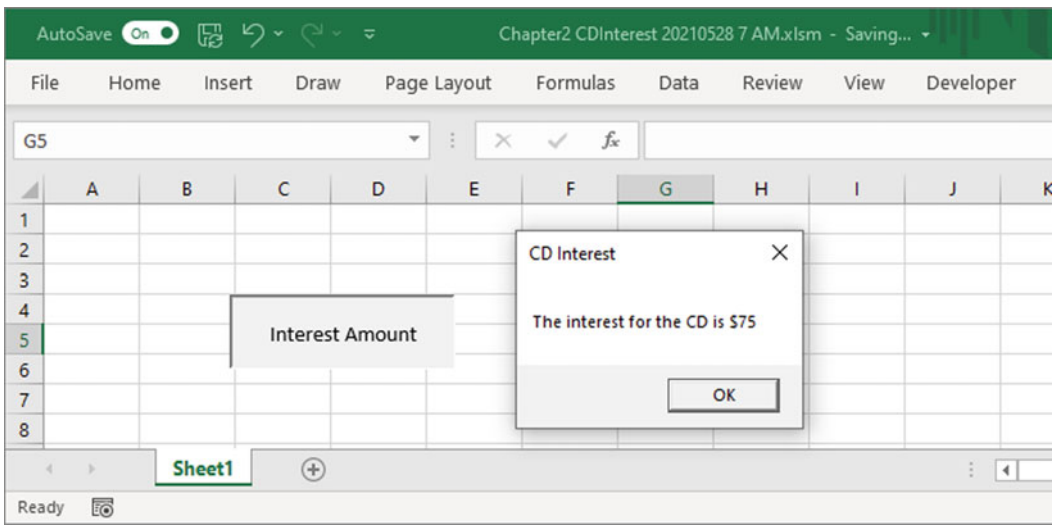
Up to this point, the VBA code that we have been writing is executed sequentially from top to bottom. When the VBA code reaches the bottom, it stops. We will now look at looping, the concept of where VBA code is executed more than once. The first looping code that we will look at is the *For* loop. The *For* loop is used when it can be determined how many times the loop should be. To demonstrate the *For* loop, we will extend our CD program in our previous section. We will add the procedure below to ask how many CDs we want to calculate.

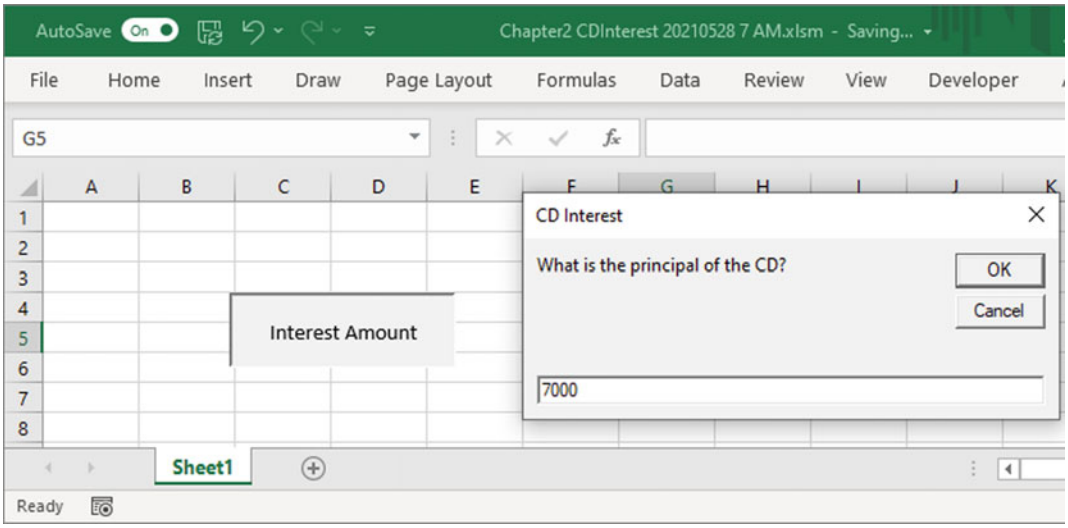


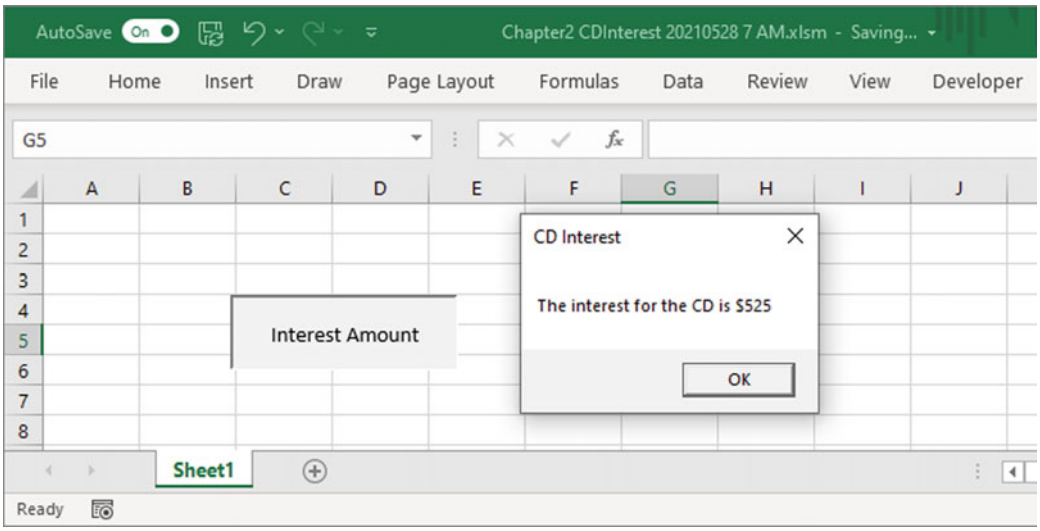
The below demonstrates the *MultiplyLoopFor* procedure.





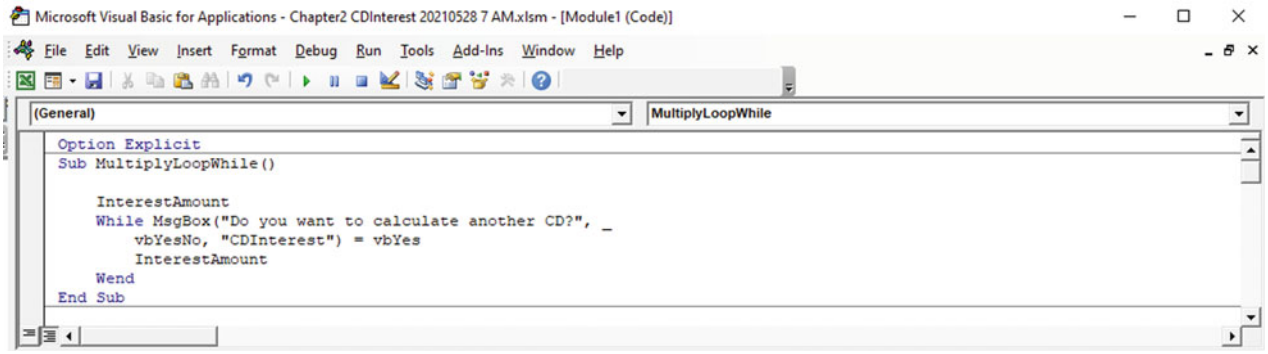




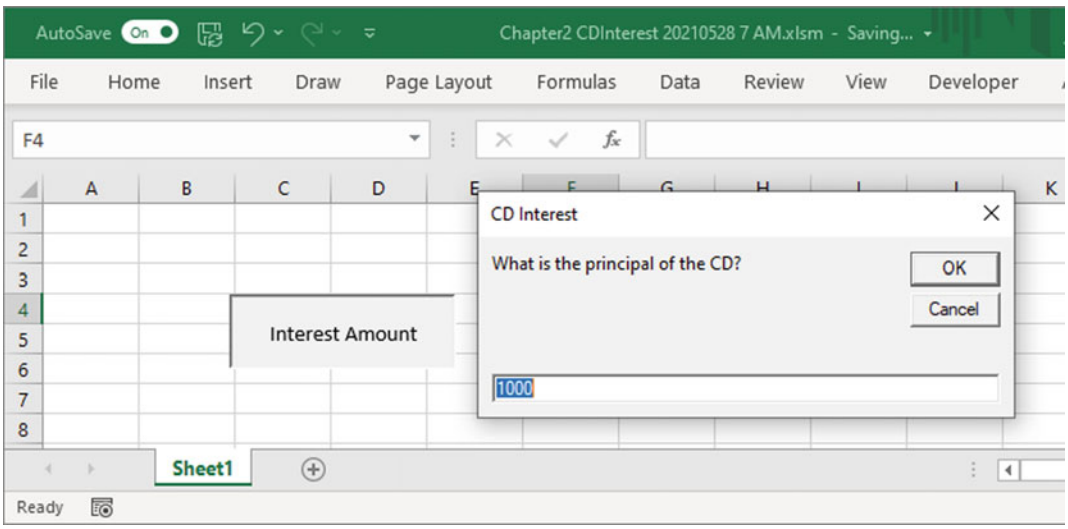


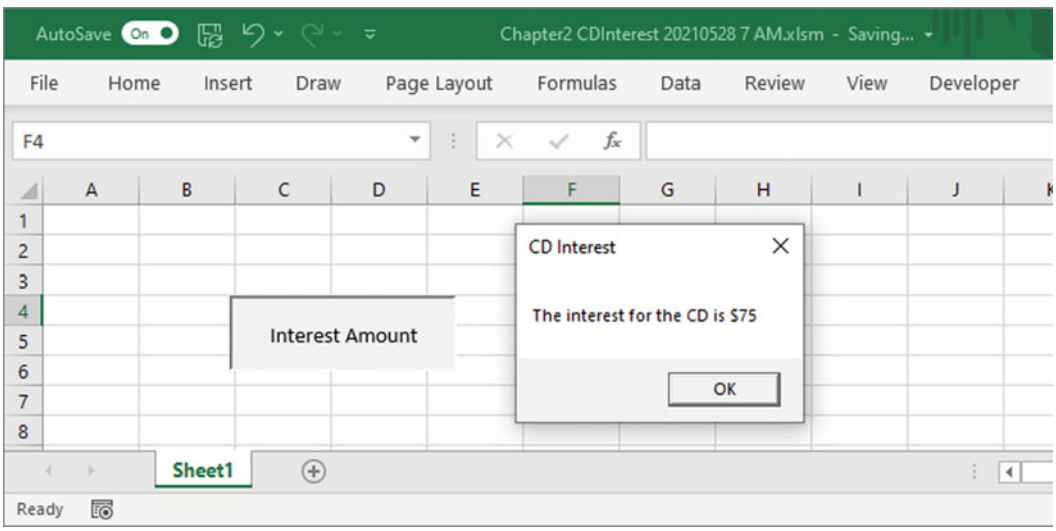
3.13 While Loop

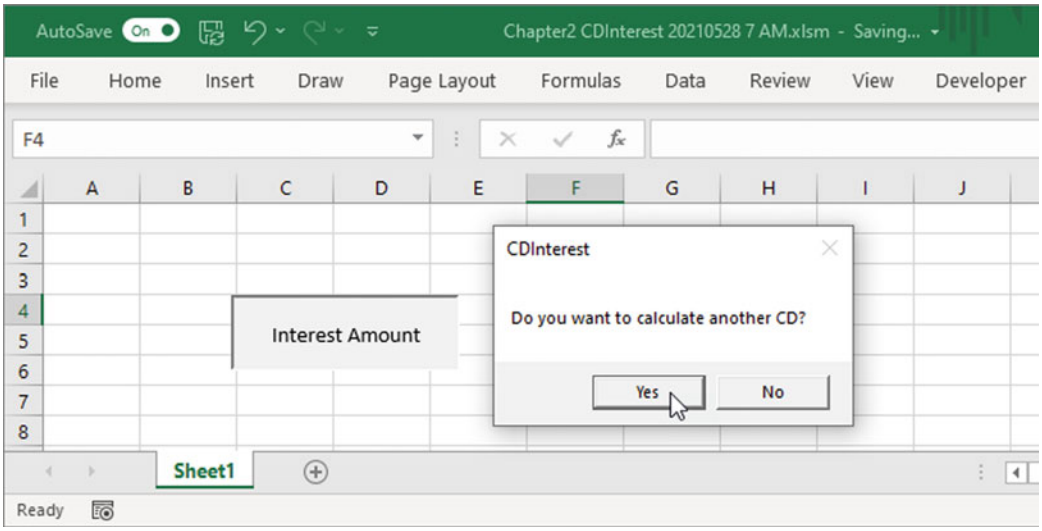
Many times, we do not know how many loops beforehand we will need. In this case, the *While* loop is used instead. The *While* loop does a conditional test during each loop to determine if a loop should be continued or not. To demonstrate the *While* loop, we will rewrite the above program to use the *While* loop instead of the *For* loop.

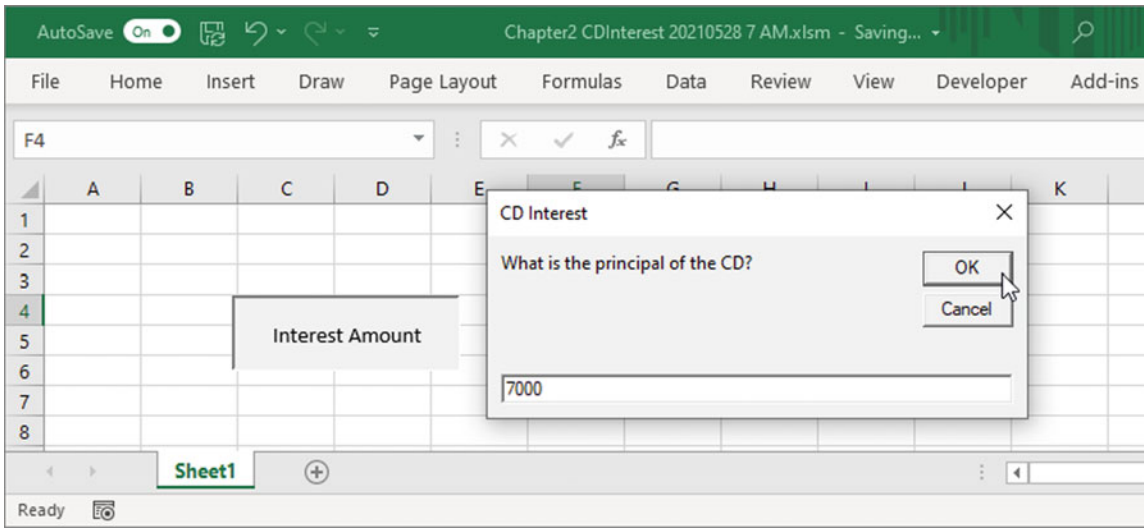


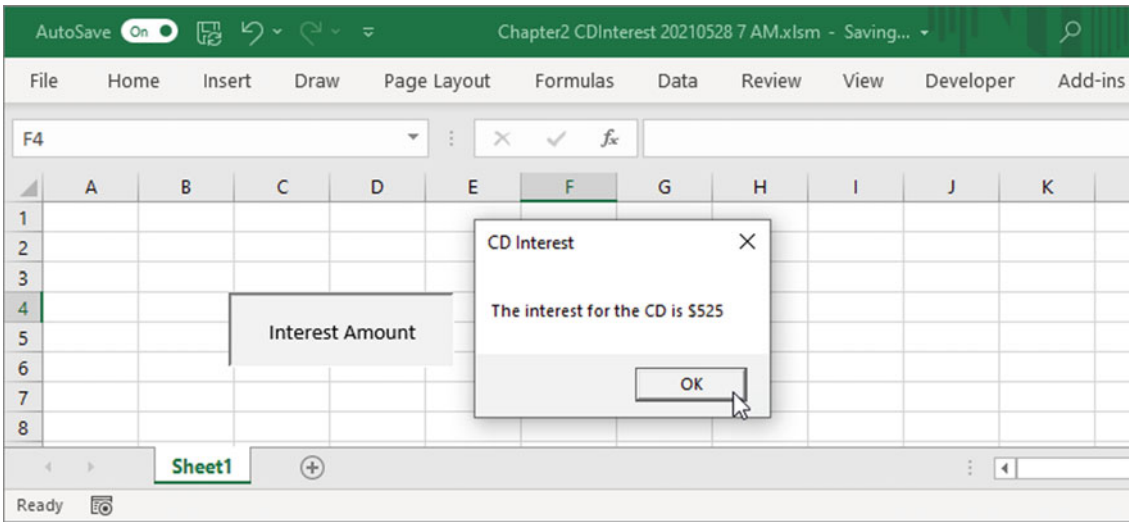
The below illustrates the *While* loop.

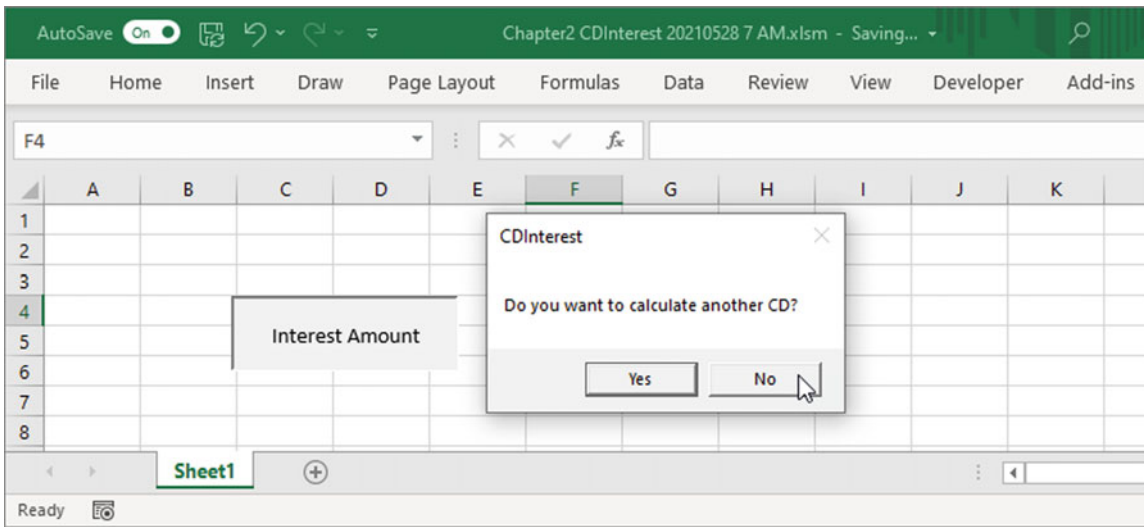





AutoSave On
Chapter2 CDInterest 20210528 7 AM.xlsm - Saving...
File Home Insert Draw Page Layout Formulas Data Review View Developer
F4
Interest Amount
CDInterest
Do you want to calculate another CD?
Yes
No
Sheet1
Ready


AutoSave On
Chapter2 CDInterest 20210528 7 AM.xlsm - Saving...
File Home Insert Draw Page Layout Formulas Data Review View Developer Add-ins
F4
Interest Amount
CD Interest
What is the principal of the CD?
OK
Cancel
7000
Sheet1
Ready





3.14 Arrays

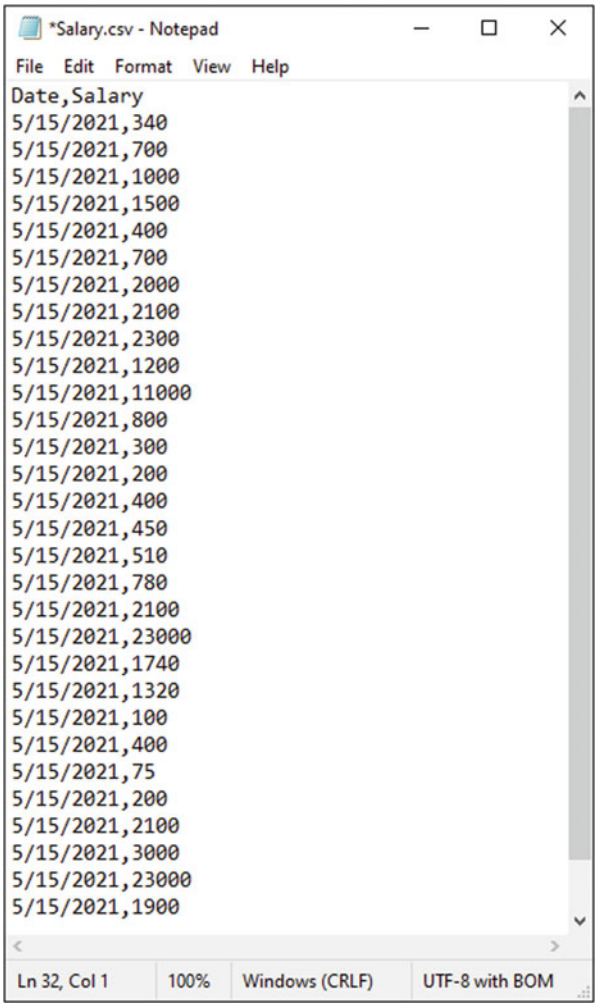
Most of the time, when we are analyzing a dataset, the dataset contains data of the same data type. For example, we may have a dataset of accounting salaries, a dataset of GM stock prices, a dataset of accounts receivables, or a dataset of certificates of deposits. We might define 50 variables if we are processing a dataset of salaries that have 50 data items. We might define the variables as "Salary1," "Salary2," "Salary3,"... "Salary50." Another alternative is to define an *Array* of salaries. An *Array* is a group or collection of like data items. We would reference a particular salary through an index. The following is how to define our salary array variable of 50 elements:

```
Dim Salary (1 to 50) As Double
```

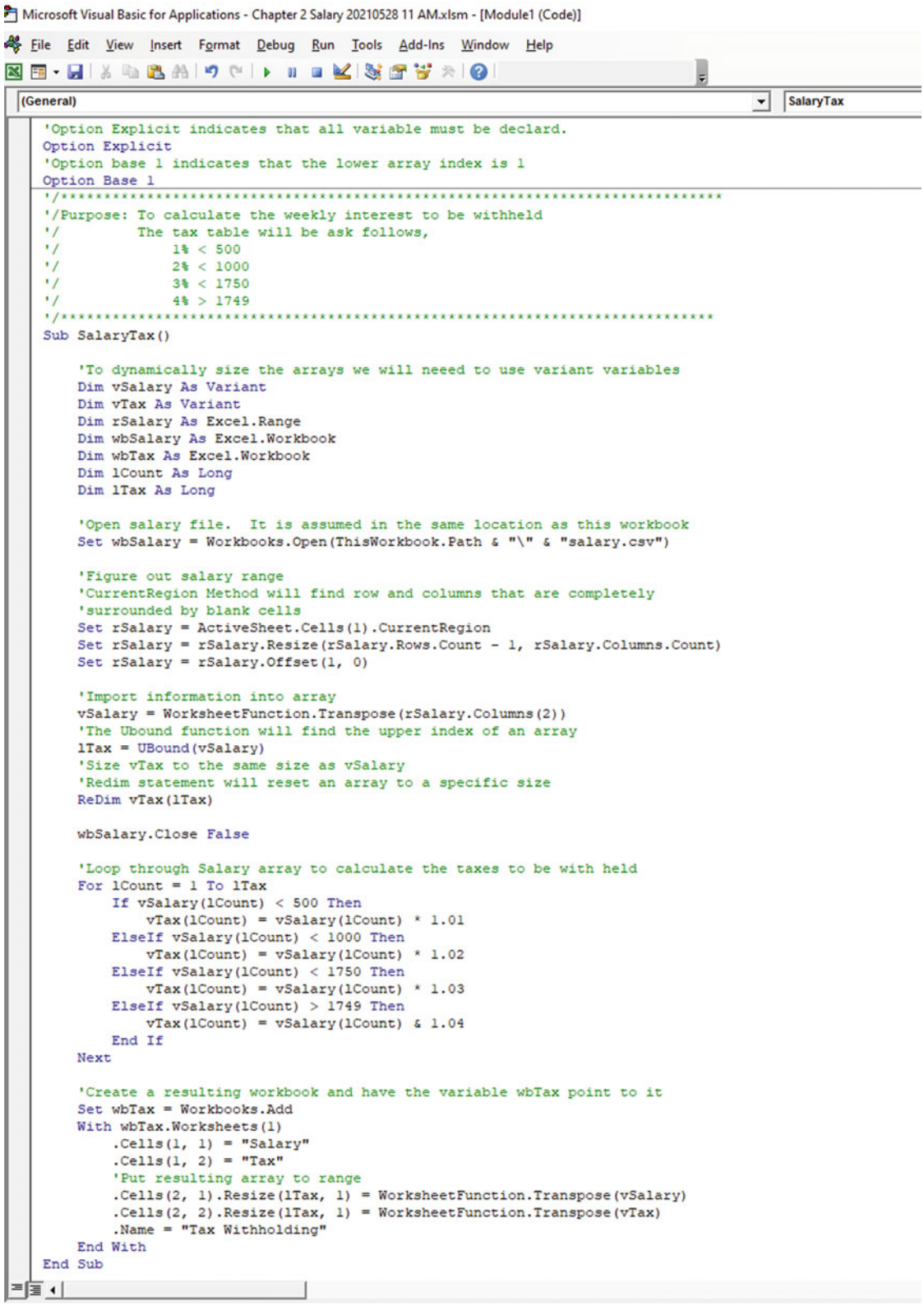
The following shows how to assign 15,000 to the 20th salary item:

```
Salary(20) = 15000
```

Suppose we need to calculate every 2 weeks the income tax to be withheld from 30 employees. This situation is very similar to our example in calculating the interest of the certificate of deposits. When we calculate the certificate of deposits, we prompted the user for the principal amount. This process is very time-consuming and very tedious. In the business world, it is common that the information of interest is already in an application. The procedure would then be to extract the information to a file to be processed. For our salary example, we will extract the salary data to a *csv* file format. A CSV file format is basically a text file that is separated by commas. A common application to read CSV files is Microsoft Windows Notepad. The below shows the "salary.csv" that we are interested in processing.



The thing to note about the *csv* file is that the first row is usually the header. The first row is the row that describes the columns of a dataset. In the salary file above, we can say that the header contains two fields. One field is the date field, and the other field is the salary field.




```
'Option Explicit indicates that all variable must be declard.
Option Explicit
'Option base 1 indicates that the lower array index is 1
Option Base 1
'/**************************************************************************
'/Purpose: To calculate the weekly interest to be withheld
'/         The tax table will be ask follows,
'/             1% < 500
'/             2% < 1000
'/             3% < 1750
'/             4% > 1749
'/**************************************************************************
Sub SalaryTax()

    'To dynamically size the arrays we will neeed to use variant variables
    Dim vSalary As Variant
    Dim vTax As Variant
    Dim rSalary As Excel.Range
    Dim wbSalary As Excel.Workbook
    Dim wbTax As Excel.Workbook
    Dim lCount As Long
    Dim lTax As Long

    'Open salary file.  It is assumed in the same location as this workbook
    Set wbSalary = Workbooks.Open(ThisWorkbook.Path & "\" & "salary.csv")

    'Figure out salary range
    'CurrentRegion Method will find row and columns that are completely
    'surrounded by blank cells
    Set rSalary = ActiveSheet.Cells(1).CurrentRegion
    Set rSalary = rSalary.Resize(rSalary.Rows.Count - 1, rSalary.Columns.Count)
    Set rSalary = rSalary.Offset(1, 0)

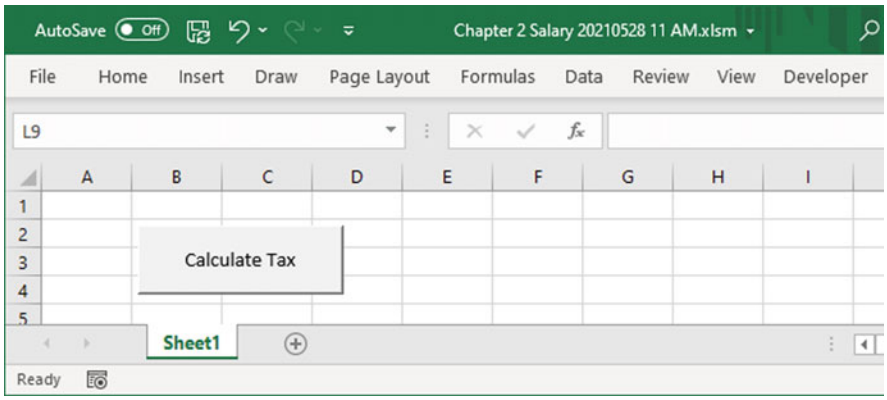
    'Import information into array
    vSalary = WorksheetFunction.Transpose(rSalary.Columns(2))
    'The Ubound function will find the upper index of an array
    lTax = UBound(vSalary)
    'Size vTax to the same size as vSalary
    'Redim statement will reset an array to a specific size
    ReDim vTax(lTax)

    wbSalary.Close False

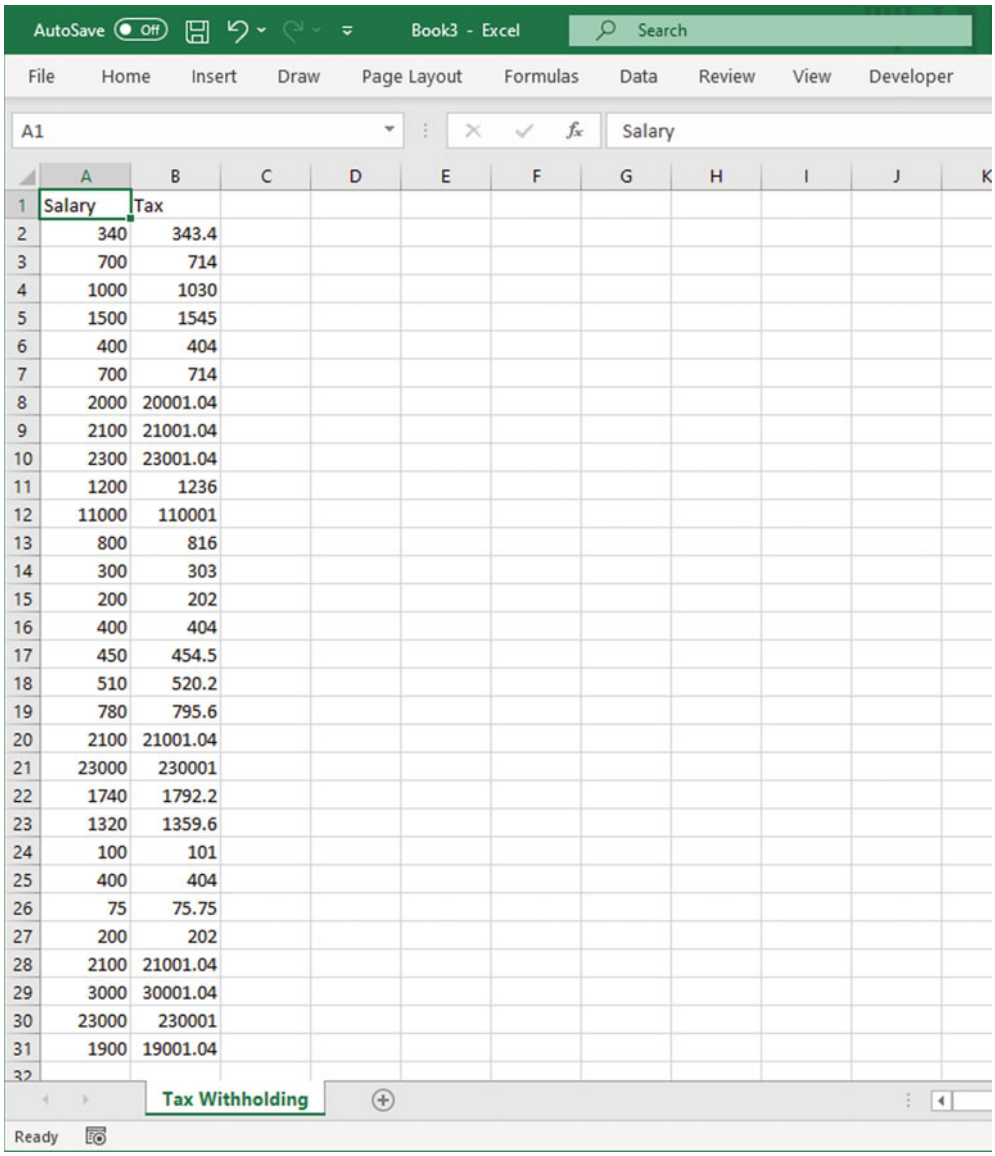
    'Loop through Salary array to calculate the taxes to be with held
    For lCount = 1 To lTax
        If vSalary(lCount) < 500 Then
            vTax(lCount) = vSalary(lCount) * 1.01
        ElseIf vSalary(lCount) < 1000 Then
            vTax(lCount) = vSalary(lCount) * 1.02
        ElseIf vSalary(lCount) < 1750 Then
            vTax(lCount) = vSalary(lCount) * 1.03
        ElseIf vSalary(lCount) > 1749 Then
            vTax(lCount) = vSalary(lCount) & 1.04
        End If
    Next

    'Create a resulting workbook and have the variable wbTax point to it
    Set wbTax = Workbooks.Add
    With wbTax.Worksheets(1)
        .Cells(1, 1) = "Salary"
        .Cells(1, 2) = "Tax"
        'Put resulting array to range
        .Cells(2, 1).Resize(lTax, 1) = WorksheetFunction.Transpose(vSalary)
        .Cells(2, 2).Resize(lTax, 1) = WorksheetFunction.Transpose(vTax)
        .Name = "Tax Withholding"
    End With
End Sub
```

The below illustrates the *SalaryTax* procedure.



Pushing the *Calculate Tax* button will result in the following workbook.



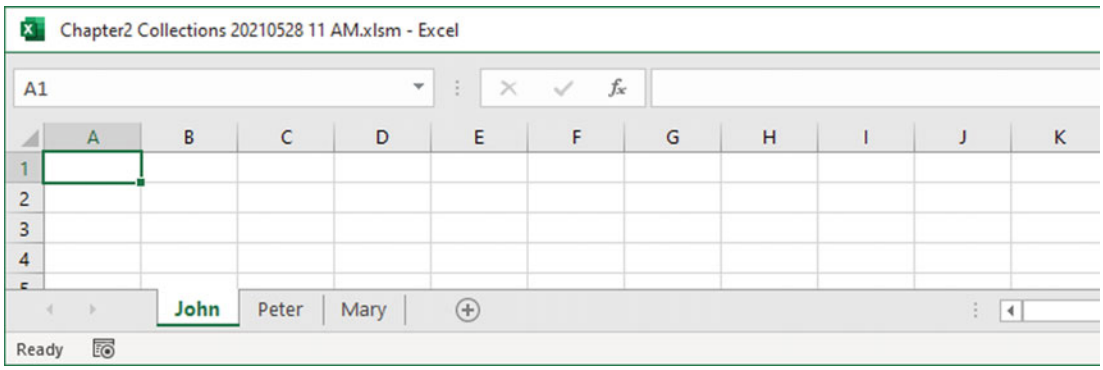
Salary	Tax
340	343.4
700	714
1000	1030
1500	1545
400	404
700	714
2000	20001.04
2100	21001.04
2300	23001.04
1200	1236
11000	110001
800	816
300	303
200	202
400	404
450	454.5
510	520.2
780	795.6
2100	21001.04
23000	230001
1740	1792.2
1320	1359.6
100	101
400	404
75	75.75
200	202
2100	21001.04
3000	30001.04
23000	230001
1900	19001.04

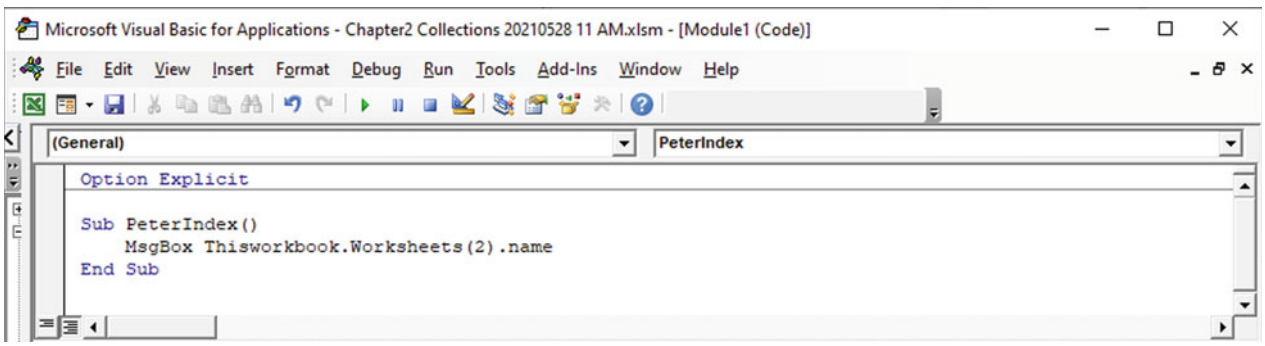
3.15 Option Base 1

When normal people think about lists, they usually start with the number 1. A lot of times, programmers begin a list with the number 0. In VBA programming, the beginning of an array index is 0. To set it so that the beginning of array index is 1, we would use the statement "Option Base 1." This was done in the procedure "SalaryTax" in the previous procedure.

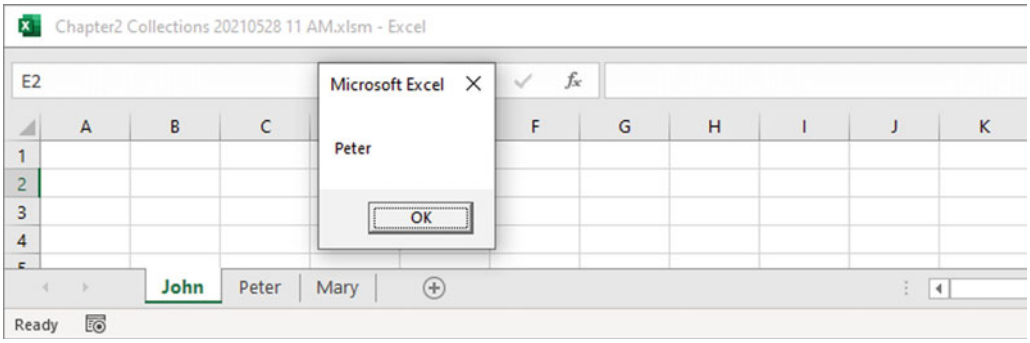
3.16 Collections

In VBA programming, there is a lot of programming with a group of like items. Groups of like items are called *Collections*. Examples are collections of workbooks, worksheets, cells, charts, and names. There are two ways to reference a collection. The first way is through an index. The second way is by name. For example, suppose we have the following workbook that contains three worksheets.

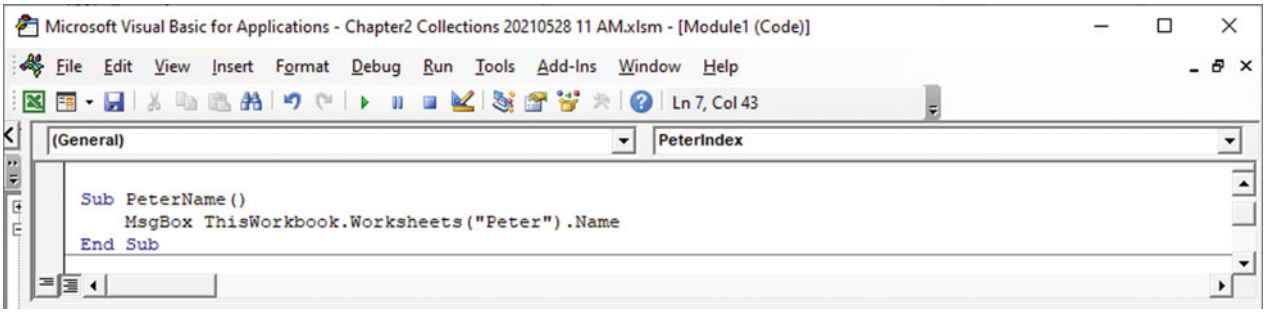




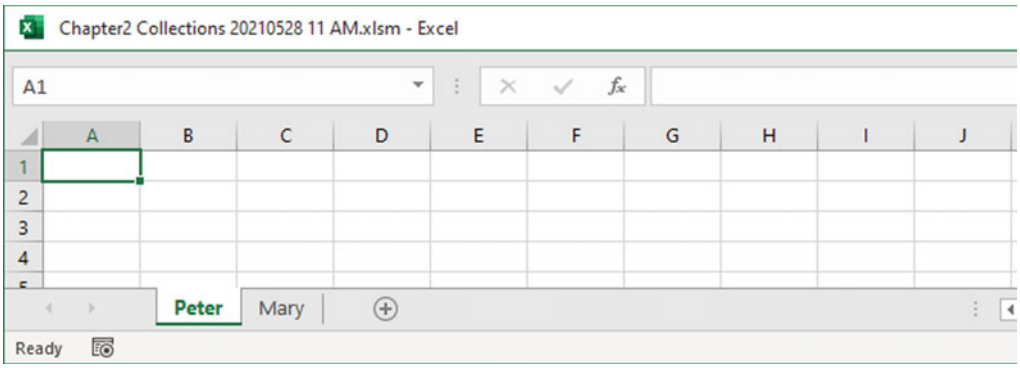
The below demonstrates the procedure “PeterIndex.”



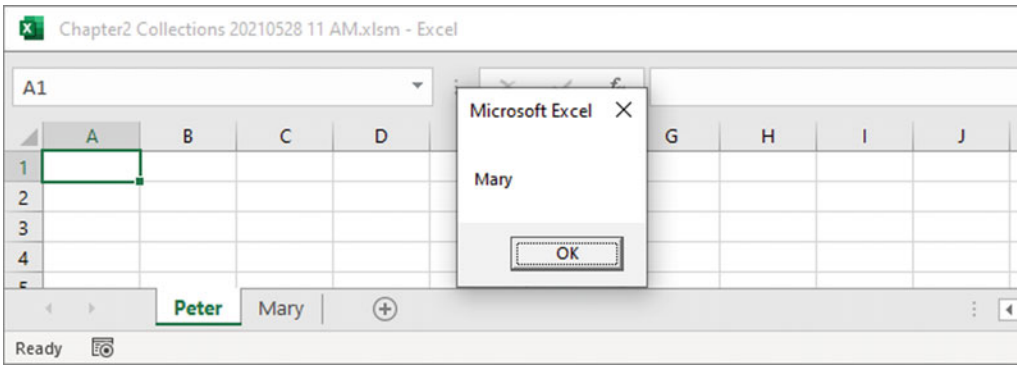
Below is a procedure that references the second worksheet by name.



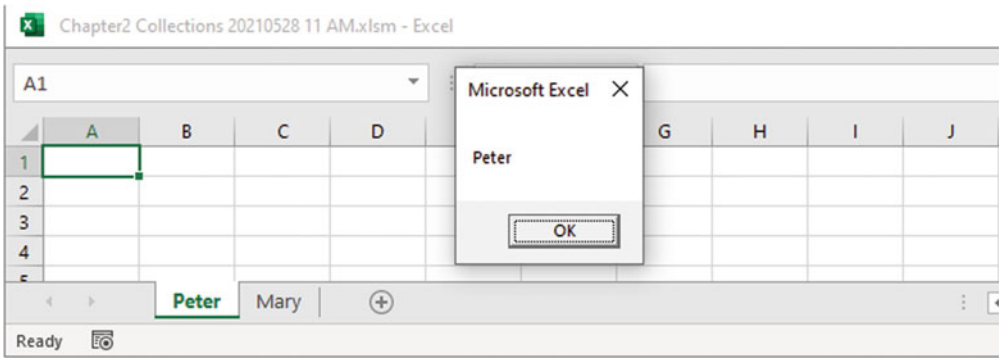
It is important to note what the effect of removing an item from a collection to a VBA code is. The below shows the workbook without the worksheet “John.”



Below is the result when executing the procedure "PeterIndex."



Below is the result when executing the procedure "PeterName."



The above demonstrates that referencing an item in a collection by name is preferable when there are additions or deletions to a collection.

3.17 Summary

In this chapter, we discussed Excel's object model, the Intellisense menu, and the object browser. We also looked at variables and talked about option explicit. We discussed object variables and functions. We discussed adding a function description and then discussed specifying a function category. We discussed conditional programming with the IF statement, for loop, and while loop. We also talked about arrays. We talked about option base 1 and collections.

References

https://www.excelcampus.com/vba/intellisense-keyboard-shortcuts/
https://docs.microsoft.com/en-us/office/vba/language/reference/user-interface-help/data-type-summary

4 Professional Techniques Used in Excel and VBA

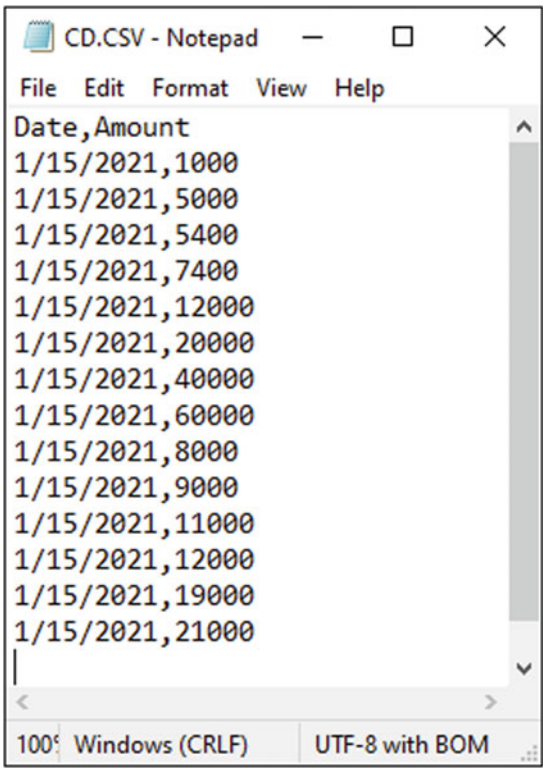
4.1 Introduction

In this chapter, we will discuss Excel and Excel VBA techniques that are useful and are not usually discussed or pointed out in Excel and Excel VBA books.

This chapter is broken down into the following sections. In Sect. 4.2 we find the range of a table with the CurrenRegion property, and in Sect. 4.3, we discuss the offset property of the range object. In Sect. 4.4, we discuss resizing the property of the range object, and in Sect. 4.5, we discuss the UsedRange property of the range object. In Sect. 4.6, we look at a special dialog box in Excel. In Sect. 4.7, we import column data into arrays, and in Sect. 4.8, we import row data into an array. In Sect. 4.9, we then transfer data from an array to a range. In Sect. 4.10, we discuss workbook names, and in Sect. 4.11, we look at dynamic range names. Section 4.12 looks at global versus local workbook names. In Sect. 4.13, we list all of the files in a directory. Finally, in Sect. 4.14, we summarize the chapter.

4.2 Finding the Range of a Table: CurrentRegion Property

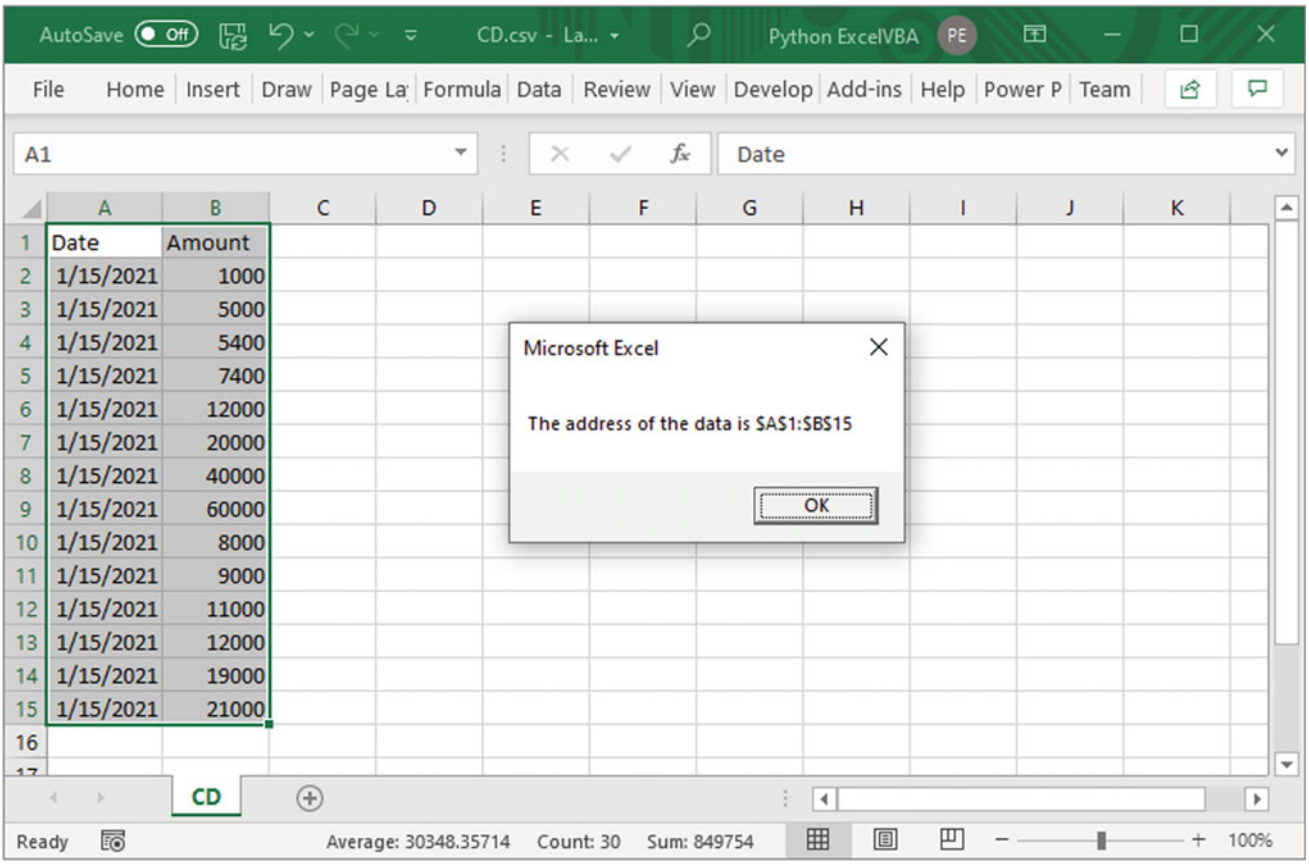
Many times we are interested in finding the range or an address of a table. A way to do this is to use the *CurrentRegion* property of the range object. One common situation where there is a need to do this is when we import data files. Usually, Excel places the data in the upper left-hand corner of the first worksheet.




```
'/
'''''''
```

```
On Error Resume Next
'Open CD file. It is assumed in same location as this workbook
Set wbCD = Workbooks.Open(ThisWorkbook.Path & ``\'' & ``CD.csv'')
If wbCD Is Nothing Then
MsgBox ``Could not find the file CD.csv in the path '' _
& ThisWorkbook.Path, vbCritical
End
End If
'Figure out salary range
'CurrentRegion Method will find row and columns that are completely
'surronded by blank cells
Set rCD = ActiveSheet.Cells(1).CurrentRegion
rCD.Select
MsgBox ``The address of the data is '' & rCD.Address
wbCD.Close False
End Sub
```

The above procedure will open the "CD.csv" file and then select the data range by using the *CurrentRegion* property of the range object and also display the address of the data range. Below demonstrates the *FindCurrentRegion* procedure.



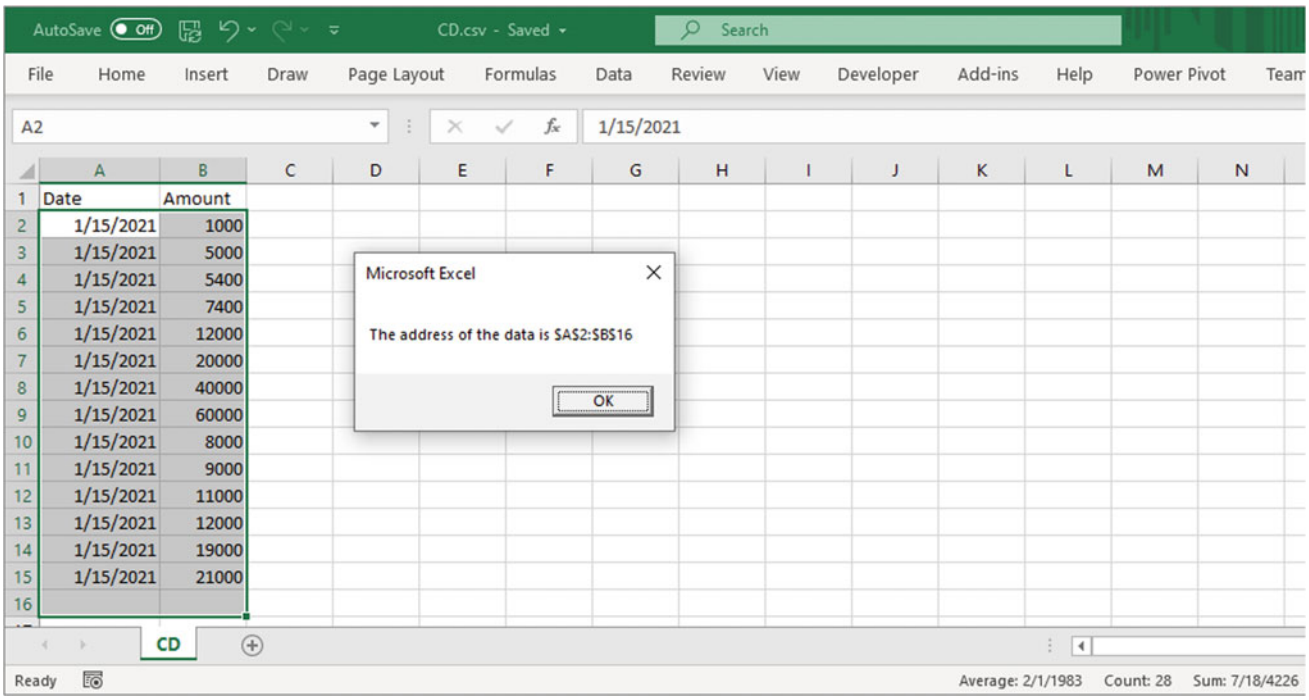
	A	B
1	Date	Amount
2	1/15/2021	1000
3	1/15/2021	5000
4	1/15/2021	5400
5	1/15/2021	7400
6	1/15/2021	12000
7	1/15/2021	20000
8	1/15/2021	40000
9	1/15/2021	60000
10	1/15/2021	8000
11	1/15/2021	9000
12	1/15/2021	11000
13	1/15/2021	12000
14	1/15/2021	19000
15	1/15/2021	21000
16		

Notice that the current region area contains the header or row 1. Many times when data is imported, we will want to exclude the header row. To solve this problem, we will look at the *offset* property of the range object in the next section.

4.3 Offset Property of the Range Object

The *offset* property is one of those properties and methods that are usually mentioned in passing in most books. The *offset* property has two arguments. The first argument is for the row offset. The second argument is for the column offset. Below is a procedure that illustrates the *offset* property.

```
'/
*************************************************
*****************************
'/Purpose: To find the data range of an imported file
'/
*************************************************
*****************************
Sub CurrentRegionOffset()
Dim rCD As Range
Dim wbCD As Workbook
On Error Resume Next
'Open CD file. It is assumed in same location as this
workbook
Set wbCD = Workbooks.Open(ThisWorkbook.Path & ``\'' &
``CD.csv'')
If wbCD Is Nothing Then
MsgBox ``Could not find the file CD.csv in the path '' _
& ThisWorkbook.Path, vbCritical
End
End If
'Figure out salary range
'CurrentRegion Method will find row and columns that are
completely
'surronded by blank cells
Set rCD = ActiveSheet.Cells(1).CurrentRegion
'Offset the current region by one row.
'The offset property has row offset argument and column
offset argument
Set rCD = rCD.Offset(rowoffset:=1, columnoffset:=0)
rCD.Select
MsgBox ``The address of the data is '' & rCD.Address
wbCD.Close False
End Sub
```



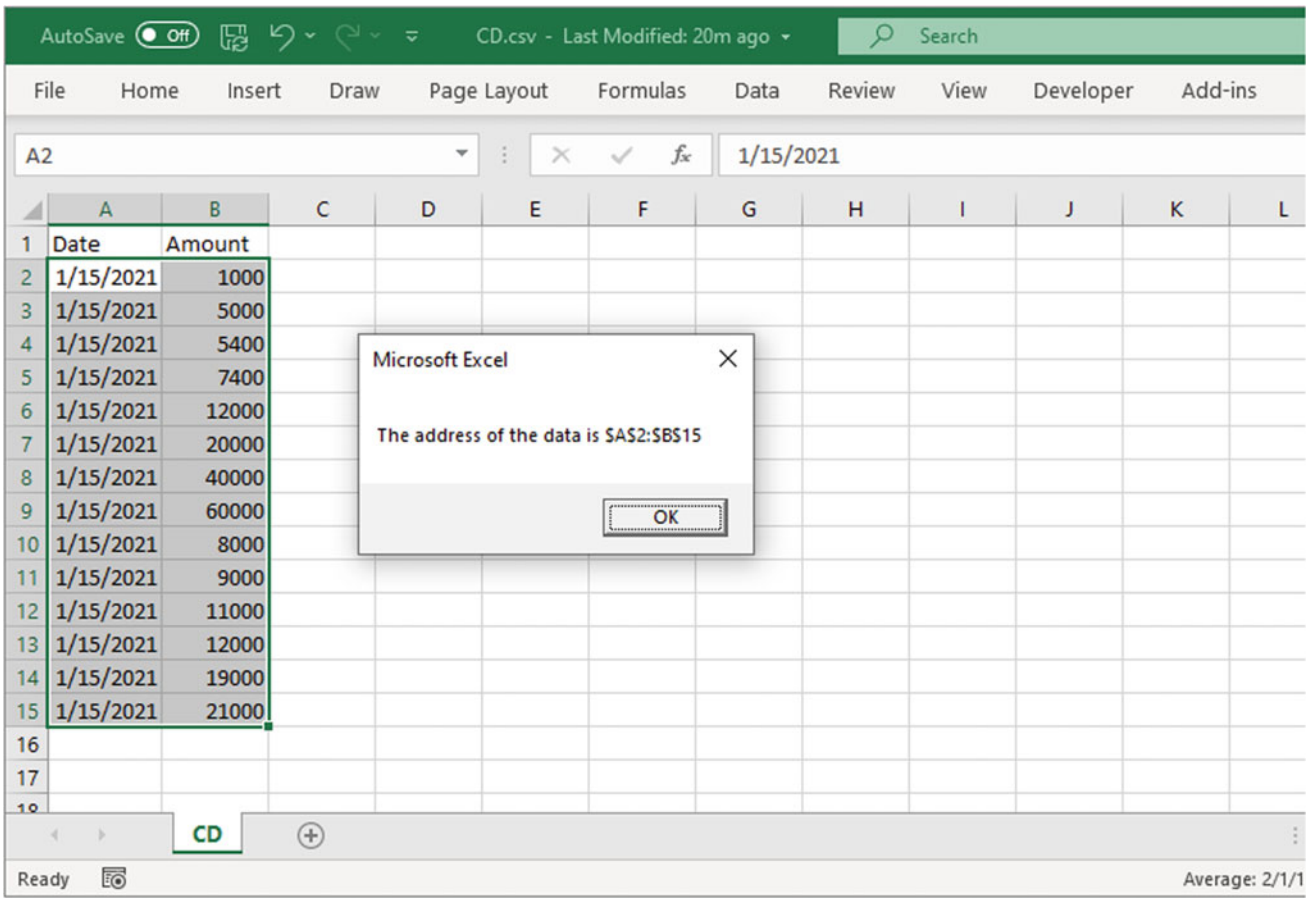
Notice that when we used the *offset* property, we shifted the whole current region by one row. As shown above, offsetting the current region by one row causes the blank row 16 to be included. To solve this problem, we will use the *resize* property of the range object. The *resize* property is discussed in the next section.

4.4 Resize Property of the Range Object

Like the *offset* property, the *resize* property is one of those properties and methods that are usually mentioned in passing in most books.

The *resize* property has two arguments. The first argument is to resize the row to a certain size. The second argument is to resize the column to a certain size. Below is a procedure that illustrates the *resize* property.

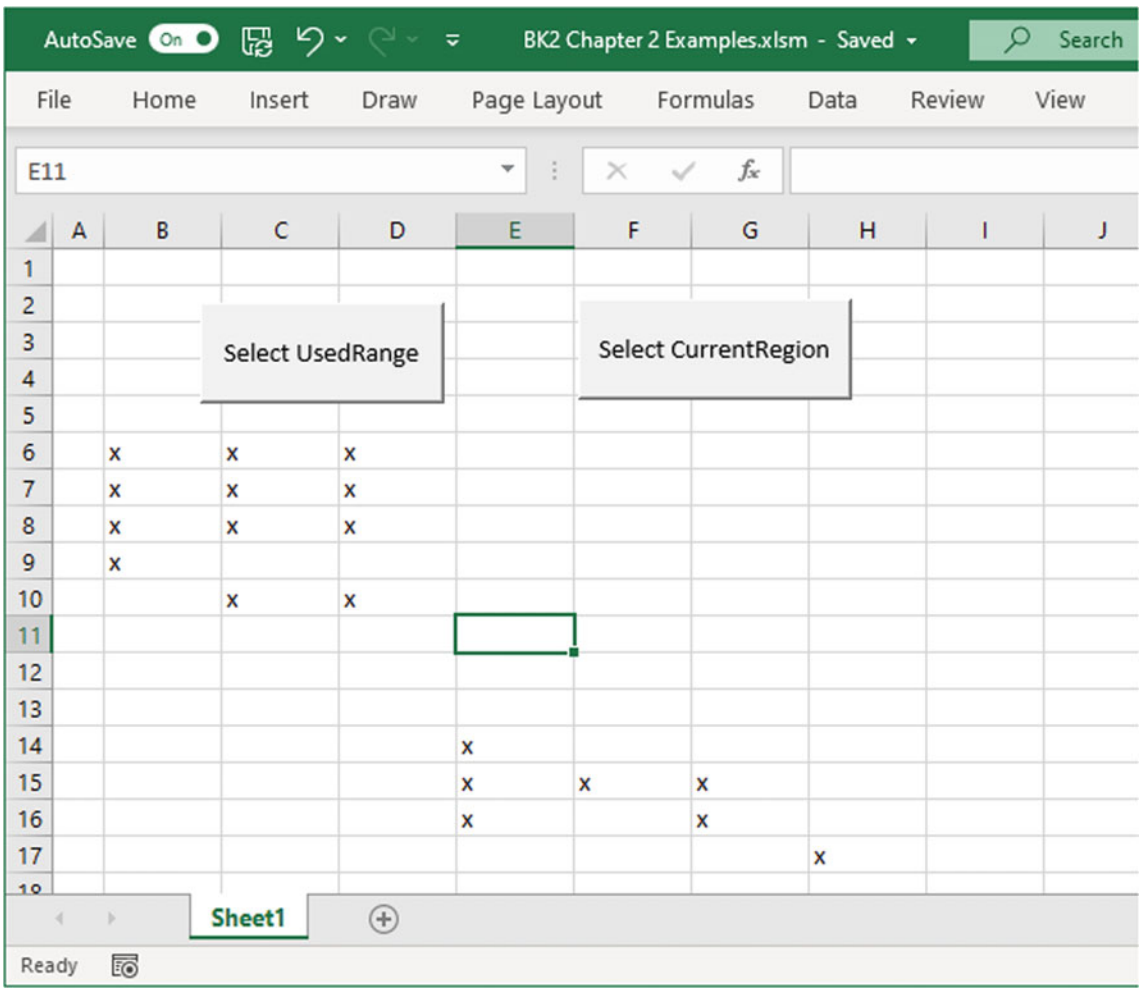
```
'/
**************************************************
*****************************
'/Purpose: To find the data range of an imported file
'/
**************************************************
***************************
Sub CurrentRegionOffsetResize()
Dim rCD As Range
Dim wbCD As Workbook
On Error Resume Next
'Open CD file. It is assumed in same location as this
workbook
Set wbCD = Workbooks.Open(ThisWorkbook.Path & ``\'' &
``CD.csv'')
If wbCD Is Nothing Then
MsgBox ``Could not find the file CD.csv in the path '' _
& ThisWorkbook.Path, vbCritical
End
End If
'Figure out salary range
'CurrentRegion Method will find row and columns that are
completely
'surronded by blank cells
Set rCD = ActiveSheet.Cells(1).CurrentRegion
'Offset the current region by one row.
'The offset property has row offset argument and column
offset argument
Set rCD = rCD.Offset(rowoffset:=1, columnoffset:=0)
'resize the range by the amount previous rows -1
'resize the columns to same number columns as previ-
ously
Set rCD = rCD.Resize(rowsize:=rCD.Rows.Count—1,
columnsize:=rCD.Columns.Count)
rCD.Select
MsgBox ``The address of the data is '' & rCD.Address
wbCD.Close False
End Sub
```



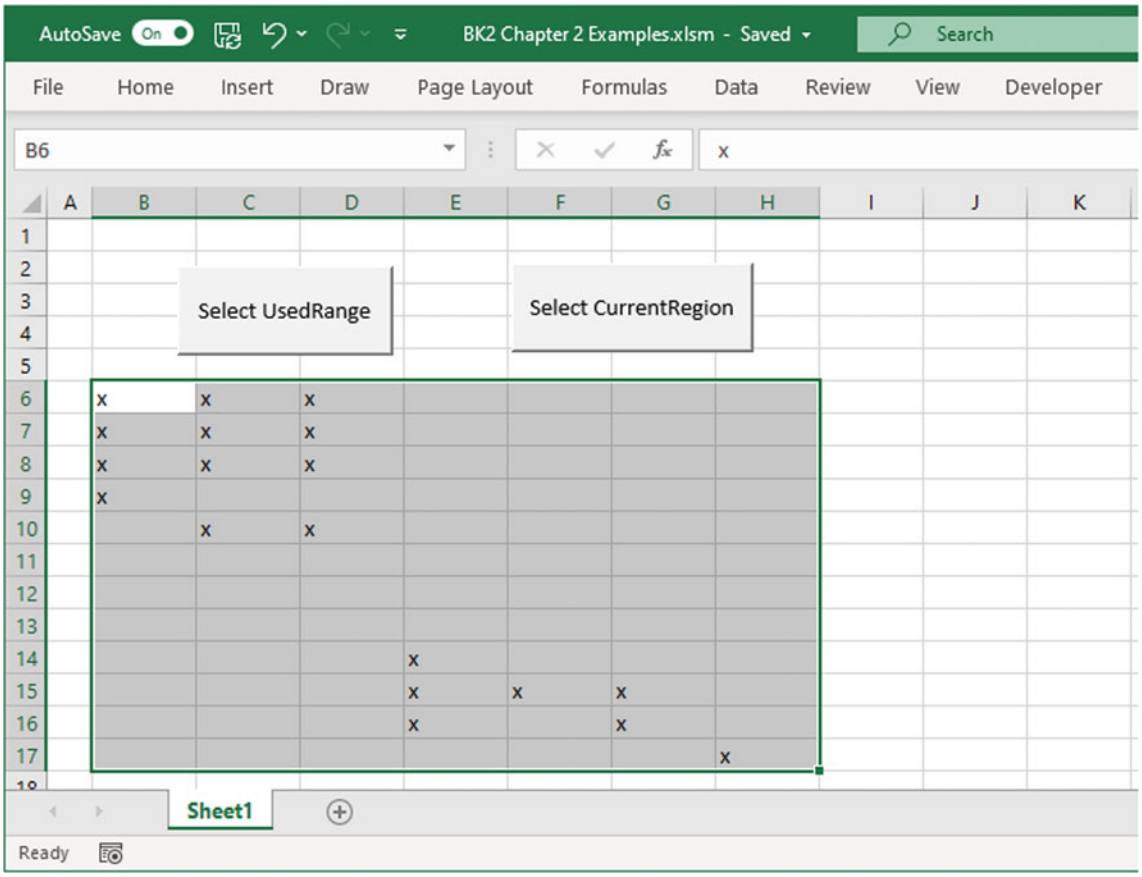
Notice that the current region of the table now only contains the data row. It does not contain the header and blank rows.

4.5 UsedRange Property of the Range Object

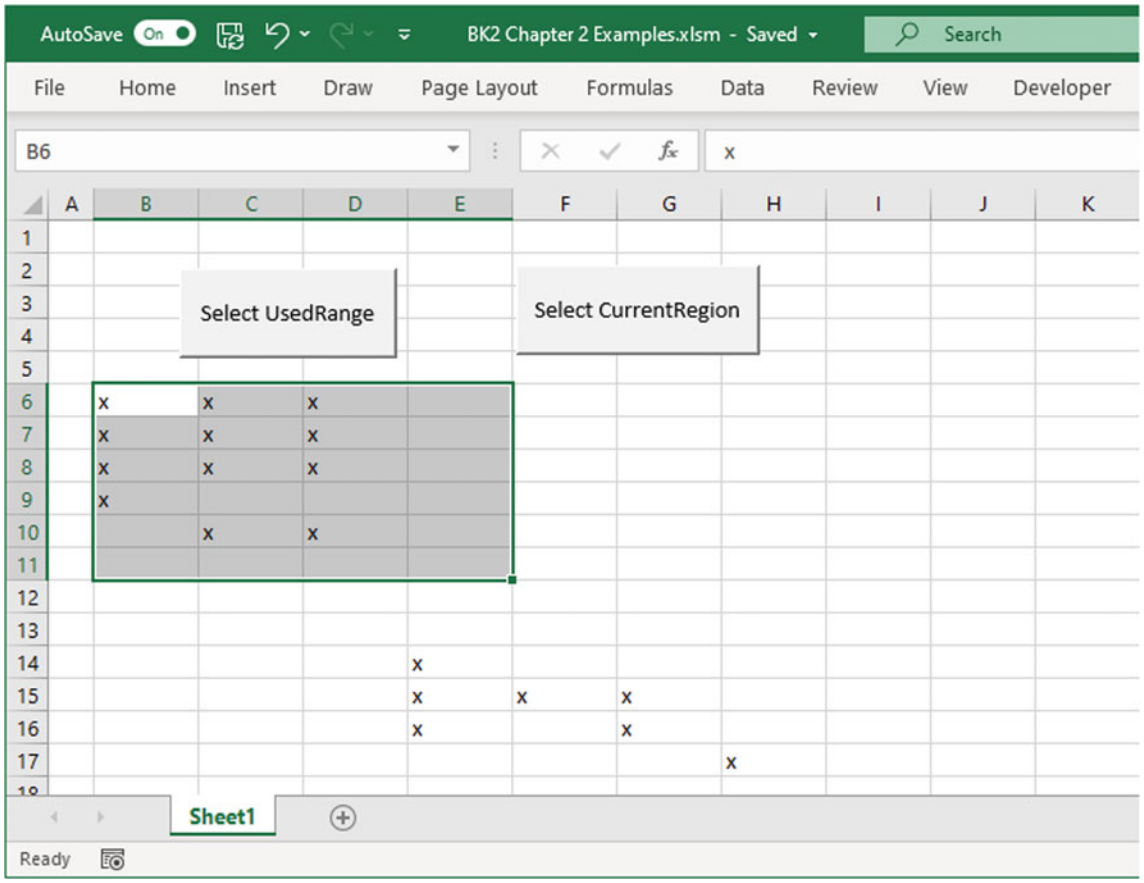
Another useful property to know is the *UsedRange* property of the Range object. The VBA Help file defines the usedrange as the used range of a specific worksheet. Below demonstrates the difference between the *usedrange* and the *currentregion*. To demonstrate both concepts, let's first select cell E11, as shown below.



Below shows what happens after pushing the *Select UsedRange* button.



Below shows what happens after pushing the *Select CurrentRegion* button.



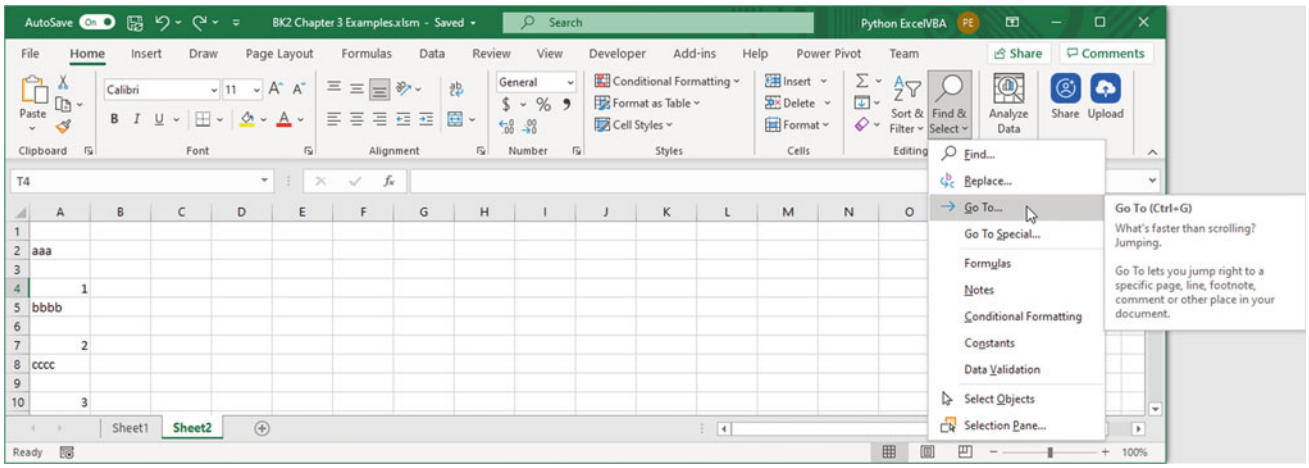
To understand the difference between the *usedrange* and the *currntregion*, it is important to know how the help file defines the *currentregion*. The VBA Help file defines the *currentregion* as "a range bounded by any combination of blank rows and blank columns."

Below are the procedures to find the used range and the current region range.

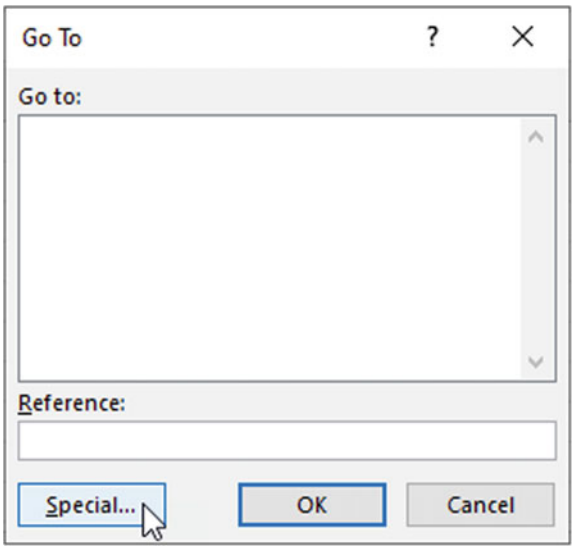
```
Sub FindUsedRange()
ActiveCell.Parent.UsedRange.Select
End Sub
Sub FindCurrentRegion()
ActiveCell.CurrentRegion.Select
End Sub
```

4.6 Go to Special Dialog Box of Excel

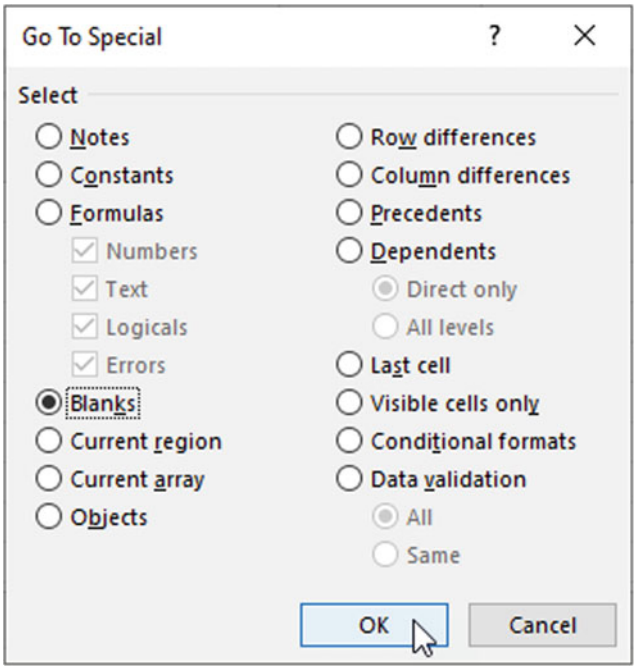
As we have seen so far, navigating around an Excel worksheet is very important. A tool to help navigate around an Excel worksheet is the *Go To Special* dialog box. To get to the *Go To Special* dialog box, we would need first to choose Home ➜ Find & Select ➜ Go To as shown below or press the *F5* key on the keyboard.



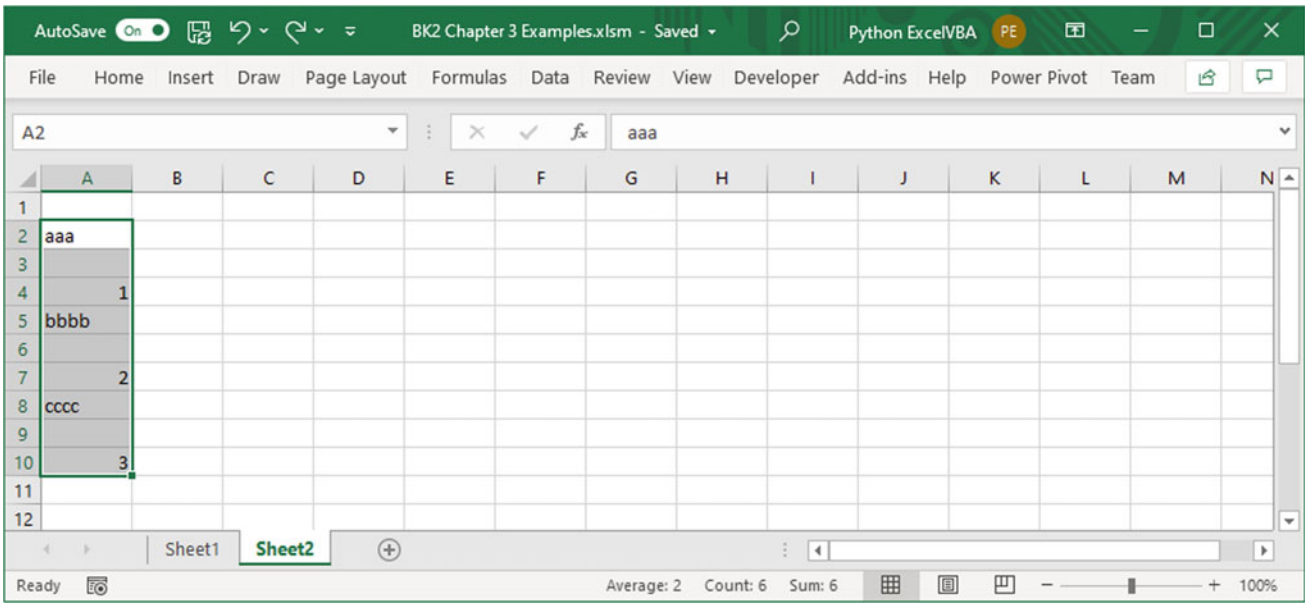
Doing this will show the *Go To* dialog box as shown below.



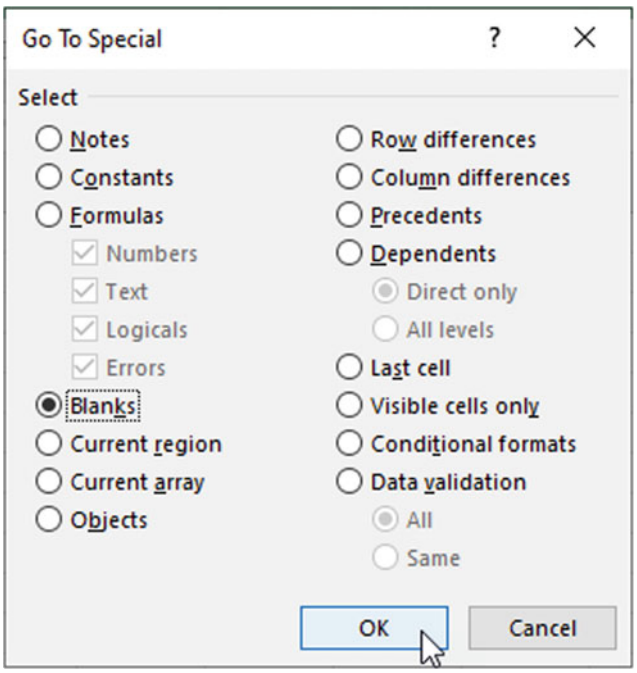
Next, press the *Special* button as shown above to get the *Go To Special* button shown below.



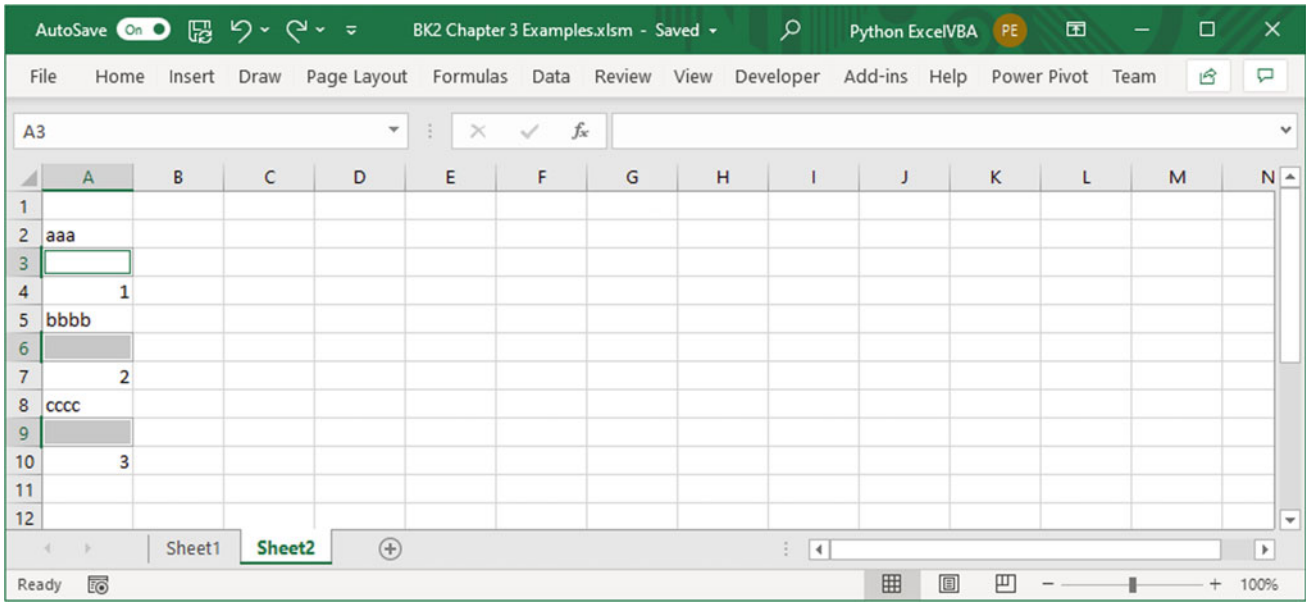
Below illustrates the *Go To Special* dialog box. Suppose we are interested in finding the blank cells inside the selected range shown below.



To find the blank cells, we would go to the *Go To Special* dialog box and then choose the *Blanks* options as shown below.

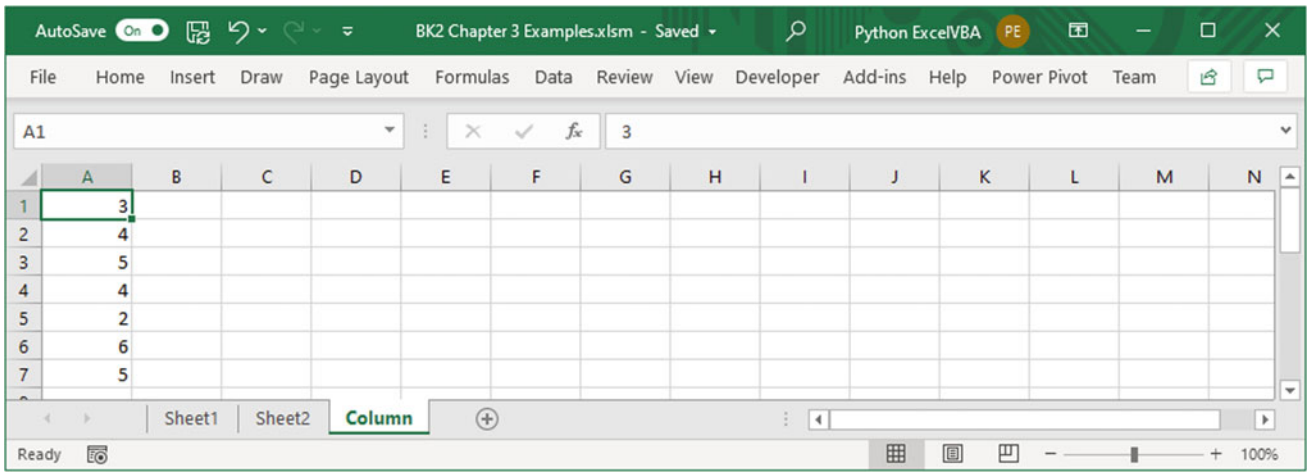


The following is the result after pressing the *OK* button on the *Go To Special* dialog box.



4.7 Importing Column Data into Arrays

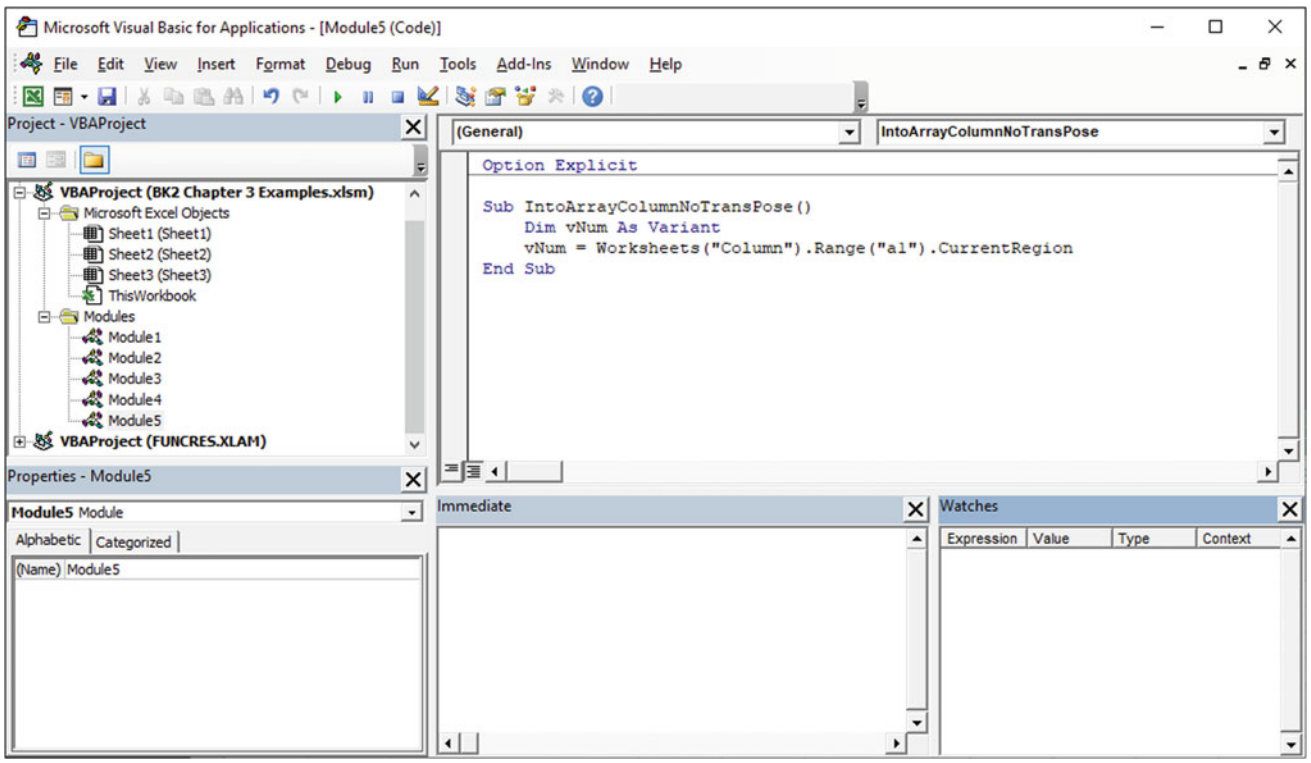
Many times we are interested in importing data into arrays. The main reason to do this is speed. When the dataset is large, there is a noticeable difference between manipulating data in arrays versus manipulating arrays in a worksheet. One way to get data into an array is to loop through every cell and put each data element individually into an array. The other way to get data into an array is shown below.



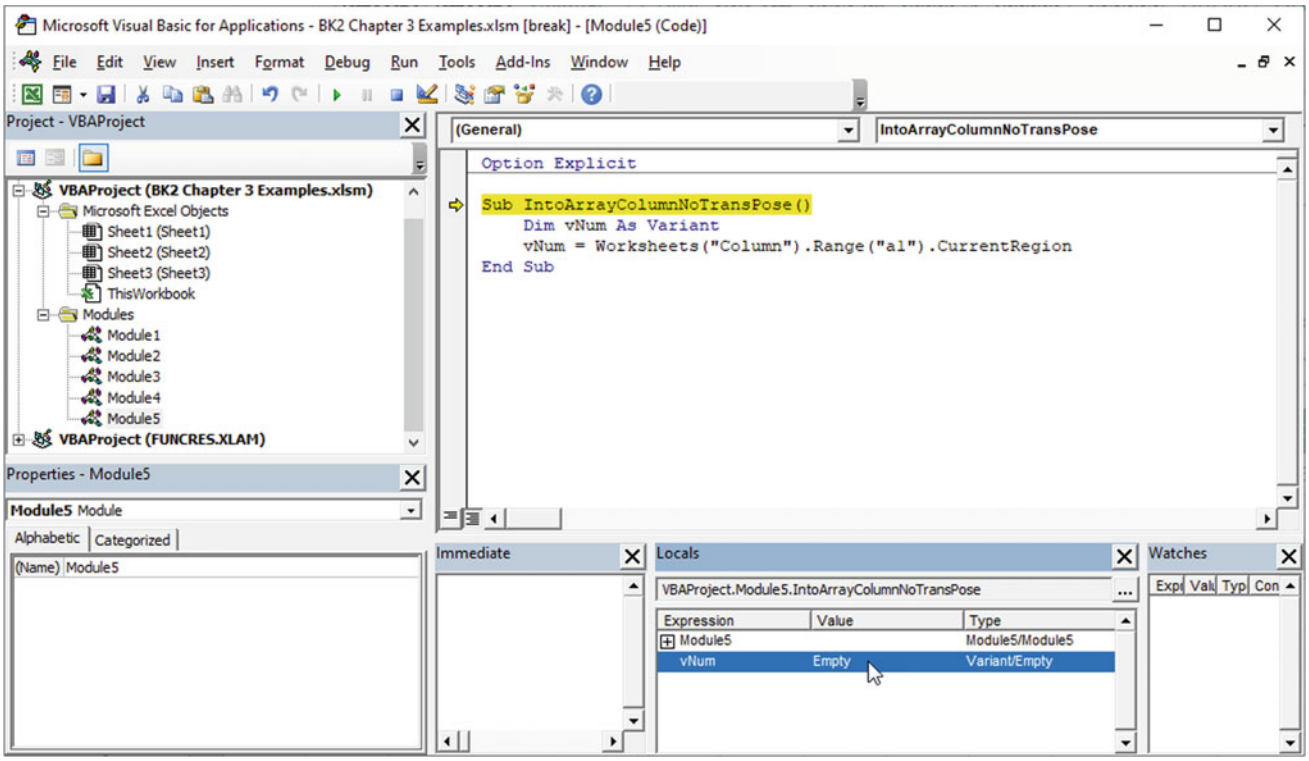
```
Sub IntoArrayColumnNoTransPose()
Dim vNum As Variant
vNum    =    Worksheets(``Column'').Range(``a1'').Cur-
rentRegion
End Sub
```

Notice that, in the above procedure, it requires only one line of VBA code to bring data into an array from a worksheet. It is important to note that for the above technique to work, the array variable "vNum" must be defined as a variant.

To illustrate that the above technique works, we will have to use the professional programming tools provided by Excel. The tools are in the Visual Basic Editor. The Visual Basic Editor is shown below.

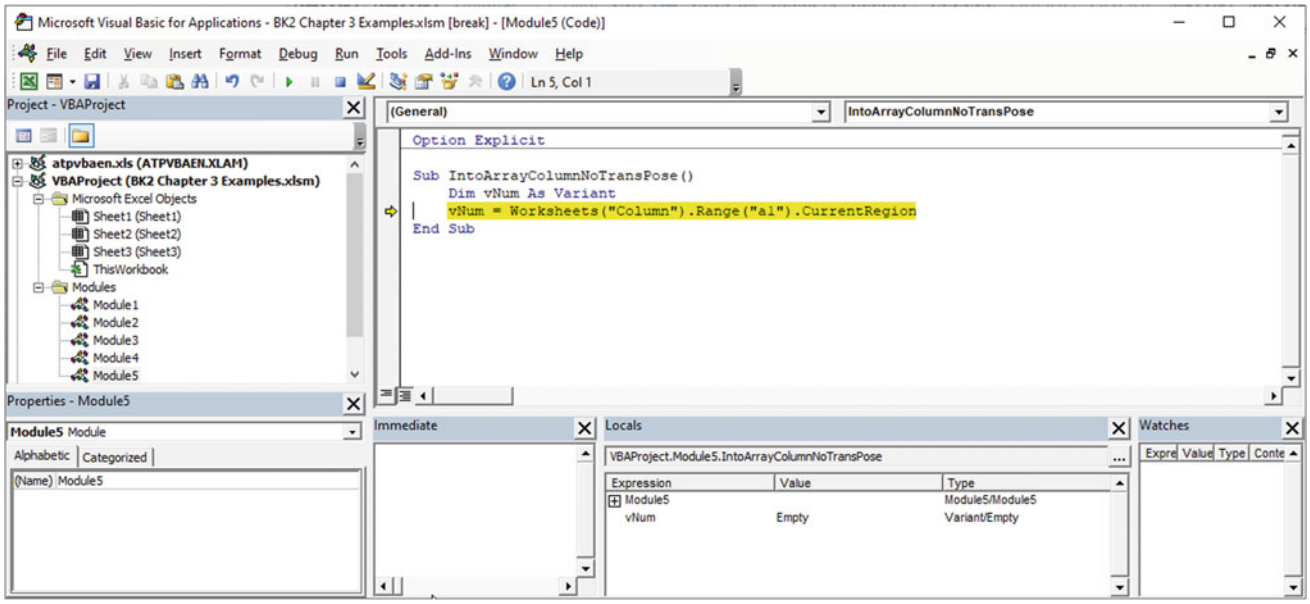


To illustrate the technique discussed in this section, we will need to run the VBA code in the procedure "IntoArrayColumnNoTranspose" one line at a time and look at the value of variables after each line. To do this, we will need first to put the cursor on the first line of the procedure, as shown above. Then we will need to press the *F8* key on the keyboard. Doing this will result in the following:

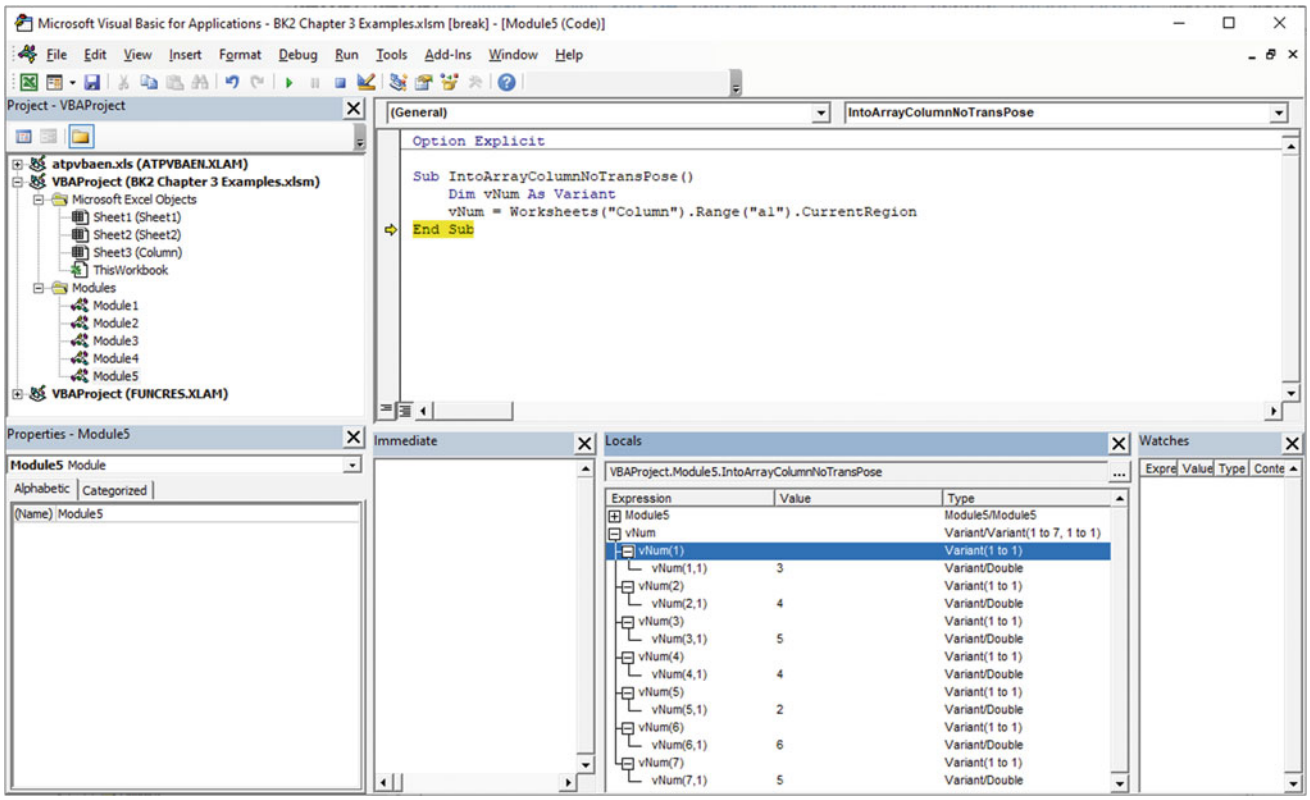


The first thing to notice is that the first line of the procedure is highlighted in yellow in the code window. The yellow highlighted line is shown above. The other thing to note is the *Locals* window. It shows the value of all the variables. At this point, it indicates that the variable "vNum" has an empty value, which means no value.

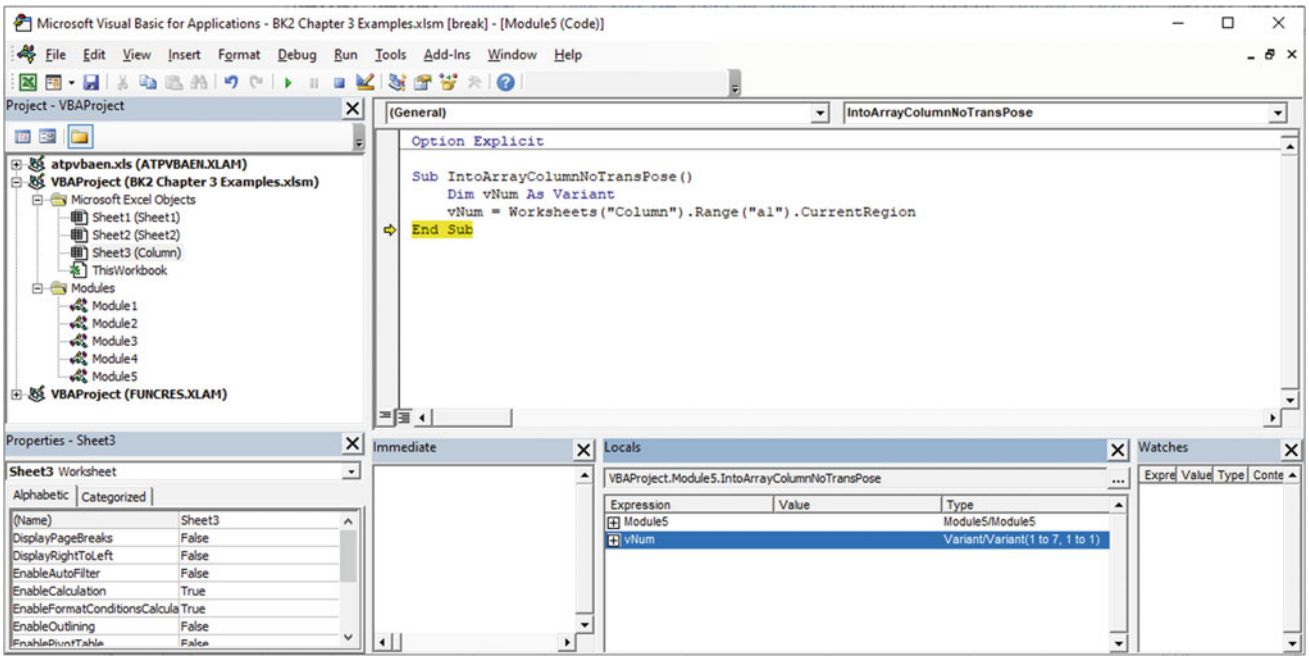
The next thing that we need to do now is to press the *F8* key on the keyboard to move to the next VBA line. Below shows what happens after pressing the *F8* key.



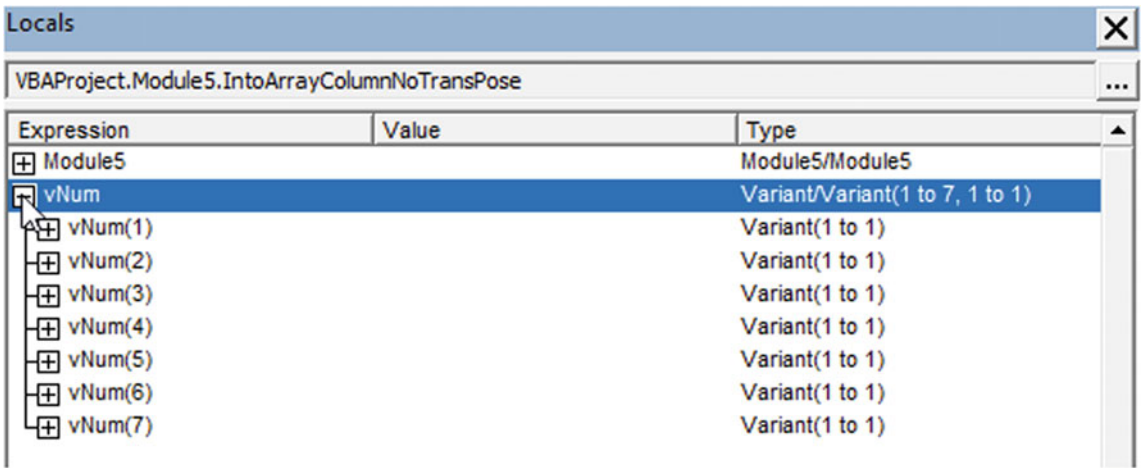
Notice at this point the variable "vNum" still has no value. Let's press the *F8* key one more time.



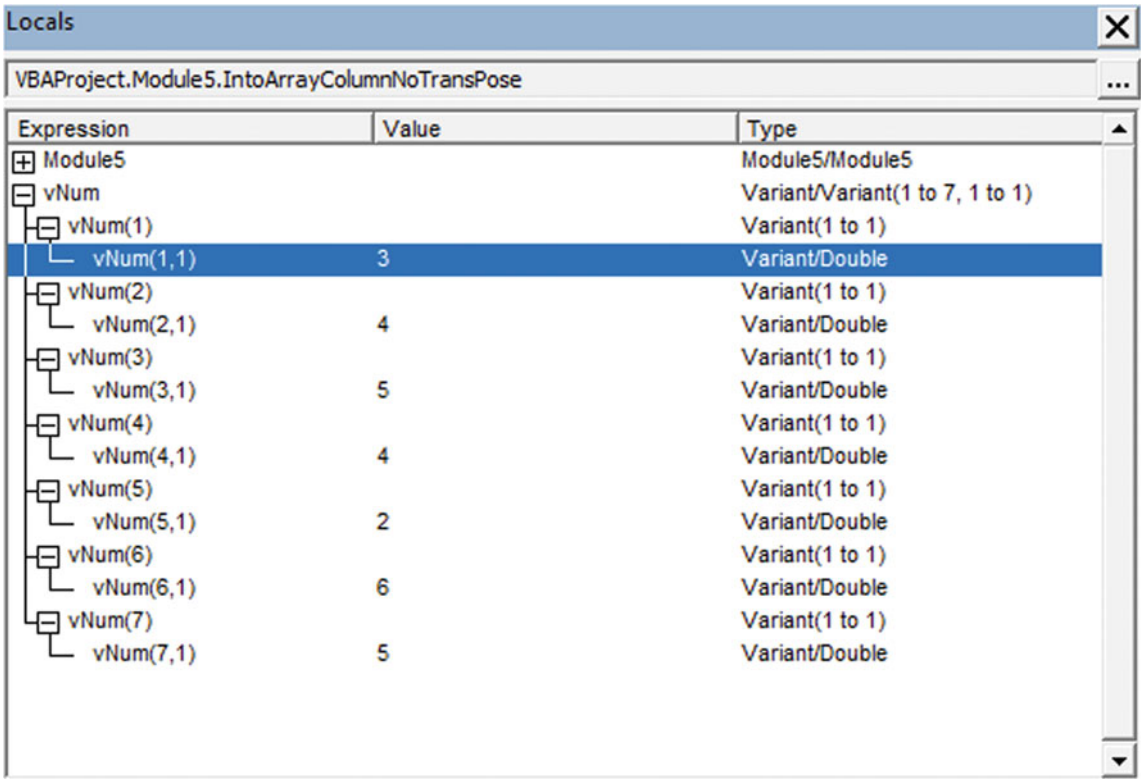
Notice at this point the variable "vNum" no longer indicates empty. There is also a symbol next to the variable. This symbol indicates that there are values for the array. We will need to click on the symbol to look at the values of the array. The following shows the result of clicking on the symbol:



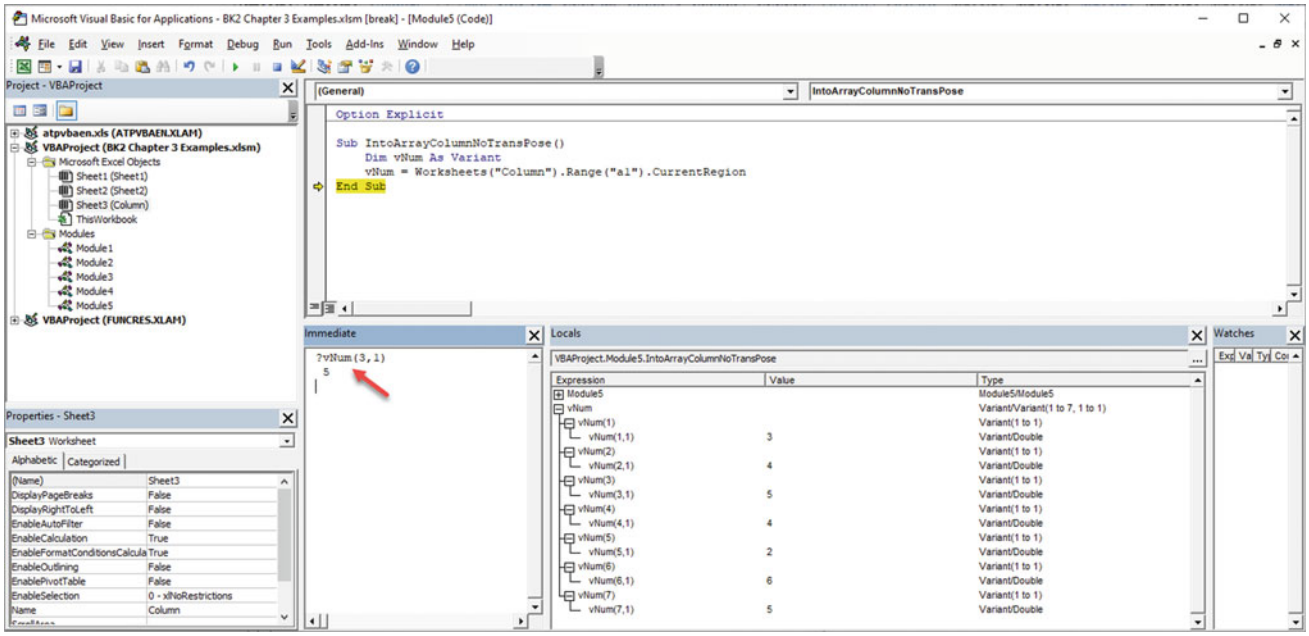
Notice at this point the variable "vNum" no longer indicates empty. There is also a symbol next to the variable. This symbol indicates that there are values for the array. We will need to click on the plus sign next to *vNum* to look at the array's values. The following shows the result of clicking on the plus sign:



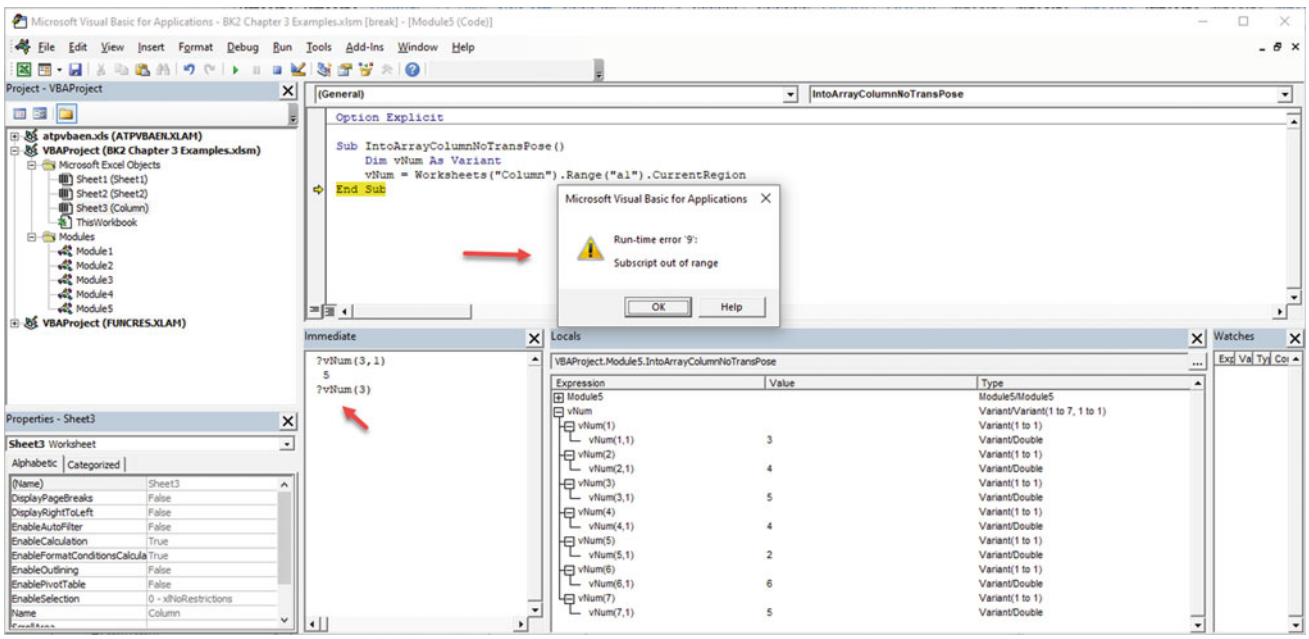
The *Locals* window now shows that there are seven elements in the array "vNum." Let's now click on each element of the array. The end result is shown below.



The *Locals* window indicates that the first element of the array is 3. The values of the rest of the elements agree with the values in the worksheet. Note that in the *Locals* window, the third element has a reference of "vNum(3,1)." This reference indicates that VBA has automatically set the variable "vNum" to a two-dimensional array. So to reference the third element, we will need to indicate "vNum(3,1)" and not "vNum(3)." This reference can be illustrated with the *Immediate* window of the Visual Basic Editor. Below shows in the *Immediate* window the value of the array element "vNum(3,1)."



Below shows what happens when we try to reference the third element as "vNum(3)." The Visual Basic Editor complains when we try to reference the third element as "vNum(3)."

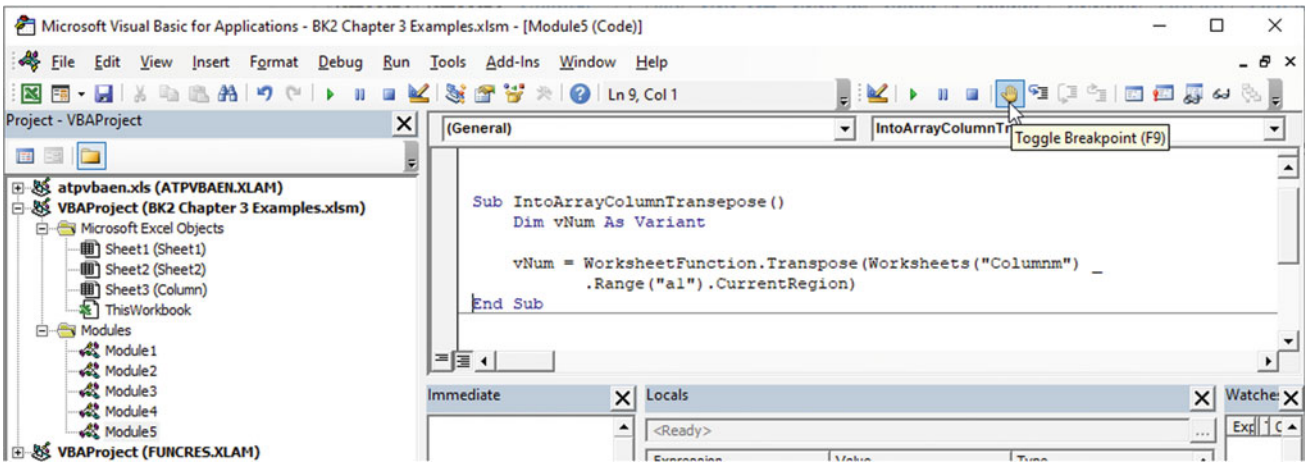


Many times we are interested in the variable being a one-dimensional array. To do this, we will use

the *Transpose* method of the *worksheetufnciton* object to create a one-dimensional array. The procedure "IntoArrayColumnTranspose," shown below, accomplishes this.

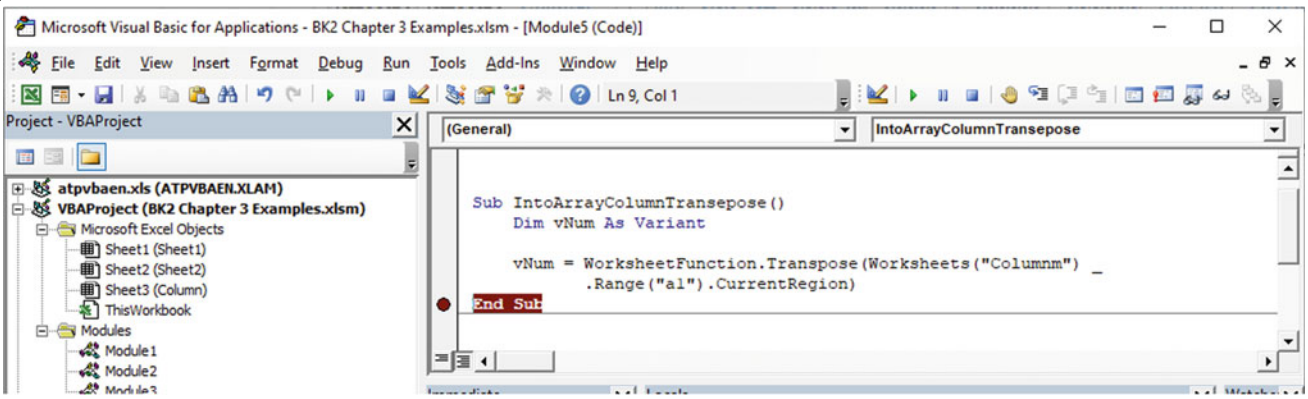
```
Sub IntoArrayColumnTransepose()
Dim vNum As Variant
vNum = WorksheetFunction.Transpose(Worksheets(``Col-
umn'') _
.Range(``a1'').CurrentRegion)
End Sub
```

Instead of stepping through the code line by line, we can tell the VBE to run the VBA code and stop at a certain point. To indicate where to stop, put the cursor at the "end Sub" VBA line as shown below. Then, press the *ToggleBreakPoint* button as shown below or press the *F9* key on the keyboard.

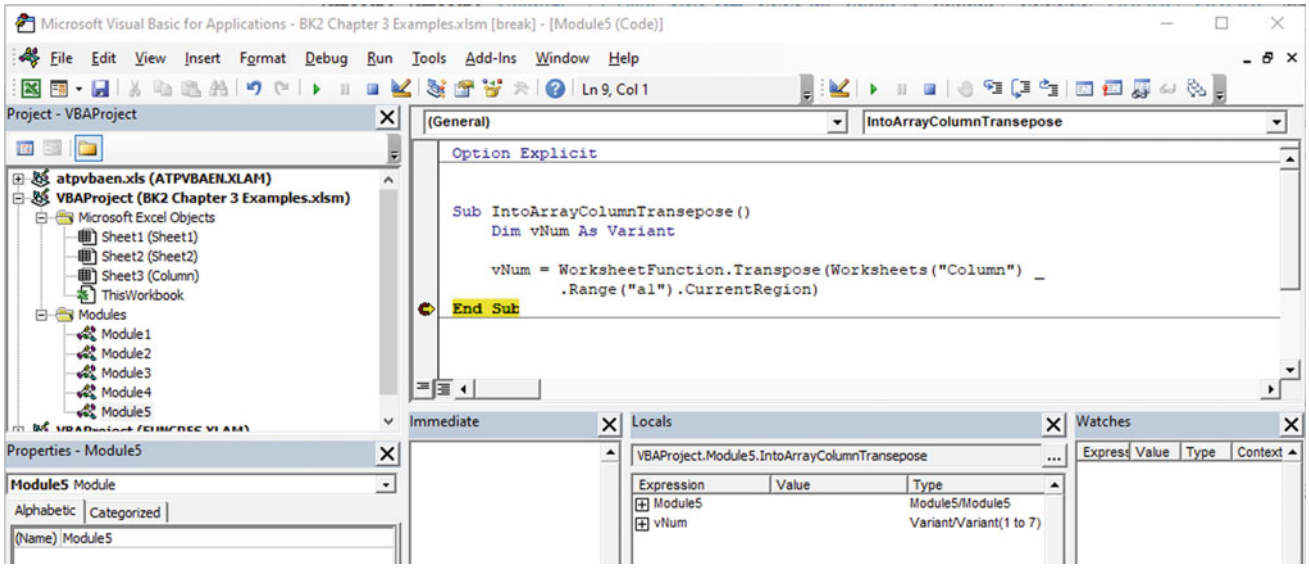


Below shows what happens after pressing the *F9* key

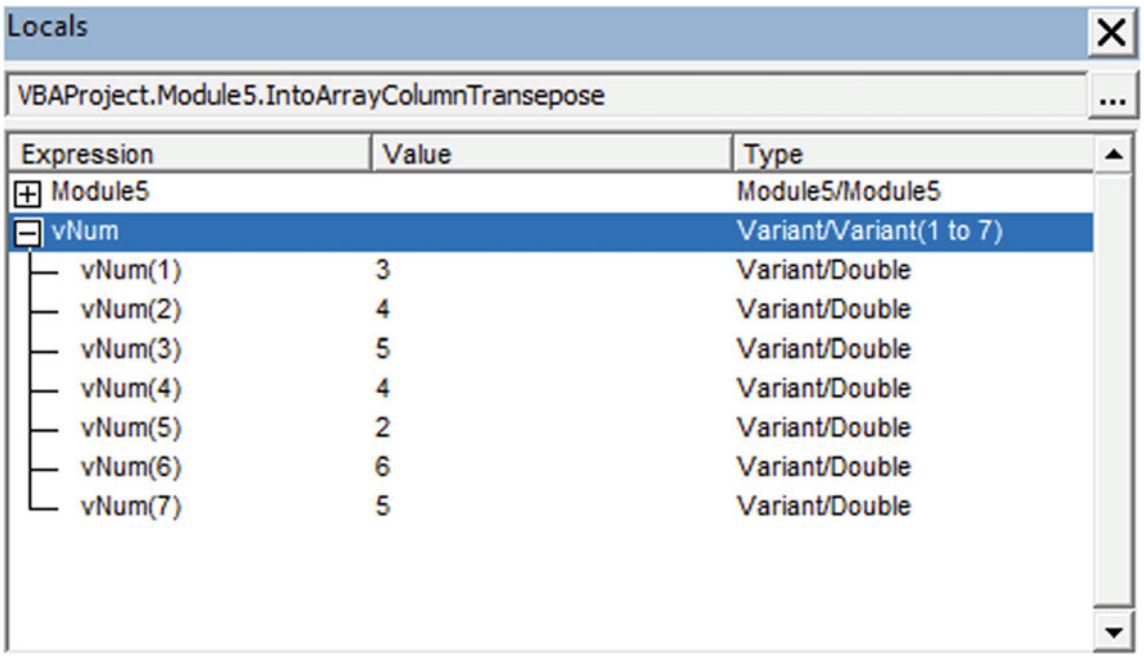
.



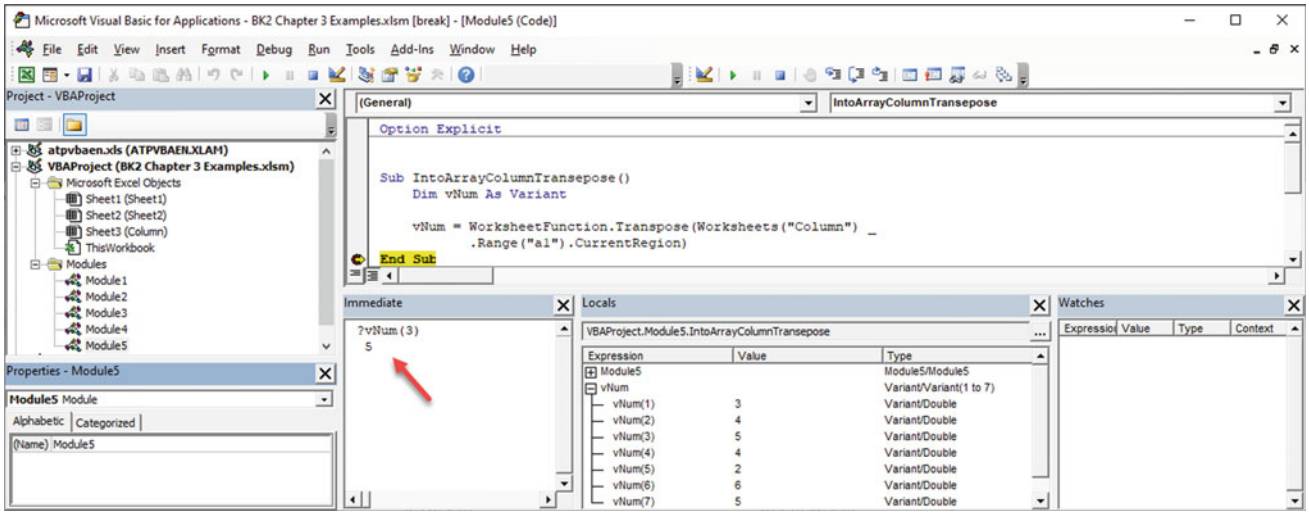
Pressing the *F5* key will run the VBA code until the breakpoint. Below shows the state of the VBE after pressing the *F5* key.



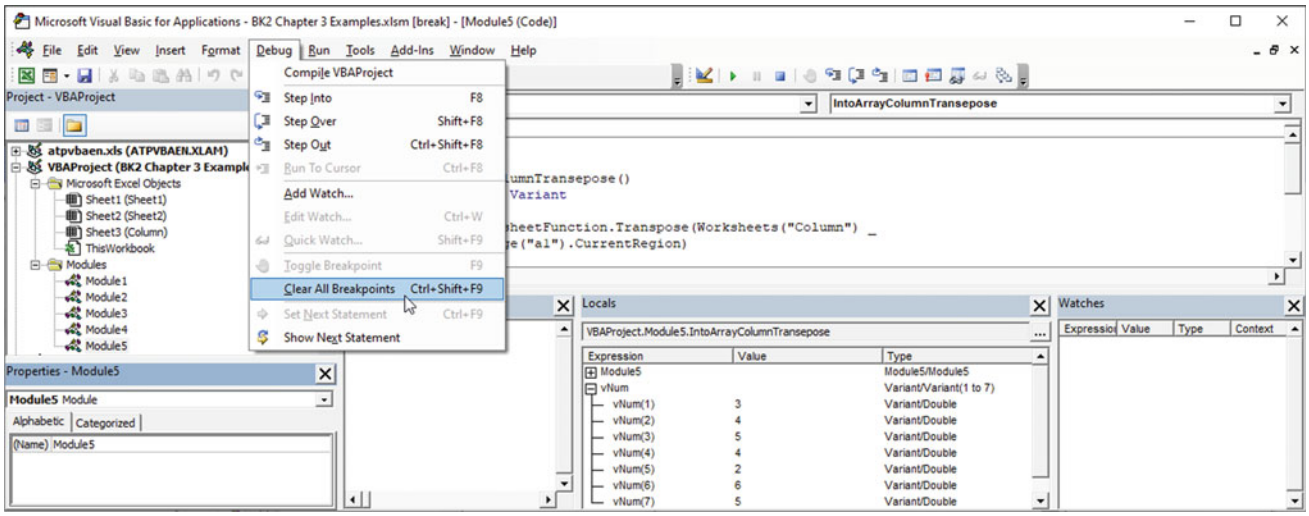
Let's now expand the variable "vNum" in the *Locals* window. Below shows the state of the *Locals* window after expanding the "vNum" variable.



The above *Locals* window shows that the variable "vNum" is one-dimensional. Below shows the *Immediate* pane referencing the third element of the variable "vNum" as a one-dimensional variable.



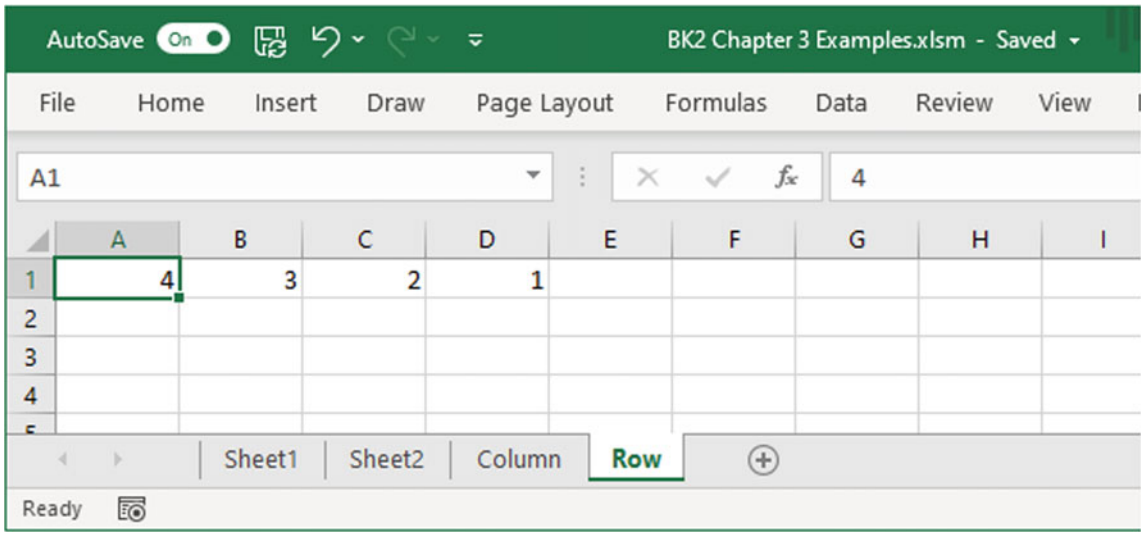
When you are finished analyzing the procedure above, choose Debug ➔ Clear All Breakpoints as shown below. This will clear out all the breakpoints.



Not clearing out the breakpoints will cause the macro to stop at this point after you reopen the workbook and then rerun the macro.

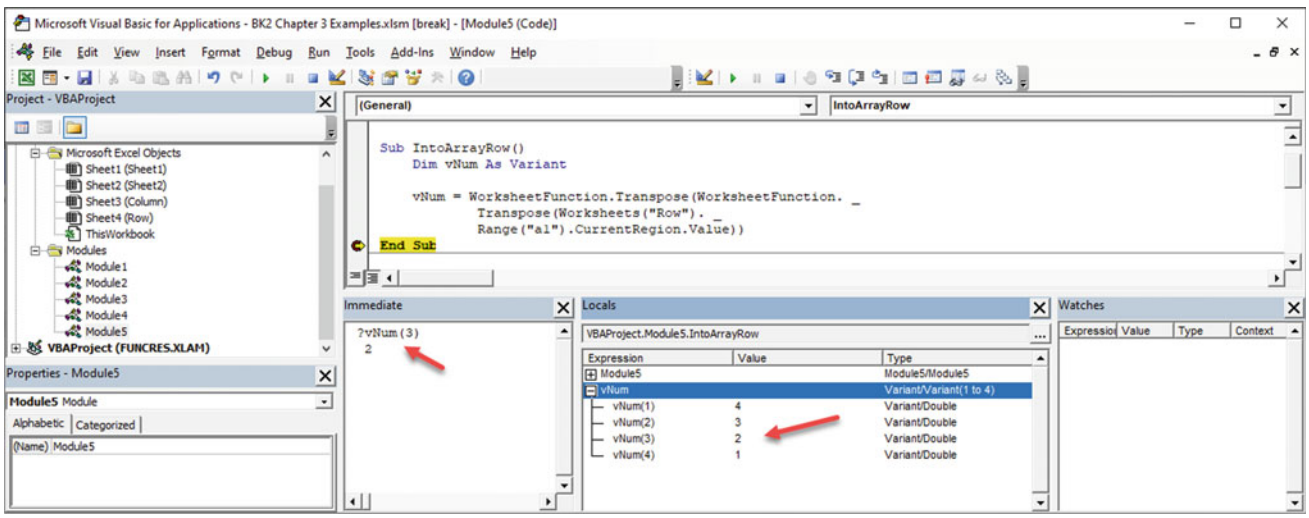
4.8 Importing Row Data into an Array

In the previous section, we used the *Transpose* property (function) to transpose the column data. We need to use the *Transpose* property twice for row data. Let's import the row data shown below to an array.




```
Sub IntoArrayRow()
Dim vNum As Variant
vNum  =  WorksheetFunction.Transpose(WorksheetFunc-
tion. _
Transpose(Worksheets(``Row''). _
Range(``a1'').CurrentRegion.Value))
End Sub
```

Below demonstrates the above procedure.

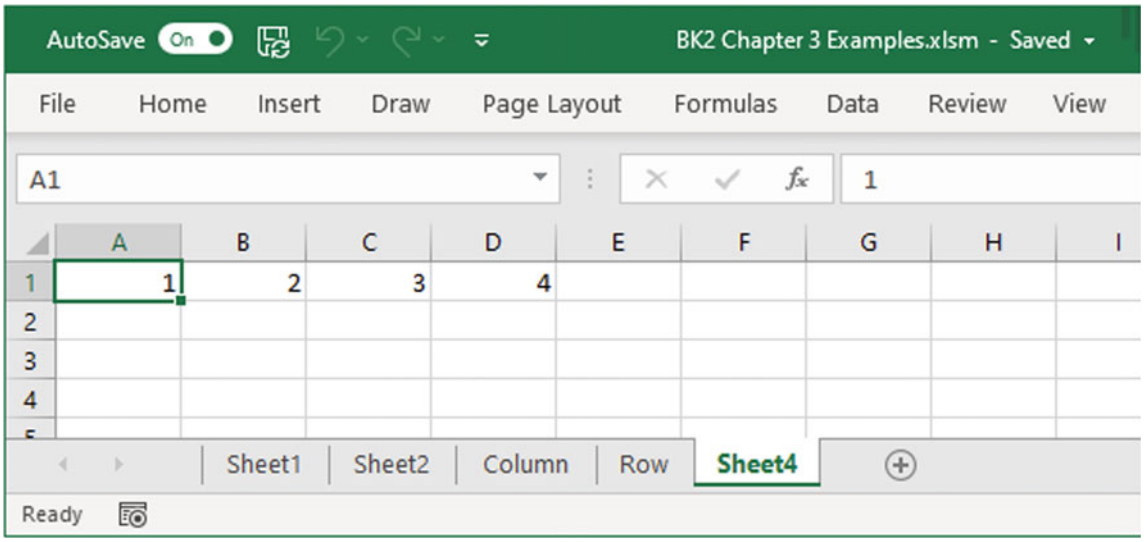


4.9 Transferring Data from an Array to a Range

In this section, we will illustrate how to transfer an array to a range. We will first illustrate how to transfer an array to a row range, and then we will illustrate how to transfer an array to a column range. The following procedure transfers an array to a row range:

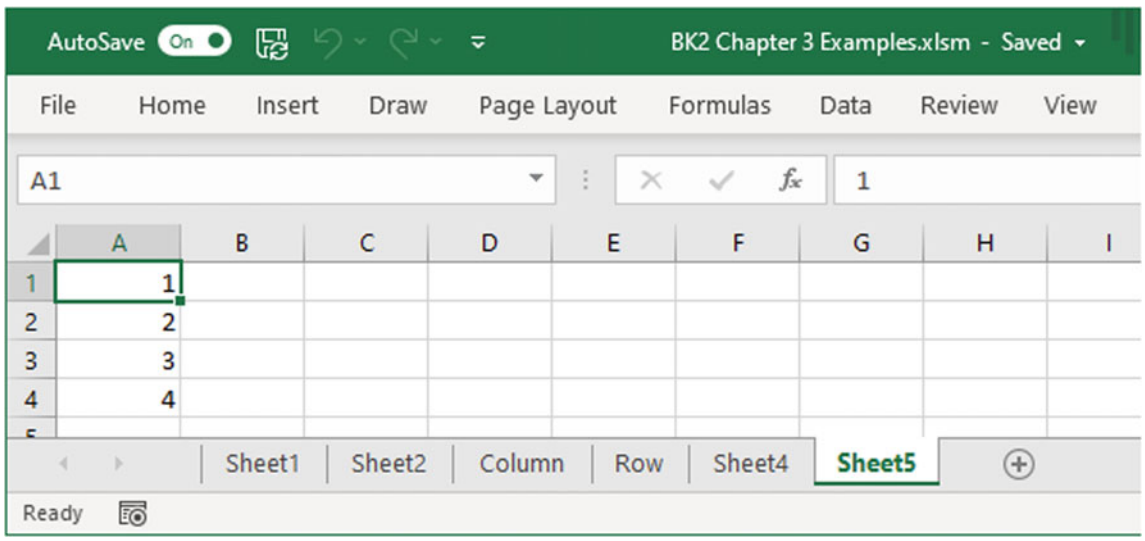
```
Sub TransferToRow()
Dim v As Variant
v = Array(1, 2, 3, 4)
With ActiveSheet.Range(``a1'')
```

```
.CurrentRegion.ClearContents
.Resize(1, 4) = (v)
End With
End Sub
```



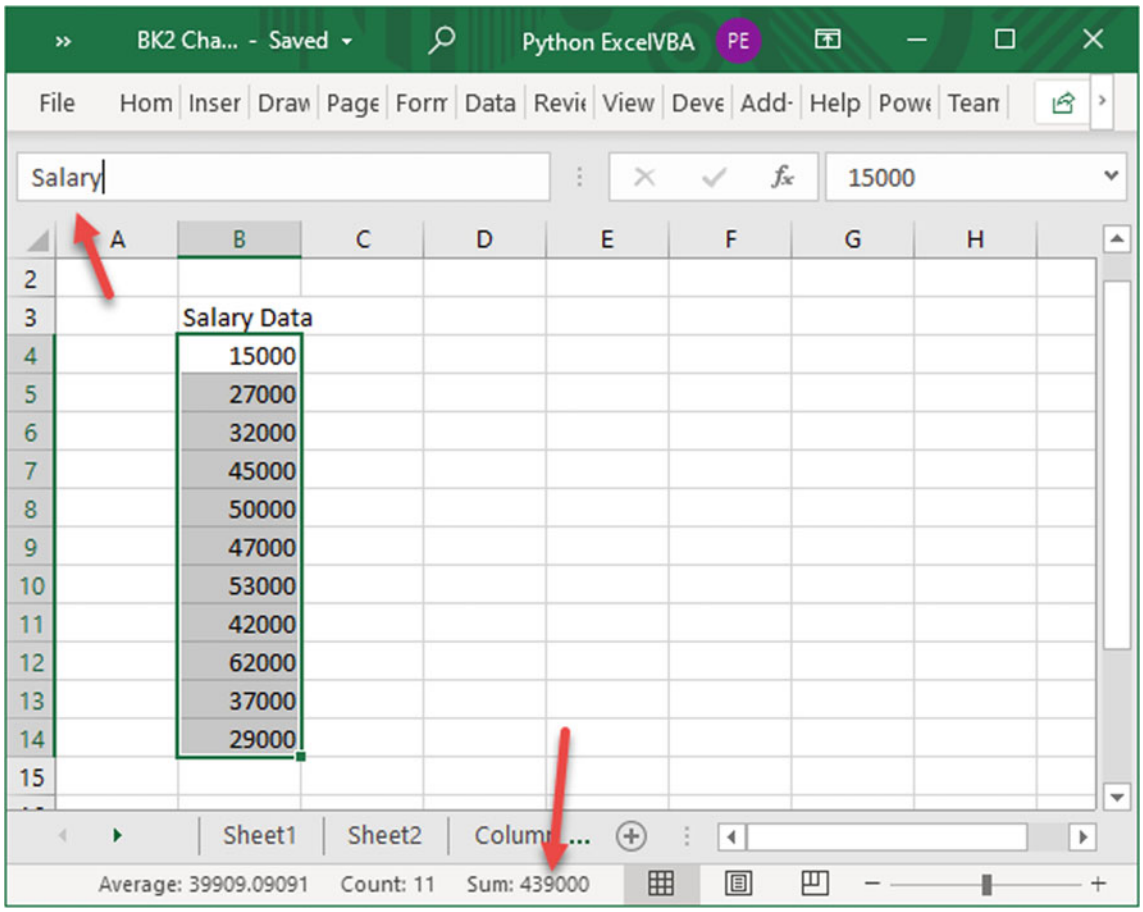
The following procedure transfers an array to a column range:

```
Sub TransferToColumn()
Dim v As Variant
v = Array(1, 2, 3, 4)
With ActiveSheet.Range(``a1'')
.CurrentRegion.ClearContents
.Resize(4, 1) = WorksheetFunction.Transpose(v)
End With
End Sub
```



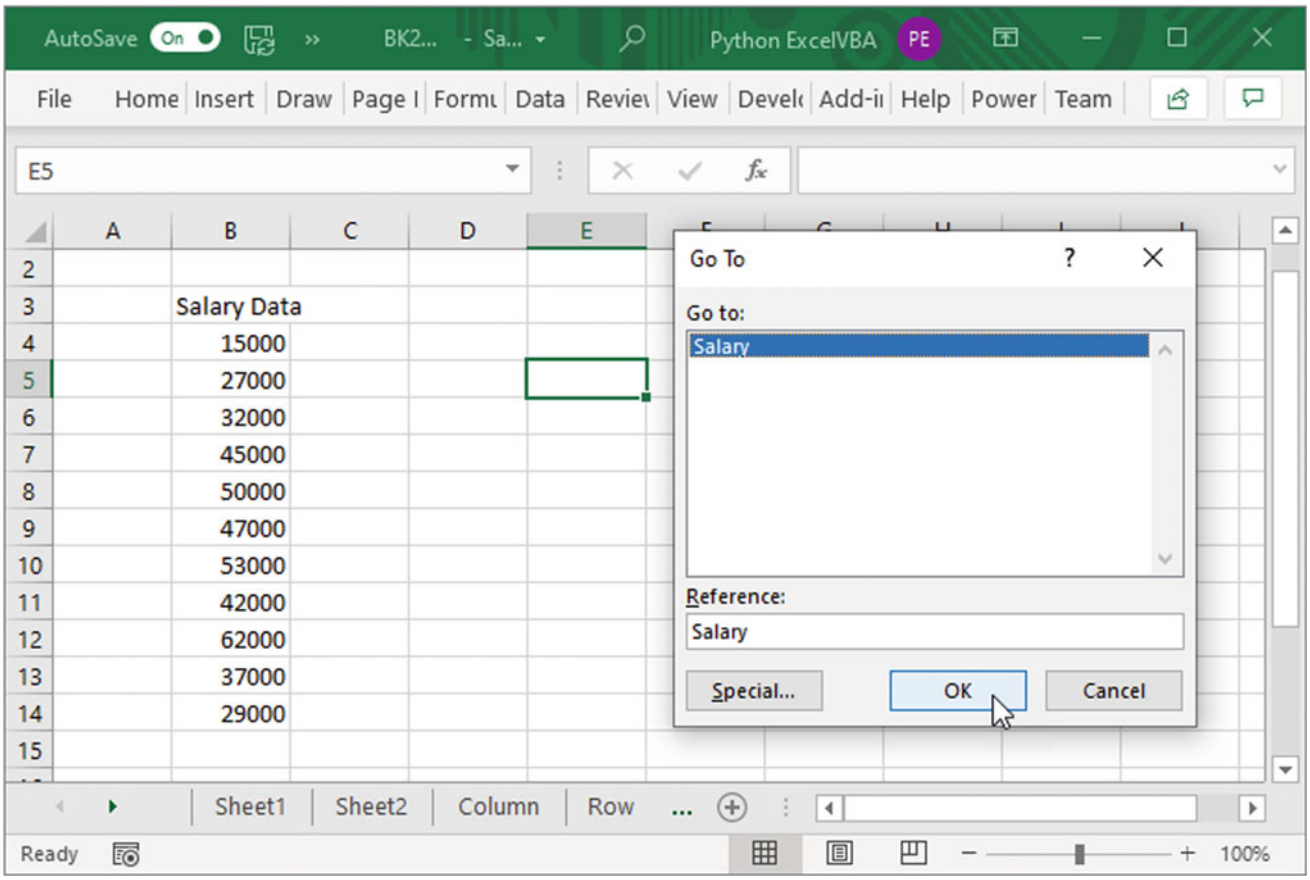
4.10 Workbook Names

We can do a lot of things with workbook names. The first thing that we will do is assign names to worksheet ranges. It is common to set a range name by first selecting the range and then typing a name in the *Name Box*. This is shown below:

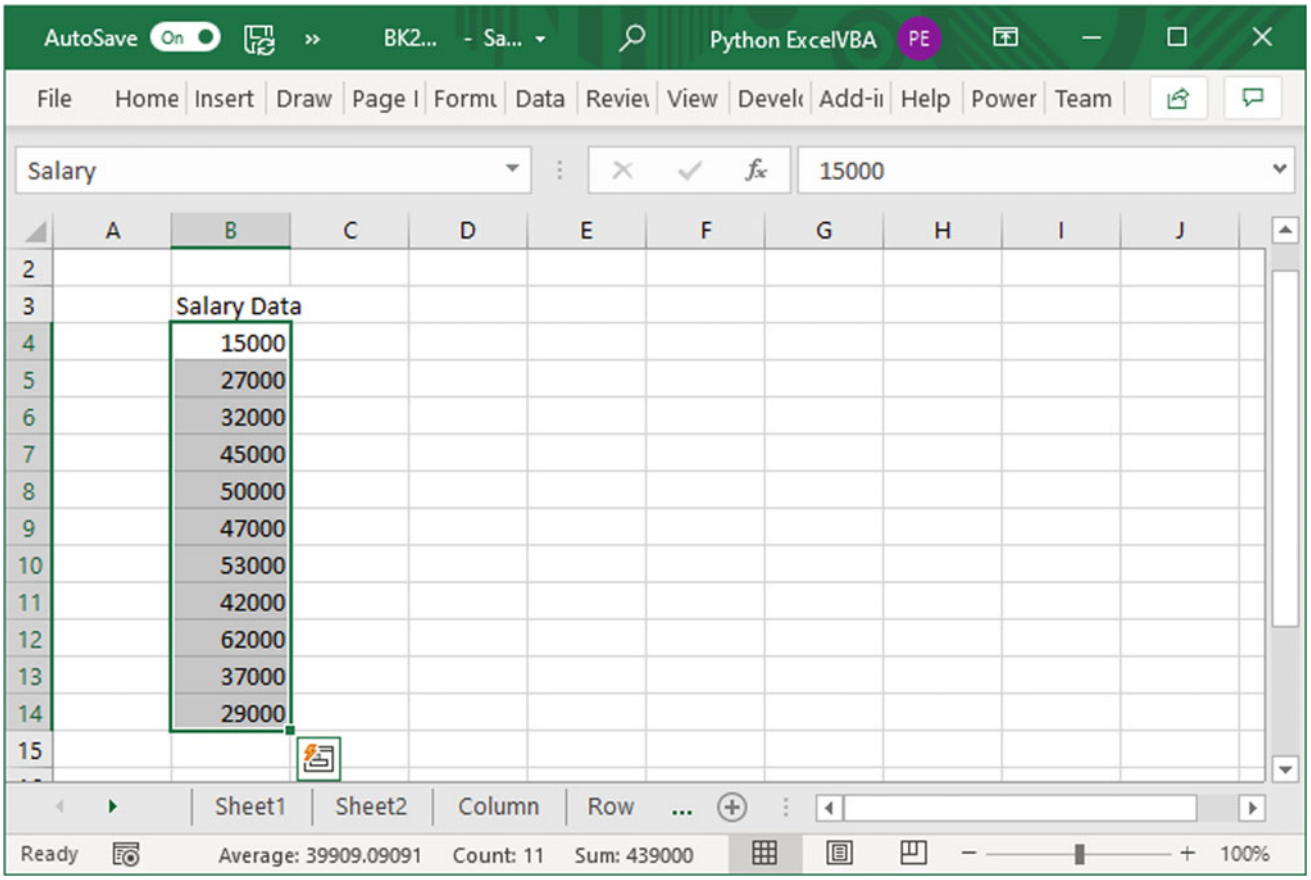


Notice as shown above that Excel will automatically sum any range selected.

One thing that can be done with workbook names is range navigation. As an illustration, let's choose cell E5 as shown below, and then press the *F5* key.



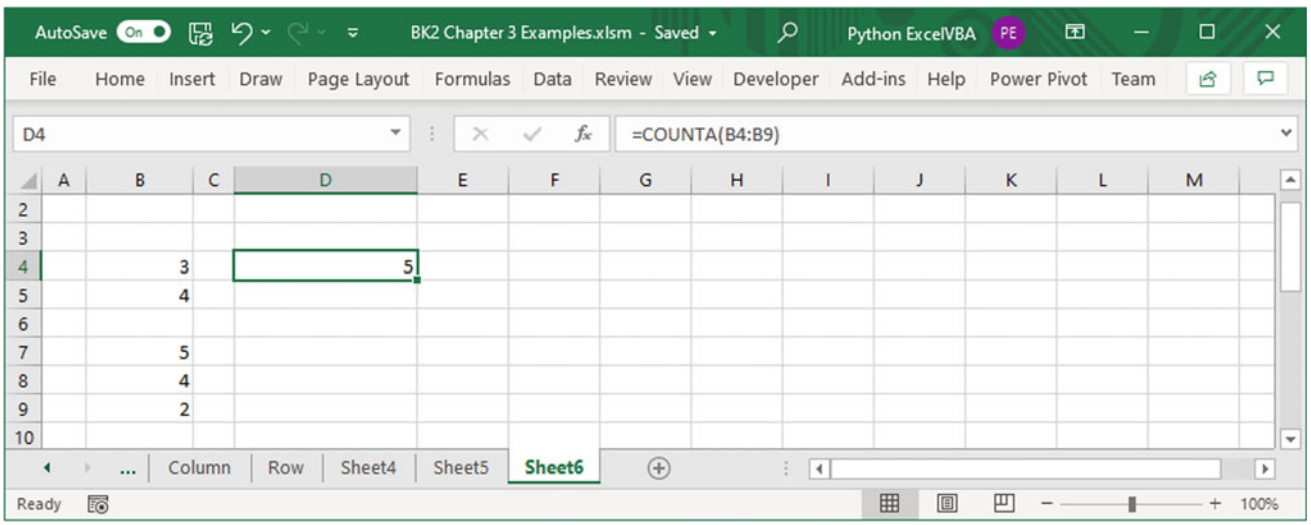
Notice that the *Go To* dialog box shows all workbook names. The next thing that we should do is highlight the *Salary* range and press the *OK* button as shown above. Pressing the *OK* button caused Excel to select the *Salary* range as shown below.



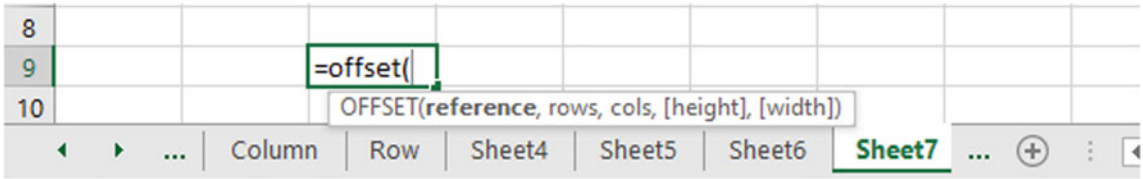
4.11 Dynamic Range Names

In this section, we will illustrate how to create dynamic range names. Dynamic range names use the worksheet function *counta* and the worksheet function *offset*.

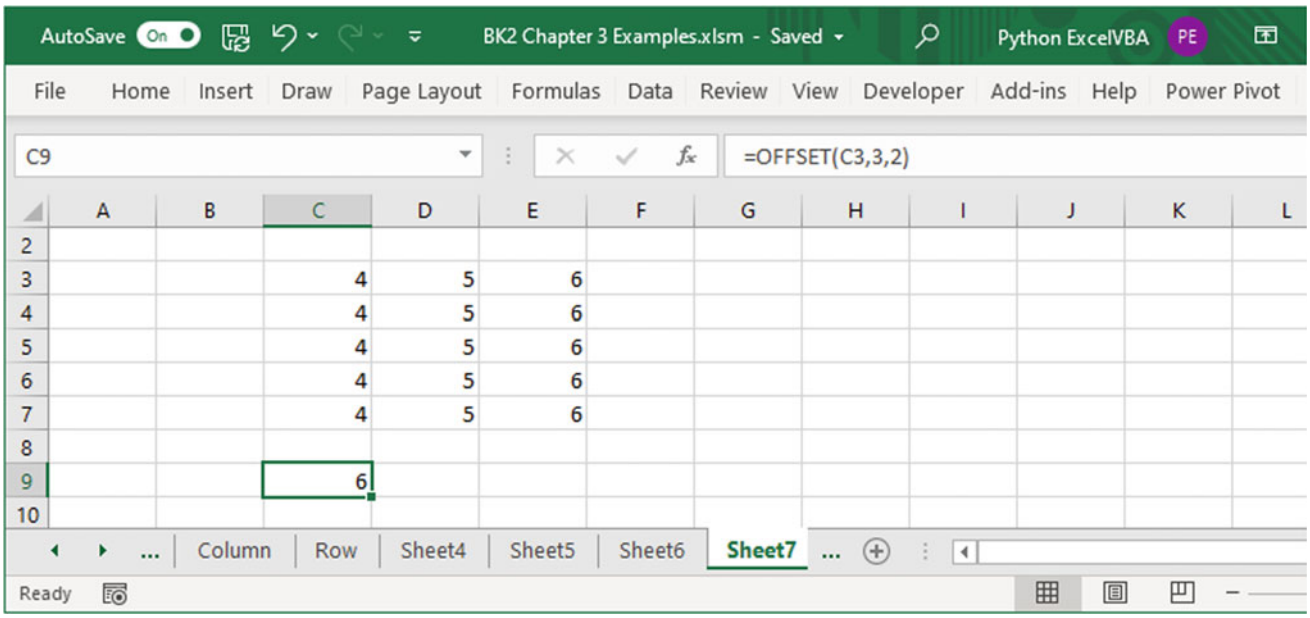
The function *counta* counts the number of cells that are not empty. This concept is illustrated below.



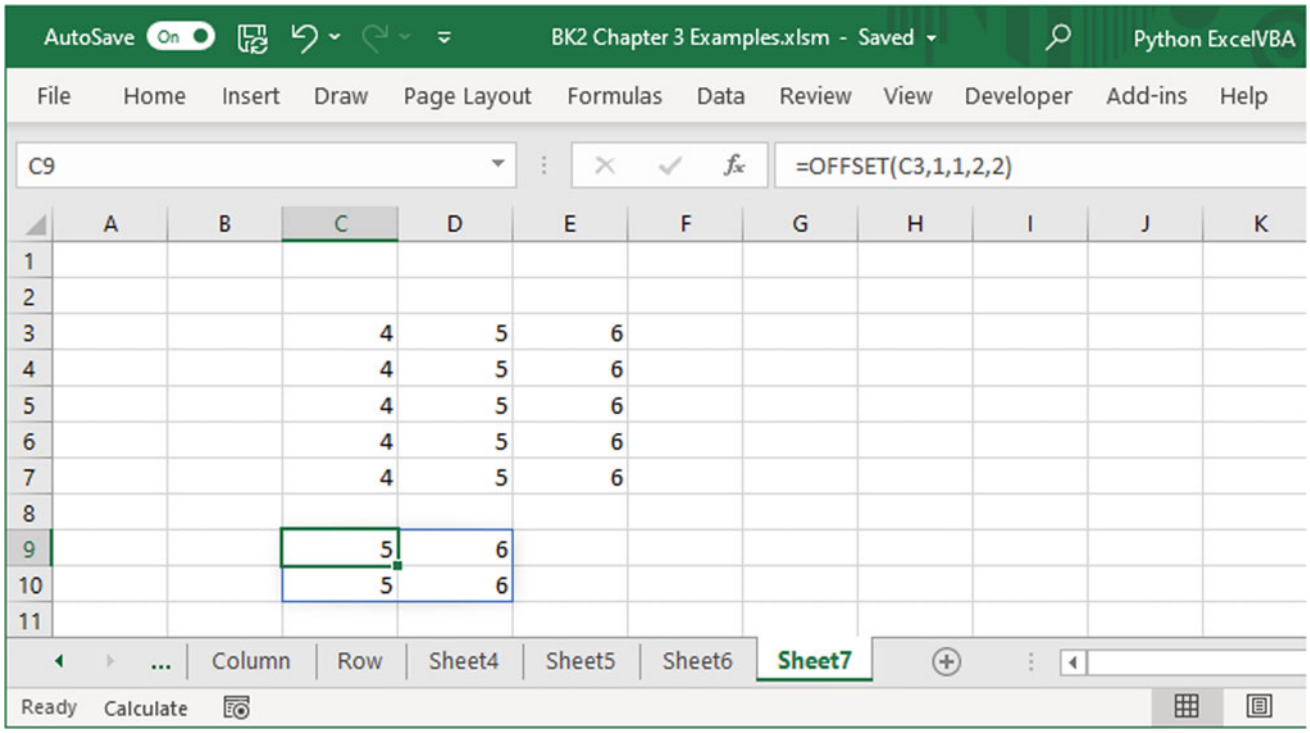
Now we will look at the worksheet function *offset*. The worksheet function *offset* takes five parameters. The first parameter is where to anchor off. The second parameter indicates the row offset. The third parameter indicates the column offset.



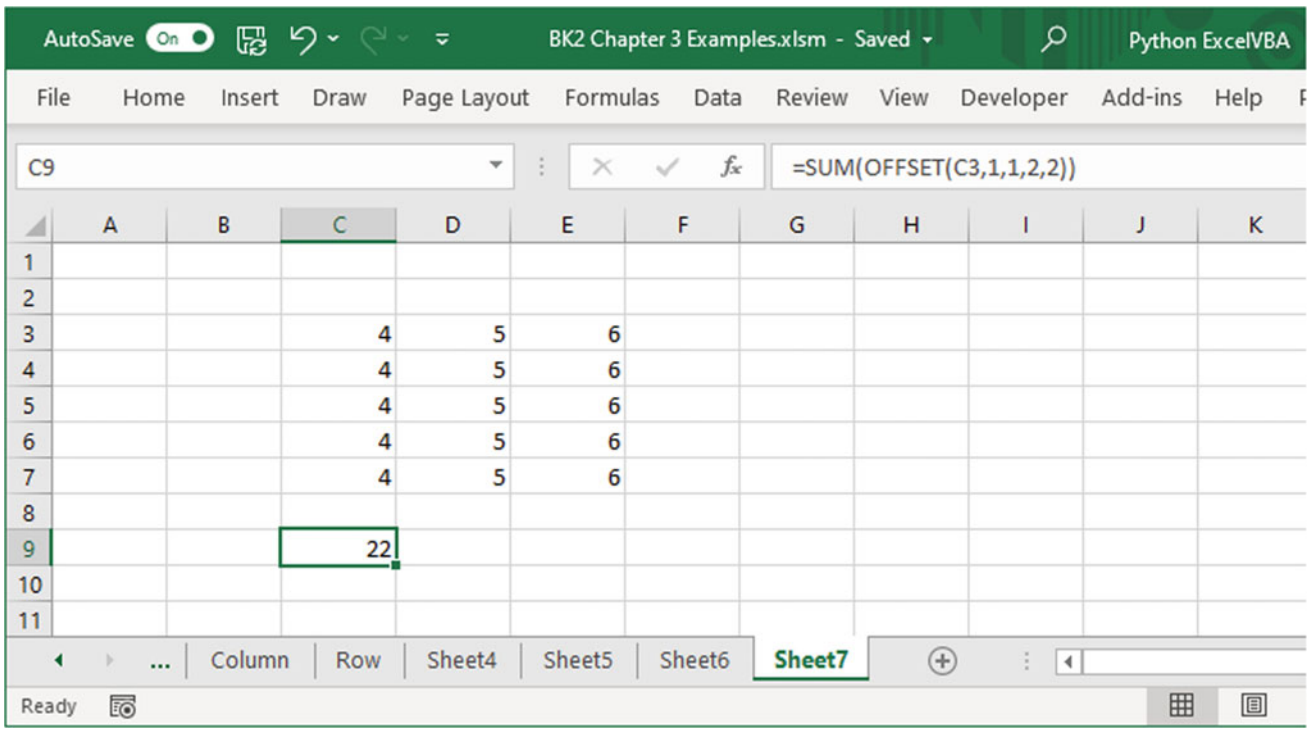
The *offset* function requires that at least the first three parameters be used. The *offset* function shown below indicates to start at cell C3 and then offset three rows and two columns. This would bring us to cell E6. The *offset* function below returns a value of 6, which agrees with the cell value of E6.



Below shows the *offset* function with all five parameters being used.

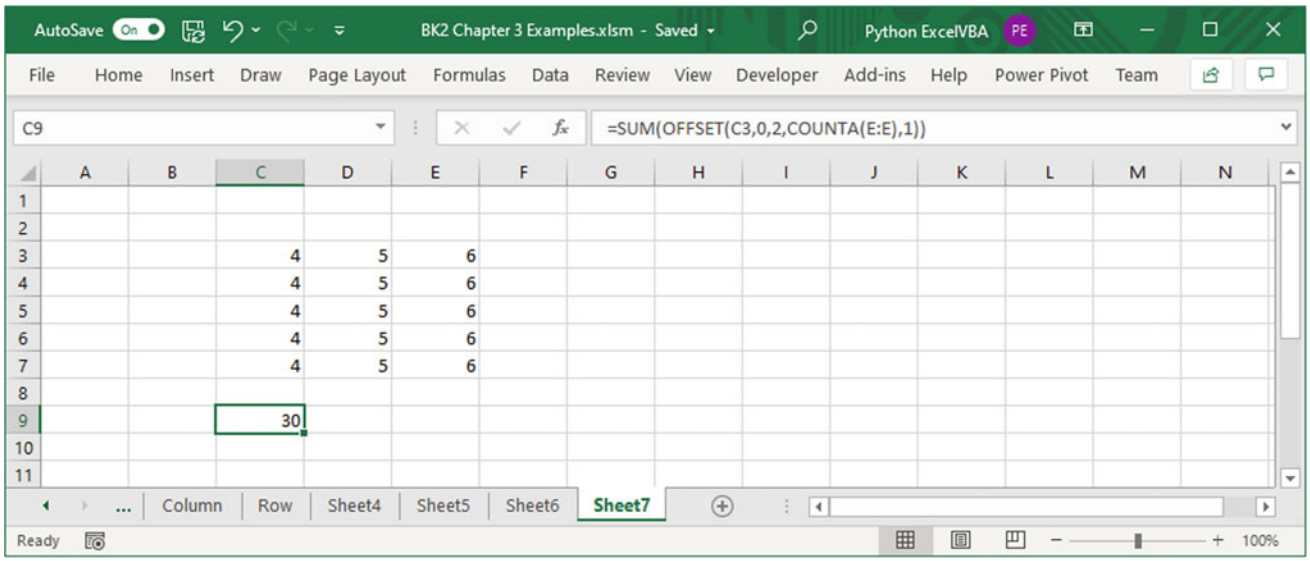


The fourth parameter indicates how many rows to resize, which in this case is 2. The fifth parameter indicates how many columns to resize to, which in this case is 2. The *offset* functions return the four values in the range D5 to E6.



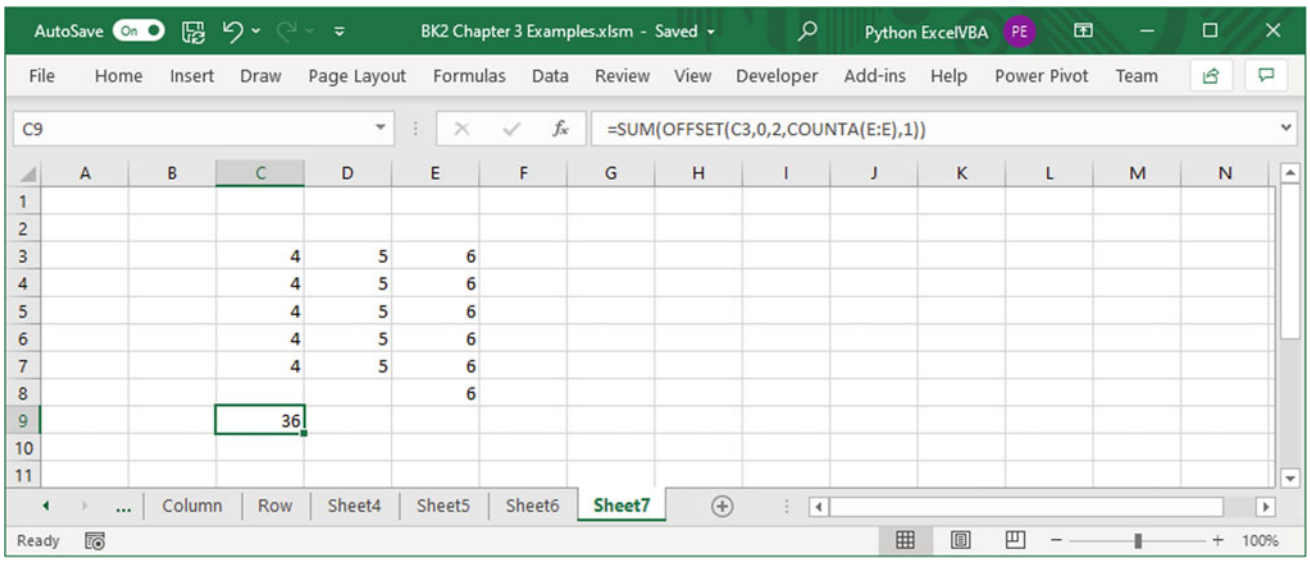
The above worksheet shows the *sum* worksheet function with the *offset* function in cell C9. The above shows a value of 22, which is the sum of the ranges from cell D4 to E5.

Next, we will illustrate how to dynamically sum column E in the above workbook. We do this by inserting a *counta* function into the fourth parameter of the *offset* function. Since we are adding column E, we change the third parameter of the *offset* to 2, which means to offset two columns to the right. This is shown below.



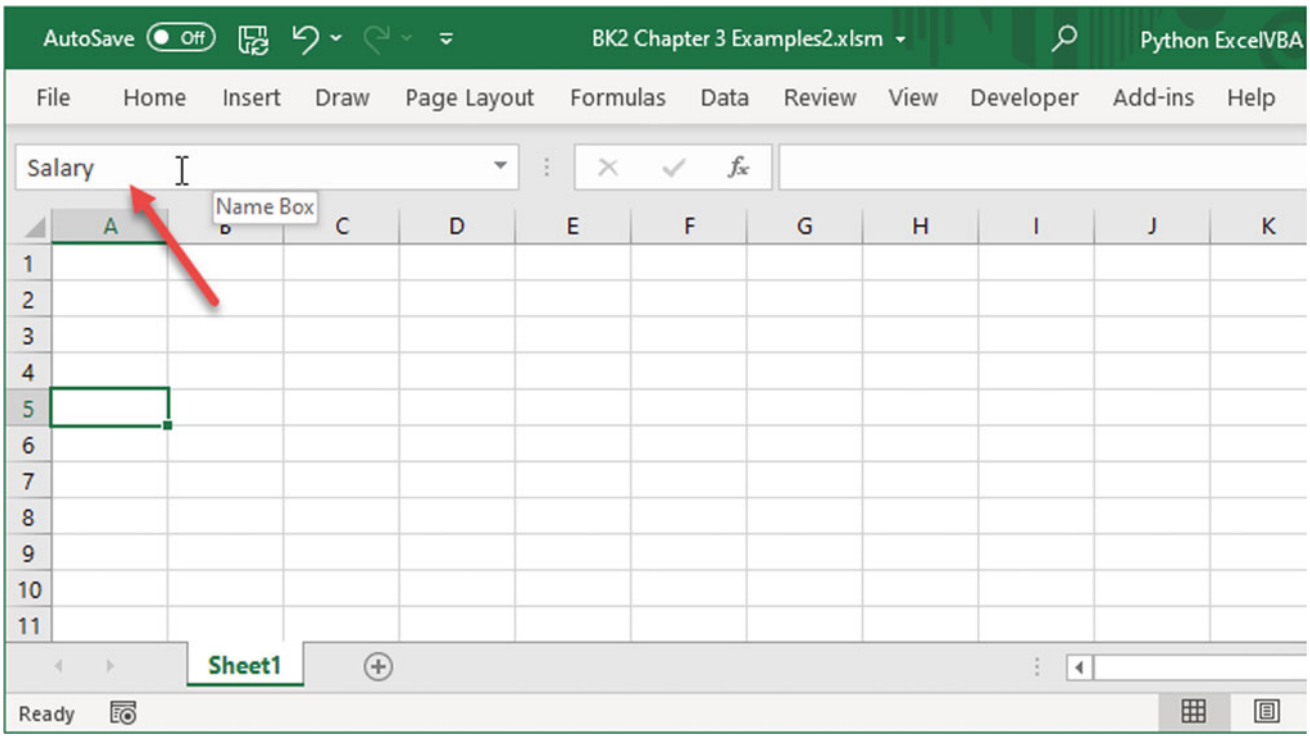
Cell C9 shows a value of 30, which agrees to the sum of the range from cell E3 to E7.

We put the function *counta* in the fourth parameter of the *offset* function. This causes the Excel formula in cell C9 to be dynamic. We can demonstrate this dynamic concept by entering a value of 6 in cell E8. Entering the value 6 in cell E8 will cause B9 to have a value of 36. This is shown below.



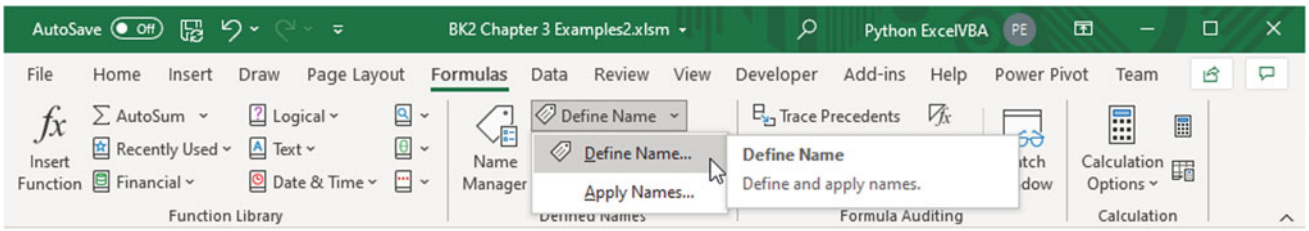
4.12 Global Versus Local Workbook Names

With workbook names, there is a distinction between "global" names versus "local" names. Not knowing the distinctions can cause a lot of problems and confusion. We will, in this section, look at many scenarios for "global" names and "local" names. By default, all workbook names are created as "global" names. Below demonstrates the consequences of names being "global" names. The first thing that we will do is define cell A1 in worksheet "Sheet1" as "Salary" through the *Name Box*. This is illustrated below.

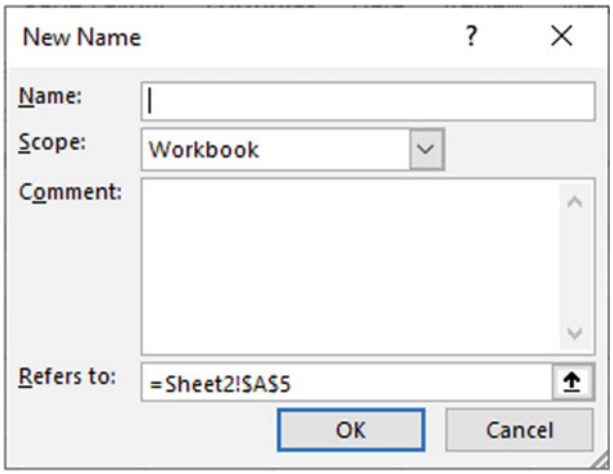


Now suppose we are also interested in defining cell A5 in worksheet "Sheet2" also as "Salary." What we will find out is when we try to define cell A5 in worksheet "Sheet2" through the *Name Box,* Excel will jump to cell A5 in worksheet "Sheet1," our first definition of "Salary." This concept illustrates the concept that there can only be one unique "global" workbook name.

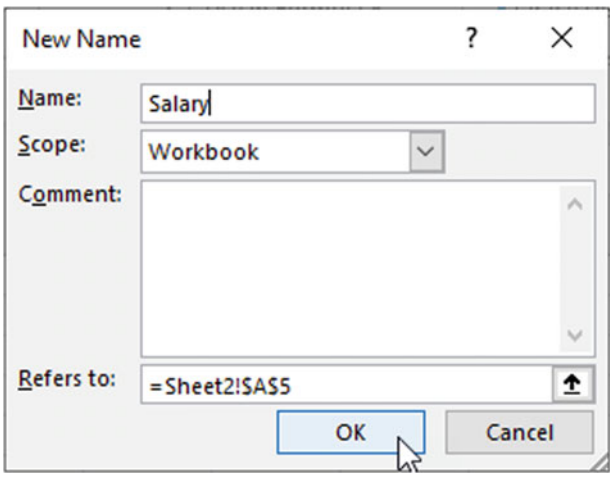
It is also possible to define names by selecting Formulas ➜ Define Name ➜ Define Name as shown below.



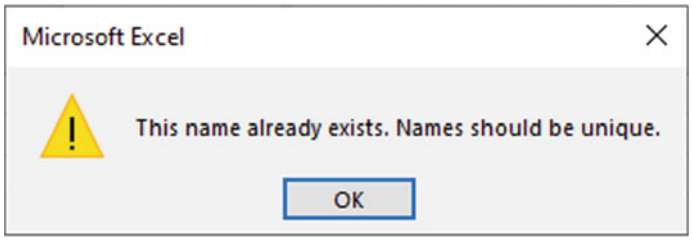
If we first choose cell A5 in worksheet "Sheet2" and then select Formulas ➔ Define Name ➔ Define Name, we will get the following *New Names* dialog box.



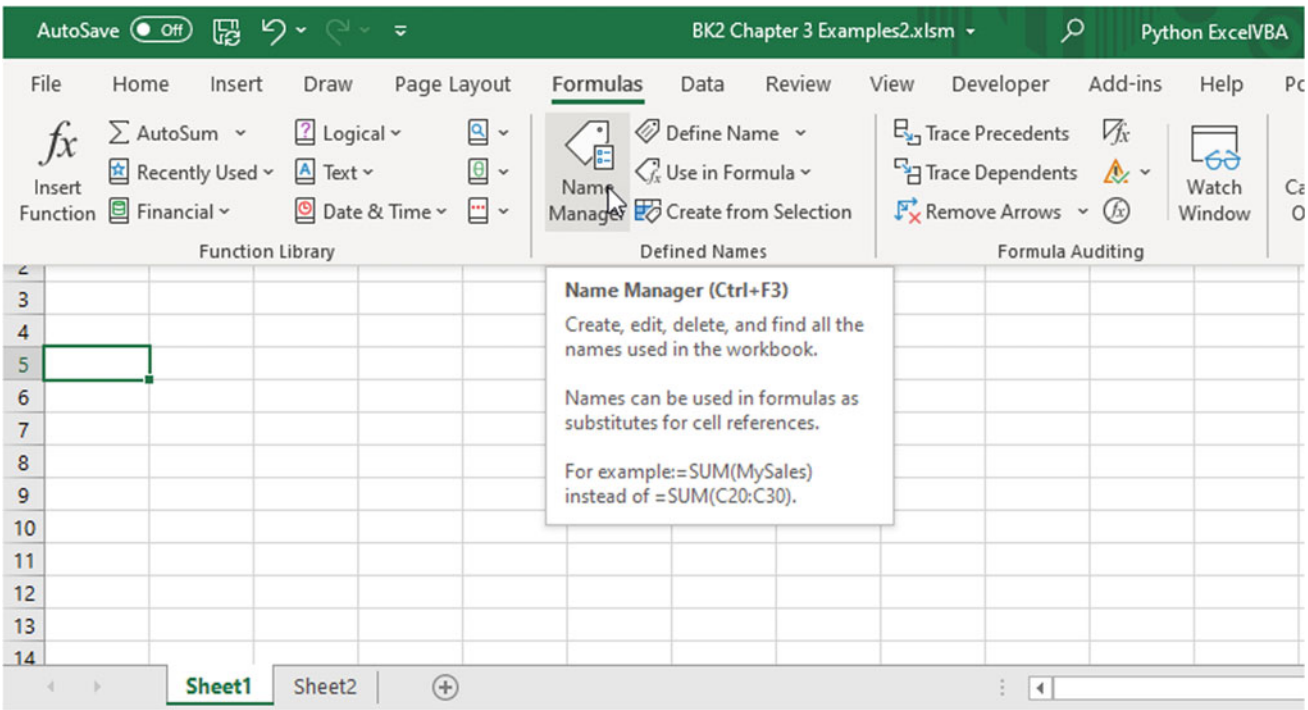
The *New Name* dialog box shows in the *Refers to*: Textbox the address of the active cell. Let's now type in "Salary" in the *Name*: Textbox and then press the *OK* button. This is shown below.



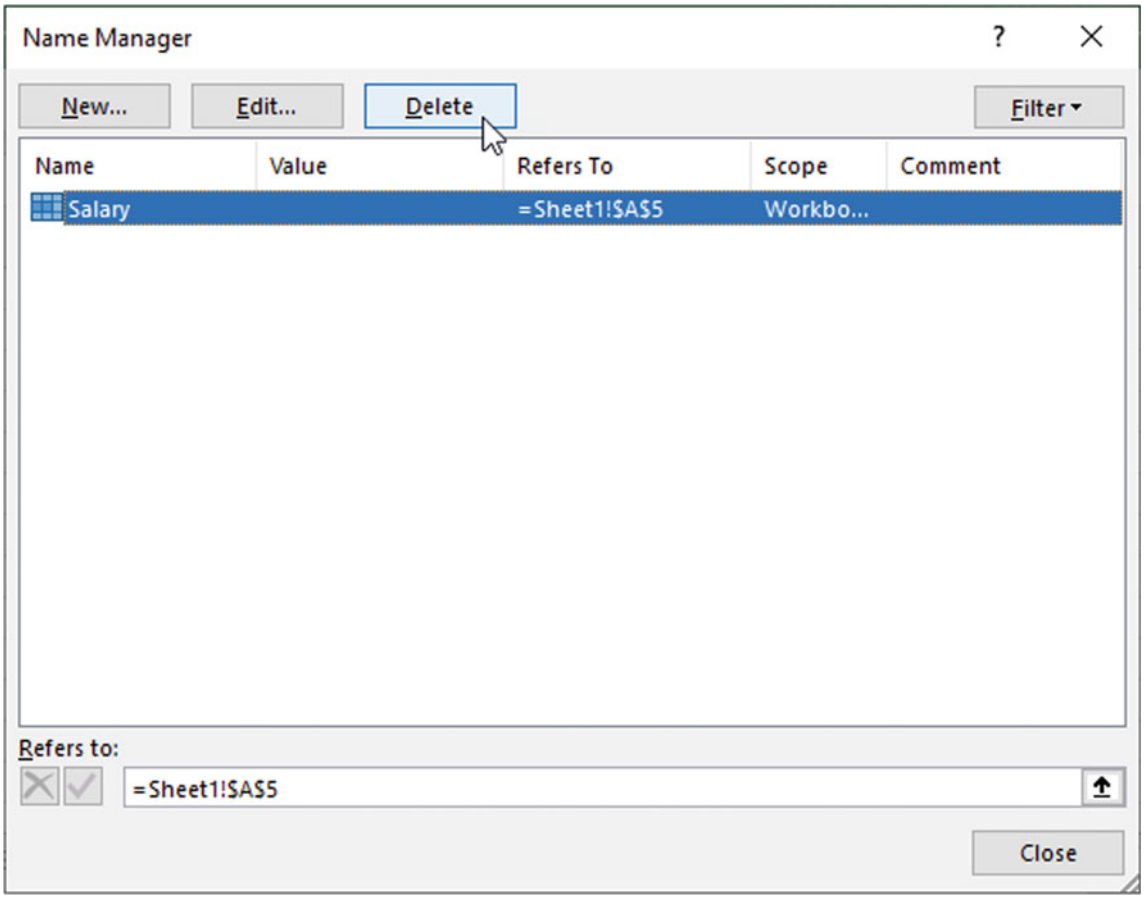
The following error message is shown after pressing the *OK* button:



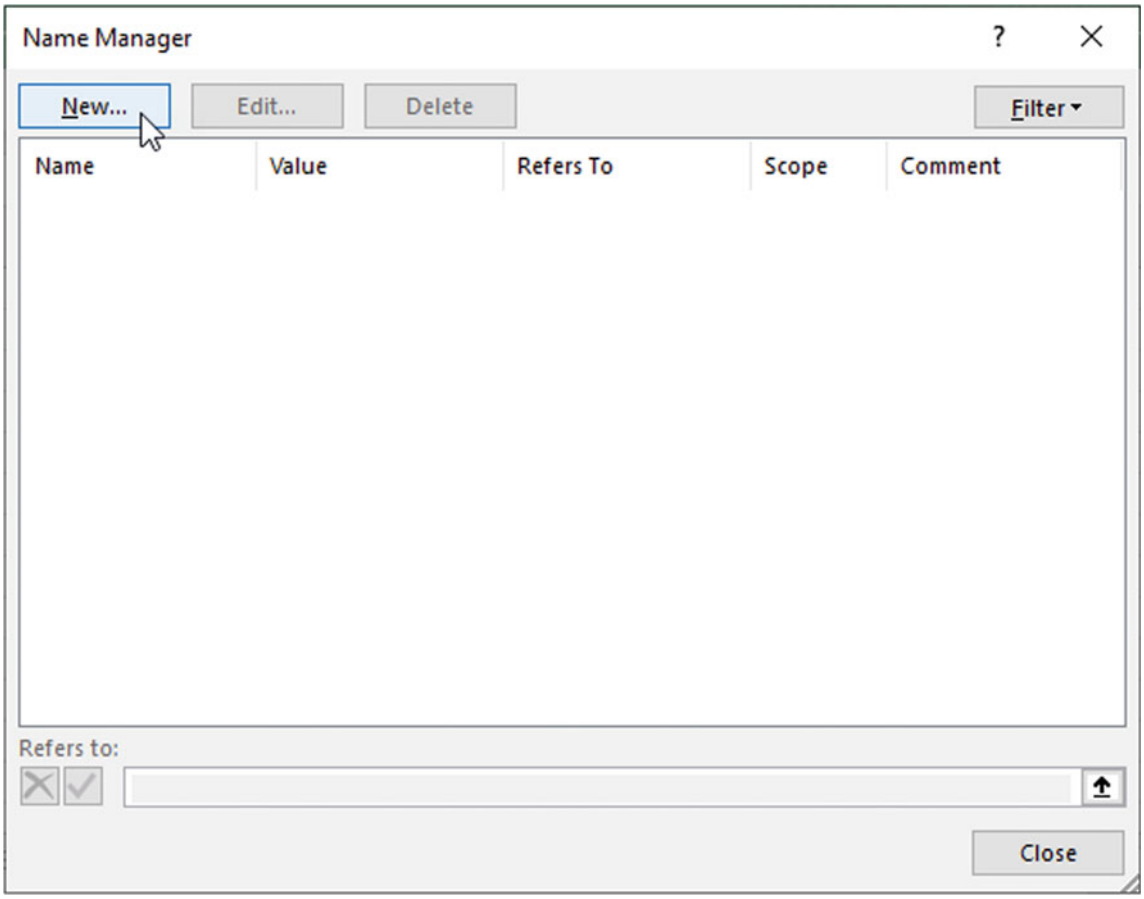
Let's now illustrate how we can have cell A5 in worksheet "Sheet1" and cell A5 in worksheet "Sheet2" both be defined as "Salary." To do this, let's press *Ctrl + F3* to get to the *Name Manager*.



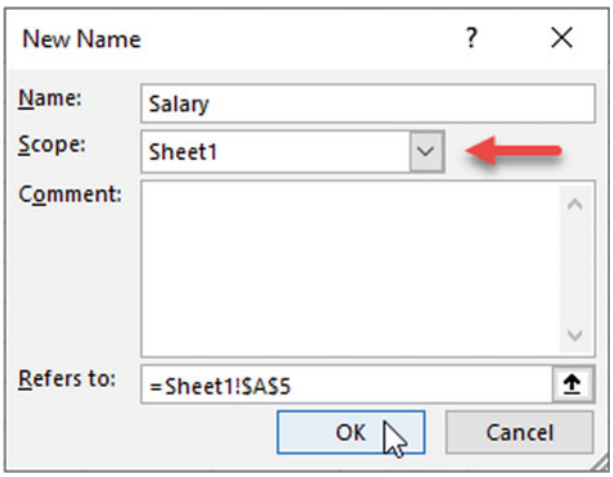
In the *Name Manager,* select "salary" and then click on the *Delete* button to delete the "salary" name.



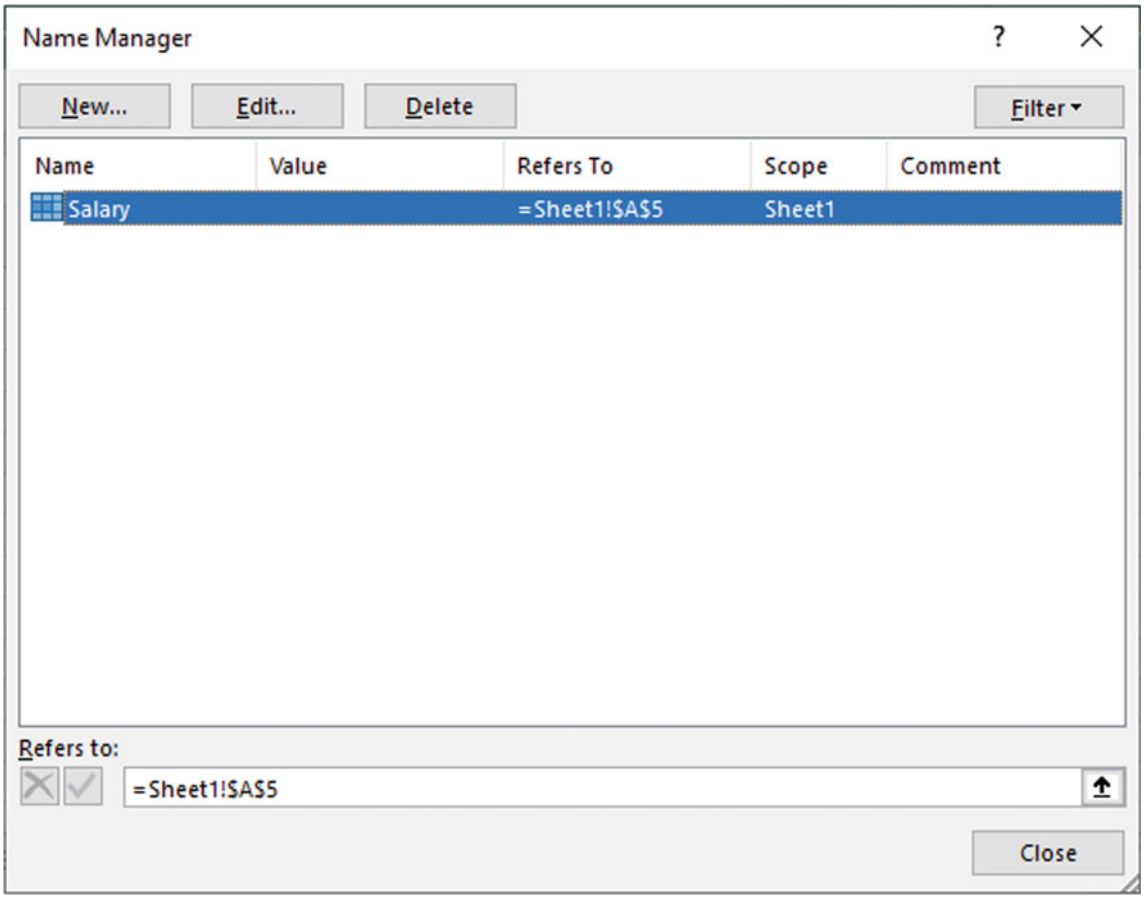
Next, click on the *New* button to create a new "Salary" name.



In the *New Name* dialog box, type in "Salary" in the *Name* textbox and change the *Scope* to "Sheet1."

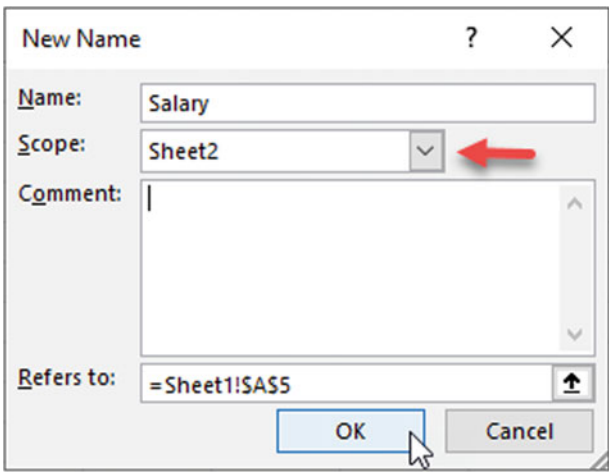


Below shows the *New Manager* after clicking on the *OK* button on the *New Name* dialog box.

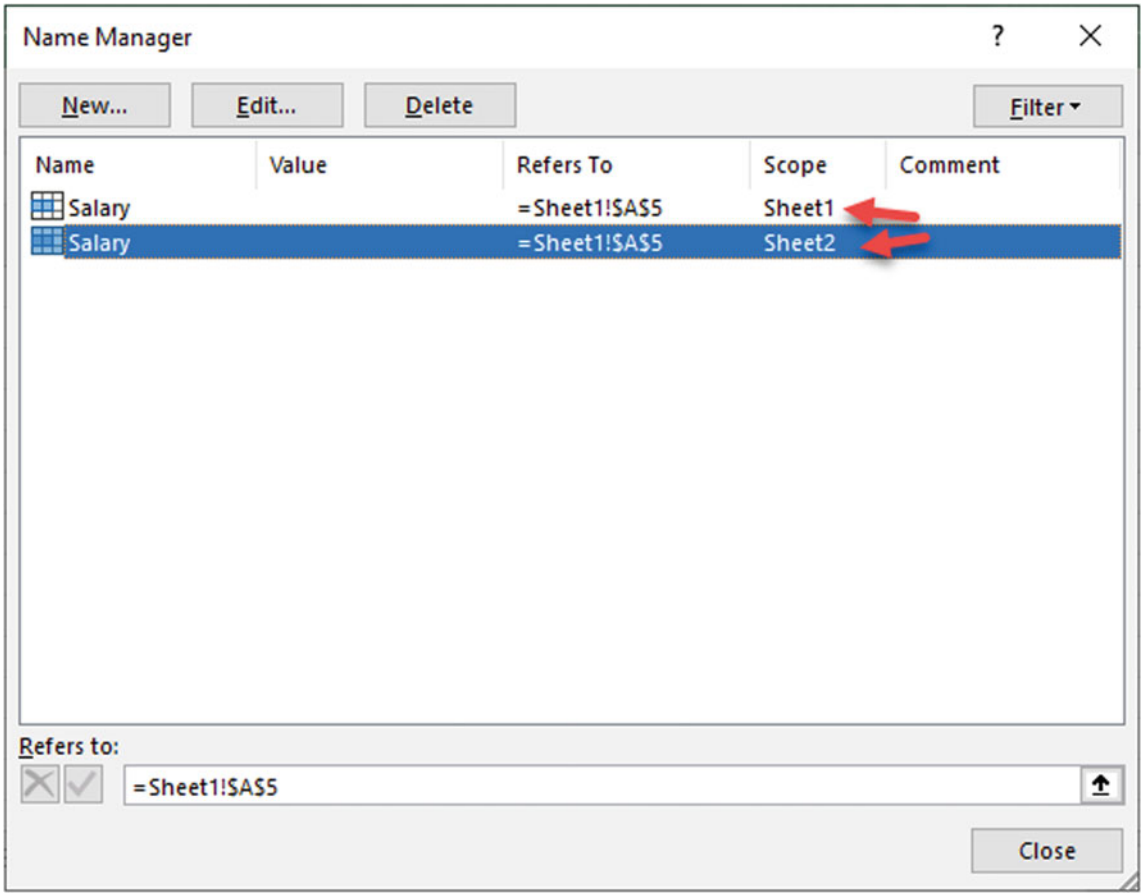


Notice that the name *Salary* is highlighted and that in the same row, the worksheet name "Sheet1" is indicated. This indicates that there is a local "Salary" range name defined for worksheet "Sheet1."

Next, press the *New* button to create the name *Salary* for "Sheet2."



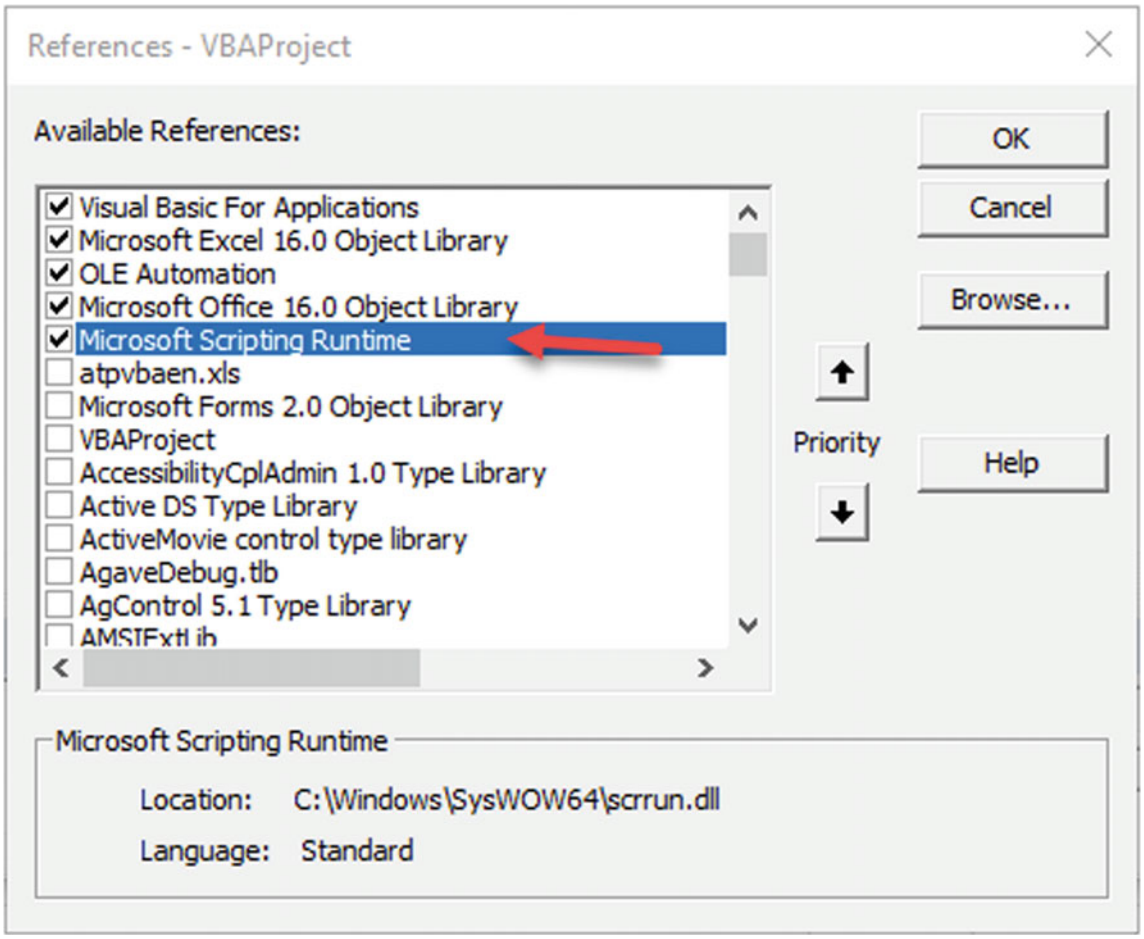
Type in "Salary" in the *Name* textbox and change the scope to "Sheet2."



The *Name* Manager now shows two *Salary* names. We are able to have two salary names because each of the salary names has different scopes.

4.13 List of All Files in a Directory

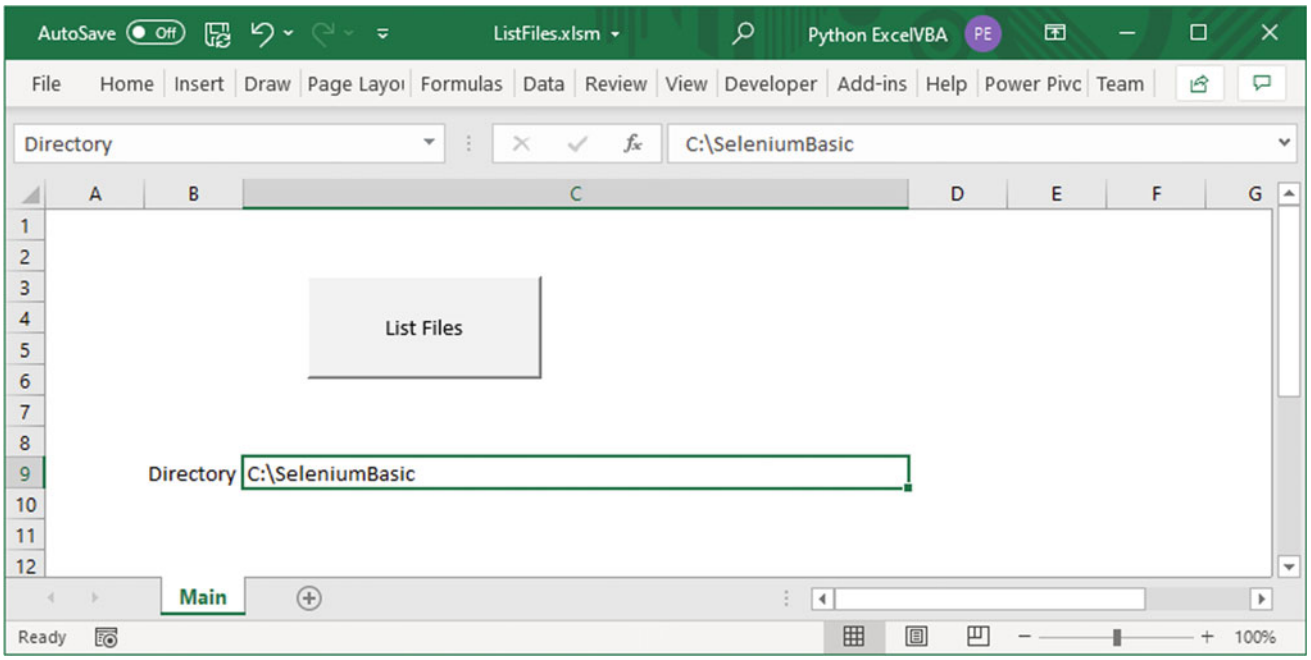
A very useful type of library is the *Microsoft Scripting Runtime type library*. This library gives you access to the *FileSystemObejct* data type. We will use this data type to list all the files in a directory. Below is a VBA macro that lists all the files in a directory. The *FileSystemObject* object is the key to accomplish this. The *FileSystemObject* requires the *Microsoft Scripting Runtime* type library. This type of library is not selected by default in the *Reference* dialog box.



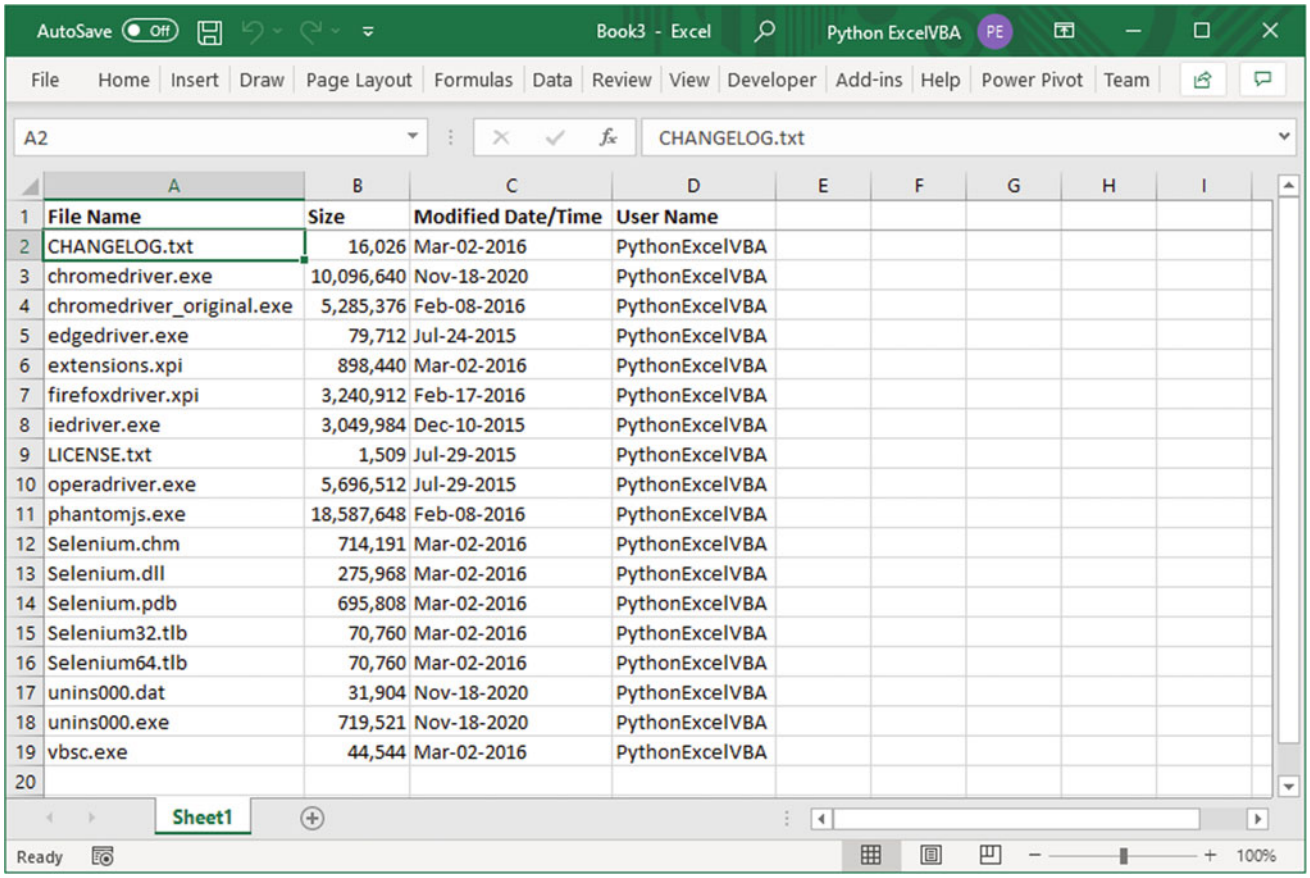

```
Sub Listfiles()
Dim FSO As New FileSystemObject
Dim objFolder As Folder
Dim objFile As File
Dim strPath As String
Dim NextRow As Long
Dim wb As Workbook
Dim ws As Worksheet
Dim wsMain As Worksheet
Set wsMain = ThisWorkbook.Worksheets(``Main'')
'Specify the path of the folder
strPath = wsMain.Range(``Directory'')
If Not FSO.FolderExists(strPath) Then
MsgBox ``The folder '' & strPath & `` does not exits.''
Exit Sub
End If
'Create the object of this folder
Set objFolder = FSO.GetFolder(strPath)
'Check if the folder is empty or not
If objFolder.Files.Count = 0 Then
MsgBox ``No files were found ...'', vbExclamation
Exit Sub
End If
Set wb = Workbooks.Add
Set ws = wb.Worksheets(1)
ws.Cells(2, 1).Select
ActiveWindow.FreezePanes = True
'Adding Column names
ws.Cells(1, ``A'').Value = ``File Name''
ws.Cells(1, ``B'').Value = ``Size''
ws.Cells(1, ``C'').Value = ``Modified Date/Time''
ws.Cells(1, ``D'').Value = ``User Name''
ws.Cells(1, 1).Resize(1, 4).Font.Bold = True
'Find the next available row
NextRow = ws.Cells(2, 1).Row
'Loop through each file in the folder
For Each objFile In objFolder.Files
'List the name of the current file
ws.Cells(NextRow, 1).Value = objFile.Name
ws.Cells(NextRow, 2).Value = Format(objFile.Size,
``#,##0'')
ws.Cells(NextRow, 3).Value = Format(objFile.
DateLastModified, ``mmm-dd-yyyy'')
ws.Cells(NextRow, 4).Value = Application.UserName
'find the next row
```

```
NextRow = NextRow + 1
Next objFile
With ws
.Cells.EntireColumn.AutoFit
End With
End Sub
```

Below demonstrates the above procedure:



Below lists all the files in the directory "c:\SeleniumBasic":



	A	B	C	D
1	File Name	Size	Modified Date/Time	User Name
2	CHANGELOG.txt	16,026	Mar-02-2016	PythonExcelVBA
3	chromedriver.exe	10,096,640	Nov-18-2020	PythonExcelVBA
4	chromedriver_original.exe	5,285,376	Feb-08-2016	PythonExcelVBA
5	edgedriver.exe	79,712	Jul-24-2015	PythonExcelVBA
6	extensions.xpi	898,440	Mar-02-2016	PythonExcelVBA
7	firefoxdriver.xpi	3,240,912	Feb-17-2016	PythonExcelVBA
8	iedriver.exe	3,049,984	Dec-10-2015	PythonExcelVBA
9	LICENSE.txt	1,509	Jul-29-2015	PythonExcelVBA
10	operadriver.exe	5,696,512	Jul-29-2015	PythonExcelVBA
11	phantomjs.exe	18,587,648	Feb-08-2016	PythonExcelVBA
12	Selenium.chm	714,191	Mar-02-2016	PythonExcelVBA
13	Selenium.dll	275,968	Mar-02-2016	PythonExcelVBA
14	Selenium.pdb	695,808	Mar-02-2016	PythonExcelVBA
15	Selenium32.tlb	70,760	Mar-02-2016	PythonExcelVBA
16	Selenium64.tlb	70,760	Mar-02-2016	PythonExcelVBA
17	unins000.dat	31,904	Nov-18-2020	PythonExcelVBA
18	unins000.exe	719,521	Nov-18-2020	PythonExcelVBA
19	vbsc.exe	44,544	Mar-02-2016	PythonExcelVBA
20				

4.14 Summary

In this chapter, we found the range of a table with the current region property, and we discussed the offset property of the range object. We also discussed resizing the property of the range object, and we discussed the UsedRange property of the range object. We looked at a special dialog box in Excel. We imported column data into arrays, imported row data in an array, and then transferred data from an array to a range. We talked about workbook names and then looked at dynamic range names. We also compared global workbook names and local workbook names. Finally, we listed all of the files in a directory.

References

https://www.ablebits.com/office-addins-blog/2017/07/11/excel-name-named-range-define-use/
https://www.automateexcel.com/vba/current-region/
https://vbaf1.com/tutorial/arrays/read-values-from-range-to-an-array/

Part II
Financial Derivatives

Binomial Option Pricing Model Decision Tree Approach

5

5.1 Introduction

Microsoft Excel is one of the most powerful and valuable tools available to business users. The financial industry in New York City has recognized this value. We can see this by going to one of the many job sites on the Internet. Two Internet sites that demonstrate the value of someone who knows Microsoft Excel very well are www.dice.com and www.indeed.com. For both of these Internet sites, search by New York City and VBA, which is Microsoft Excel's programming language, and you will see many jobs posting requiring VBA.

The academic world has begun to realize the value of Microsoft Excel. There are now many books that use Microsoft Excel to do statistical analysis and financial modeling. This can be shown by going to the Internet site www.amazon.com and searching for books by "Data Analysis Microsoft Excel" and by "Financial Modeling Microsoft Excel".

The binomial option pricing model is one the most famous models used to price options. Only the Black and Scholes model is more famous. One problem with learning the binomial option pricing model is that it is computationally intensive. This results in a very complicated formula to price an option.

The complexity of the binomial option pricing model makes it a challenge to learn the model. Most books teach the binomial option model by describing the formula. This is not very effective because it usually requires the learner to mentally keep track of many details, many times to the point of information overload. There is a well-known principle in psychology that the average number of things that a person can remember at one time is seven.

As a teaching aid, many books include decision trees. Because of the computational intensity of the model, most books do not present decision trees with more than three periods. One problem with this is that the binomial option model is best when the periods are large.

This chapter will do two things. It will first demonstrate the power of Microsoft Excel. It will do this by demonstrating that it is possible to create large decision trees for the Binomial Pricing Model using Microsoft Excel. A ten-period decision tree would require 2047 call calculations and 2047 put calculations. This paper will also show the decision tree for the price of a stock and the price of a bond, each requiring 2047 calculations. Therefore, there would be 2,047 * 4 = 8,188 calculations for a complete set of ten-period decision trees.

The second thing that this paper will do is present the binomial option model in a less mathematical matter. It will try to make it so that the reader will not have to keep track of many things at one time. It will do this by using decision trees to price call and put options.

In this chapter, we show how the binomial distribution is combined with some basic finance concepts to generate a model for determining the price stock of options.

This chapter is broken down into the following sections. In Sect. 5.2, we discuss call and put options; in Sect. 5.3, we discuss option pricing in one period; and, in Sect. 5.4, we discuss put option pricing in one period. In Sect. 5.5, we look at option pricing in two periods, and in Sect. 5.6, we look at option pricing in four periods. In Sect. 5.7, we use Microsoft Excel to create the binomial option call trees. Section 5.8 discusses American options, and Sect. 5.9 looks at alternative tree methods. Finally, in Sect. 5.10, we retrieve option prices from Yahoo Finance.

5.2 Call and Put Options

A *call option* gives the owner the right but not the obligation to buy the underlying security at a specified price. The price at which the owner can buy the underlying price is called the *exercise price*. A call option becomes valuable when the exercise price is less than the current price of the underlying stock price.

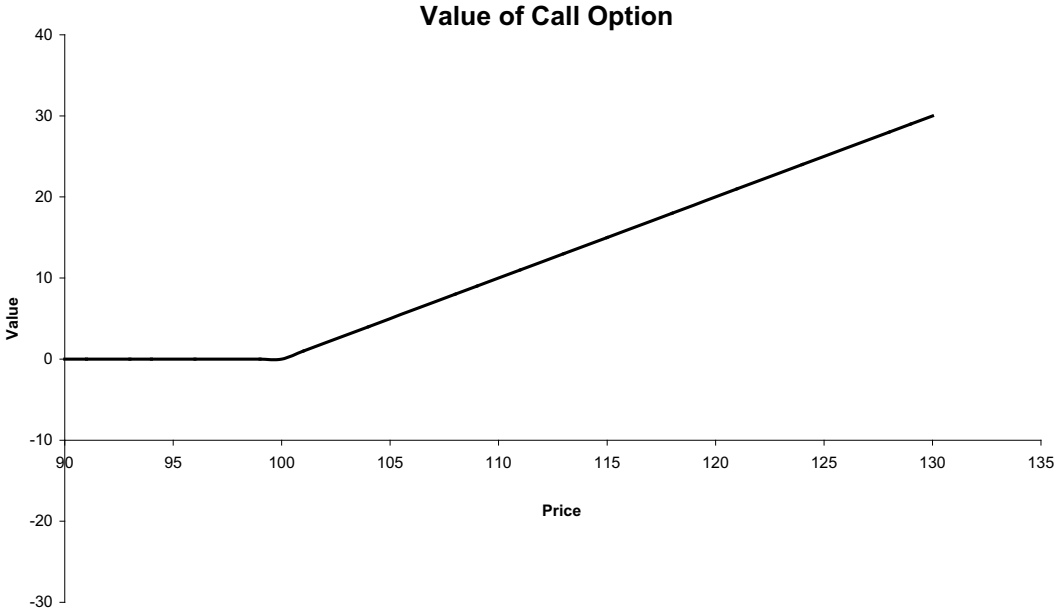
© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

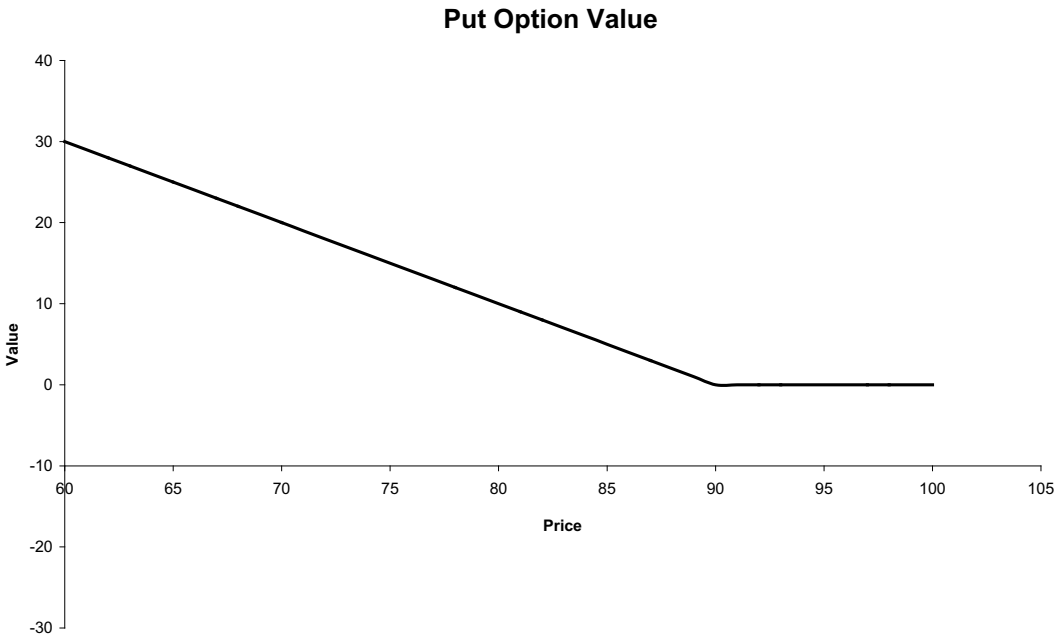
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_5

For example, a call option on an IBM stock with an exercise price of $100 when the stock price of an IBM stock is $110 is worth $10. The reason it is worth $10 is because a holder of the call option can buy the IBM stock at $100 and then sell the IBM stock at the prevailing price of $110 for a profit of $10. Also, a call option on an IBM stock with an exercise price of $100 when the stock price of an IBM stock is $90 is worth $0.

A *put option* gives the owner the right but not the obligation to sell the underlying security at a specified price. A put option becomes valuable when the exercise price is more than the current price of the underlying stock price.

For example, a put option on an IBM stock with an exercise price of $100 when the stock price of an IBM stock is $90 is worth $10. The reason it is worth $10 is because a holder of the put option can buy the IBM stock at the prevailing price of $90 and then sell the IBM stock at the put price of $100 for a profit of $10. Also, a put option on an IBM stock with an exercise price of $100 when the stock price of the IBM stock is $110 is worth $0.

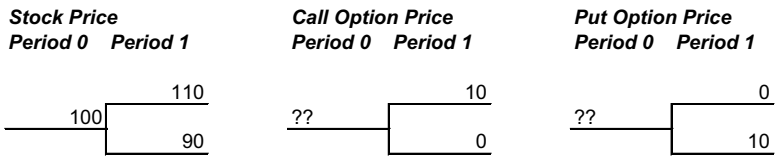




Below are the charts showing the value of call and put options of the above IBM stock at varying prices:

5.3 Option Pricing—One Period

What should be the value of these options? Let's look at a case where we are only concerned with the value of options for one period. In the next period, a stock price can either go up or go down. Let's look at a case where we know for certain that a stock with a price of \$100 will either go up 10% or go down 10% in the next period and the exercise after one period is \$100. Below shows the decision tree for the stock price, the call option price, and the put option price.



Let's first consider the issue of pricing a call option. Using a one-period decision tree, we can illustrate the price of a stock if it goes up and the price of a stock if it goes down. Since we know the possible endings values of a stock, we can derive the possible ending values of a call option. If the stock price increases to \$110, the price of the call option will then be \$10 (\$110−\$100). If the stock price decreases to \$90, the value of the call option will worth \$0 because it would be below the exercise price of \$100. We have just discussed the possible ending value of a call option in period 1. But, what we are really interested is what is the value now of the call option knowing the two resulting values of a call option.

To help determine the value of a one-period call option, it's useful to know that it is possible to replicate the resulting two states of the value of the call option by buying a combination of stocks and bonds. Below is the formula to replicate the situation where the price increases to \$110. We will assume that the interest rate for the bond is 7%.

$$\begin{aligned} 110S + 1.07B &= 10, \\ 90S + 1.07B &= 0. \end{aligned}$$

We can use simple algebra to solve for both S and B. The first thing that we need to do is to rearrange the second equation as follows:

$$1.07B = -90S.$$

With the above equation, we can rewrite the first equation as

$$\begin{aligned} 110S + (-90S) &= 10, \\ 20S &= 10, \\ S &= .5 \end{aligned}$$

We can solve for B by substituting the value 0.5 for S in the first equation as follows:

$$\begin{aligned} 110(.5) + 1.07B &= 10, \\ 55 + 1.07B &= 10, \\ 1.07B &= -45, \\ B &= -42.05607. \end{aligned}$$

Therefore, from the above simple algebraic exercise, we should at period 0 buy 0.5 shares of IBM stock and borrow 42.05607 at 7 percent to replicate the payoff of the call option. This means the value of a call option should be 0.5*100−42.05607 = 7.94393.

If this were not the case, there would then be arbitrage profits. For example, if the call option were sold for \$8 there would be a profit of 0.056607. This would result in an increase in the selling of the call option. The increase in the supply of call options would push the price down for the call options. If the call option were sold for \$7, there would be a saving of 0.94393. This saving would result in the increase demand for the call option. This increase demand would

result in the price of the call option to increase. The equilibrium point would be 7.94393.

Using the above-mentioned concept and procedure, Benninga (2000) derived a one-period call option model as

$$C = q_u \text{Max}[\text{S}(1+\text{u}) - \text{X}, 0] + q_d \text{Max}[\text{S}(1+\text{d}) - \text{X}, 0], \quad (5.1)$$

where

$$q_u = \frac{i-d}{(1+i)(u-d)},$$

$$q_d = \frac{u-i}{(1+i)(u-d)},$$

u = increase factor,
d = down factor,
i = interest rate.

If we let $i = r$, $p = (r\text{-}d)/(u\text{-}d)$, 1—$p = (u\text{-}r)/(u\text{-}d)$, R = 1/(1 + r), C_u = Max[S(1 + u)—X, 0] and C_d = Max[S(1 + d)—X, 0], then we have

$$C = [pC_u + (1-p)C_d]/R, \quad (5.2)$$

where

C_u = call option price after increase,
C_d = call option price after decrease.

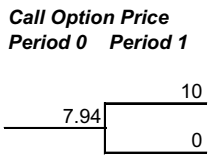
Equation (5.2) represents one-period call option value.

Below calculates the value of the above one-period call option, where the strike price, X, is \$100 and the risk-free interest rate is 7%. We will assume that the price of a stock for any given period will either increase or decrease by 10%.

$$X = \$100,$$
$$\text{S} = \$100,$$
$$u = 1.10,$$
$$\text{d} = .9,$$
$$\text{R} = 1 + \text{r} = 1 + .07,$$
$$\text{p} = (1.07 - .90)/(1.10 - .90),$$
$$C = [.85(10) + .15(0)]/1.07 = \$7.94.$$

Therefore, from the above calculations, the value of the call option is \$7.94.

From the above calculations, the call option pricing decision tree should look like the following:



5.4 Put Option Pricing—One Period

Like the call option, it is possible to replicate the resulting two states of the value of the put option by buying a combination of stocks and bonds. Below is the formula to replicate the situation where the price decreases to \$90:

$$110\text{S} + 1.07\text{B} = 0,$$
$$90\text{S} + 1.07\text{B} = 10.$$

We will use simple algebra to solve for both S and B. The first thing we will do is to rewrite the second equation as follows:

$$1.07\text{B} = 10 - 90\text{S}.$$

The next thing to do is to substitute the above equation to the first put option equation. Doing this would result in the following:

$$110\text{S} + 10 - 90\text{S} = 0.$$

The following solves for S:

$$20\text{S} = -10,$$
$$\text{S} = -.5.$$

Now let's solve for B by putting the value of S into the first equation. This is shown below:

$$110(-.5) + 1.07\text{B} = 0,$$
$$1.07\text{B} = 55,$$
$$\text{B} = 51.04.$$

From the above simple algebra exercise, we have S = -0.5 and B = 51.04. This tells us that we should in period 0 lend \$51.04 at 7% and sell 0.5 shares of stock to replicate the put option payoff for period 1. And, the value of the put option should be 100*(-0.5) + 51.40 = -50 + 51.40 = 1.40.

Using the same arbitrage argument that we used in the discussion of call option, 1.40 has to be the equilibrium price of the put option.

As with the call option, Benninga (2000) has derived a one-period put option model as

$$P = q_u \text{Max}[\text{X} - \text{S}(1+\text{u}), 0] + q_d \text{Max}[\text{X} - \text{S}(1+\text{d}), 0], \quad (5.3)$$

where

$$q_u = \frac{i-d}{(1+i)(u-d)},$$

$$q_d = \frac{u - i}{(1+i)(u-d)},$$

u = increase factor,
d = down factor,
i = interest rate.

If we let $i = r$, $p = (r\text{-}d)/(u\text{-}d)$, $1-p = (u\text{-}r)/(u\text{-}d)$, R = 1/(1 + r), P_u = Max[X−S(1 + u), 0] and P_d = Max[X−S(1 + d), 0], then we have

$$P = [pP_u + (1-p)P_d]/\mathrm{R}, \tag{5.4}$$

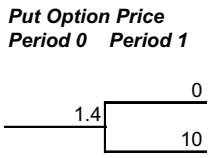
where

P_u = put option price after increase,
P_d = put option price after decrease.

Below calculates the value of the above one-period call option, where the strike price, X, is \$100 and the risk-free interest rate is 7%.

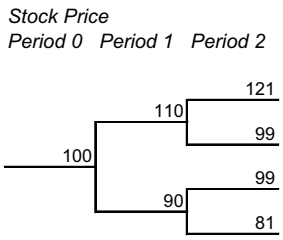
$$P = [.85(0) + .15(10)]/1.07 = \$1.40$$

From the above calculation, the put option pricing decision tree would look like the following:



5.5 Option Pricing—Two Period

We now will look at pricing options for two periods. Below shows the stock price decision tree based on the parameters indicated in the last section.



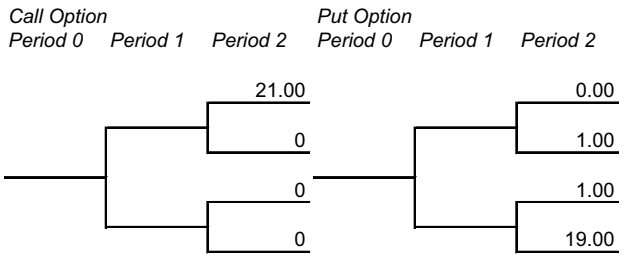
This decision tree was created based on the assumption that a stock price will either increase by 10% or decrease by 10%.

How do we price the value of a call and put options for two periods?

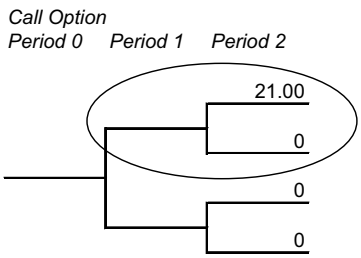
The highest possible value for our stock based on our assumption is \$121. We get this value first by multiplying the stock price at period 0 by 110% to get the resulting value of \$110 for period 1. We then again multiply the stock price in period 1 by 110% to get the resulting value of \$121. In period two, the value of a call option, when a stock price is \$121, is the stock price minus the exercise price, \$121−100, or \$21. In period two, the value of a put option, when a stock price is \$121, is the exercise price minus the stock price, \$100−\$121, or -\$21. A negative value has no value to an investor so the value of the put option would be \$0.

The lowest possible value for our stock based on our assumptions is \$81. We get this value first by multiplying the stock price at period 0 by 90% (decreasing the value of the stock by 10%) to get the resulting value of \$90 for period 1 and then multiplying the stock price in period 1 by 90% to get the resulting value of \$81. In period 2, the value of a call option, when a stock price is \$81, is the stock price minus the exercise price, \$81−\$100, or -\$19. A negative value has no value to an investor so the value of a call option would be \$0. In period 2, the value of a put option when a stock price is \$81 is the exercise price minus the stock price, \$100−\$ 81, or \$19. We can derive the call and put option values for the other possible value of the stock in period 2 in the same fashion.

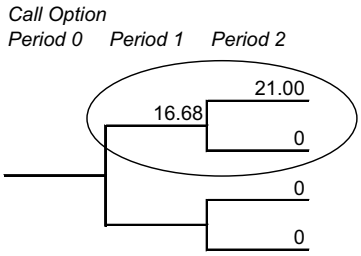
The following shows the possible call and put option values for period 2.



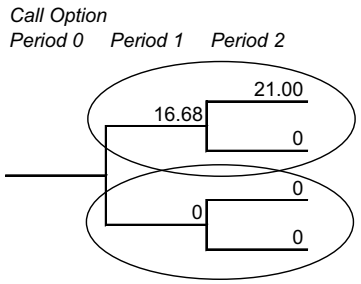
We cannot calculate the value of the call and put options in period 1 the same way we did in period 2 because it's not the ending value of the stock. In period 1, there are two possible call values. One value is when the stock price increases and one value is when the stock price decreases. The call option decision tree shown above shows two possible values for a call option in period 1. If we just focus on the value of a call option when the stock price increases from period 1, we will notice that it is like the decision tree for a call option for one period. This is shown below.



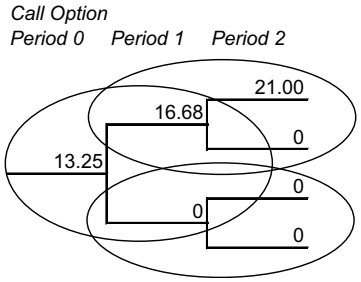
Using the same method for pricing a call option for one period, the price of a call option when the stock price increases from period 0 will be \$16.68. The resulting decision tree is shown below.



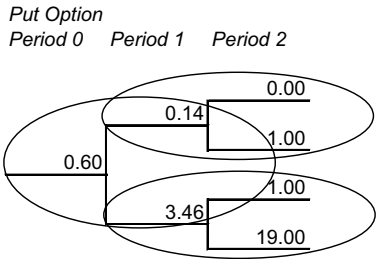
In the same fashion, we can price the value of a call option when a stock price decreases. The price of a call option when a stock price decreases from period 0 is \$0. The resulting decision tree is shown below.



In the same fashion, we can price the value of a call option in period 0. The resulting decision tree is shown below.

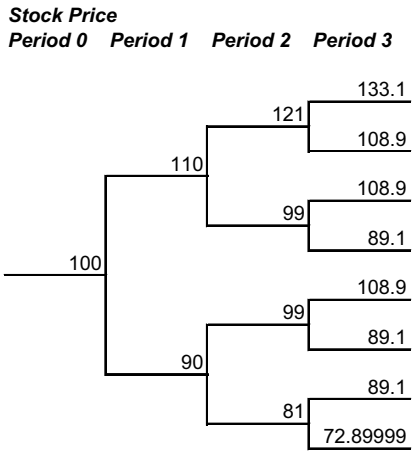


We can calculate the value of a put option in the same manner as we did in calculating the value of a call option. The decision tree for a put option is shown below.

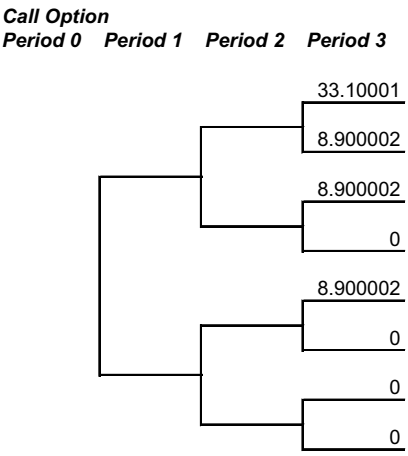


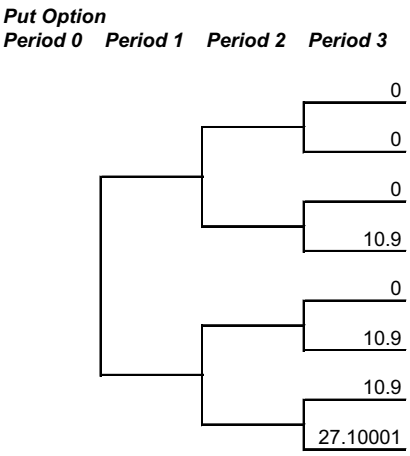
5.6 Option Pricing—Four Period

We now will look at pricing options for three periods. Below shows the stock price decision tree based on the parameters indicated in the last section.



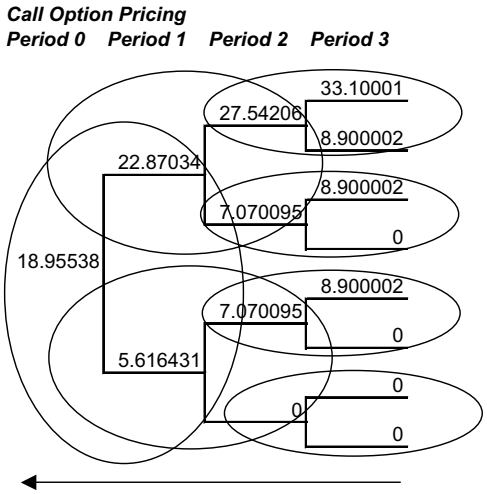
From the above stock price decision tree, we can figure out the values for the call and put options for period 3. The values for the call and put options are shown below.

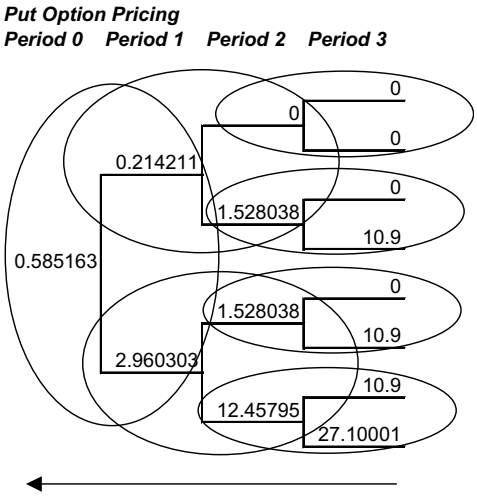




The value is $33.10 for the topmost call option because the stock price is $133.1 and the exercise price is $100. In other words, $133.1−$100 = $33.10.

To get the price of the call and put options at period 0, we will need to price backwards from period 3 to period 0 as shown below. Each circled calculation below is basically a one-period calculation shown in the previous section.





5.7 Using Microsoft Excel to Create the Binomial Option Call Trees

In the previous section, we priced the value of a call and put option by pricing backwards, from the last period to the first period. This method of pricing call and put options will work for any n period. To price the value of a call option for two periods required seven sets of calculations. The number of calculations increases dramatically as n increases. Table 1 lists the number of calculations for a specific number of periods.

Periods	Calculations
1	3
2	7
3	15
4	31
5	63
6	127
7	255
8	511
9	1023
10	2047
11	4065
12	8191

After two periods, it becomes very cumbersome to calculate and create the decision trees for a call and put option. In the previous section, we saw that calculations were very repetitive and mechanical. To solve this problem, this paper will use Microsoft Excel to do the calculations and create the decision trees for the call and put options. We will also use Microsoft Excel to calculate and draw the related decision trees for the underlying stock and bond.

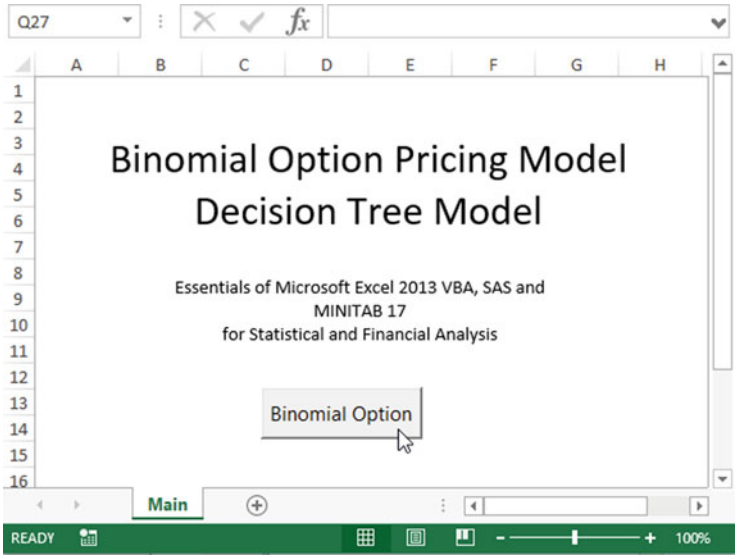
To solve this repetitive and mechanical calculation of the binomial option pricing model, we will look at a Microsoft Excel file called *binomialoptionpricingmodel.xlsm*.

We will use this Excel file to produce four decision trees for the IBM stock that was discussed in the previous sections. The four decision trees are given below:

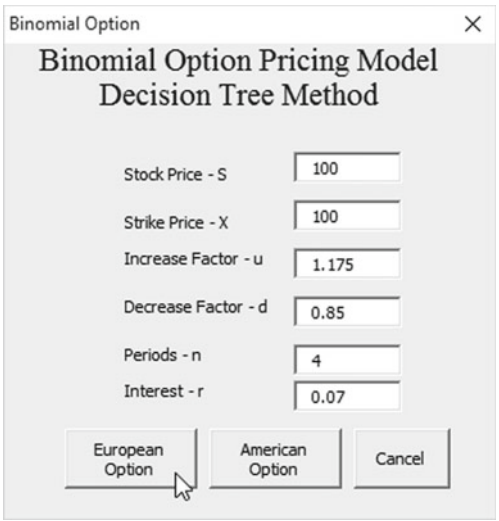
(1) Stock Price,
(2) Call Option Price,
(3) Put Option Price, and
(4) Bond Price.

This section will demonstrate how to use the *binomialoptionpricingmodel.xlsm* Excel file to create the four decision trees.

The following shows the Excel file *binomialoptionpricingmodel.xlsm* after the file is opened.

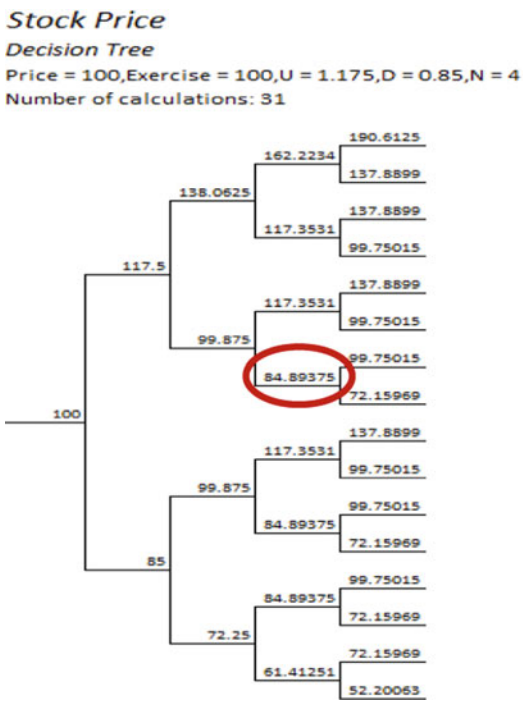


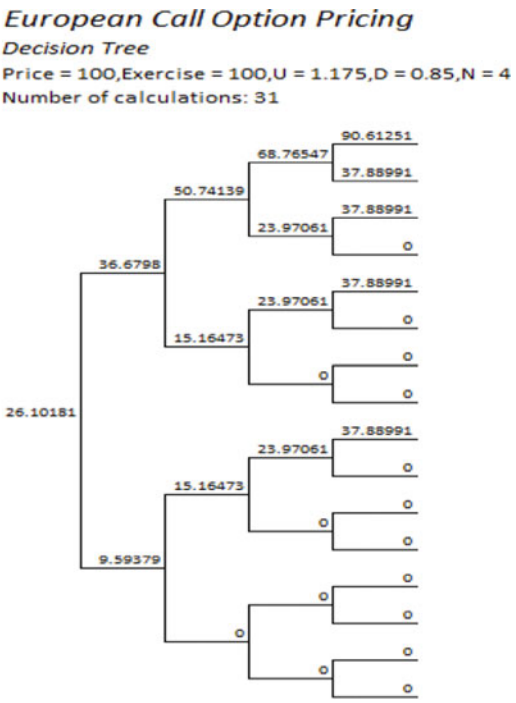
Pushing the binomial option button shown above will get the dialog box shown below.

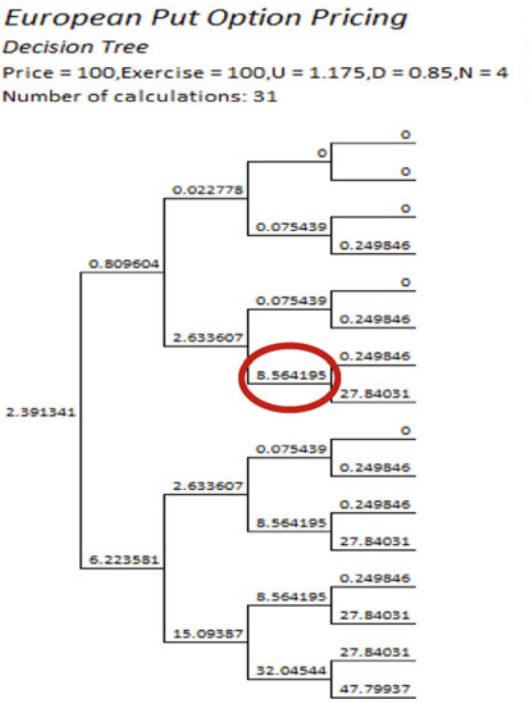


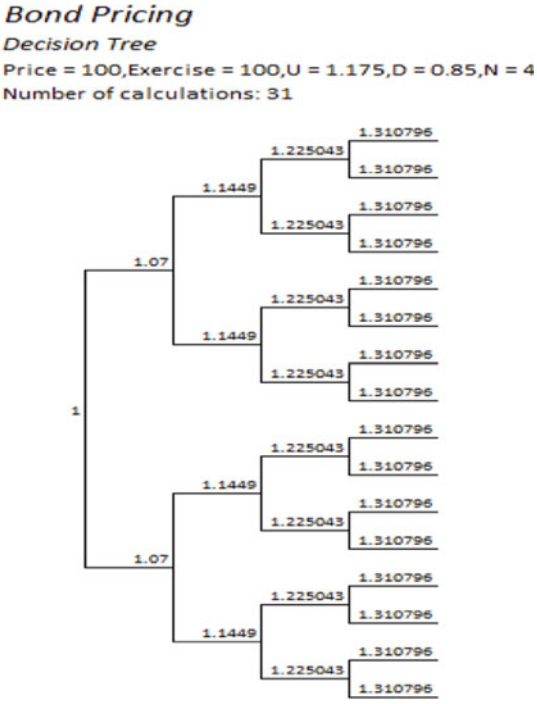
The dialog box shown above shows the parameters for the *binomial option pricing model*. These parameters are changeable. The dialog box shows the default values.

Pushing the *European Option* button produces four binomial option decision trees.









The table at the beginning of this section indicated that 31 calculations were required to create a decision tree that has four periods. This section showed four decision trees. Therefore, the Excel file did 31 * 4 = 121 calculations to create the four decision trees.

Benninga (2000, p260) defined the price of a call option in a binomial option pricing model with n periods as

$$C = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[S(1+u)^i(1+d)^{n-i} - X, 0] \qquad (5.5)$$

and the price of a put option in a binomial option pricing model with n periods as

$$P = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[X - S(1+u)^i (1+d)^{n-i}, 0]. \tag{5.6}$$

Lee et al. (2000,p237) defined the pricing of a call option in a binomial option pricing model with n period as

$$C = \frac{1}{R^n} \sum_{k=0}^{n} \frac{n!}{k!(n-k!)} p^k (1-p)^{n-k} \max[0, (1+u)^k (1+d)^{n-k} S - X]. \tag{5.7}$$

The definition of the pricing of a put option in a binomial option pricing model with n period would then be defined as

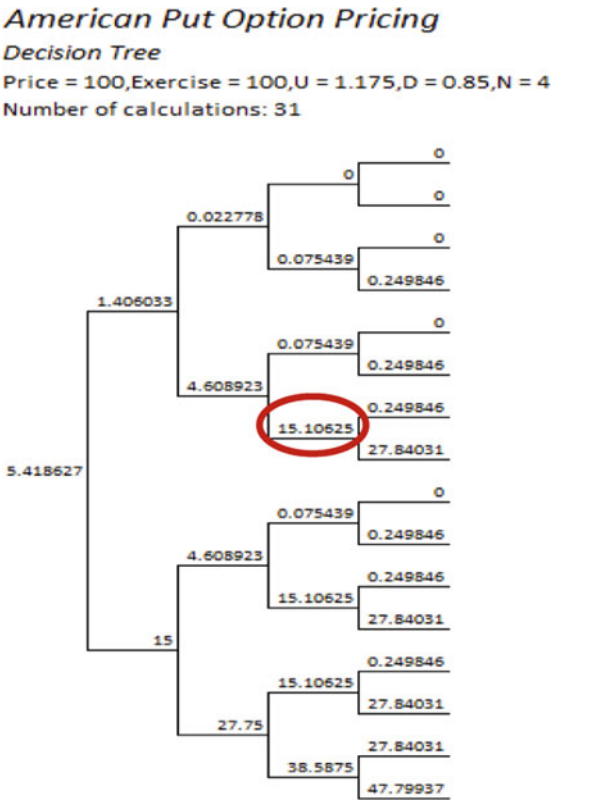
$$P = \frac{1}{R^n} \sum_{k=0}^{n} \frac{n!}{k!(n-k!)} p^k (1-p)^{n-k} \max[0, X - (1+u)^k (1+d)^{n-k} S]. \tag{5.8}$$

5.8 American Options

An American option is an option that the holder may exercise at any time between the start date and the maturity date. Therefore, the holder of an American option faces the dilemma of deciding when to exercise. Binomial tree valuation can be adapted to include the possibility of exercise at intermediate dates and not only the maturity date This feature needs to be incorporated into the pricing of American options.

The first step of pricing an American option is the same as a European option. For a nAmerican put option, the second step is taken as the maximum of the difference between the strike price of the stock and the price of the stock at that node N and the value of the European put option at node N. The value of a Eurpean put option is shown in Eq. 5.4.

Below shows the American put option binomial tree. This American put option has the same parameters as the European put option.



With the same input parameters, we can see that the value of the European put option and the value of the American put option are different. The value of the European put option is 2.391341, while the value of the American put option is 5.418627.

The red circle in the American put option binomial tree is one reason why. At this node, the American put option has a value of 15.10625, while, at the same node, the European put option has a value of 8.564195. At this node, the value of the put option is the maximum of the difference between the strike stock's strike price and stock price at this node and the value of the European put option at this node. At this node, the stock price is 84.09375 and the stock strike price is 100.

Mathematically, the price of the American put option at this node is

$$\begin{aligned} \text{Max}(\text{X} - \text{St}, 8.564195) &= \text{Max}(100 - 84.09375, 8.56195) \\ &= 15.10625. \end{aligned}$$

5.9 Alternative Tree Methods

In this section, we will introduce three binomial tree methods and one trinomial tree method to price option values. Three binomial tree methods include Cox, Ross, and Rubinstein (1979), Jarrow and Rudd (1983), and Leisen and Reimer (1996). These methods will generate different kinds of underlying asset trees to represent different trends of asset movement. Kamrad and Ritchken (1991) extended binomial tree method to multinomial approximation models. Trinomial tree method is one of the multinomial models.

5.9.1 Cox, Ross, and Rubinstein

Cox, Ross, and Rubinstein (1979) (hereafter CRR) propose an alternative choice of parameters that also creates a risk-neutral valuation environment. The price multipliers, u and d, depend only on volatility σ and on δt, not on drift as shown below:

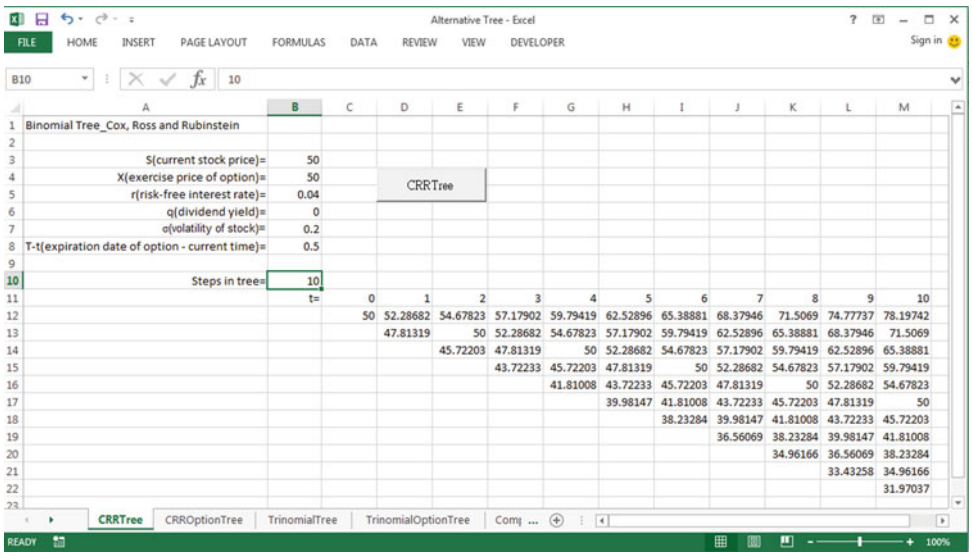
$$u = e^{\sigma\sqrt{\delta t}}$$

$$d = \frac{1}{u}$$

To offset the absence of a drift component in u and d, the probability of an up move in the CRR tree is usually greater than 0.5 to ensure that the expected value of the price increases by a factor of exp[(r-q)δt] on each step. The formula for p is

$$p = \frac{e^{(r-q)\delta t} - d}{u - d}$$

Below is the asset price tree base on CRR binomial tree model.



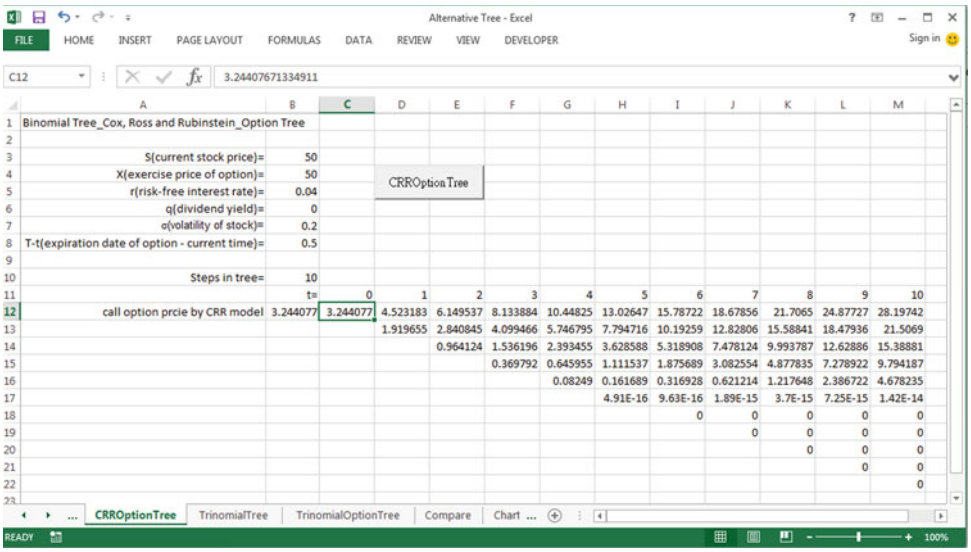
We can see that CRR tree is symmetric to its initial asset price, in this case, is 50. Next, we want to create option tree in the worksheet. For example, a call option value is on this asset price. Let fi,j denotes the option value in node (i,j), where j refers to period j (j = 0,1,2,…,N) and i denotes the ith node in period j (in the binomial tree model, node numbers increase going up in the lattice, so i = 0,…,j). With these assumptions, the underlying asset price in node (i,j) is Su^jd^{i-j}. At the expiration, we have

$$f_{i,N} = \max\left[Su^i d^{N-i} - X, 0\right] \qquad i = 0, 1, \ldots, N$$

Going backward in time (decreasing j), we get

$$f_{i,j} = e^{-r\delta t}\left[pf_{i+1,j+1} + (1-p)f_{i,j+1}\right]$$

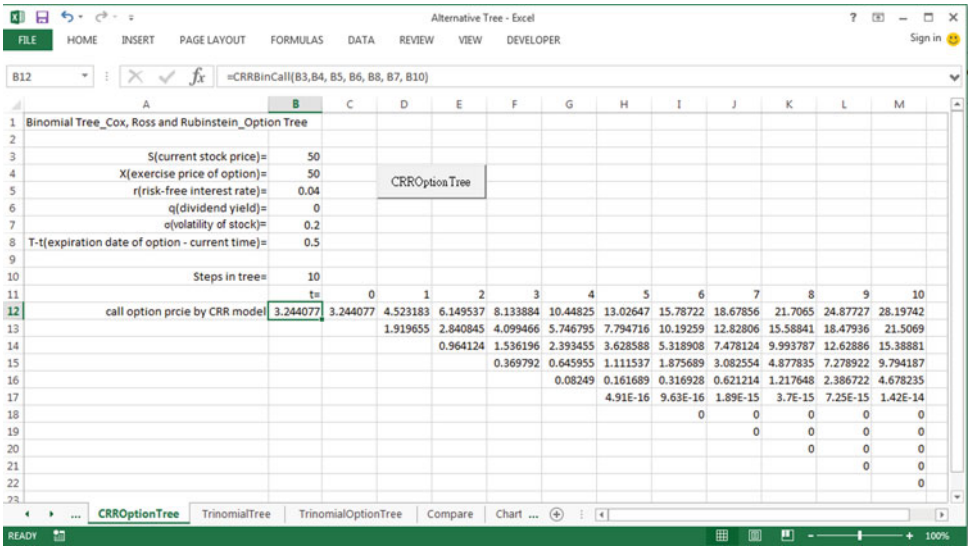
The CRR option value tree is shown below.



We can see the call option value at time zero is equal to 3.244077 in Cell C12. We also can write a VBA function to price call option. Below is the function:

```
'  Returns CRR Binomial Option Value
Function CRRBinCall(S, X, r, q, T, sigma, Nstep)
  Dim dt, erdt, ermqdt, u, d, p
  Dim i As Integer, j As Integer
  Dim vvec() As Variant
  ReDim vvec(Nstep)
  dt = T / Nstep
  erdt = Exp(r * dt)
  ermqdt = Exp((r - q) * dt)
  u = Exp(sigma * Sqr(dt))
  d = 1 / u
  p = (ermqdt - d) / (u - d)
  For i = 0 To Nstep
    vvec(i) = Application.Max(S * (u ^ i) * (d ^ (
Nstep - i)) - X, 0)
  Next i
    For j = Nstep - 1 To 0 Step -1
      For i = 0 To j
        vvec(i) = (p * vvec(i + 1) + (1 - p) * vvec
(i)) / erdt
      Next i
    Next j
  CRRBinCall = vvec(0)
End Function
```

Using this function and putting parameters in the function, we can get call option value under different steps. This result is shown below.



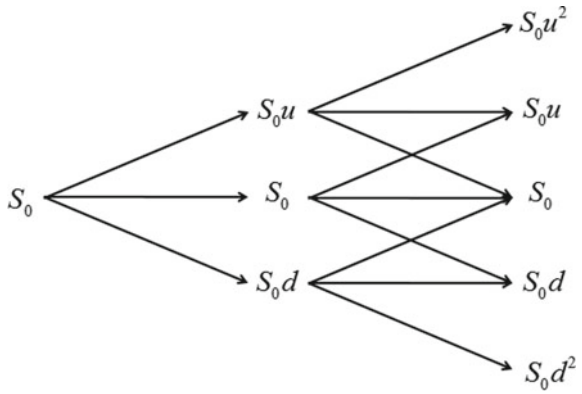
The function in cell B12 is

$$= \text{CRRBinCall}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}, \text{B10})$$

We can see the result in B12 is equal to C12.

5.9.2 Trinomial Tree

Because binomial tree methods are computationally expensive, Kamrad and Ritchken (1991) propose multinomial models. New multinomial models include as special cases existing models. The more general models are shown to be computationally more efficient.



Expressed algebraically, the trinomial tree parameters are

$$\text{u} = e^{\lambda\sigma\sqrt{\delta t}}$$

$$\text{d} = \frac{1}{u}$$

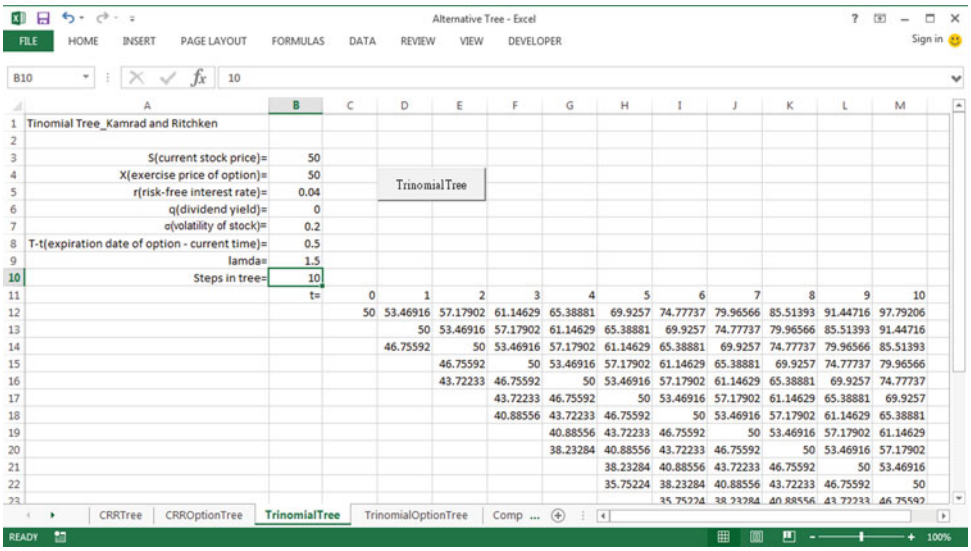
The formula for probability p is given as follows:

$$p_u = \frac{1}{2\lambda^2} + \frac{(r - \sigma^2/2)\sqrt{\delta t}}{2\lambda\sigma}$$

$$p_m = 1 - \frac{1}{\lambda^2}$$

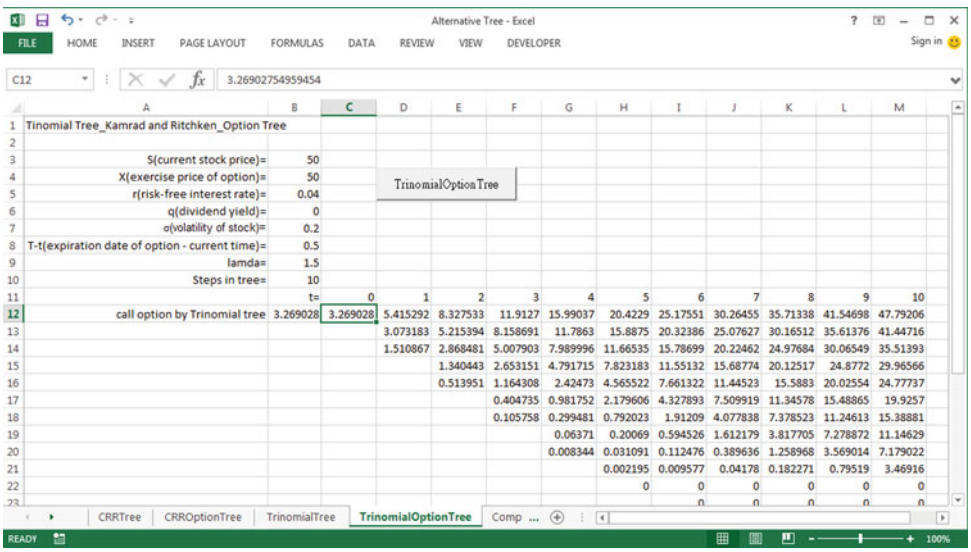
$$p_d = 1 - p_u - p_m$$

If parameter λ is equal to 1, then trinomial tree model reduces to a binomial tree model. Below is the underlying asset price pattern base on trinomial tree model.



We can see this trinomial tree model is also a symmetric tree. The middle price in each period is the same as the initial asset price, 50.

Through the similar rule, we can use this tree to price a call option. At first, we can draw the option tree based on trinomial underlying asset price tree. The result is shown below.

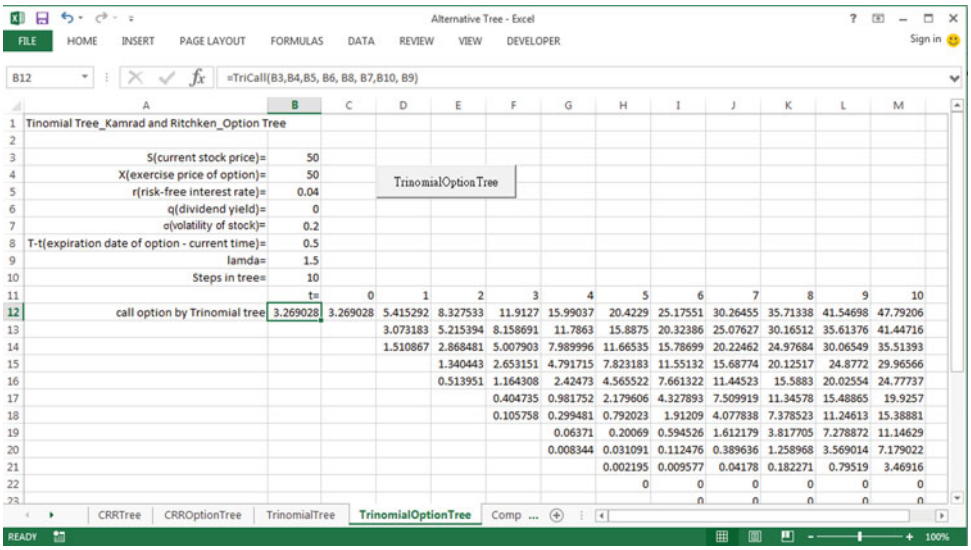


The call option value at time zero is 3.269028 in cell C12. In addition, we also can write a function to price a call option based on trinomial tree model. The function is shown below.

```
'  Returns Trinomial Option Value
Function TriCall(S, X, r, q, T, sigma, Nstep, lamda)
   Dim dt, erdt, ermqdt, u, d, pu, pm, pd
   Dim i As Integer, j As Integer
   Dim vvec() As Variant
   ReDim vvec(2 * Nstep)
   dt = T / Nstep
   erdt = Exp(r * dt)
   ermqdt = Exp((r - q) * dt)
   u = Exp(lamda * sigma * Sqr(dt))
   d = 1 / u
   pu = 1 / (2 * lamda ^ 2) + (r - sigma ^ 2 / 2) * Sqr
(dt) / (2 * lamda * sigma)
   pm = 1 - 1 / (lamda ^ 2)
```

```
  pd = 1 - pu - pm
  For i = 0 To 2 * Nstep
   vvec(i) = Application.Max(S * (d ^ Nstep) * (u ^
i) - X, 0)
  Next i
    For j = Nstep - 1 To 0 Step -1
      For i = 0 To 2 * j
        vvec(i) = (pu * vvec(i + 2) + pm * vvec
(i + 1) + pd * vvec(i)) / erdt
      Next i
    Next j
  TriCall = vvec(0)
End Function
```

Similar data can use in this function and get the same call option at today's price.

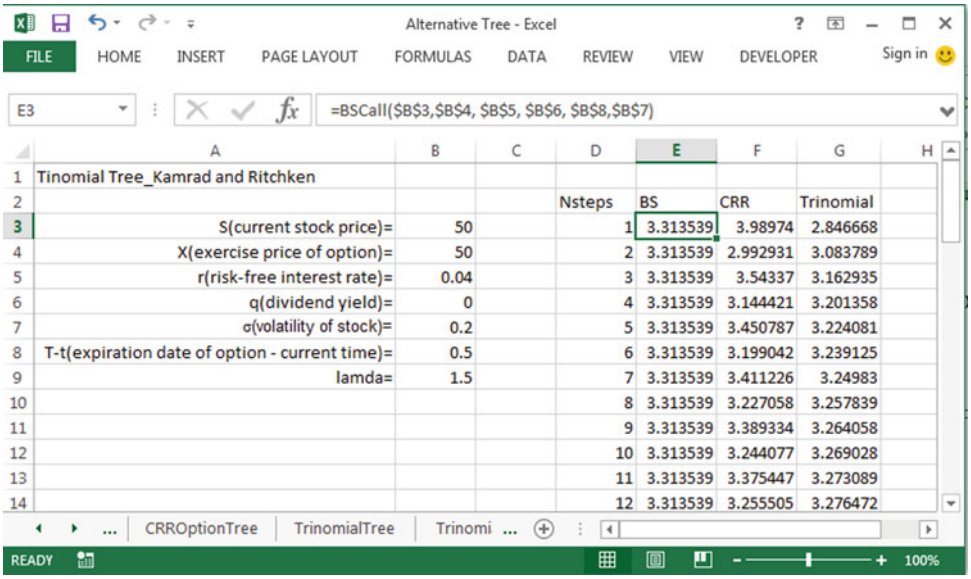


The function in cell B12 is equal to

$$= \text{TriCall}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}, \text{B10}, \text{B9})$$

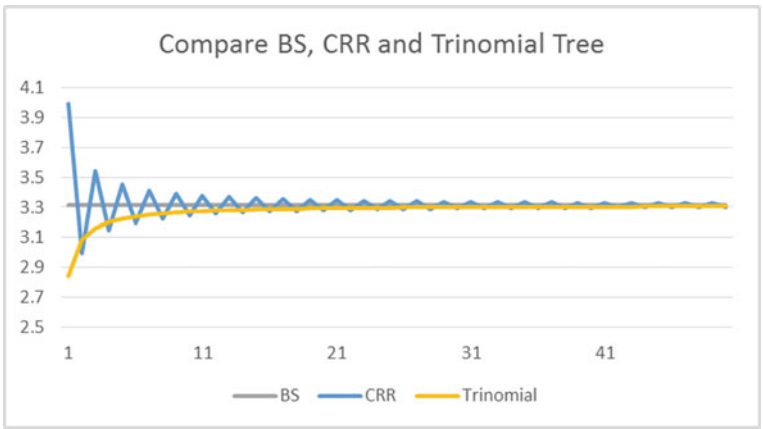
5.9.3 Compare the Option Price Efficiency

In this section, we would like to compare the efficiency between these two methods. In the table below, we represent different numbers of steps 1,2,…,50. And, we represent Black and Scholes, CRR binominal tree, and trinomial tree method results. The following figure is the result.



	A	B	C	D	E	F	G	H
1	Tinomial Tree_Kamrad and Ritchken							
2				Nsteps	BS	CRR	Trinomial	
3	S(current stock price)=	50		1	3.313539	3.98974	2.846668	
4	X(exercise price of option)=	50		2	3.313539	2.992931	3.083789	
5	r(risk-free interest rate)=	0.04		3	3.313539	3.54337	3.162935	
6	q(dividend yield)=	0		4	3.313539	3.144421	3.201358	
7	σ(volatility of stock)=	0.2		5	3.313539	3.450787	3.224081	
8	T-t(expiration date of option - current time)=	0.5		6	3.313539	3.199042	3.239125	
9	lamda=	1.5		7	3.313539	3.411226	3.24983	
10				8	3.313539	3.227058	3.257839	
11				9	3.313539	3.389334	3.264058	
12				10	3.313539	3.244077	3.269028	
13				11	3.313539	3.375447	3.273089	
14				12	3.313539	3.255505	3.276472	

In order to see the result more deeply, we draw the result in the picture. The picture is shown below.



After we increase the number of steps, we can see that trinomial tree method is more quickly convergence to Black and Scholes than CRR binomial tree method.

5.10 Retrieving Option Prices from Yahoo Finance

The following is the URL to retrieve Coca-Cola's option prices:

http://finance.yahoo.com/q/op?s=KO+Options

The following is the URL to retrieve Home Depot's option prices:

http://finance.yahoo.com/q/op?s=HD+Options

The following is the URL to retrieve Microsoft's option prices:

http://finance.yahoo.com/q/op?s=MSFT+Options.

5.11 Summary

In this paper, we demonstrated why Microsoft Excel is a very powerful application and why the Financial Industry in New York City value people that know Microsoft Excel very well. Microsoft Excel gives the business user the ability to create powerful application quickly without relying on the Information Technology (IT) department. Prior to Microsoft Excel, business users would have to rely heavily on the Information Technology department. There are two problems with relying on the IT department. The first problem is that the tools that the IT department was using resulted in a longer development time. The second problem was that the IT department was not as familiar with the business processes as the business users.

Simultaneously, this paper demonstrated, with the aid of Microsoft Excel and decision trees, the binomial option model in a less mathematical fashion. This paper allowed the reader to focus more on the concepts by studying the associated decision trees, which were created by Microsoft Excel. This paper also demonstrates that using Microsoft Excel releases the reader from the computation burden of the binomial option model.

This paper also published the Microsoft Excel VBA code that created the binomial option decision trees. This allows for those who are interested to study the many advanced Microsoft Excel VBA programming concepts that were used to create the decision trees. One major computer science programming concept used by Microsoft Excel VBA is recursive programming. Recursive programming is the ideal of a procedure calling itself many times. Inside the procedure, there are statements to decide when not to call itself.

Appendix 5.1: EXCEL CODE—Binomial Option Pricing Model

```
'/
**************************************************
**********************
'/Essentials of Microsoft Excel 2013 VBA, SAS
'/         and MINITAB 17
'/  for Statistical and Financial Analysis
'/
'/
**************************************************
**********************
Option Explicit
Dim mwbTreeWorkbook As Workbook
Dim mwsTreeWorksheet As Worksheet
Dim mwsCallTree As Worksheet
Dim mwsPutTree As Worksheet
Dim mwsBondTree As Worksheet
Dim mdblPFactor As Double
Dim mBinomialCalc As Long
Dim mOptionType As String
'/
**********************************************
'/Purpose: Keep track the numbers of binomial calc
'/**********************************************
Property Let OptionType(t As String)
  mOptionType = t
End Property
Property Get OptionType() As String
  OptionType = mOptionType
End Property
Property Let BinomialCalc(l As Long)
  mBinomialCalc = l
End Property
Property Get BinomialCalc() As Long
  BinomialCalc = mBinomialCalc
End Property
Property Set TreeWorkbook(wb As Workbook)
  Set mwbTreeWorkbook = wb
End Property
Property Get TreeWorkbook() As Workbook
  Set TreeWorkbook = mwbTreeWorkbook
End Property
Property Set TreeWorksheet(ws As Worksheet)
  Set mwsTreeWorksheet = ws
End Property
Property Get TreeWorksheet() As Worksheet
  Set TreeWorksheet = mwsTreeWorksheet
End Property
Property Set CallTree(ws As Worksheet)
  Set mwsCallTree = ws
End Property
Property Get CallTree() As Worksheet
  Set CallTree = mwsCallTree
End Property
Property Set PutTree(ws As Worksheet)
  Set mwsPutTree = ws
End Property
Property Get PutTree() As Worksheet
  Set PutTree = mwsPutTree
End Property
Property Set BondTree(ws As Worksheet)
  Set mwsBondTree = ws
End Property
Property Get BondTree() As Worksheet
  Set BondTree = mwsBondTree
End Property
Property Let PFactor(r As Double)
  Dim dRate As Double
  dRate = ((1 + r) - Me.txtBinomialD) / (Me.
txtBinomialU - Me.txtBinomialD)
  Let mdblPFactor = dRate
End Property
Property Get PFactor() As Double
  Let PFactor = mdblPFactor
End Property
Private Sub cmdCalculate_Click()
  Me.Hide
  BinomialOption
  Unload Me
End Sub
Private Sub cmdCalculateAmerican_Click()
  Me.Hide
  Me.OptionType = ``American''
  BinomialOption
  Unload Me
End Sub
```

```
Private Sub cmdCalculateEuropean_Click()
  Me.Hide
  Me.OptionType = ``European''
  BinomialOption
  Unload Me
End Sub
Private Sub cmdCancel_Click()
  Unload Me
End Sub
Private Sub UserForm_Initialize()
 With Me
    .txtBinomialS = 100
    .txtBinomialX = 100
    .txtBinomialD = 0.85
    .txtBinomialU = 1.175
    .txtBinomialN = 4
    .txtBinomialr = 0.07
  End With
  Me.Hide
End Sub
Sub BinomialOption()
  Dim wbTree As Workbook
  Dim wsTree As Worksheet
  Dim rColumn As Range
  Dim ws As Worksheet
  Set Me.TreeWorkbook = Workbooks.Add
  Set Me.BondTree = Me.TreeWorkbook.Worksheets.Add
  Set Me.PutTree = Me.TreeWorkbook.Worksheets.Add
  Set Me.CallTree = Me.TreeWorkbook.Worksheets.Add
  Set Me.TreeWorksheet = Me.TreeWorkbook.Worksheets.
Add
  Set rColumn = Me.TreeWorksheet.Range(``a1'')
  With Me
    .BinomialCalc = 0
    .PFactor = Me.txtBinomialr
    .CallTree.Name = ``American Call Option Price''
    .PutTree.Name = ``American Put Option Price''
    .TreeWorksheet.Name = ``Stock Price''
    .BondTree.Name = ``Bond''
  End With
  DecitionTree rCell:=rColumn, nPeriod:=Me.txtBino-
mialN + 1, _
        dblPrice:=Me.txtBinomialS, sngU:=Me.
txtBinomialU, _
        sngD:=Me.txtBinomialD
  DecitionTreeFormat
  TreeTitle wsTree:=Me.TreeWorksheet, sTitle:
=``Stock Price ''
  TreeTitle wsTree:=Me.CallTree, sTitle:=Me.
OptionType & `` Call Option Pricing''
  TreeTitle wsTree:=Me.PutTree, sTitle:=Me.
OptionType & `` Put Option Pricing''
  TreeTitle wsTree:=Me.BondTree, sTitle:
=``Bond Pricing''
  Application.DisplayAlerts = False
  For Each ws In Me.TreeWorkbook.Worksheets
    If Left(ws.Name, 5) = ``Sheet'' Then
      ws.Delete
    Else
      ws.Activate
      ActiveWindow.DisplayGridlines = False
    End If
  Next
  Application.DisplayAlerts = True
  Me.TreeWorksheet.Activate
End Sub
Sub TreeTitle(wsTree As Worksheet, sTitle As String)
  wsTree.Range(``A1:a5'').EntireRow.Insert (
xlShiftDown)
  With wsTree
    With .Cells(1)
       .Value = sTitle
       .Font.Size = 20
       .Font.Italic = True
    End With
    With .Cells(2, 1)
      .Value = ``Decision Tree''
      .Font.Size = 16
      .Font.Italic = True
    End With
    With .Cells(3, 1)
      .Value = ``Price = '' & Me.txtBinomialS & _
          ``,Exercise = '' & Me.txtBinomialX & _
          ``,U = '' & Me.txtBinomialU & _
          ``,D = '' & Me.txtBinomialD & _
          ``,N = '' & Me.txtBinomialN
      .Font.Size = 14
    End With
    With .Cells(4, 1)
      .Value = ``Number of calculations: '' & Me.
BinomialCalc
      .Font.Size = 14
    End With
  End With
End Sub
Sub BondDecisionTree(rPrice As Range, arCell As Vari-
ant, iCount As Long)
  Dim rBond As Range
  Dim rPup As Range
  Dim rPDown As Range
  Set rBond = Me.BondTree.Cells(rPrice.Row, rPrice.
Column)
  Set rPup = Me.BondTree.Cells(arCell(iCount - 1).
```

```
Row, arCell(iCount - 1).Column)
   Set rPDown = Me.BondTree.Cells(arCell(iCount).
Row, arCell(iCount).Column)
   If rPup.Column = Me.TreeWorksheet.UsedRange.Col-
umns.Count Then
     rPup.Value = (1 + Me.txtBinomialr) ^ (rPup.
Column - 1)
     rPDown.Value = rPup.Value
   End If
   rBond.Value = (1 + Me.txtBinomialr) ^ (rBond.
Column - 1)
       rPDown.Borders(xlBottom).LineStyle = xlCon-
tinuous
       With rPup
          .Borders(xlBottom).LineStyle = xlContinu-
ous
          .Offset(1, 0).Resize((rPDown.Row - rPup.
Row), 1). _
             Borders(xlEdgeLeft).LineStyle = xlCon-
tinuous
       End With
End Sub

Sub PutDecisionTree(rPrice As Range, arCell As Vari-
ant, iCount As Long)
   Dim rCall As Range
   Dim rPup As Range
   Dim rPDown As Range
   Set rCall = Me.PutTree.Cells(rPrice.Row, rPrice.
Column)
   Set rPup = Me.PutTree.Cells(arCell(iCount - 1).
Row, arCell(iCount - 1).Column)
   Set rPDown = Me.PutTree.Cells(arCell(iCount).Row,
arCell(iCount).Column)
   If rPup.Column = Me.TreeWorksheet.UsedRange.Col-
umns.Count Then
     rPup.Value = WorksheetFunction.Max(Me.
txtBinomialX - arCell(iCount - 1), 0)
     rPDown.Value = WorksheetFunction.Max(Me.
txtBinomialX - arCell(iCount), 0)
   End If
   If Me.OptionType = ``American'' Then
      'American Option
      'Striket price - put option price for perion N
      rCall.Value = WorksheetFunction.Max(Me.
txtBinomialX - arCell(iCount - 1) / Me.txtBinomialU, _
               (Me.PFactor * rPup + (1 - Me.PFac-
tor) * rPDown) / (1 + Me.txtBinomialr))
   Else
      'European Option
      rCall.Value = (Me.PFactor * rPup + (1 - Me.
PFactor) * rPDown) / (1 + Me.txtBinomialr)
   End If
       rPDown.Borders(xlBottom).LineStyle = xlCon-
tinuous
       With rPup
          .Borders(xlBottom).LineStyle = xlContinu-
ous
          .Offset(1, 0).Resize((rPDown.Row - rPup.
Row), 1). _
             Borders(xlEdgeLeft).LineStyle = xlCon-
tinuous
       End With
End Sub
Sub CallDecisionTree(rPrice As Range, arCell As Vari-
ant, iCount As Long)
   Dim rCall As Range
   Dim rCup As Range
   Dim rCDown As Range
   Set rCall = Me.CallTree.Cells(rPrice.Row, rPrice.
Column)
   Set rCup = Me.CallTree.Cells(arCell(iCount - 1).
Row, arCell(iCount - 1).Column)
   Set rCDown = Me.CallTree.Cells(arCell(iCount).
Row, arCell(iCount).Column)
   If rCup.Column = Me.TreeWorksheet.UsedRange.Col-
umns.Count Then
     rCup.Value = WorksheetFunction.Max(arCell
(iCount - 1) - Me.txtBinomialX, 0)
     rCDown.Value = WorksheetFunction.Max(arCell
(iCount) - Me.txtBinomialX, 0)
   End If
   If Me.OptionType = ``American'' Then
       'Call option price for Period N - strike price
       rCall.Value = WorksheetFunction.Max(arCell
(iCount - 1) / Me.txtBinomialU - Me.txtBinomialX, _
                (Me.PFactor * rCup + (1 - Me.PFac-
tor) * rCDown) / (1 + Me.txtBinomialr))
   Else
       'European
       rCall.Value = (Me.PFactor * rCup + (1 - Me.
PFactor) * rCDown) / (1 + Me.txtBinomialr)
   End If
       rCDown.Borders(xlBottom).LineStyle = xlCon-
tinuous
       With rCup
          .Borders(xlBottom).LineStyle = xlContinu-
ous
          .Offset(1, 0).Resize((rCDown.Row - rCup.
Row), 1). _
             Borders(xlEdgeLeft).LineStyle = xlCon-
tinuous
       End With
```

```
End Sub
Sub DecitionTreeFormat()
  Dim rTree As Range
  Dim nColumns As Integer
  Dim rLast As Range
  Dim rCell As Range
  Dim lCount As Long
  Dim lCellSize As Long
  Dim vntColumn As Variant
  Dim iCount As Long
  Dim lTimes As Long
  Dim arCell() As Range
  Dim sFormatColumn As String
  Dim rPrice As Range
  Application.StatusBar = ``Formatting Tree.. ''
  Set rTree = Me.TreeWorksheet.UsedRange
  nColumns = rTree.Columns.Count
  Set rLast = rTree.Columns(nColumns).EntireColumn.
SpecialCells(xlCellTypeConstants, 23)
  lCellSize = rLast.Cells.Count
  For lCount = nColumns To 2 Step -1
    sFormatColumn = rLast.Parent.Columns(lCount).
EntireColumn.Address
    Application.StatusBar = ``Format-
ting column '' & sFormatColumn
    ReDim vntColumn(1 To (rLast.Cells.Count / 2), 1)
    Application.StatusBar = ``Assigning val-
ues to array for column '' & _
      rLast.Parent.Columns(lCount).EntireColumn.
Address
    vntColumn = rLast.Offset(0, -1).EntireColumn.
Cells(1).Resize(rLast.Cells.Count / 2, 1)
    rLast.Offset(0, -1).EntireColumn.ClearContents
    ReDim arCell(1 To rLast.Cells.Count)
    lTimes = 1
    Application.StatusBar = ``Assigning cells to ar-
rays. Total number of cells: '' & lCellSize
    For Each rCell In rLast.Cells
      Application.Sta-
tusBar = ``Array to column '' & sFor-
matColumn & `` Cells '' & rCell.Row
      Set arCell(lTimes) = rCell
      lTimes = lTimes + 1
    Next
    lTimes = 1
    Application.StatusBar = ``Format-
ting leaves for column '' & sFormatColumn
    For iCount = 2 To lCellSize Step 2
      Application.StatusBar = ``Format-
ting leaves for cell '' & arCell(iCount).Row
      If rLast.Cells.Count <> 2 Then
        Set rPrice = arCell(iCount).Offset(-
1 * ((arCell(iCount).Row - arCell(iCount - 1).Row) /
 2), -1)
        rPrice.Value = vntColumn(lTimes, 1)
      Else
        Set rPrice = arCell(iCount).Offset(-
1 * ((arCell(iCount).Row - arCell(iCount - 1).Row) /
 2), -1)
        rPrice.Value = vntColumn
      End If
      arCell(iCount).Borders(xlBottom).
LineStyle = xlContinuous
      With arCell(iCount - 1)
        .Borders(xlBottom).LineStyle = xlContinu-
ous
        .Offset(1, 0).Resize((arCell(iCount).
Row - arCell(iCount - 1).Row), 1). _
          Borders(xlEdgeLeft).LineStyle = xlCon-
tinuous
      End With
      lTimes = 1 + lTimes
      CallDecisionTree rPrice:=rPrice, arCell:
=arCell, iCount:=iCount
      PutDecisionTree rPrice:=rPrice, arCell:
=arCell, iCount:=iCount
      BondDecisionTree rPrice:=rPrice, arCell:
=arCell, iCount:=iCount
    Next
    Set rLast = rTree.Columns(lCount - 1).EntireCol-
umn.SpecialCells(xlCellTypeConstants, 23)
    lCellSize = rLast.Cells.Count
  Next ' / outer next
  rLast.Borders(xlBottom).LineStyle = xlContinuous
  Application.StatusBar = False
End Sub
'/
**************************************************
*****************
'/Purpse: To calculate the price value of ev-
ery state of the binomial
'/     decision tree
'/
**************************************************
*****************
Sub DecitionTree(rCell As Range, nPeriod As Integer, _
        dblPrice As Double, sngU As Sin-
gle, sngD As Single)
  Dim lIteminColumn As Long
```

```
  If Not nPeriod = 1 Then
    'Do Up
    DecitionTree rCell:=rCell.Offset(0, 1), nPer-
iod:=nPeriod - 1, _
         dblPrice:=dblPrice * sngU, sngU:=sngU, _
         sngD:=sngD
    'Do Down
    DecitionTree rCell:=rCell.Offset(0, 1), nPer-
iod:=nPeriod - 1, _
         dblPrice:=dblPrice * sngD, sngU:=sngU, _
         sngD:=sngD
  End If
  lIteminColumn = WorksheetFunction.CountA(rCell.
EntireColumn)
  If lIteminColumn = 0 Then
    rCell = dblPrice
  Else
    If nPeriod <> 1 Then
      rCell.EntireColumn.Cells(lIteminColumn + 1) =
dblPrice
    Else
      rCell.EntireColumn.Cells(((lIteminColumn + 1)
 * 2) - 1) = dblPrice
    End If
  End If
  Me.BinomialCalc = Me.BinomialCalc + 1
  Application.StatusBar = ``The number of bino-
mial calcs are : '' & Me.BinomialCalc
End Sub
```

References

Benninga, S. *Financial Modeling*. Cambridge, MA: MIT Press, 2000.

Benninga, S. *Financial Modeling*. Cambridge, MA: MIT Press, 2008.

Black, F. and M. Scholes. "The Pricing of Options and Corporate Liabilities." *Journal of Political Economy*, v. 31 (May–June 1973), pp. 637–659.

Cox, J., S. A. Ross and M. Rubinstein. "Option Pricing: A Simplified Approach." *Journal of Financial Economics*, v. 7 (1979), pp. 229–263.

Daigler, R. T. *Financial Futures and Options Markets Concepts and Strategies*. New York: Harper Collins, 1994.

Jarrow, R. and S. TurnBull. *Derivative Securities*. Cincinnati: South-Western College Publishing, 1996.

Lee, C. F., AC Lee and John Lee. *Handbook of Quantitative Finance and Risk management*. New York, NY: Springer, 2010.

Lee, C. F. and A. C. Lee. *Encyclopedia of Finance*. 2nd edition. New York, NY: Springer, 2013.

Lee, C. F., J. C. Lee and A. C. Lee (2000). *Statistics for Business and Financial Economics*. 3rd edition. Springer, New York, 2000.

Lee, J. C., C. F. Lee, R. S. Wang and T. I. Lin. "On the Limit Properties of Binomial and Multinomial Option Pricing Models: Review and Integration," in *Advances in Quantitative Analysis of Finance and Accounting New Series*, Vol. 1. Singapore: World Scientific, 2004.

Lee, C. F., C. M. Tsai and A. C. Lee, "Asset pricing with disequilibrium price adjustment: theory and empirical evidence." *Quantitative Finance*. Volume 13, Number 2, Pages 227–240.

Lee, J. C., "Using Microsoft Excel and Decision trees to Demonstrate the Binomial Option Pricing Model." *Advances in Investment Analysis and Portfolio Management*, v. 8 (2001), pp. 303–329.

Lo, A. W. and J. Wang. "Trading Volume: Definition, Data Analysis, and Implications of Portfolio Theory." *Review of Financial Studies*, v. 13 (2000), pp. 257–300.

Rendleman, R. J., Jr. and B. J. Barter. "Two-State Option Pricing." *Journal of Finance*, v. 34(5) (December 1979), pp. 1093–1110.

Wells, E. and S. Harshbarger. *Microsoft Excel 97 Developer's Handbook*. Redmond, WA: Microsoft Press, 1997.

Walkenbach, J. *Excel 2003 Power Programming with VBA*. Indianapolis, IN: Wiley Publishing, Inc., 2003.

Microsoft Excel Approach to Estimating Alternative Option Pricing Models

6

6.1 Introduction

This chapter shows how Microsoft Excel can be used to estimate call and put options for (a) Black–Scholes model for individual stock, (b) Black–Scholes model for stock indices, and (c) Black–Scholes model for currencies. In addition, we are going to present how an Excel program can be used to estimate American options. Section 6.2 presents an option pricing model for Individual Stocks, Sect. 6.3 presents an option pricing model for Stock Indices, Sect. 6.4 presents option pricing model for Currencies, Sect. 6.5 presents Bivariate Normal Distribution Approach to calculate American call options, Sect. 6.6 presents the Black's approximation method to calculate American Call options, Sect. 6.6 presents how to evaluate American call option when dividend yield is known, and Sect.6.9 summarizes this chapter. Appendix 6.1 defines the Bivariate Normal Probability Density Function and Appendix 6.2 presents the Excel program to calculate the American call option when dividend payments are known.

6.2 Option Pricing Model for Individual Stock

The call option formula for an individual stock can be defined as

$$C = SN(d_1) - Xe^{-r(T)}N(d_2), \tag{6.1}$$

where

$$d_1 = \frac{\ln\left(\frac{S}{X}\right) + \left(r + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}}$$

$$d_2 = \frac{\ln\left(\frac{S}{X}\right) + \left(r - \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}$$

C = price of the call option.
S = current price of the stock.
X = exercise price of the option.
e = 2.71828…
r = short-term interest rate (T-Bill rate) = R_f.
T = time to expiration of the option, in years
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1,2)
σ^2 = variance of the stock rate of return.

The put option formula can be defined as

$$P = Xe^{-r(T)}N(-d_2) - SN(-d_1), \tag{6.2}$$

where

P = price of the put option.

The other notations have been defined in Eq. (6.1).

Assume S = 42, X = 40, r = 0.1, σ = 0.2, and T = 0.5. The following shows how to set up Microsoft Excel to solve the problem:

This chapter was written by Professor Cheng F. Lee and Dr. Ta-Peng Wu of Rutgers University.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_6

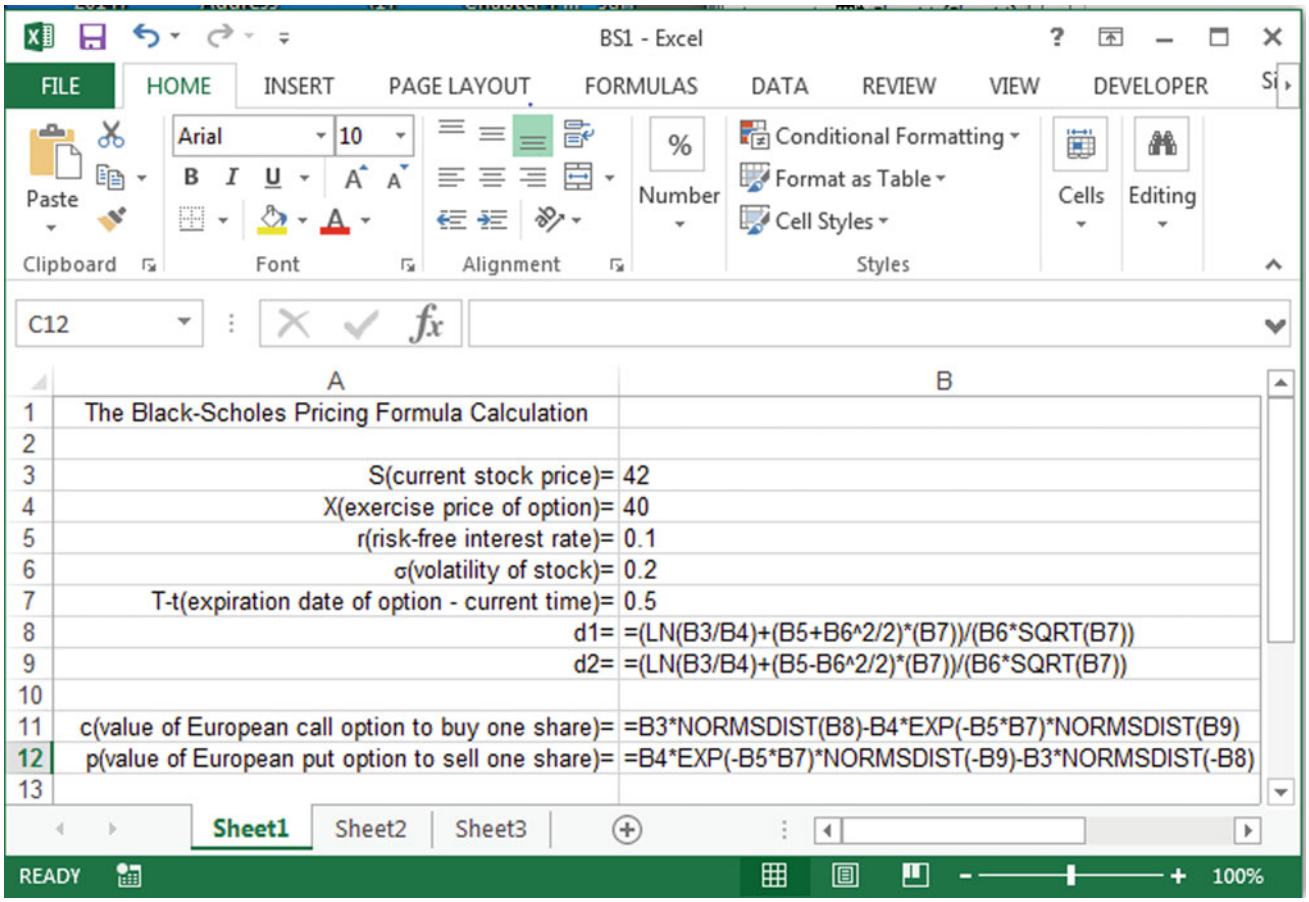


Fig. 6.1 The inputs and excel functions of European call and put options

The following shows the answer to the problem in Microsoft Excel: (Fig. 6.2)

From the Excel output, we find that the prices of a call option and a put option are $4.76 and $0.81, respectively.

6.3 Option Pricing Model for Stock Indices

The call option formula for a stock index can be defined as

$$C = Se^{-q(T)}N(d_1) - Xe^{-r(T)}N(d_2), \tag{6.3}$$

where

$$d_1 = \frac{\ln(S/X) + \left(r - q + \frac{\sigma^2}{2}\right)(T)}{\sigma\sqrt{T}}$$

$$d_2 = \frac{\ln(S/X) + \left(r - q - \frac{\sigma^2}{2}\right)(T)}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}$$

q = dividend yield;
S = value of index;
X = exercise price;
r = short-term interest rate (T-Bill rate) = R_f;
T = time to expiration of the option, in years;
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1,2);
σ^2 = variance of the stock rate of return.

The put option formula for a stock index can be defined as

$$P = Xe^{-r(T)}N(-d_2) - Se^{-q(T)}N(-d_1), \tag{6.4}$$

where

P = the price of the put option.

The other notations have been defined in Eq. (6.3).

Assume that S = 950, X = 900, r = 0.06, σ = 0.15, q = 0.03, and T = 2/12. The following shows how to set up Microsoft Excel to solve the problem:

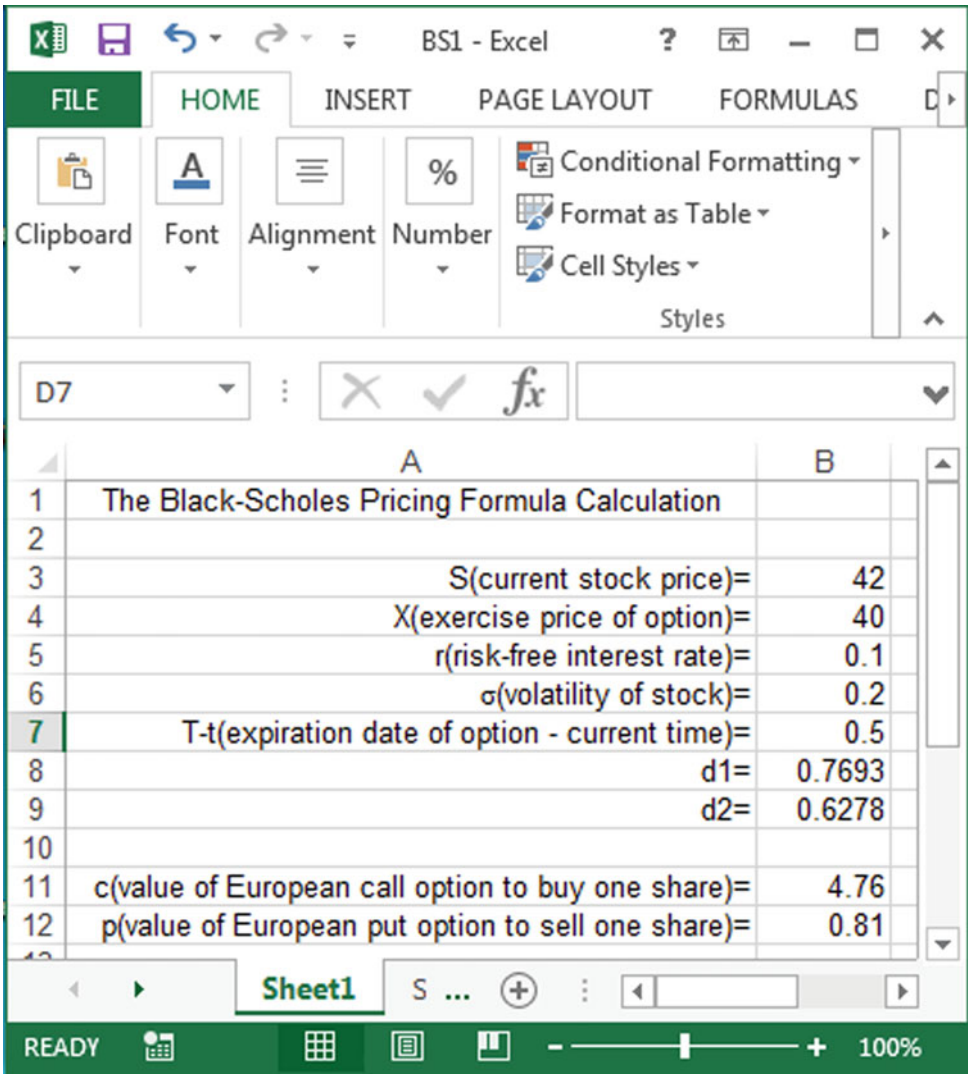


Fig. 6.2 Results for functions contained in Fig. 6.1

The following shows the answer to the problem in Microsoft Excel: (Fig. 6.4).

From the Excel output, we find that the prices of a call option and a put option are \$59.26 and \$5.01, respectively.

6.4 Option Pricing Model for Currencies

The call option formula for a currency can be defined as

$$C = Se^{-r_f(T)}N(d_1) - Xe^{-r(T)}N(d_2),$$

where

$$d_1 = \frac{\ln(S/X) + \left(r - r_f + \frac{\sigma^2}{2}\right)(T)}{\sigma\sqrt{T}}$$

$$d_2 = \frac{\ln(S/X) + \left(r - r_f - \frac{\sigma^2}{2}\right)(T)}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}$$

S = spot exchange rate;
r = risk-free rate for domestic country;
X = exercise price;
T = time to expiration of the option, in years;
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1,2);
σ = standard deviation of spot rate.

The put option formula for a currency can be defined as

$$P = Xe^{-r(T)}N(-d_2) - Se^{-r_f(T)}N(-d_1), \qquad (6.5)$$

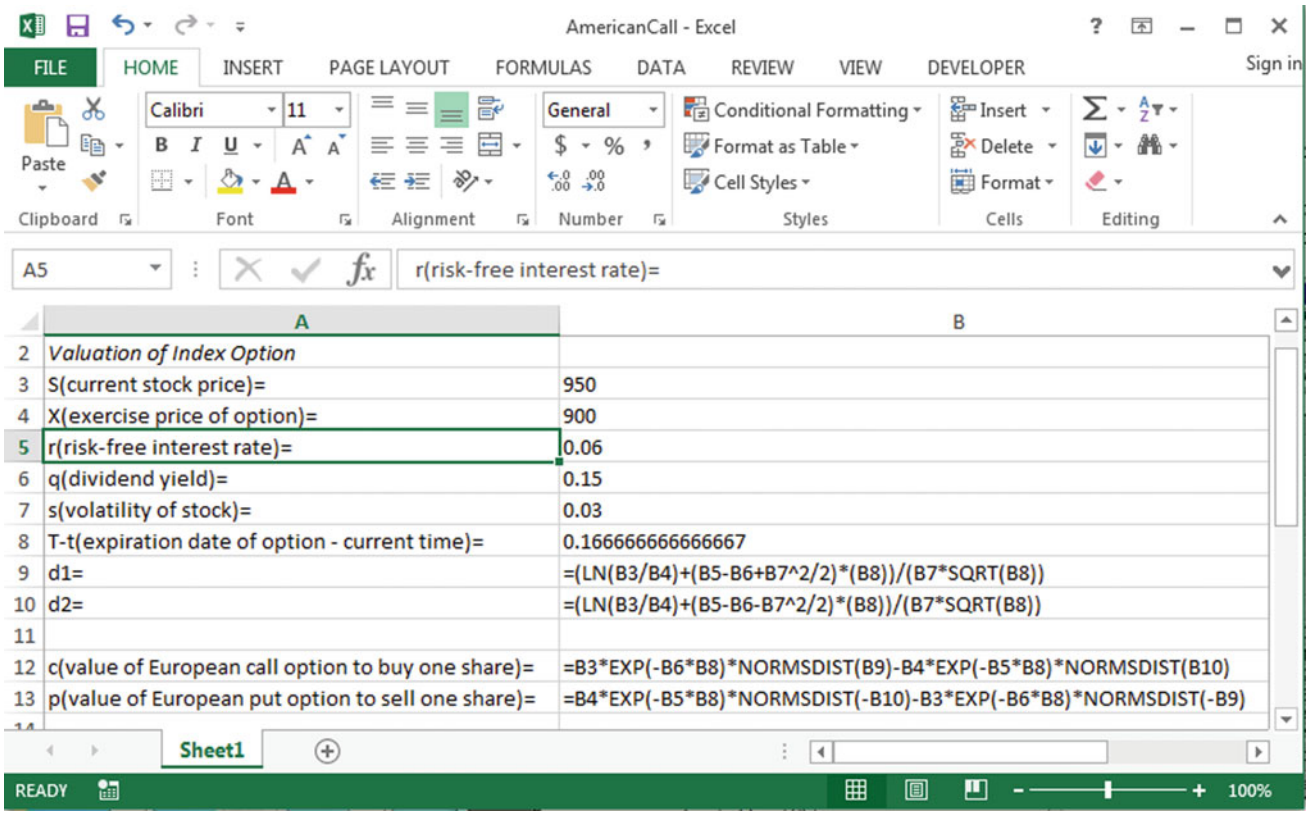


Fig. 6.3 The inputs and Excel functions of European call and put options

where

$P =$ the price of the put option.

The other notations have been defined in Eq. (6.5).

Assume that S = 130, X = 125, r = 0.06, $r_f = 0.02$, $\sigma = 0.15$, and T = 4/12. The following shows how to set up Microsoft Excel to solve the problem:

The following shows the answer to the problem in Microsoft Excel: (Fig. 6.6).

From the Excel output, we find that the prices of a call option and a put option are $8.43 and $1.82, respectively.

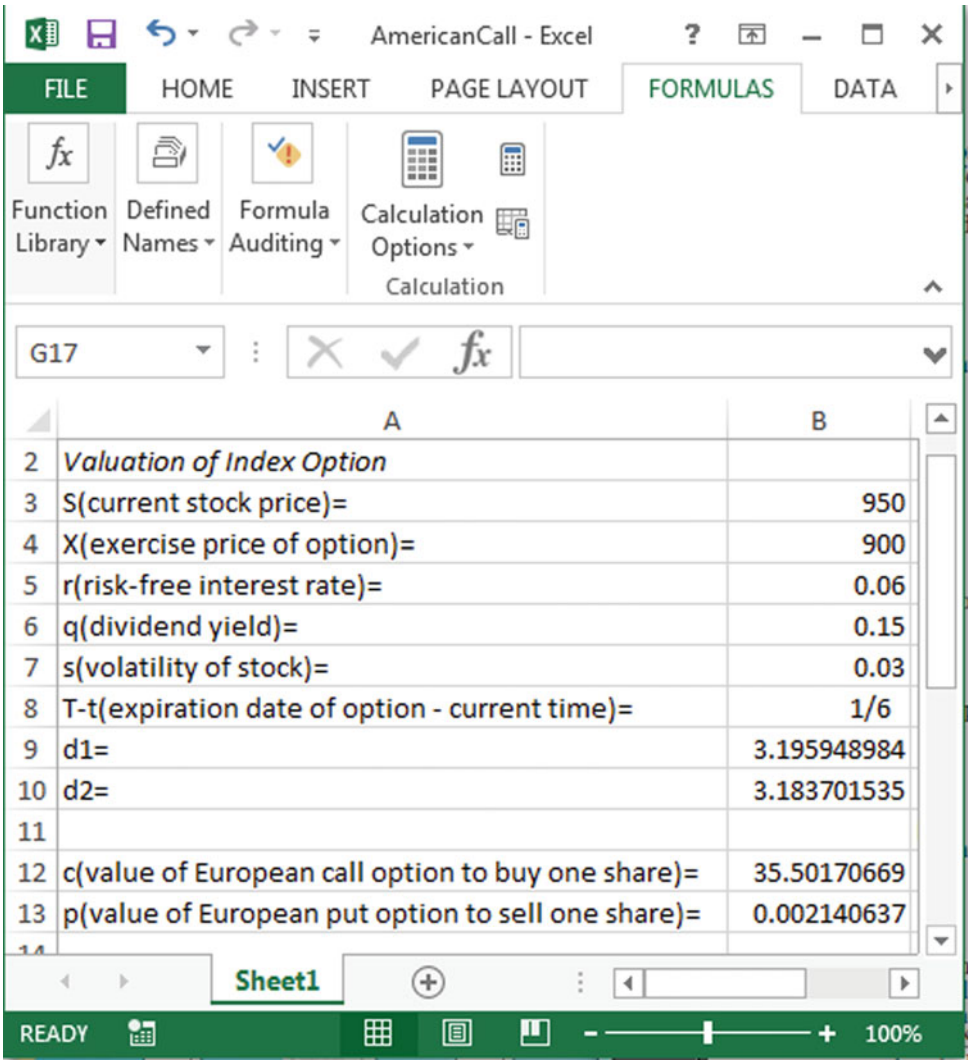
6.5 Futures Options

Black (1976) showed that the original call option formula for stocks can be easily modified to be used in pricing call options on futures. The formula is

$$C(T, F, \sigma^2, X, r) = e^{-rT}[FN(d_1) - XN(d_2)], \tag{6.6}$$

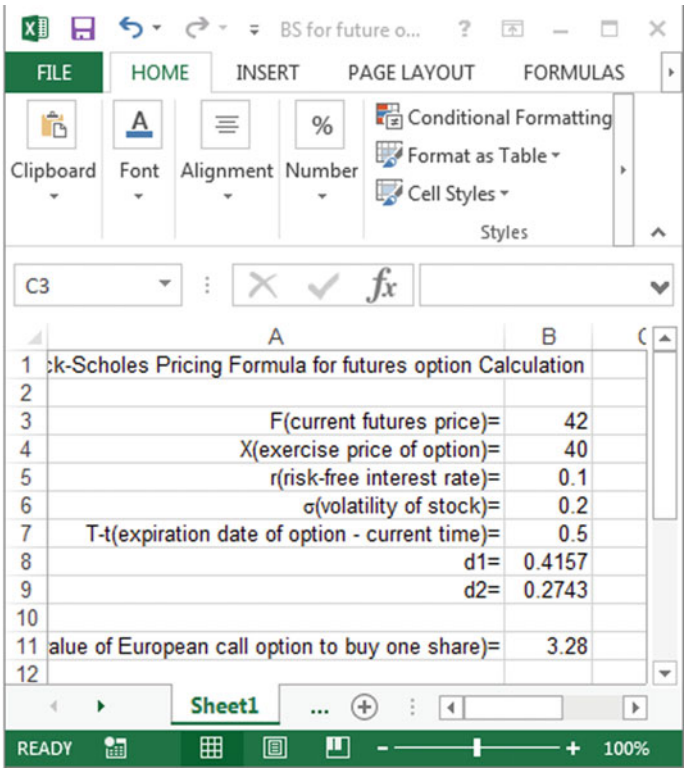
$$d_1 = \frac{\ln(F/X) + \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}}, \tag{6.7}$$

Fig. 6.4 Results for functions contained in Fig. 6.3



$$d_2 = \frac{\ln(F/X) - \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}}. \tag{6.8}$$

In Eq. (6.7), F now denotes the current futures price. The other four variables are as before—time-to-maturity, volatility of the underlying futures price, exercise price, and risk-free rate. Note that Eq. (6.7) differs from Eq. (6.1) only in one respect: by substituting $e^{-rT}F$ for S in the original Eq. (6.1), Eq. (6.7) is obtained. This holds because the investment in a futures contract is zero, which causes the interest rate in Eqs. (6.8) and (6.9) to drop out. The following Excel results are obtained by substituting F = 42, X = 40, r = 0.1,σ = 0.2, T-t = 0.5, d1 = 0.4157,and d2 = 0.2743 into Eqs. (6.7), (6.8), and (6.9).



6.6 Using Bivariate Normal Distribution Approach to Calculate American Call Options

Following Chap. 19 of Lee et.al (2013), the call option formula for American options for a stock that pays a dividend, and there is at least one known dividend, can be defined as

$$\begin{aligned} C(S,T,X) = {} & S^x[N_1(b_1) + N_2(a_1, -b_1; -\sqrt{t/T})] \\ & - Xe^{-rt}[N_1(b_2)e^{r(T-t)} + N_2(a_2, -b_2; -\sqrt{t/T})] \\ & + De^{-rt}N_1(b_2), \end{aligned} \tag{6.9}$$

where

$$a1 = \frac{\ln\left(\frac{S^x}{X}\right) + \left(r + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}}, a2 = a1 - \sigma\sqrt{T} \tag{6.10}$$

$$b1 = \frac{\ln\left(\frac{S^x}{S_t^*}\right) + \left(r + \frac{1}{2}\sigma^2\right)t}{\sigma\sqrt{t}}, b2 = b1 - \sigma\sqrt{t} \tag{6.11}$$

$$S^x = S - De^{-rT}, \tag{6.12}$$

S^x represents the corrected stock net price of the present value of the promised dividend per share (D); t represents the time dividend to be paid.

S_t^* is the ex-dividend stock price for which

$$C(S_t^*, T-t) = S_t^* + D - X.$$

Both $N_1(b_1)$ and $N_2(b_2)$ represent the cumulative univariate normal density function. $N_2(a, b; \rho)$ is the cumulative bivariate normal density function with upper integral limits a and b and correlation coefficient $\rho = -\sqrt{t/T}$.

If we want to calculate the call option value of the American option, we need first to calculate a_1 and b_1. For calculating a_1 andb_1, we need to first calculate S^x and S_t^*. The calculation of S^x can be found in Eq. 6.9. The calculation will be explained in the following example from Chap. 19 of Lee et.al (2013).

An American call option whose exercise price is $48 has an expiration time of 90 days. Assume the risk-free rate of interest is 8% annually, the underlying price is $50, the standard deviation of the rate of return of the stock is 20%, and the stock pays a dividend of $2 exactly for 50 days. (a) What is the European call value? (b) Can the early exercise price predicted? (c) What is the value of the American call?

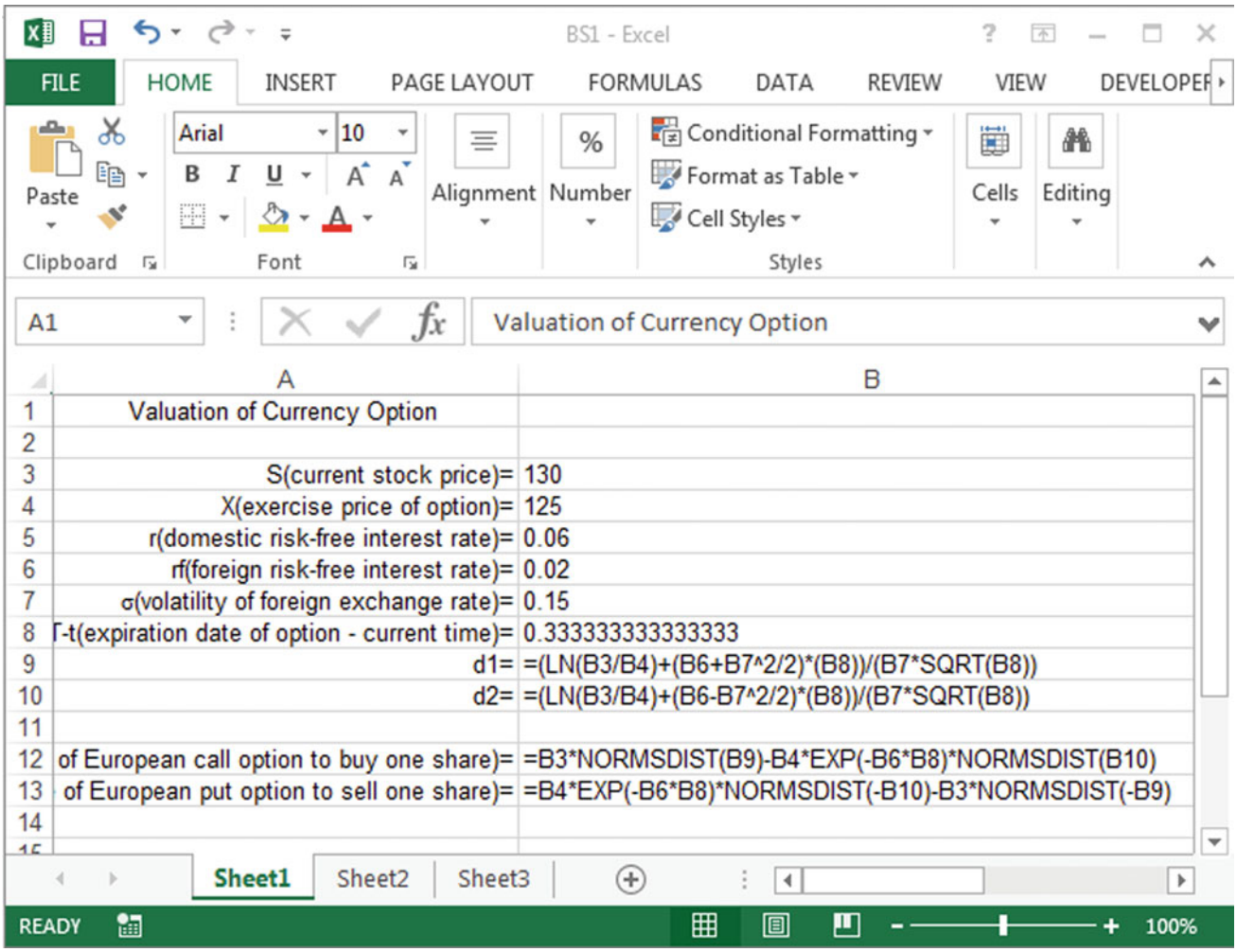


Fig. 6.5 The inputs and Excel functions of European Call and Put options

(a) The current stock net price of the present value of the promised dividend is

$$S^x = 50 - 2e^{-0.08(50/365)} = 48.0218.$$

The European call value can be calculated as

$$C = (48.0218)N(d_1) - 48e^{-0.08(90/365)}N(d_2),$$

where

$$d_1 = \frac{[\ln(48.208/48) + (0.08 + 0.5(0.20)^2)(90/365)]}{.20\sqrt{90/365}} = 0.25285$$

$$d_2 = 0.292 - 0.0993 = 0.15354.$$

From the standard normal table, we obtain

$$N(0.25285) = 0.5 + .3438 = 0.599809$$
$$N(0.15354) = 0.5 + .3186 = 0.561014.$$

So the European call value is.

$$C = (48.516)(0.599809) - 48(0.980)(0.561014)$$
$$= 2.40123.$$

(b) The present value of the interest income that would be earned by deferring exercise until expiration is

$$X(1 - e^{-r(T-t)}) = 48(1 - e^{-0.08(90-50)/365})$$
$$= 48(1 - 0.991) = 0.432.$$

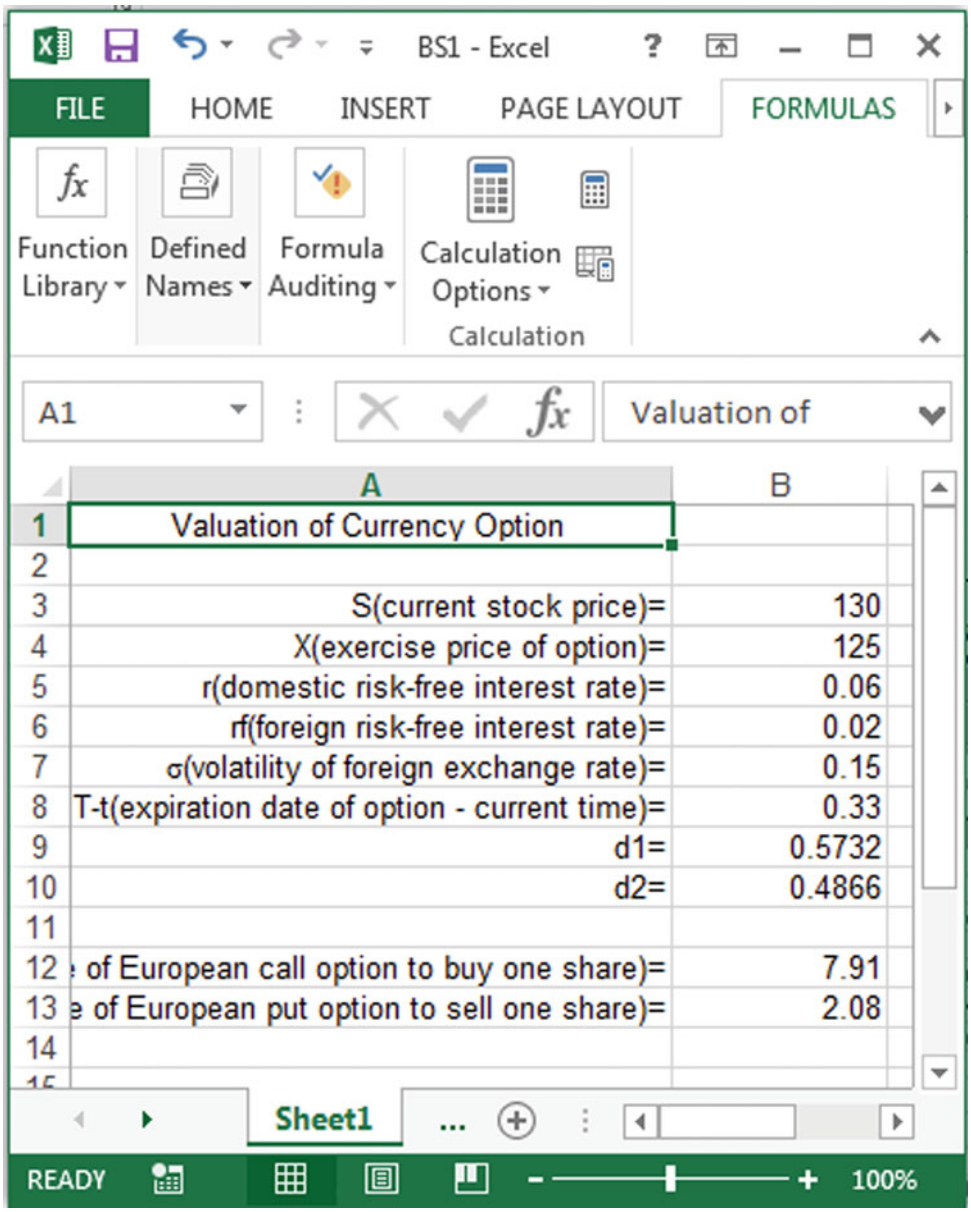


Fig. 6.6 Results for functions contained in Fig. 6.5

Since $d = 2 > 0.432$, therefore, the early exercise is not precluded.

(c) The value of the American call is now calculated as

$$C = 48.208[N_1(b_1) + N_2(a_1, -b_1; \sqrt{50/90})] - 48e^{-0.08(90/365)}[N_1(b_2)e^{-0.08(40/365)} + N_2(a_2, -b_2; -\sqrt{50/90})] + 2e^{-0.08(50/365)}N_1(b_2) \quad (6.13)$$

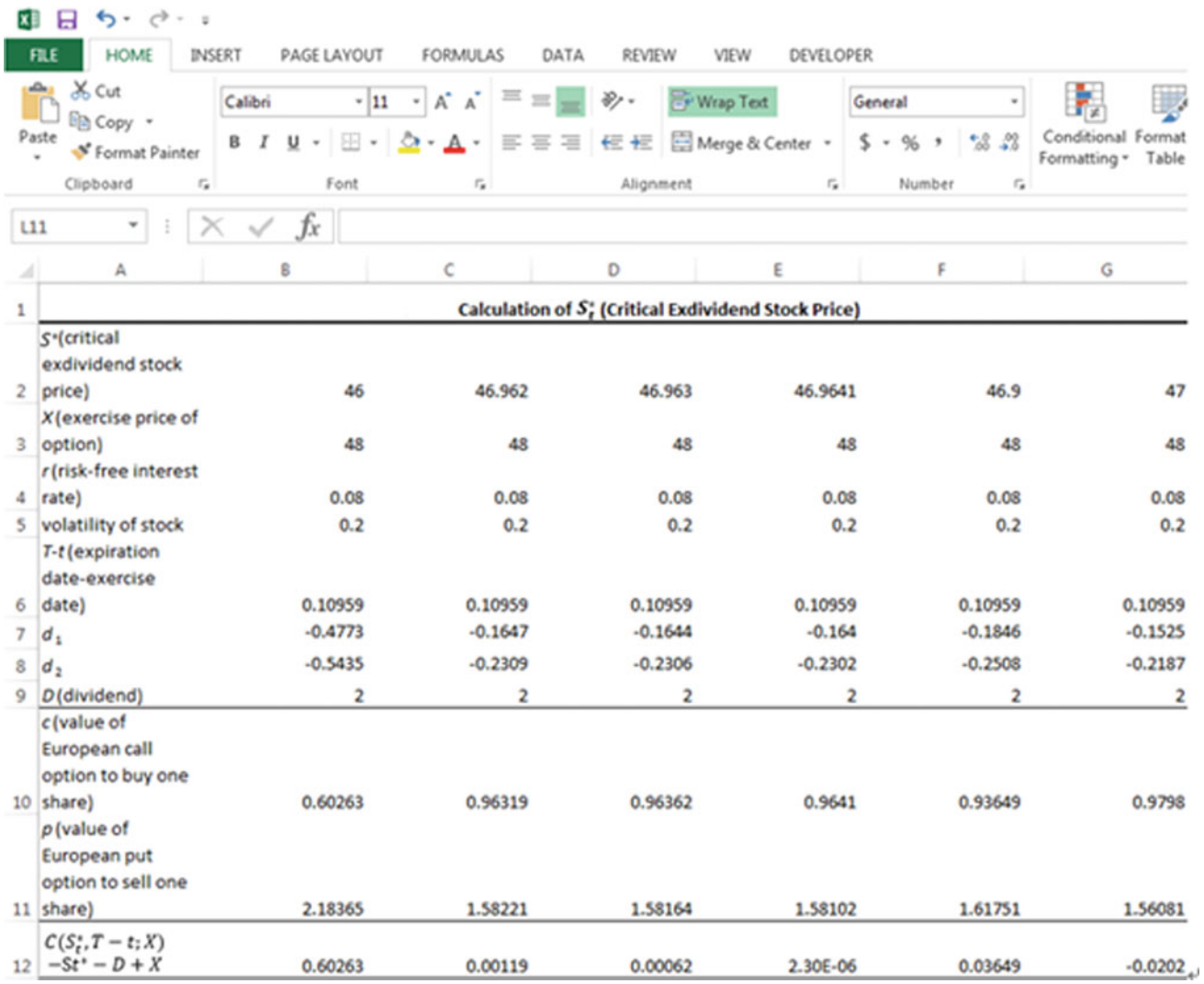
since both b_1 and b_2 depend on the critical ex-dividend stock price S_t^*, which can be determined by

$$C(S_t^*, 40/365; 48) = S_t^* + 2 - 48.$$

By using trial and error, we find that $S_t^* = 46.9641$. An Excel program used to calculate this value is presented in Fig. 6.7.

Substituting $S^x = 48.208$, $X = \$48$ and S_t^* into Eqs. (6.8) and (6.9), we can calculate a_1, a_2, b_1, and b_2:

$$a_1 = d_1 = 0.25285.$$
$$a_2 = d_2 = 0.15354.$$



	A	B	C	D	E	F	G
1	Calculation of S_t^* (Critical Exdividend Stock Price)						
2	S^*(critical exdividend stock price)	46	46.962	46.963	46.9641	46.9	47
3	X(exercise price of option)	48	48	48	48	48	48
4	r(risk-free interest rate)	0.08	0.08	0.08	0.08	0.08	0.08
5	volatility of stock	0.2	0.2	0.2	0.2	0.2	0.2
6	T-t(expiration date-exercise date)	0.10959	0.10959	0.10959	0.10959	0.10959	0.10959
7	d_1	-0.4773	-0.1647	-0.1644	-0.164	-0.1846	-0.1525
8	d_2	-0.5435	-0.2309	-0.2306	-0.2302	-0.2508	-0.2187
9	D(dividend)	2	2	2	2	2	2
10	c(value of European call option to buy one share)	0.60263	0.96319	0.96362	0.9641	0.93649	0.9798
11	p(value of European put option to sell one share)	2.18365	1.58221	1.58164	1.58102	1.61751	1.56081
12	$C(S_t^*, T-t; X)$ $-St^* - D + X$	0.60263	0.00119	0.00062	2.30E-06	0.03649	-0.0202

Fig. 6.7 Calculation of S_t^* (critical ex-dividend stock price)

$$b_1 = \frac{\ln\left(\frac{48.208}{46.9641}\right) + \left(0.08 + \frac{0.2^2}{2}\right)\left(\frac{50}{365}\right)}{(.20)\sqrt{50/365}} = 0.4859.$$

$$b_2 = 0.485931 - 0.074023 = 0.4119.$$

In addition, we also know $\rho = -\sqrt{50/90}$ = -0.7454.

From the above information, we now calculate related normal probability as follows:

$$N_1(b_1) = N_1(0.4859) = 0.6865$$
$$N_1(b_2) = N_1(0.7454) = 0.6598.$$

Following Equation (6.A2), we now calculate the value of $N_2(0.25285,-0.4859;\ -0.7454)$ and N_2 $(0.15354,\ -0.4119;\ -0.7454)$ as follows:

Since $ab\rho > 0$ for both cumulative bivariate normal density function, we can use equation N2 $(a, b;\rho) = N_2$ $(a,0;\rho_{ab})$ + $N_2(b, 0;\rho_{ba})$ -δ to calculate the value of both $N_2(a,b;\rho)$ as follows:

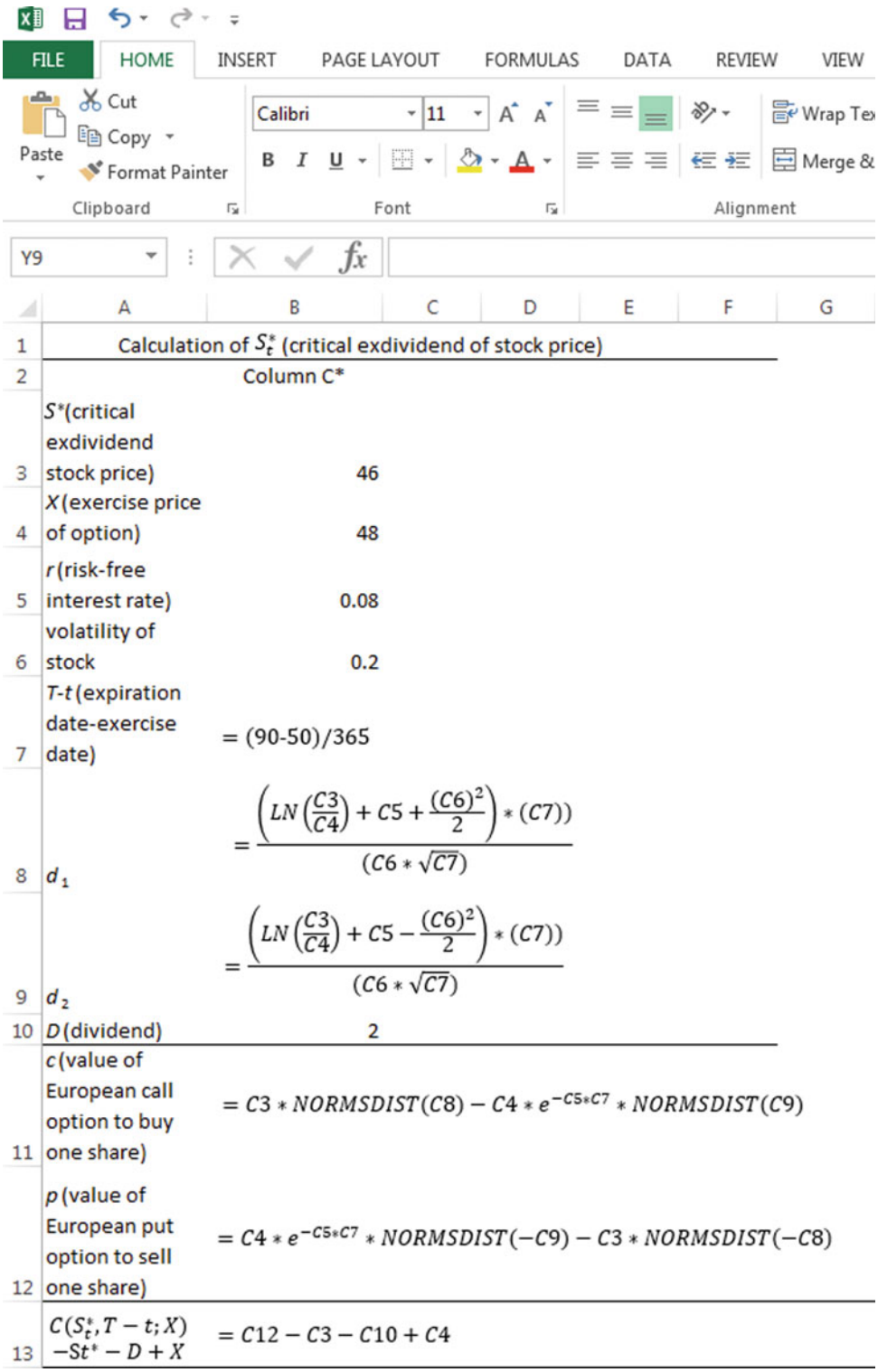
$$\rho_{ab} = \frac{[(-0.7454)(0.25285) + 0.4859](1)}{\sqrt{(0.25285)^2 - 2(0.7454)(0.25285)(-0.4859) + (0.4859)^2}} = 0.87002$$

$$\rho_{ba} = \frac{[(-0.7454)(-0.4859) - 0.25285](1)}{\sqrt{(0.25285)^2 - 2(0.7454)(0.25285)(-0.4859) + (0.4859)^2}} = -0.31979$$

$$\delta = (1 - (1)(-1))/4 = 1/2$$

$$\begin{aligned} &\mathrm{N}_2(0.292, -0.4859 - 0.7454) = \mathrm{N}_2(0.292, 0.0844) \\ &+ \mathrm{N}_2(-0.5377, 0.0656) - 0.5 = \mathrm{N}_1(0) \\ &+ \mathrm{N}_1(-0.5377) - \Phi(-0.292, 0; -0.0844) \\ &- \Phi(-0.5377, 0; -0.0656) - 0.5 = 0.07525 \end{aligned}$$

Fig. 6.7 (continued)



FILE HOME INSERT PAGE LAYOUT FORMULAS DATA REVIEW VIEW

Cut Copy Format Painter Paste Clipboard

Calibri 11 Font

Wrap Tex Merge & Alignment

Y9

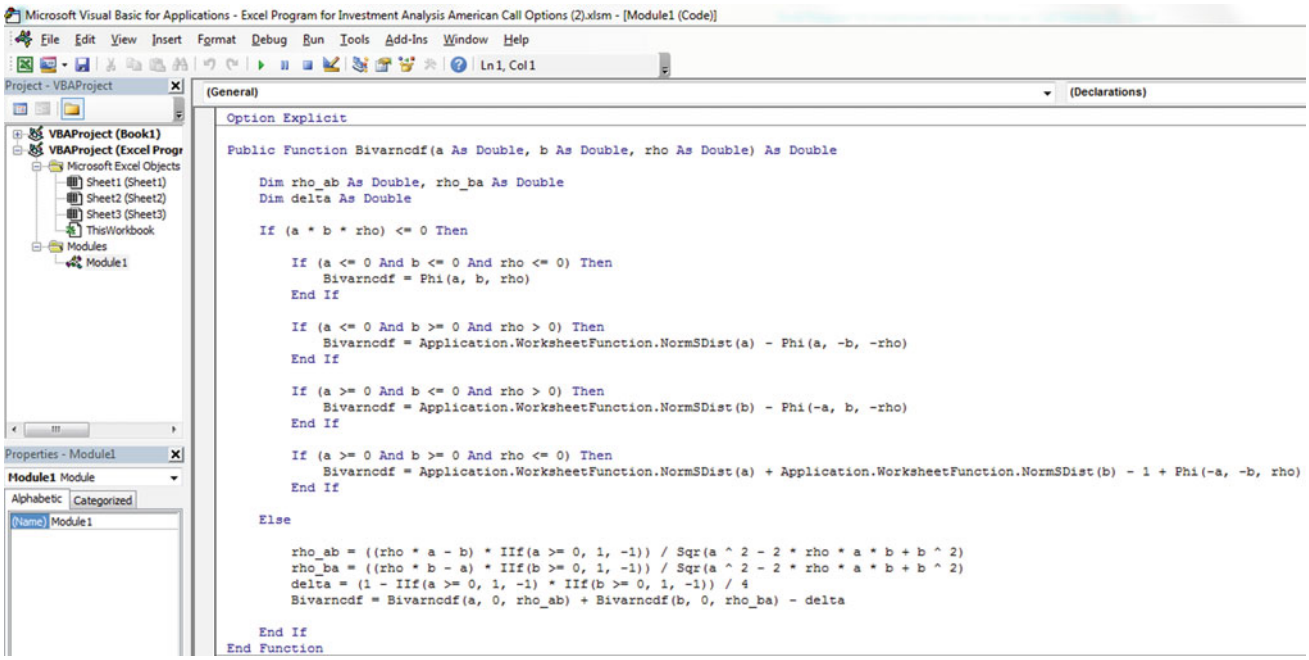
	A	B	C	D	E	F	G
1	Calculation of S_t^* (critical exdividend of stock price)						
2		Column C*					
3	S^*(critical exdividend stock price)	46					
4	X(exercise price of option)	48					
5	r(risk-free interest rate)	0.08					
6	volatility of stock	0.2					
7	T-t(expiration date-exercise date)	=(90-50)/365					
8	d_1	$=\frac{\left(LN\left(\frac{C3}{C4}\right)+C5+\frac{(C6)^2}{2}\right)*(C7))}{(C6*\sqrt{C7})}$					
9	d_2	$=\frac{\left(LN\left(\frac{C3}{C4}\right)+C5-\frac{(C6)^2}{2}\right)*(C7))}{(C6*\sqrt{C7})}$					
10	D(dividend)	2					
11	c(value of European call option to buy one share)	$=C3*NORMSDIST(C8)-C4*e^{-C5*C7}*NORMSDIST(C9)$					
12	p(value of European put option to sell one share)	$=C4*e^{-C5*C7}*NORMSDIST(-C9)-C3*NORMSDIST(-C8)$					
13	$C(S_t^*,T-t;X)$ $-St^*-D+X$	$=C12-C3-C10+C4$					

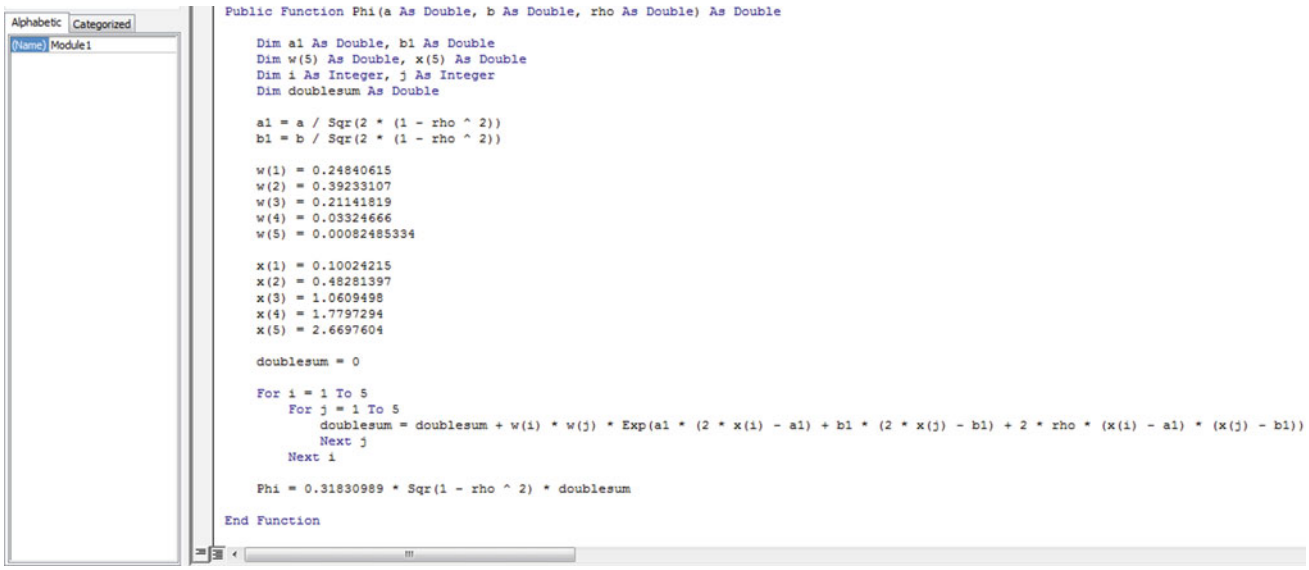
Using a Microsoft Excel programs presented in Appendix 6.2, we obtain

$$N_2(0.1927, -0.4119; -0.7454) = 0.06862.$$

Then substituting the related information into Equation (6.11), we obtain C=\$3.08238 and all related results are presented in Appendix 6.2.

The following is the VBA code necessary for Microsoft Excel to run the bivariate normal distribution approach to calculating an American call option:





6.7 Black's Approximation Method for American Option with One Dividend Payment

By using the same data as the bivariate normal distribution (from Sect. 6.4) we will show how Black's approximation method can be used to calculate the value of an American option. The first step is to calculate the stock price minus the current value of the dividend and then calculate d_1 and d_2 to calculate the call price at time T (the time of maturity).

$$2e^{-0.13699(0.08)} + 2e^{-0.24658(0.08)} = 0 = 1.9782.$$

- The option price can therefore be calculated from the Black–Scholes formula with S_0=48.0218, K = 48, r = 0.08, σ = 0.2, and T = 0.24658. We have

$$d_1 = \frac{\ln\left(\frac{48.0218}{48}\right) + \left(0.08 + \frac{0.2^2}{2}\right)(0.24658)}{0.8\sqrt{0.24658}} = 0.2529$$

$$d_1 = \frac{\ln\left(\frac{48.0218}{48}\right) + \left(0.08 - \frac{0.2^2}{2}\right)(0.24658)}{0.8\sqrt{0.24658}} = 0.1535.$$

- We can get from the normal table

$$N(d_1) = 0.5998, N(d_2) = 0.5610.$$

- And the call price is

$$48.0218(0.5998) - 48e^{-0.08(0.24658)}(0.5610) = \$2.40.$$

You then calculate the call price at time t (the time of the dividend payment) using the current stock price.

$$d_1 = \frac{\ln\left(\frac{50}{48}\right) + \left(0.08 + \frac{0.2^2}{2}\right)(0.13699)}{0.8\sqrt{0.13699}} = 0.7365$$

$$d_1 = \frac{\ln\left(\frac{50}{48}\right) + \left(0.08 - \frac{0.2^2}{2}\right)(0.13699)}{0.8\sqrt{0.13699}} = 0.6625.$$

- We can get from the normal table

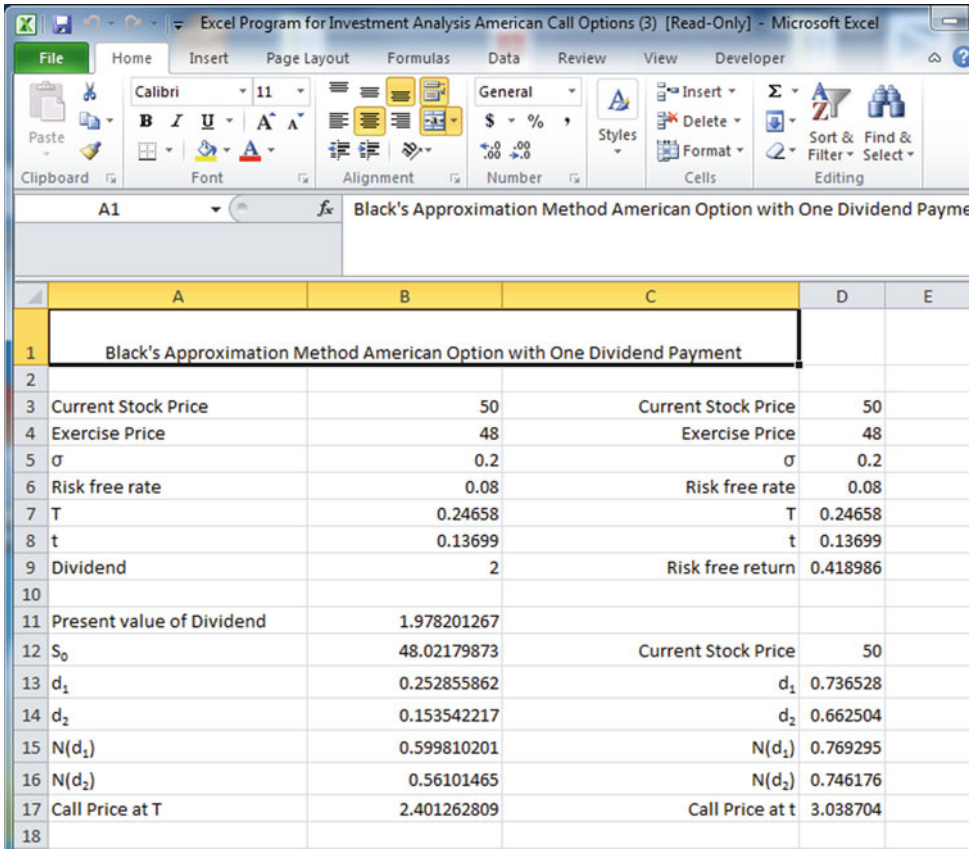
$$N(d_1) = 0.7693, N(d_2) = 0.7462.$$

- And the call price is

$$50(0.7693) - 48e^{-0.08(0.24658)}(0.7462) = \$3.04.$$

Comparing the greater of the two call option values will show if it is worth waiting until the time-to-maturity or exercising at the dividend payment.

$$\$3.04 > \$2.40.$$



	A	B	C	D	E
1	Black's Approximation Method American Option with One Dividend Payment				
2					
3	Current Stock Price	50	Current Stock Price	50	
4	Exercise Price	48	Exercise Price	48	
5	σ	0.2	σ	0.2	
6	Risk free rate	0.08	Risk free rate	0.08	
7	T	0.24658	T	0.24658	
8	t	0.13699	t	0.13699	
9	Dividend	2	Risk free return	0.418986	
10					
11	Present value of Dividend	1.978201267			
12	S_0	48.02179873	Current Stock Price	50	
13	d_1	0.252855862	d_1	0.736528	
14	d_2	0.153542217	d_2	0.662504	
15	$N(d_1)$	0.599810201	$N(d_1)$	0.769295	
16	$N(d_2)$	0.56101465	$N(d_2)$	0.746176	
17	Call Price at T	2.401262809	Call Price at t	3.038704	
18					

Current Stock Price	50
Exercise Price	48
σ	0.2
Risk free rate	0.08
T	0.24658
t	0.13699
Dividend	2
Present value of Dividend	=B9*EXP(-B8*B6)
S_0	=B3-B11
d_1	=(LN(B12/B4)+(B6+((B5^2)/2))*(B7))/(B5*SQRT(B7))
d_2	=(LN(B12/B4)+(B6-((B5^2)/2))*(B7))/(B5*SQRT(B7))
$N(d_1)$	=NORMSDIST(B13)
$N(d_2)$	=NORMSDIST(B14)
Call Price at T	=B12*(B15)-(B4*(EXP(-B6*B7))*B16)

Current Stock Price	50
Exercise Price	48
σ	0.2
Risk free rate	0.08
T	0.24658
t	0.13699
Risk free return	=$B$4*(1-EXP(-$B$6*($B$7-$B$8)))
Current Stock Price	=B3
d_1	=(LN(D12/D4)+(D6+((D5^2)/2))*(D8))/(D5*SQRT(D8))
d_2	=(LN(D12/D4)+(D6-((D5^2)/2))*(D8))/(D5*SQRT(D8))
$N(d_1)$	=NORMSDIST(D13)
$N(d_2)$	=NORMSDIST(D14)
Call Price at t	=D12*(D15)-(D4*(EXP(-D6*D8))*D16)

6.8 American Call Option When Dividend Yield is Known

Sections 6.5 and 6.6 discuss American option valuation procedure when the dividend payment amounts are known. In this section, we discuss the American option valuation when dividend yield instead of dividend payment is known.

Following Technical Note No.8* named "Options, Futures, and Other Derivatives, Ninth Edition" by John Hull, we use the following procedures to calculate the American call options value. Hull method is derived from Barone-Adesi and Whaley (1987). In our words, Hull replaces Barone-Adesi and Whaley's commodity option model in terms of stock option model. They use a quadratic approximation to get an analytic approximation for American option.

6.8.1 Theory and Method

Consider an option written on a stock providing a dividend yield equal to q. The European call prices at time t will be denoted by c(S, t), where S is the stock price, and the corresponding American call will be denoted by C(S, t). The relationship between American option and European option can be represented as

$$C(S,t) = \begin{cases} c(S,t) + A_2\left(\frac{S}{S^*}\right)^{\gamma_2} & \text{when } S < S^* \\ S - K & \text{when } S \geq S^* \end{cases},$$

where

$$A_2 = \frac{S^*}{\gamma_2}\left\{1 - e^{-q(T-t)}N[d_1(S^*)]\right\}$$

$$\gamma_2 = \left[-(\beta-1) + \sqrt{(\beta-1)^2 + \frac{4\alpha}{h}}\right]/2$$

$$d_1 = \frac{\ln\left(\frac{S}{K}\right) + \left(r - q + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$\alpha = \frac{2r}{\sigma^2}$$

$$\beta = \frac{2(r-q)}{\sigma^2}$$

$$h = 1 - e^{-r(T-t)}$$

To find the critical stock price S^*, it is necessary to solve

$$S^* - K = c(S^*,t) + \frac{S^*}{\gamma_2}\left\{1 - e^{-q(T-t)}N[d_1(S^*)]\right\}.$$

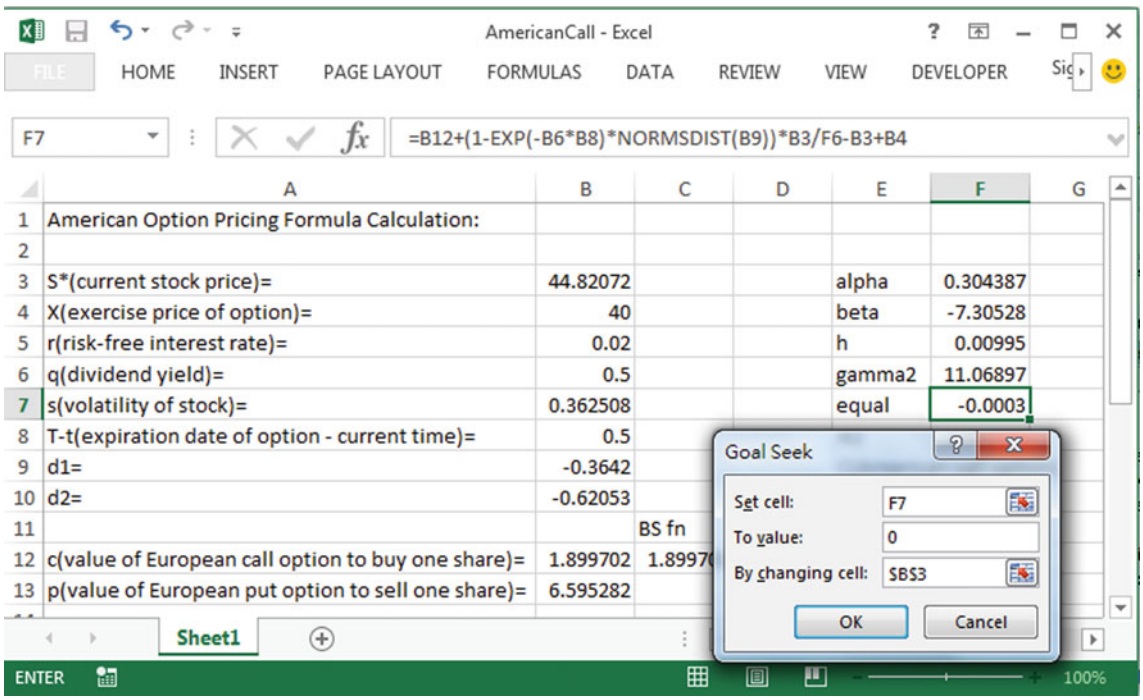
Since this cannot be done directly, an iterative procedure must be developed.

6.8.2 VBA Program for Calculating American Option When Dividend Yield is Known

WE can use Excel Goal Seek tool to develop the iterative process. We set Cell F7 equal to zero by changing Cell B3 to find S^*. The function in Cell F7 is

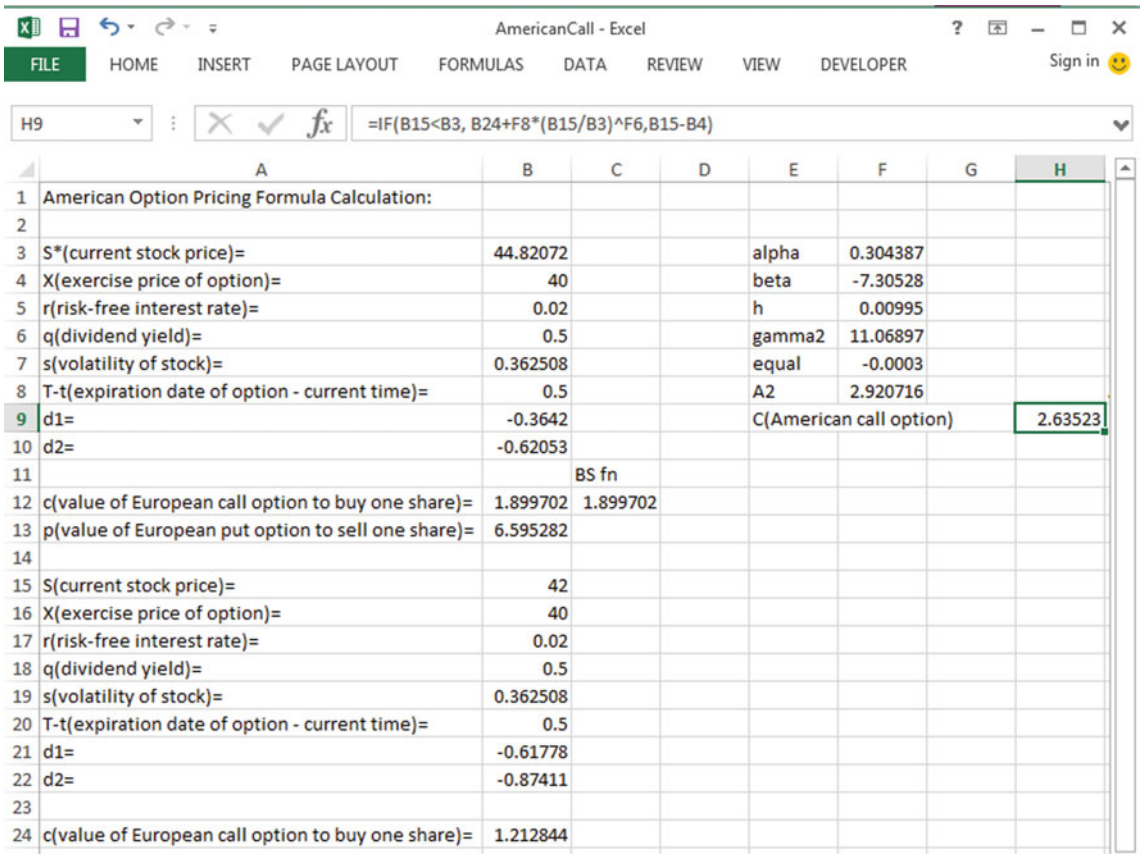
$$= B12 + (1 - EXP(-B6*B8)*NORMSDIST(B9))*\frac{B3}{F6} - B3 + B4.$$

After doing the iterative procedure, the result shows that S^* is equal to 44.82072.



After we get S^*, we can calculate the value of American call option when S is equal to 42 in Cell B15. The function to calculate American call option in Cell H9 is

$$= \text{IF}\left(\text{B15} < \text{B3},\ \text{B24} + \text{F8} * \left(\frac{\text{B15}}{\text{B3}}\right)^{\text{F6}}, \text{B15} - \text{B4}\right).$$



	A	B	C	D	E	F	G	H
1	American Option Pricing Formula Calculation:							
2								
3	S*(current stock price)=	44.82072			alpha	0.304387		
4	X(exercise price of option)=	40			beta	-7.30528		
5	r(risk-free interest rate)=	0.02			h	0.00995		
6	q(dividend yield)=	0.5			gamma2	11.06897		
7	s(volatility of stock)=	0.362508			equal	-0.0003		
8	T-t(expiration date of option - current time)=	0.5			A2	2.920716		
9	d1=	-0.3642			C(American call option)			2.63523
10	d2=	-0.62053						
11			BS fn					
12	c(value of European call option to buy one share)=	1.899702	1.899702					
13	p(value of European put option to sell one share)=	6.595282						
14								
15	S(current stock price)=	42						
16	X(exercise price of option)=	40						
17	r(risk-free interest rate)=	0.02						
18	q(dividend yield)=	0.5						
19	s(volatility of stock)=	0.362508						
20	T-t(expiration date of option - current time)=	0.5						
21	d1=	-0.61778						
22	d2=	-0.87411						
23								
24	c(value of European call option to buy one share)=	1.212844						

In addition to Goal Seek tool, we also can write a user-defined function to calculate this value of American call option. The VBA function is given below:

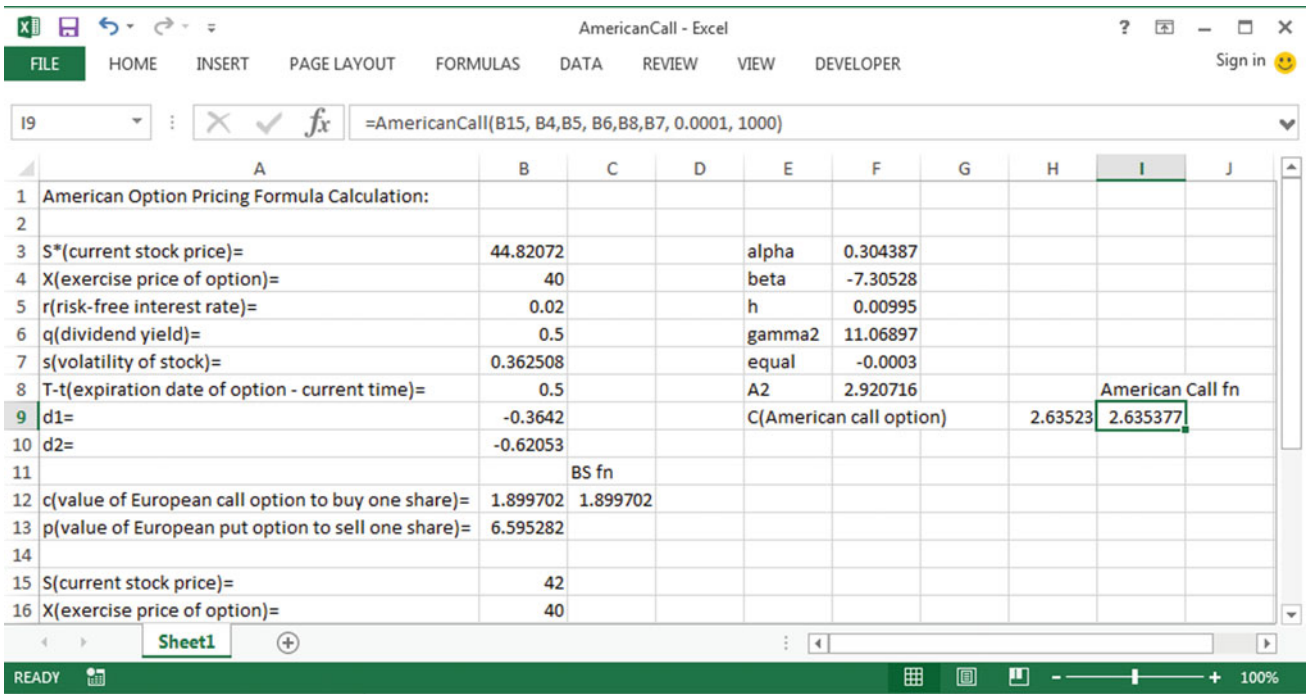
```
Function AmericanCall(S, X, r, q, T, sigma, a, b)
'  Estimate implied volatility by Bisection
'  Uses BSCall fn
  Dim yb, ya, c, yc, alpha, beta, h, gamma2, d1, A2, Sa
  alpha = 2 * r / sigma ^ 2
  beta = 2 * (r - q) / sigma ^ 2
  h = 1 - Exp(-r * T)
  gamma2 = (-(beta - 1) + Sqr
((beta - 1) ^ 2 + 4 * alpha / h)) / 2
  d1 = (Log
(b / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr
(T))
  yb = BSCall(b, X, r, q, T, sigma) + (1 - Exp(-q * T) *
Application.NormSDist(d1)) * b / gamma2 - b + X
  d1 = (Log
(a / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr
(T))
  ya = BSCall(a, X, r, q, T, sigma) + (1 - Exp(-q * T) *
Application.NormSDist(d1)) * a / gamma2 - a + X
  If yb * ya > 0 Then
   BSIVBisection = CVErr(xlErrValue)
  Else
    Do While Abs(a - b) > 0.000000001
     c = (a + b) / 2
     d1 = (Log
(c / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr
(T))
     yc = BSCall(c, X, r, q, T, sigma) + (1 - Exp(-q *
T) * Application.NormSDist(d1)) * c / gamma2 - c + X
     d1 = (Log
(a / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr
(T))
     ya = BSCall(a, X, r, q, T, sigma) + (1 - Exp(-q *
```

```
T) * Application.NormSDist(d1)) * a / gamma2 - a + X
      If ya * yc < 0 Then
        b = c
      Else
        a = c
      End If
     Loop
    Sa = (a + b) / 2
  End If
    d1 = (Log
(Sa / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr
(T))
    A2 = (Sa / gamma2) * (1 - Exp(-q * T) * Application.
NormSDist(d1))
  If S < Sa Then
    AmericanCall = BSCall
(S, X, r, q, T, sigma) + A2 * (S / Sa) ^ gamma2
  Else
    AmericanCall = S - X
  End If
End Function
```

The function in Cell I9 is

$$= \text{AmericanCall}(\text{B15},\ \text{B4}, \text{B5},\ \text{B6}, \text{B8}, \text{B7}, 0.0001, 1000).$$

After putting the parameters in function of the Cell I9, the result is similar to the value of American call option calculated by Goal Seek in Cell H9.



6.9 Summary

This chapter has shown how Microsoft Excel can be used to estimate European call and put options for (a) Black–Scholes model for Individual Stock, (b) Black–Scholes model for Stock Indices, and (c) Black–Scholes model for Currencies. In addition, we also discuss alternative methods to evaluate American call option when either dividend payment or dividend yield is known.

Appendix 6.1: Bivariate Normal Distribution

We have shown how the cumulative univariate normal density function can be used to evaluate a European call option in previous sections of this chapter. If a common stock pays a discrete dividend during the option's life, the American call option valuation equation requires the evaluation of a cumulative bivariate normal density function. While there are many available approximations for the cumulative bivariate normal distribution, the approximation provided here relies on Gaussian quadratures. The approach is straightforward and efficient, and its maximum absolute error is 0.00000055.

The probability that x' is less than a and that y' is less than b for the standardized cumulative bivariate normal distribution can be defined as

$$P(X'<a, Y'<b) = \frac{1}{2\pi\sqrt{1-\rho^2}} \int_{-\infty}^{a} \int_{-\infty}^{b} \exp\left[\frac{2x'^2 - 2\rho x'y' + y'^2}{2(1-\rho^2)}\right] dx'dy',$$

where $x' = \frac{x-\mu_x}{\sigma_x}$, $y' = \frac{y-\mu_y}{\sigma_y}$, and p is the correlation between the random variables x' and y'.

The first step in the approximation of the bivariate normal probability $N_2(a, b; \rho)$ is given below:

$$\phi(a, b; \rho) \approx .31830989\sqrt{1-\rho^2} \sum_{i=1}^{5} \sum_{j=1}^{5} w_i w_j f(x'_i, x'_j), \tag{6.A1}$$

where

$$f(x'_i, x'_j) = \exp[a_1(2x'_i - a_1) + b_1(2x'_j - b_1) + 2\rho(x'_i - a_1)(x'_j - b_1)].$$

The pairs of weights (w) and corresponding abscissa values (x') are.

i,j	w	x'
1	0.24840615	0.10024215
2	0.39233107	0.48281397
3	0.21141819	1.0609498
4	0.033246660	1.7797294
5	0.00082485334	2.6697604

(This portion is based on Appendix 13.1 of Stoll H. R. and R. E Whaley. *Futures and Options*. Cincinnati, OH: South Western Publishing, 1993.)

and the coefficients a_1 and b_1 are computed using

$$a_1 = \frac{a}{\sqrt{2(1-\rho^2)}} \text{ and } b_1 = \frac{b}{\sqrt{2(1-\rho^2)}}.$$

The second step in the approximation involves computing the product $ab\ \rho$; if $ab\ \rho \le 0$, compute the bivariate normal probability, $N_2(a, b; \rho)$, using the following rules:

(1) If $a \le 0, b \le 0$ and $\rho \le 0$, then $N_2(a, b; \rho) = \phi(a, b; \rho)$;
(2) If $a \le 0, b \ge 0$ and $\rho > 0$, then $N_2(a, b; \rho) = N_1(a) - \phi(a, -b; -\rho)$;
(3) If $a \ge 0, b \le 0$ and $\rho > 0$, then $N_2(a, b; \rho) = N_1(b) - \phi(-a, b; -\rho)$;
(4) If $a \ge 0, b \ge 0$ and $\rho \le 0$, then $N_2(a, b; \rho) = N_1(a) + N_1(b) - 1 + \phi(-a, -b; \rho)$. (6.A2)

If $ab\ \rho > 0$, compute the bivariate normal probability, $N_2(a, b; \rho)$, as

$$N_2(a, b; \rho) = N_2(a, 0; \rho_{ab}) + N_2(b, 0; \rho_{ab}) - \delta, \tag{6.A3}$$

where the values of $N_2(\bullet)$ on the right-hand side are computed from the rules, for $ab\ \rho \le 0$

$$\rho_{ab} = \frac{(\rho a - b)Sgn(a)}{\sqrt{a^2 - 2\rho ab + b^2}}, \rho_{ba} = \frac{(\rho b - a)Sgn(b)}{\sqrt{a^2 - 2\rho ab + b^2}}, \delta = \frac{1 - Sgn(a) \times Sgn(b)}{4},$$

and

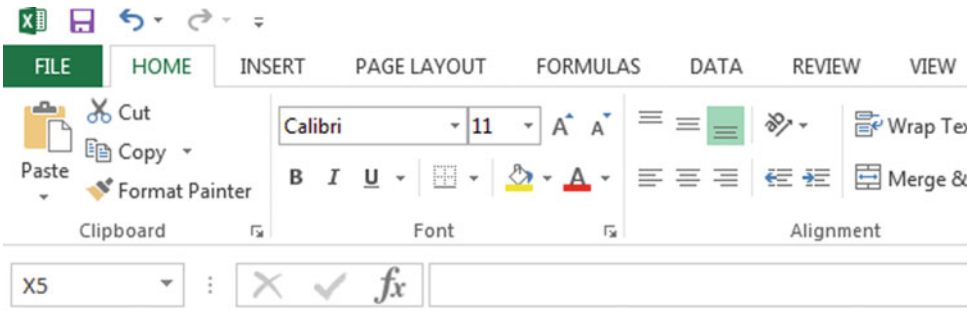
$$Sgn(x) = \begin{cases} 1 & x \ge 0 \\ -1 & x < 0 \end{cases},$$

$N_1(d)$ is the cumulative univariate normal probability.

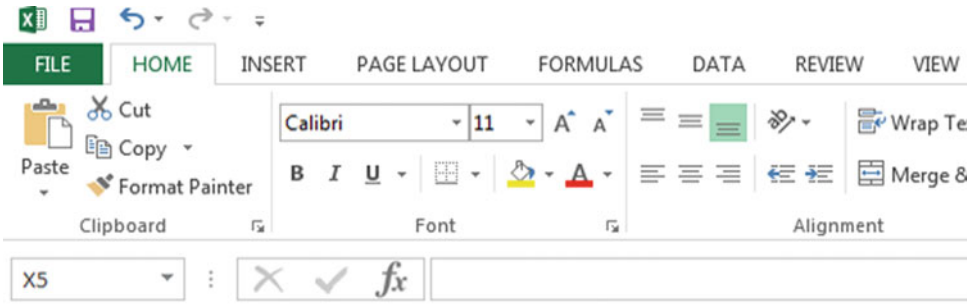
Appendix 6.2: Excel Program to Calculate the American Call Option When Dividend Payments are Known

The following is a Microsoft Excel program which can be used to calculate the price of an American call option using the bivariate normal distribution method: (Table B1)

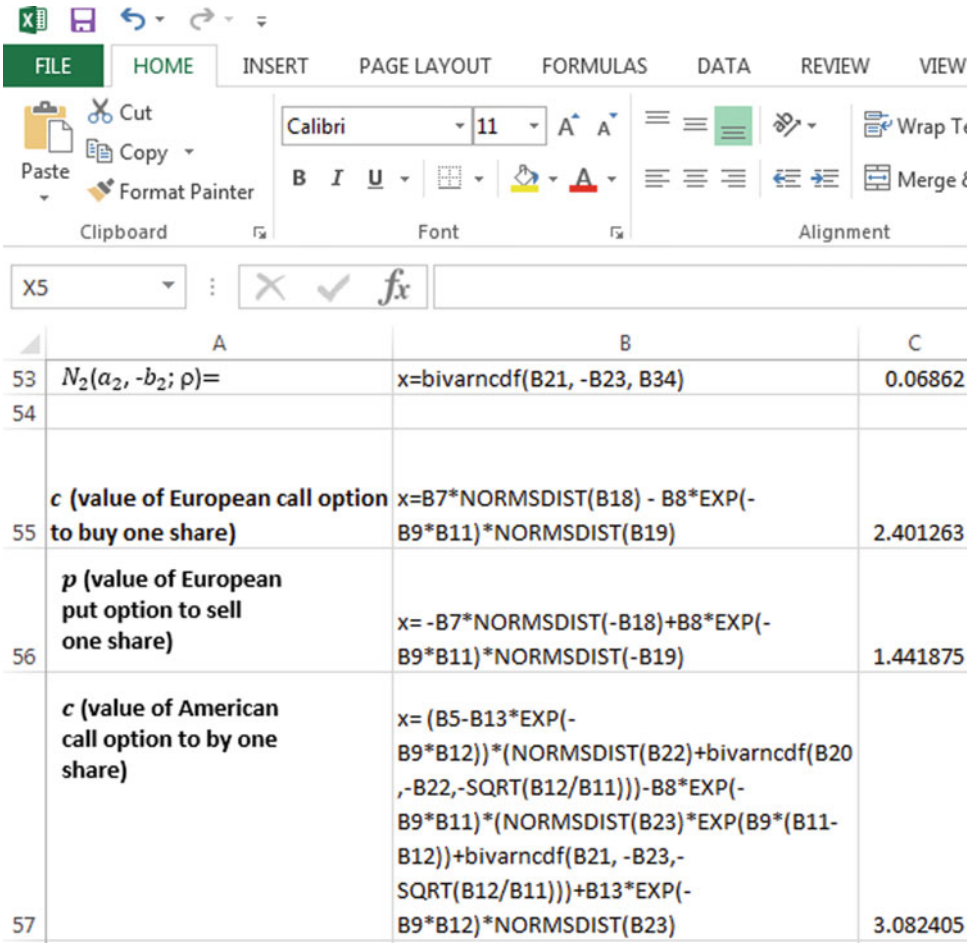
Table 6.1 Microsoft Excel program for calculating the American call options



	A	B	C
1	A	B	C*
2			
3	Option Pricing Calculation		
4			
5	S (current stock price) =	50	50
6	S_t^* (critical exdividend stock price) =	46.9641	46.9641
7	S (current stock price NPV of promised dividend) =	x=B5-B13*EXP(-B9*B12)	48.0218
8	X (exercise price of option) =	48	48
9	r (risk-free interest rate) =	0.08	0.08
10	σ (volatility of stock) =	0.2	0.2
11	T *(expiration date)* =	0.24658	0.24658
12	t (Exercise date) =	0.13699	0.13699
13	D (Dividend) =	2	2
14	d_1 (nondividend-paying)=	x=(LN(B5/B8)+(B9+0.5*B10^2)*B11)/(B10*SQRT(B11))	0.659325
15	d_2 (nondividend-paying)=	x=B14-B10*SQRT(B11)	0.560012
16	d_1* (critical exdividend stock price) =	x=(LN(B6/B8)+(B9+0.5*B10^2)*(B11-B12))/(B10*SQRT(B11-B12))	-0.164
17	d_2 *(critical exdividend stock price) =	x=B16 - B10*SQRT(B11-B12)	-0.23021
18	d_1 (dividend- paying) =	x=(LN(B7/B8)+(B9+0.5*B10^2)*(B11))/(B10*SQRT(B11))	0.252856
19	d_2 (dividend- paying)=	x=B18-B10*SQRT(B11)	0.153542
20	$a_1 =$	x=(LN((B5-B13*EXP(-B9*B12))/B8)+(B9+0.5*B10^2)*(B11))/(B10*SQRT((B11)))	0.252856
21	$a_2 =$	x=B20-B10*SQRT(B11)	0.153542
22	$b_1 =$	x=(LN((B5-B13*EXP(-B9*B12))/B6)+(B9+0.5*B10^2)*(B12))/(B10*SQRT(B12))	0.485929
23	$b_2 =$	x=B22-B10*SQRT(B12)	0.411905
24			
25	$C(S_t^*, T-t; X)$	x=B6*NORMSDIST(B16)-B8*EXP(-B9*(B11-B12))*NORMSDIST(B17)	0.964112



	A	B	C
26	$C(S_t^*, T-t; X)$ $-S_t^* - D + X =$	x=B25 - B6 - B13 + B8	1.15E-05
27			
28	$N_1(a_1) =$	x=NORMSDIST(B20)	0.59981
29	$N_1(a_2) =$	x=NORMSDIST(B21)	0.561015
30	$N_1(b_1) =$	x=NORMSDIST(B22)	0.686491
31	$N_1(b_2) =$	x=NORMSDIST(B23)	0.659795
32	$N_1(-b_1) =$	x=NORMSDIST(-B22)	0.313509
33	$N_1(-b_2) =$	x=NORMSDIST(-B23)	0.340205
34	ρ =	x=-SQRT(B12/B11)	-0.74536
35	$a = a_1 ; b = -b_1$		
36	$\phi(a, -b; -\rho) =$	x=phi(-B22, 0, -B39)	0.202591
37	$\phi(-a, b; -\rho) =$	x=phi(-B20, 0, -B38)	0.040844
38	$\rho ab =$	x=((B34*B20-(-B22))*IF(B20 >= 0, 1, -1))/SQRT(B20^2 - 2*B34*B20*-B22+(-B22)^2)	0.870011
39	$\rho ba =$	x=((B34*-B22-(B20))*IF(-B22 >= 0, 1, -1))/SQRT(B20^2 - 2*B34*B20*-B22+(-B22)^2)	-0.31978
40	$N_2(a, 0; \rho ab) =$	x=bivarncdf(B20,-B22,B34)	0.070074
41	$N_2(b, 0; \rho ba) =$	x=bivarncdf(-B22, 0, B39)	0.110918
42	δ =	x=(1 - IF(B20 >= 0, 1, -1)* IF(-B22 >= 0, 1, -1))/4	0.5
43	$a = a_2 ; b = b_2$		
44	$\phi(a, -b; -\rho) =$	x=phi(-B23, 0, -B47)	0.244011
45	$\phi(-a, b; -\rho) =$	x=phi(-B21, 0, -B46)	0.027572
46	$\rho ab =$	x=((B34*B21-(-B23))*IF(B21 >= 0, 1, -1))/SQRT(B21^2 - 2*B34*B21*-B23+(-B23)^2)	0.94558
47	$\rho ba =$	x=((B34*-B23-(B21))*IF(-B23 >= 0, 1, -1))/SQRT(B21^2 - 2* B34*B21* -B23 + (-B23)^2)	-0.48787
48	$N_2(a, 0; \rho ab) =$	x=bivarncdf(B21,0,B46)	0.472428
49	$N_2(b, 0; \rho ba) =$	x=bivarncdf(-B23,0,B47)	0.096192
50	δ=	x=(1 - IF(B21 >= 0,1,-1)*IF(-B23 >= 0, 1, -1))/4	0.5
51			
52	$N_2(a_1, -b_1; \rho)$=	x=bivarncdf(B20,-B22,B34)	0.070074



	A	B	C
53	$N_2(a_2, -b_2; \rho)=$	x=bivarncdf(B21, -B23, B34)	0.06862
54			
55	***c* (value of European call option to buy one share)**	x=B7*NORMSDIST(B18) - B8*EXP(-B9*B11)*NORMSDIST(B19)	2.401263
56	***p* (value of European put option to sell one share)**	x= -B7*NORMSDIST(-B18)+B8*EXP(-B9*B11)*NORMSDIST(-B19)	1.441875
57	***c* (value of American call option to by one share)**	x= (B5-B13*EXP(-B9*B12))*(NORMSDIST(B22)+bivarncdf(B20,-B22,-SQRT(B12/B11)))-B8*EXP(-B9*B11)*(NORMSDIST(B23)*EXP(B9*(B11-B12))+bivarncdf(B21, -B23,-SQRT(B12/B11)))+B13*EXP(-B9*B12)*NORMSDIST(B23)	3.082405

References

Anderson, T. W. *An Introduction to Multivariate Statistical Analysis*, 3rd ed. New York: Wiley-Interscience, 2003.

Black, F. "The Pricing of Commodity Contracts." *Journal of Financial Economics*, v. 3 (January-March 1976), pp.167–178.

Cox, J. C. and S. A. Ross. "The valuation of options for alternative stochastic processes." *Journal of Financial Economics*, v. 3 (January–March 1976), pp. 145–166.

Cox, J., S. Ross and M. Rubinstein. "Option Pricing: A Simplified Approach." *Journal of Financial Economics*, v. 7 (1979), pp. 229–263.

Johnson, N. L. and S. Kotz. *Distributions in Statistics: Continuous Multivariate Distributions*. New York: Wiley, 1972.

Johnson, N. L. and S. Kotz. *Distributions in Statistics: continuous Univariate Distributions 2*. New York: Wiley, 1970.

Rubinstein, M. "The Valuation of Uncertain Income Streams and the Pricing of Options." *Bell Journal of Economics and Management Science*, v. 7 (1976), 407–425.

Stoll, H. R. "The Relationship between Put and Call Option Prices." *Journal of Finance*, v. 24 (December 1969), pp. 801–824.

Whaley, R. E. "On the Valuation of American Call Options on Stocks with Known Dividends." *Journal of Financial Economics*, v. 9 (1981), pp. 207–211.

7 Alternative Methods to Estimate Implied Variance

7.1 Introduction

In this chapter, we will introduce how to use Excel to estimate implied volatility. First, we use approximate linear function to derive the volatility implied by Black–Merton–Scholes model. Second, we use nonlinear method, which include Goal Seek and Bisection method, to calculate implied volatility. Third, we demonstrate how to get the volatility smile using IBM data. Fourth, we introduce constant elasticity volatility (CEV) model and use bisection method to calculate the implied volatility of CEV model. Finally, we calculate the 52-week historical volatility of a stock. We used the Excel function *webserivce* to retrieve the 52 historical stock prices.

This chapter is broken down into the following sections. In Sect. 7.2, we use Excel to estimate the implied variance with Black–Scholes option pricing model. In Sect. 7.3, we discuss volatility smile, and in Sect. 7.4 we use Excel to estimate implied variance with CEV model. Section 7.5 looks at the web service Excel function. In Sect. 7.6, we look at retrieve a stock price for a specific date. In Sect. 7.7, we look at a calculated holiday list, and in Sect. 7.8 we calculate historical volatility. Finally, in Sect. 7.9, we summarize the chapter.

7.2 Excel Program to Estimate Implied Variance with Black–Scholes Option Pricing Model

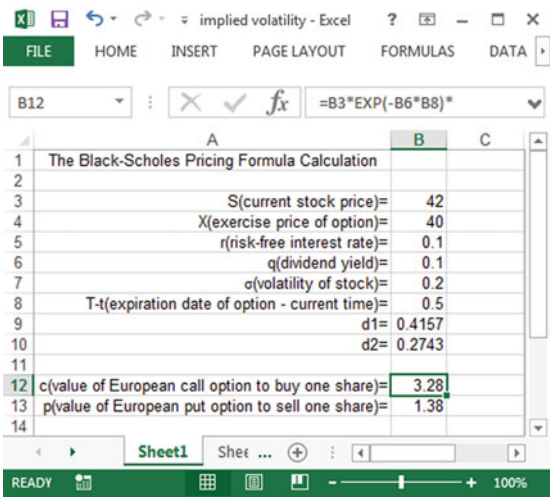
7.2.1 Black, Scholes, and Merton Model

In a classic option pricing developed by Black and Scholes (1973) and Merton (1973), the value of a European call option on a stock is stated

$$c = Se^{-qT}N(d) + Xe^{-rT}N(d - \sigma\sqrt{T})$$

$$d = \frac{\ln\left(\frac{S}{X}\right) + \left(r - q + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}$$

where the stock price, exercise price, interest rate, dividend yield, and time until option expiration are denoted by S, K, r, q, and T, respectively. The instantaneous standard deviation of the log stock price is represented by σ, and $N(.)$ is the standard normal distribution function. If we can get the parameter in the model, we can calculate the option price. The Black–Scholes formula in the spreadsheet is shown below:



For a call option on a stock, the Black–Scholes formula in cell B12 is

$$= \text{B3} * \text{EXP}(-\text{B6} * \text{B8}) * \text{NORMSDIST}(\text{B9}) - \text{B4} * \text{EXP}(-\text{B5} * \text{B8}) * \text{NORMSDIST}(\text{B10}),$$

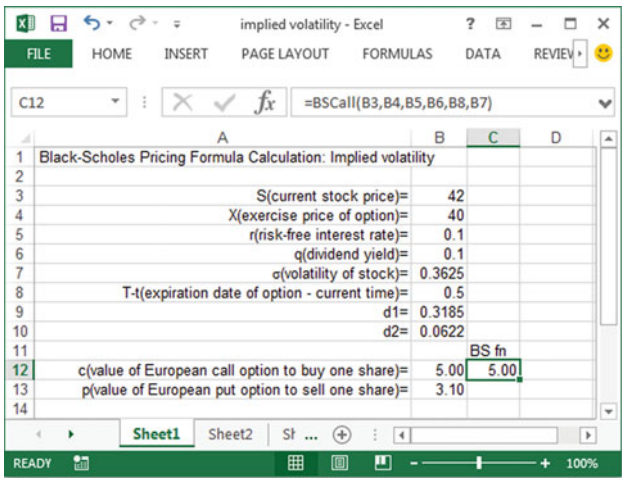
where NORMSDIST takes care of the cumulative distribution function of standard normal distribution.

It is easy to write a function to price a call function using Black and Scholes formula. The VBA function program is given below:

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_7

```
'   BS Call Option Value
Function BSCall(S, X, r, q, T, sigma)
    Dim d1, d2, Nd1, Nd2
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    d2 = d1 - sigma * Sqr(T)
    Nd1 = Application.NormSDist(d1)
    Nd2 = Application.NormSDist(d2)
    BSCall = Exp(-q * T) * S * Nd1 - Exp(-r * T) * X * Nd2
End Function
```

If we use this function to calculate, we just put the parameters into the function. And we can get the result. We don't need to write the Black and Scholes formula again. This is show below:



The user-defined VBA function in cell C12 is

$$= \text{BSCall}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}).$$

The call value in cell C12 is 5.00 which is equal to B12 calculated by spreadsheet.

7.2.2 Approximating Linear Function for Implied Volatility

All model parameters except the log stock price standard deviation are directly observable from market data. This allows a market-based estimate of a stock's future price volatility to be obtained by inverting Eq. (7.1), thereby yielding an implied volatility.

Unfortunately, there is no closed-form solution for an implied standard deviation from Eq. (7.1). We have to solve a nonlinear equation. Corrado and Miller (1996) have suggested an analytic formula that produces an approximation for the implied volatility. They start by approximating N(z) as a linear function:

$$\text{N}(\text{z}) = \frac{1}{2} + \frac{1}{\sqrt{2\pi}}\left(z - \frac{z^3}{6} + \frac{z^5}{40} + \ldots\right).$$

Substituting expansions of the normal cumulative probabilities N(d) and $\text{N}(\text{d} - \sigma\sqrt{T})$ into Black–Scholes call option price

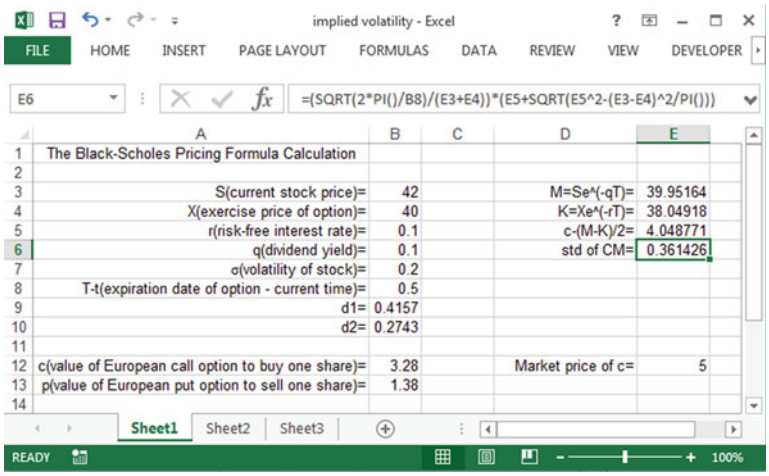
$$\text{c} = \text{S}e^{-qT}\left(\frac{1}{2} + \frac{d}{\sqrt{2\pi}}\right) + Xe^{-rT}\left(\frac{1}{2} + \frac{d - \sigma\sqrt{T}}{\sqrt{2\pi}}\right).$$

After solving the quadratic equation and some approximations, we can get

$$\sigma = \frac{\sqrt{2\pi/T}}{M+K}\left(c - \frac{M-K}{2} + \sqrt{\left(c - \frac{M-K}{2}\right)^2 - \frac{(M-K)^2}{\pi}}\right),$$

where $M = Se^{-qT}$ and $K = Xe^{-rT}$.

After typing Corrado and Miller's formula into excel worksheet, we can get the approximation of implied volatility easily. This is shown below:



	A	B	C	D	E
1	The Black-Scholes Pricing Formula Calculation				
2					
3	S(current stock price)=	42		M=Se^(-qT)=	39.95164
4	X(exercise price of option)=	40		K=Xe^(-rT)=	38.04918
5	r(risk-free interest rate)=	0.1		c-(M-K)/2=	4.048771
6	q(dividend yield)=	0.1		std of CM=	0.361426
7	σ(volatility of stock)=	0.2			
8	T-t(expiration date of option - current time)=	0.5			
9	d1=	0.4157			
10	d2=	0.2743			
11					
12	c(value of European call option to buy one share)=	3.28		Market price of c=	5
13	p(value of European put option to sell one share)=	1.38			
14					

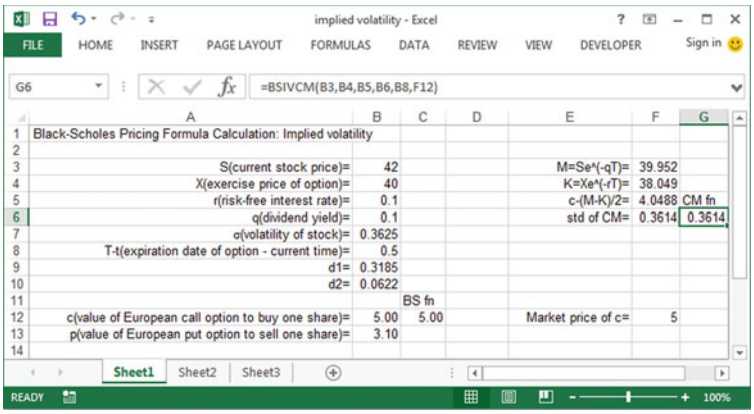
If the market price of call option is E12, the approximation value of implied volatility using the Carrodo and Miller's formula shown in E6 is

$$= (\text{SQRT}(2^*\text{PI}()/\text{B8})/(\text{F3}+\text{F4}))^*(\text{F5}+\text{SQRT}(\text{F5}\text{\^{}}2 - (\text{F3}-\text{F4})\text{\^{}}2/\text{PI}())).$$

If we want to write a function to calculate implied volatility of Corrado and Miller, here is the VBA function:

```
'   Estimate implied volatility by Corrando and Miller
Function BSIVCM(S, X, r, q, T, callprice)
    Dim M, K, p, diff, sqrtest
    M = S * Exp(-q * T)
    K = X * Exp(-r * T)
    p = Application.Pi()
    diff = callprice - 0.5 * (M - K)
    sqrtest = (diff ^ 2) - ((M - K) ^ 2) / p
    If sqrtest < 0 Then
        BSIVCM = -1
    Else
    BSIVCM = (Sqr(2 * p / T) / (M + K)) * (diff + Sqr(sqrtest))
End If
End Function
```

Using this function, it's easy to calculate an approximation of implied volatility. The output is shown below:



The Corrado and Miller implied volatility formula in G6 is

$$= \text{BSIVCM}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{F12}).$$

The approximation value in G6 is 0.3614 which is equal to F6.

7.2.3 Nonlinear Method for Implied Volatility

There are two nonlinear methods for implied volatility. The first one is Newton–Raphson method. The second one is bisection. Using the slope to improve the accuracy of subsequent guesses is known as the Newton–Raphson method.

7.2.3.1 Newton–Raphson Method

Newton–Raphson method is a method for finding successively better approximations to the roots of a nonlinear function.

$$\text{x} : \text{f}(\text{x}) = 0.$$

The Newton–Raphson method in one variable is accomplished as follows:

Given a function *f(x)* and its derivative *f'(x)*, we begin with a first guess x_0 for a root of the function *f*. The process is iterated as

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

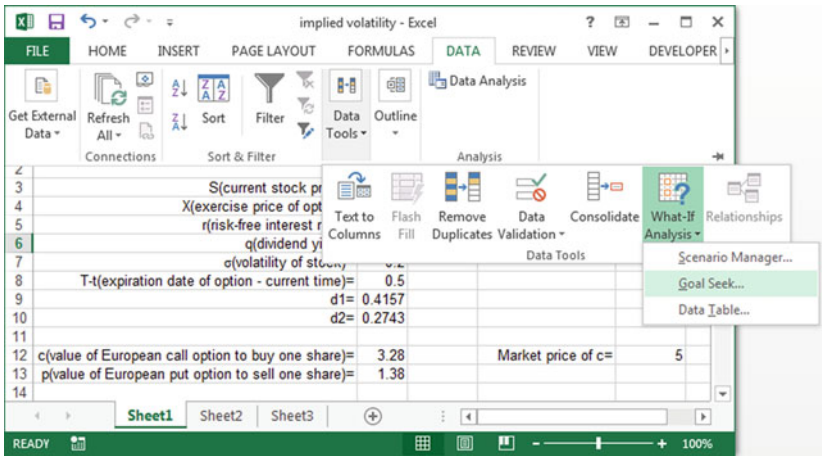
until a sufficiently accurate value is approached.

In order to use Newton–Raphson to estimate implied volatility, we need f'(.), in option pricing model is Vega.

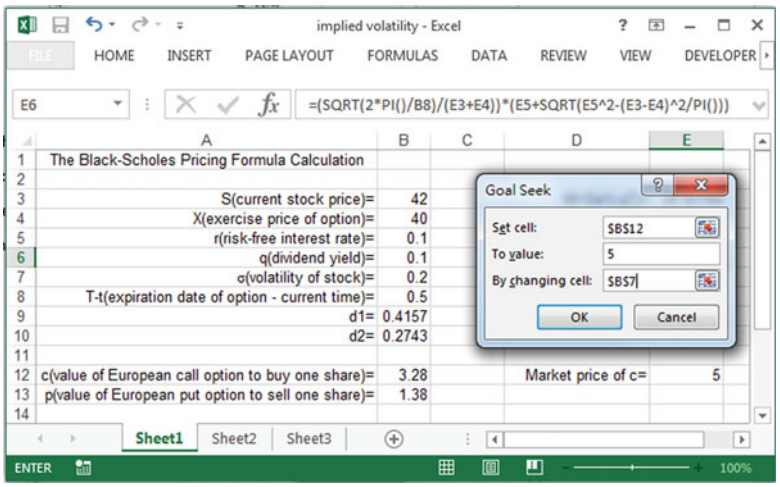
$$\text{v} = \frac{\partial \text{C}}{\partial \sigma} = \text{Se}^{-\text{qT}}\sqrt{T}N'(d_1).$$

Goal Seek is a procedure in Excel. It uses the Newton–Raphson method to solve the root of nonlinear equation. In figure given below, we would like to show how to use Goal Seek procedure to find the implied volatility. The details of our vanilla option are set out (cells B3–B8). Suppose the observed call option market value is 5.00. Our work is to choose a succession of volatility estimates in cell B6 until the BSM call option value in cell B11 equals to the observed price, 5.00. This can be done by applying the Goal Seek command in the Data part of Excel's menu.

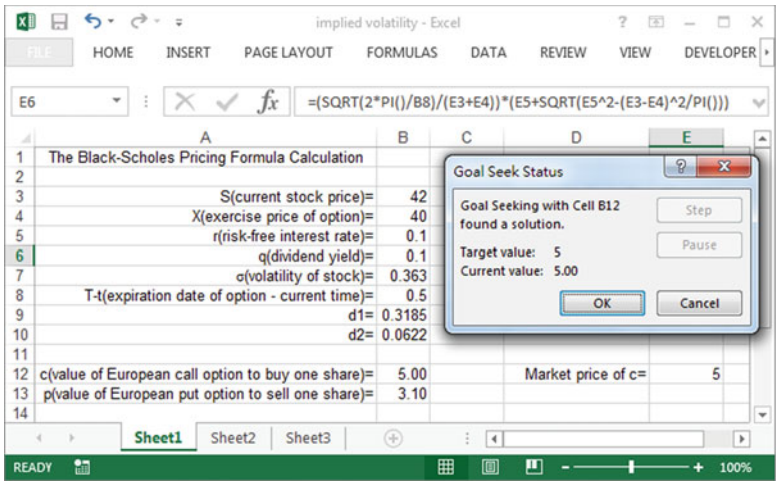
[Data] → [What If Analysis] → [Goal Seek]



Insert the following data into [Goal Seek] dialogue box:
Set cell: B12
To value: 5.00
By changing cell: $B$7

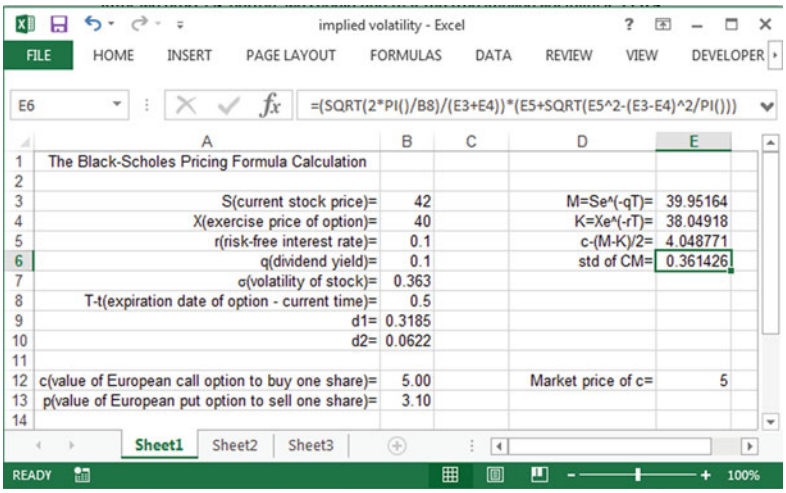


After we press OK button, we should find that the true implied volatility is 36.3%.



	A	B	C	D	E
1	The Black-Scholes Pricing Formula Calculation				
2					
3	S(current stock price)=	42			
4	X(exercise price of option)=	40			
5	r(risk-free interest rate)=	0.1			
6	q(dividend yield)=	0.1			
7	σ(volatility of stock)=	0.363			
8	T-t(expiration date of option - current time)=	0.5			
9	d1=	0.3185			
10	d2=	0.0622			
11					
12	c(value of European call option to buy one share)=	5.00		Market price of c=	5
13	p(value of European put option to sell one share)=	3.10			
14					

We can find that Corrado and Miller (1996) analytical, 0.361, which is near the Goal Seek solution 0.363.



	A	B	C	D	E
1	The Black-Scholes Pricing Formula Calculation				
2					
3	S(current stock price)=	42		M=Se^(-qT)=	39.95164
4	X(exercise price of option)=	40		K=Xe^(-rT)=	38.04918
5	r(risk-free interest rate)=	0.1		c-(M-K)/2=	4.048771
6	q(dividend yield)=	0.1		std of CM=	0.361426
7	σ(volatility of stock)=	0.363			
8	T-t(expiration date of option - current time)=	0.5			
9	d1=	0.3185			
10	d2=	0.0622			
11					
12	c(value of European call option to buy one share)=	5.00		Market price of c=	5
13	p(value of European put option to sell one share)=	3.10			
14					

7.2.3.2 Bisection Method

In addition to Newton–Raphson method, we have another method to solve the root of nonlinear equation. This is bisection method. Start with two numbers, a and b, where $a < b$ and $f(a) * f(b) < 0$. If we evaluate f and midpoint $c = (a + b)/2$, then

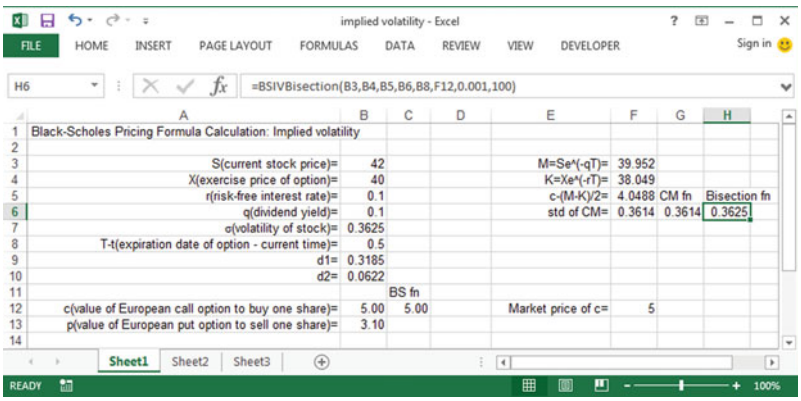
(1) $f(c) = 0$,
(2) $f(a) * f(c) < 0$, or
(3) $f(c) * f(b) < 0$.

In call option example, f(.) = BSCall(.)—market price of call option and a, b, and c are the candidates of implied volatility.

Although this method is a little slower than Newton–Raphson method, it will not run down when we give a bad initial value like Newton–Raphson method. We also can create a function to estimate implied volatility by using bisection method. The VBA function is shown below:

```
'   Estimate implied volatility by Bisection
'   Uses BSCall fn
Function BSIVBisection(S, X, r, q, T, callprice, a, b)
    Dim yb, ya, c, yc
    yb = BSCall(S, X, r, q, T, b) - callprice
    ya = BSCall(S, X, r, q, T, a) - callprice
    If yb * ya > 0 Then
      BSIVBisection = CVErr(xlErrValue)
    Else
       Do While Abs(a - b) > 0.000000001
       c = (a + b) / 2
       yc = BSCall(S, X, r, q, T, c) - callprice
       ya = BSCall(S, X, r, q, T, a) - callprice
       If ya * yc < 0 Then
          b = c
       Else
          a = c
       End If
      Loop
     BSIVBisection = (a + b) / 2
   End If
End Function
```

When we use this function to estimate implied volatility, the result is shown below:



The bisection formula of implied volatility in H6 is

$$= \text{BSIVBisection}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{F12}, 0.001, 100).$$

Implied volatility, 0.3625, estimated from bisection method is much closer to Newton–Raphson method of Goal Seek, 0.3625, than Corrado and Miller's approximation, 0.3614.

7.2.3.3 Compare Newton–Raphson Method and Bisection Method

Before we write a user-defined function for Newton–Raphson method, we need a Vega function for vanilla call option. Below is the function for Vega.

```
'   BS Call Option Vega

Function BSCallVega(S, X, r, q, T, sigma)

    Dim d1, Ndash1

    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))

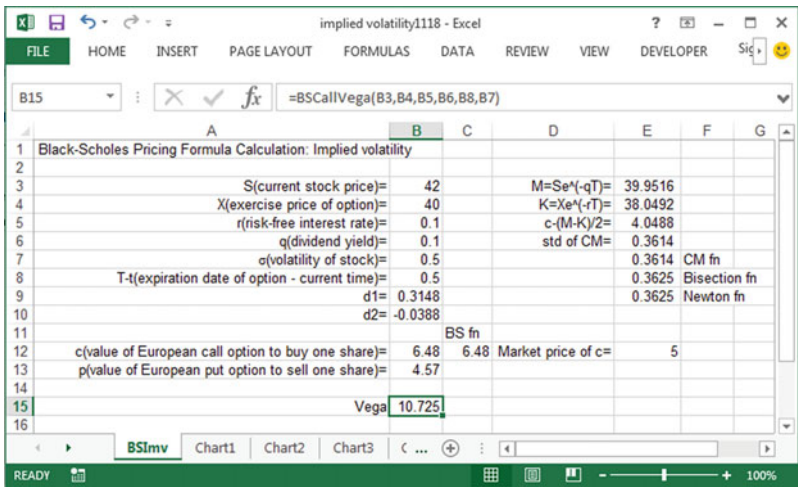
    Ndash1 = (1 / Sqr(2 * Application.Pi)) * Exp(-d1 ^ 2 / 2)

    BSCallVega = Exp(-q * T) * S * Sqr(T) * Ndash1

End Function
```

In the figure given below, we can see in Cell B15 the function to calculate Vega.

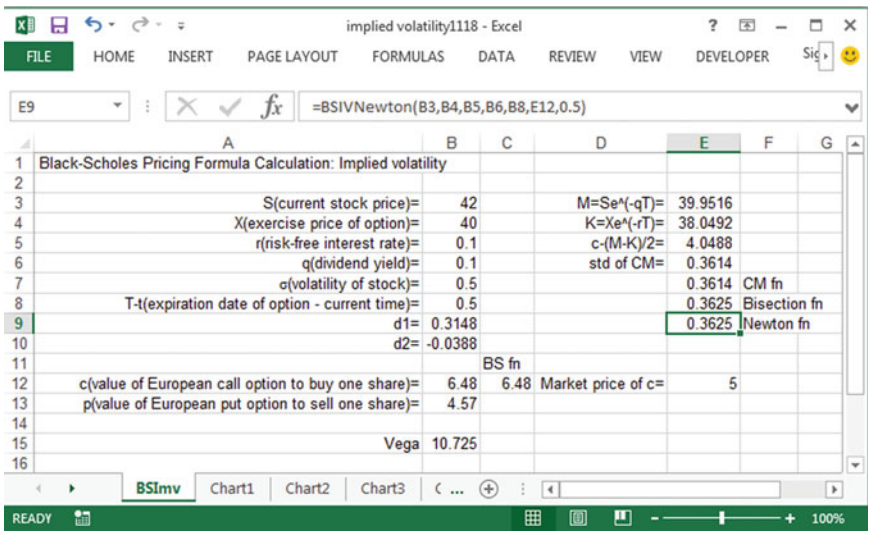
$$= \text{BSCallVega}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}).$$



In order to compare Newton–Raphson method and Bisection method, we have to write a user-defined function of Newton–Raphson. According to the methodology in Sect. 7.2.3.1, the VBA function is given below:

```
'   Estimate implied volatility by Newton
'   Uses BSCall fn & BSCallVega
Function BSIVNewton(S, X, r, q, T, callprice, initial)
    Dim bias, iv, ya, ydasha
    bias = 0.0001
    iv = initial
    Do
      ya = BSCall(S, X, r, q, T, iv) - callprice
      ydasha = BSCallVega(S, X, r, q, T, iv)
      iv = iv - ya / ydasha
    Loop While Abs(ya / ydasha) > bias
    BSIVNewton = iv
End Function
```

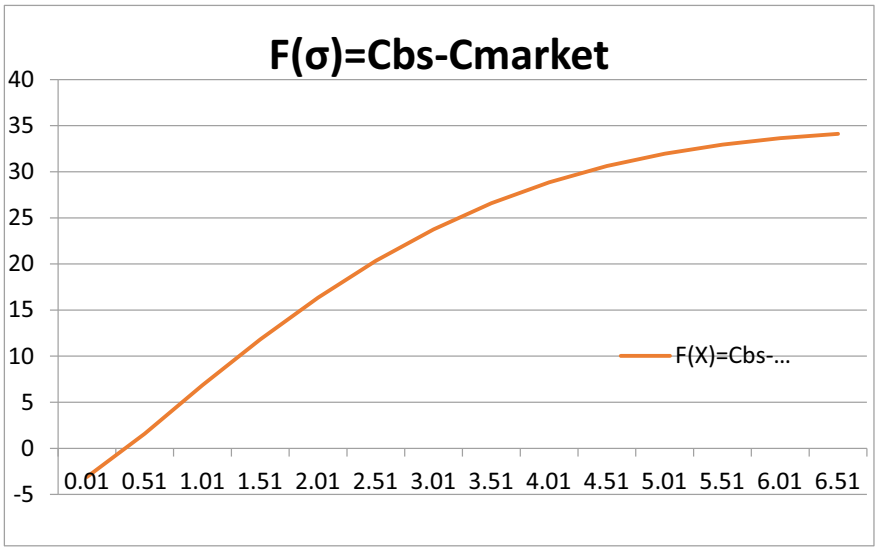
Use this function we can calculate to implied volatility by Newton–Raphson method.



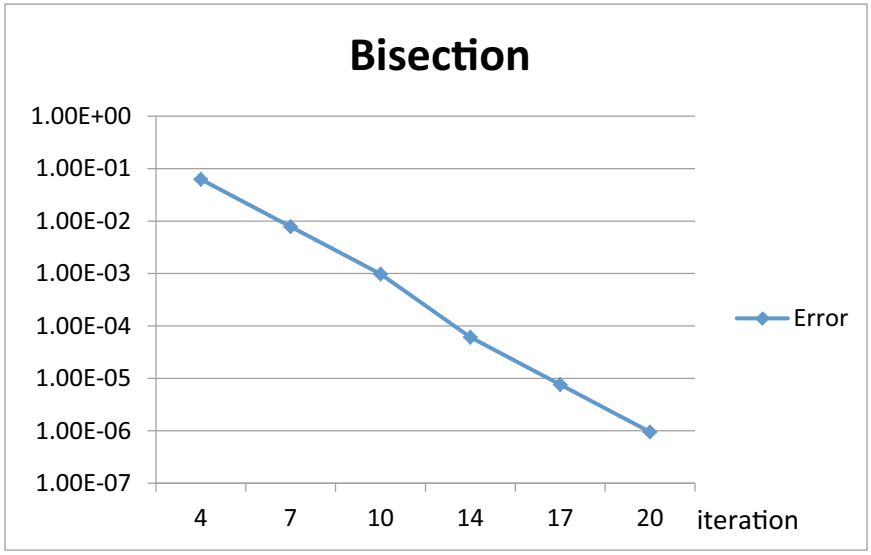
In the Cell E9, we can see the function is

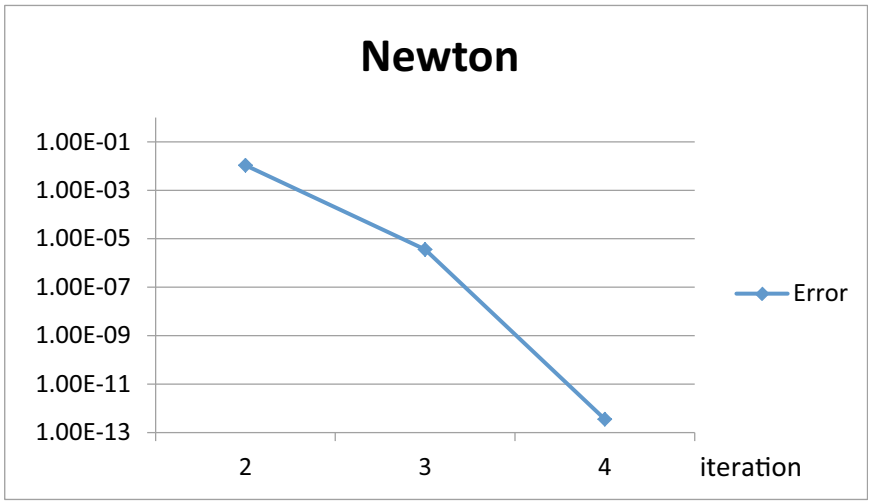
$$= \text{BSIVNewton}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{E12}, 0.5).$$

And the output is 0.3625 which is equal to output of Bisection method. The last input, 0.5, is the initial value. The most important input in the Newton–Raphson method is initial value. If we change the initial value to 0.01 or 5, we can find that the output is #VALUE! This is the biggest problem of Newton–Raphson method. If the initial is not suitable, we will not find the correct result. However, if we use a suitable initial value, then we can get a correct solution no matter how big or small initial value. The figure given below shows the F(σ) = Cbs-Cmarket. We can find that there exists a unit solution at F(σ) = 0.



Although bisection method has less initial value problem, it still has a problem of more iterations. We calculate iterations and errors for these two methods and plot the figures given below:





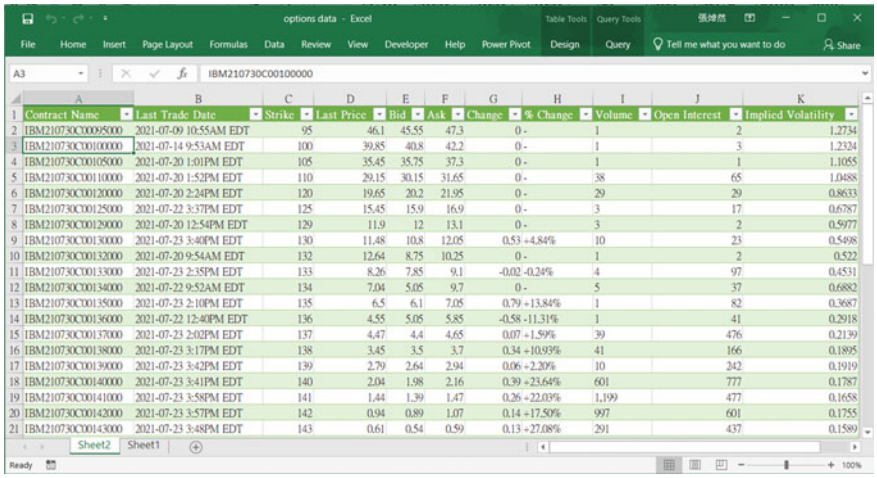
We can find that Bisection method needs 20 iterations to reduce an error of around 10^{-6}. However, Newton–Raphson method only needs four iterations to produce an error of around 10^{-13}. This problem may occur in the past but today's computer is more efficient. So, we don't need to care about this problem too much now.

7.3 Volatility Smile

The existence of volatility smile is due to Black–Scholes formula which cannot precisely evaluate the either call or put option value. The main reason is that the Black–Scholes formula assumes the stock price per share is log-normally distributed. If we introduce extra distribution parameters into the option pricing determination formula, we can obtain the constant elasticity volatility (CEV) option pricing formula. This formula can be found in Sect. 7.4 of this chapter. Lee et al. (2004) show that the CEV model performs better than the Black–Scholes model in evaluating either call or put option value.

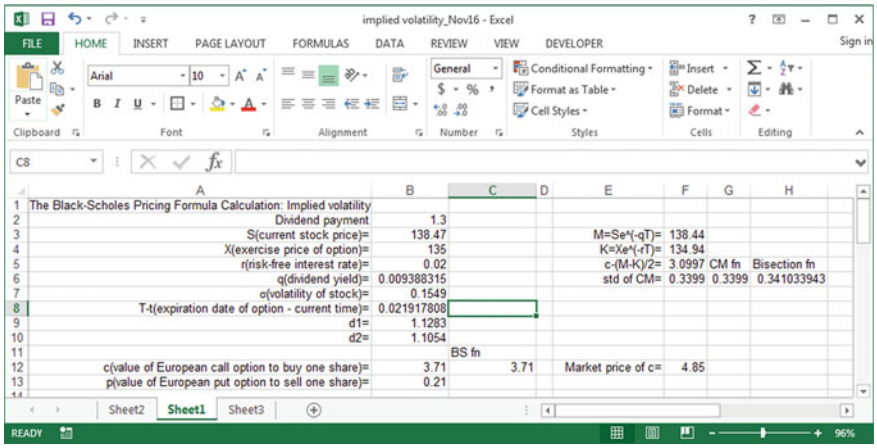
A plot of the implied volatility of an option as a function of its strike price is known as a *volatility smile*. Now we use IBM's data to show the volatility smile. The call option data listed in table given below can be found from Yahoo Finance http://finance.yahoo.com/q/op?s=IBM&date=1450396800.

We use the IBM option contract with expiration date on July 30.



Contract Name	Last Trade Date	Strike	Last Price	Bid	Ask	Change	% Change	Volume	Open Interest	Implied Volatility
IBM210730C00095000	2021-07-09 10:55AM EDT	95	46.1	45.55	47.3	0 -		1	2	1.2734
IBM210730C00100000	2021-07-14 9:53AM EDT	100	39.85	40.8	42.2	0 -		1	3	1.2324
IBM210730C00105000	2021-07-20 1:01PM EDT	105	35.45	35.75	37.3	0 -		1	1	1.1055
IBM210730C00110000	2021-07-20 1:52PM EDT	110	29.15	30.15	31.65	0 -		38	65	1.0488
IBM210730C00120000	2021-07-20 2:24PM EDT	120	19.65	20.2	21.95	0 -		29	29	0.8633
IBM210730C00125000	2021-07-22 3:37PM EDT	125	15.45	15.9	16.9	0 -		3	17	0.6787
IBM210730C00129000	2021-07-20 12:54PM EDT	129	11.9	12	13.1	0 -		3	2	0.5977
IBM210730C00130000	2021-07-23 3:40PM EDT	130	11.48	10.8	12.05	0.53	+4.84%	10	23	0.5498
IBM210730C00132000	2021-07-20 9:54AM EDT	132	12.64	8.75	10.25	0 -		1	2	0.522
IBM210730C00133000	2021-07-23 2:35PM EDT	133	8.26	7.85	9.1	-0.02	-0.24%	4	97	0.4531
IBM210730C00134000	2021-07-22 9:52AM EDT	134	7.04	5.05	9.7	0 -		5	37	0.6882
IBM210730C00135000	2021-07-23 2:10PM EDT	135	6.5	6.1	7.05	0.79	+13.84%	1	82	0.3687
IBM210730C00136000	2021-07-22 12:40PM EDT	136	4.55	5.05	5.85	-0.58	-11.31%	1	41	0.2918
IBM210730C00137000	2021-07-23 2:02PM EDT	137	4.47	4.4	4.65	0.07	+1.59%	39	476	0.2139
IBM210730C00138000	2021-07-23 3:17PM EDT	138	3.45	3.5	3.7	0.34	+10.93%	41	166	0.1895
IBM210730C00139000	2021-07-23 3:42PM EDT	139	2.79	2.64	2.94	0.06	+2.20%	10	242	0.1919
IBM210730C00140000	2021-07-23 3:41PM EDT	140	2.04	1.98	2.16	0.39	+23.64%	601	777	0.1787
IBM210730C00141000	2021-07-23 3:58PM EDT	141	1.44	1.39	1.47	0.26	+22.03%	1,199	477	0.1658
IBM210730C00142000	2021-07-23 3:57PM EDT	142	0.94	0.89	1.07	0.14	+17.50%	997	601	0.1755
IBM210730C00143000	2021-07-23 3:48PM EDT	143	0.61	0.54	0.59	0.13	+27.08%	291	437	0.1589

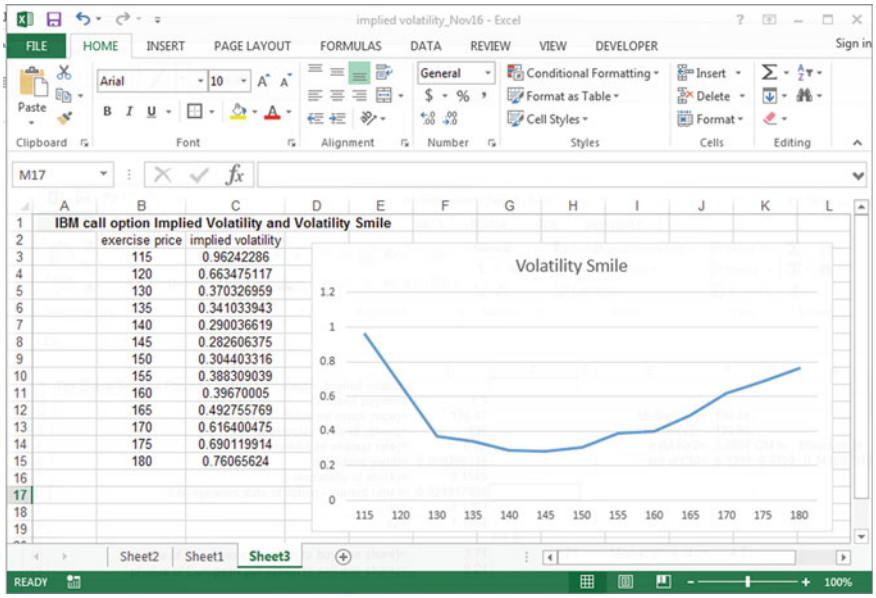
Then we use the implied volatility Excel program in last section to calculate the implied volatility with a specific exercise price list in table given above.



The Black-Scholes Pricing Formula Calculation: Implied volatility					
Dividend payment	1.3				
S(current stock price)=	138.47		M=Se^(-qT)=	138.44	
X(exercise price of option)=	135		K=Xe^(-rT)=	134.94	
r(risk-free interest rate)=	0.02		c-(M-K)/2=	3.0997	CM fn Bisection fn
q(dividend yield)=	0.009388315		std of CM=	0.3399	0.3399 0.341033943
σ(volatility of stock)=	0.1549				
T-t(expiration date of option - current time)=	0.021917808				
d1=	1.1283				
d2=	1.1054				
		BS fn			
c(value of European call option to buy one share)=	3.71	3.71	Market price of c=	4.85	
p(value of European put option to sell one share)=	0.21				

In this table, there are many inputs including dividend payment, current stock price per share, exercise price per share, risk-free interest rate, and volatility of stock and time-to-maturity. Dividend yield is calculated by dividend payment divided by current stock price. By using different methods discussed in Sect. 7.2, given the market price of the call option, we can calculate the implied volatility by using Corrado and Miller's formula and Bisection methods. In this example, we use \$135 as our exercise price for call option, the correspondent market ask price is \$4.85. The implied volatilities calculated by those two methods are 0.3399 and 0.3410, respectively.

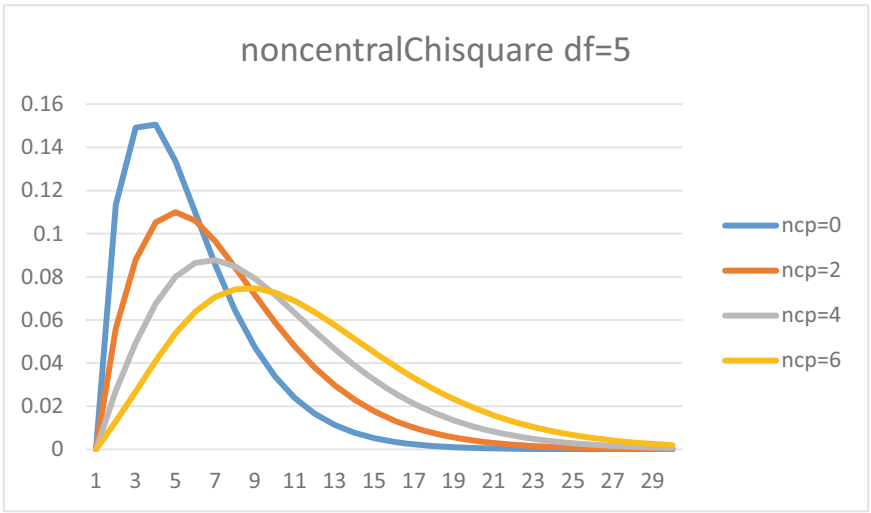
Now we calculate the implied volatility by using different exercise price and correspondent different market price.



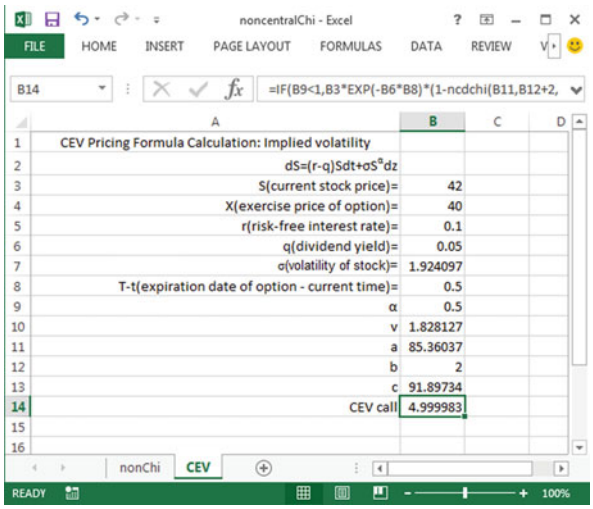
In the Excel table given above, we calculate the implied volatility for correspondent different exercise price by using Bisection method. Then by plotting the implied volatility, we can get the volatility smile as given above.

7.4 Excel Program to Estimate Implied Variance with CEV Model

In order to price a European option under a CEV model, we need a non-central chi-square distribution. The following figure shows the charts of the non-central chi-square distribution with five degrees of freedom for non-central parameter $\delta = 0, 2, 4, 6$.



Under the theory in this chapter, we can write a call option price under CEV model. The figure to do this is given below:



Hence, the formula for CEV call option in B14 is

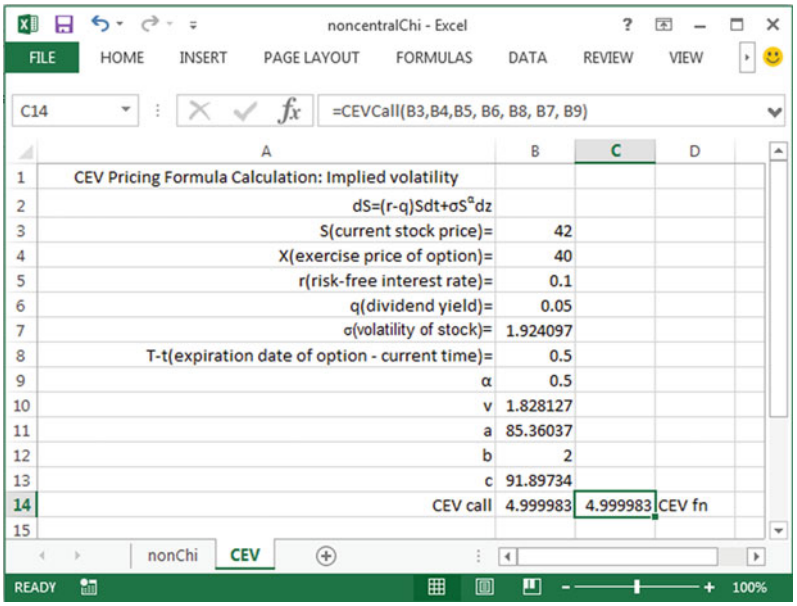
$$= \text{IF}(\text{B9} < 1, B3 * EXP(-B6 * B8) * (1 - ncdchi(B11, B12 + 2, B13)) \\ -\text{B4} * \text{EXP}(-\text{B5} * \text{B8}) * \text{ncdchi}(\text{B13}, \text{B12}, \text{B11}), \\ \text{B3} * \text{EXP}(-\text{B6} * \text{B8}) * (1 - \text{ncdchi}(\text{B13}, -\text{B12}, \text{B11})) \\ -\text{B4} * \text{EXP}(-\text{B5} * \text{B8}) * \text{ncdchi}(\text{B11}, 2 - \text{B12}, \text{B13})).$$

The ncdchi is the non-central chi-square cumulative distribution function. The function, IF, is used to separate the two conditions for this formula, $0 < \alpha < 1$ and $\alpha > 1$.

We can write a function to price the call option under CEV model. The code to accomplish this is given below:

```
'   CEV Call Option Value
Function CEVCall(S, X, r, q, T, sigma, alpha)
    Dim v As Double
    Dim aa As Double
    Dim bb As Double
    Dim cc As Double
    v = (Exp(2 * (r - q) * (alpha - 1) * T) - 1) * (sigma ^ 2) / (2 * (r - q) * (alpha
- 1))
    aa = ((X * Exp(-(r - q) * T)) ^ (2 * (1 - alpha))) / (((1 - alpha) ^ 2) * v)
    bb = 1 / (1 - alpha)
    cc = (S ^ (2 * (1 - alpha))) / (((1 - alpha) ^ 2) * v)
    If alpha < 1 Then
    CEVCall = Exp(-q * T) * S * (1 - ncdchi(aa, bb + 2, cc)) - Exp(-r * T) * X *
ncdchi(cc, bb, aa)
    Else
    CEVCall = Exp(-q * T) * S * (1 - ncdchi(cc, -bb, aa)) - Exp(-r * T) * X *
ncdchi(aa, 2 - bb, cc)
    End If
End Function
```

Use this function to value the call option which is shown below:



The CEV call option formula in C14 is

$$= \text{CEVCall}(B3, B4, B5, B6, B8, B7, B9).$$

The value of CEV call option in C14 is equal to B14.

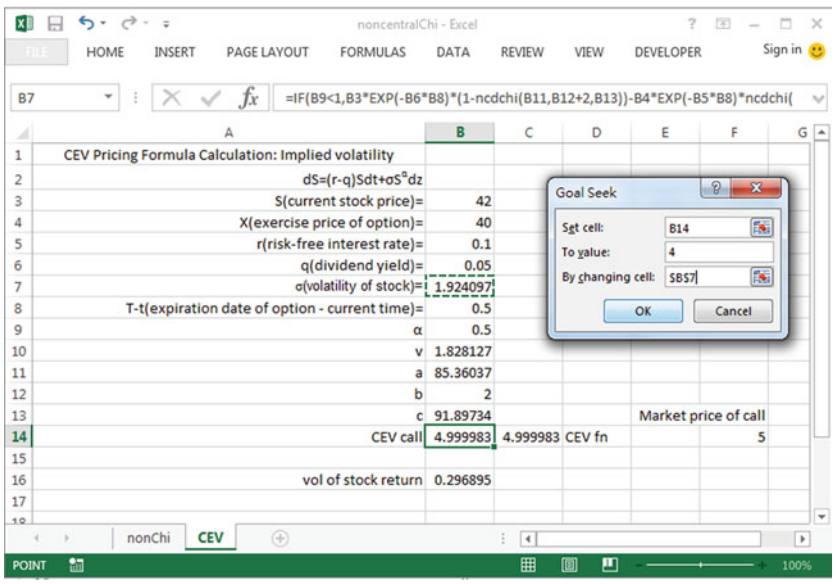
Next, we want to use Goal Seek procedure to calculate the implied volatility. To do this, we can see the figure given below:

Set cell: B14

To value: 4

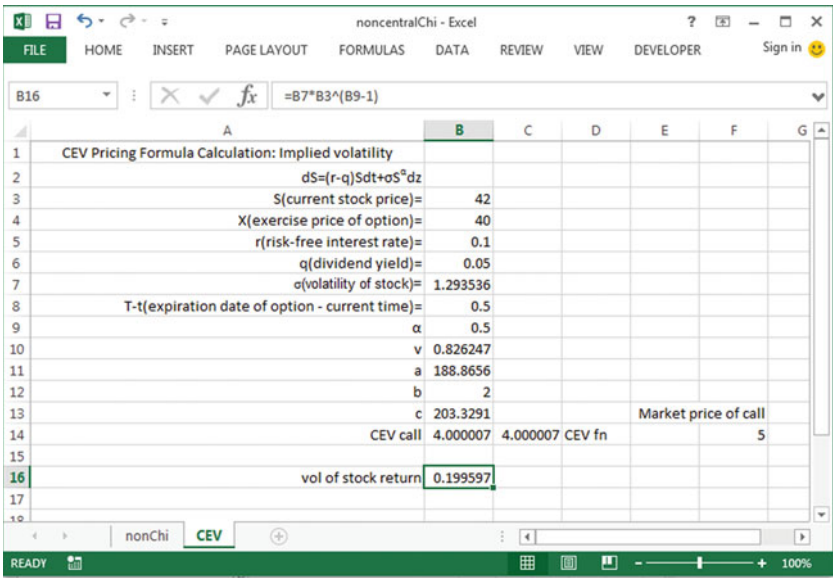
By changing cell: $B$7

After pressing the OK button, we can get the sigma value in B7.



If we want to calculate implied volatility of stock return, we show this result in B16 of the figure given below. The formula of implied volatility of stock return in B16 is

$$= \mathrm{B7} * \mathrm{B3}\hat{}(\mathrm{B9} - 1).$$



We use bisection method to write a function to calculate the implied volatility of CEV model. Following code can accomplish this task:

```
'   Estimate implied volatility by Bisection

'   Uses BSCall fn

Function CEVIVBisection(S, X, r, q, T, alpha, callprice, a, b)

    Dim yb, ya, c, yc

    yb = CEVCall(S, X, r, q, T, b, alpha) - callprice

    ya = CEVCall(S, X, r, q, T, a, alpha) - callprice

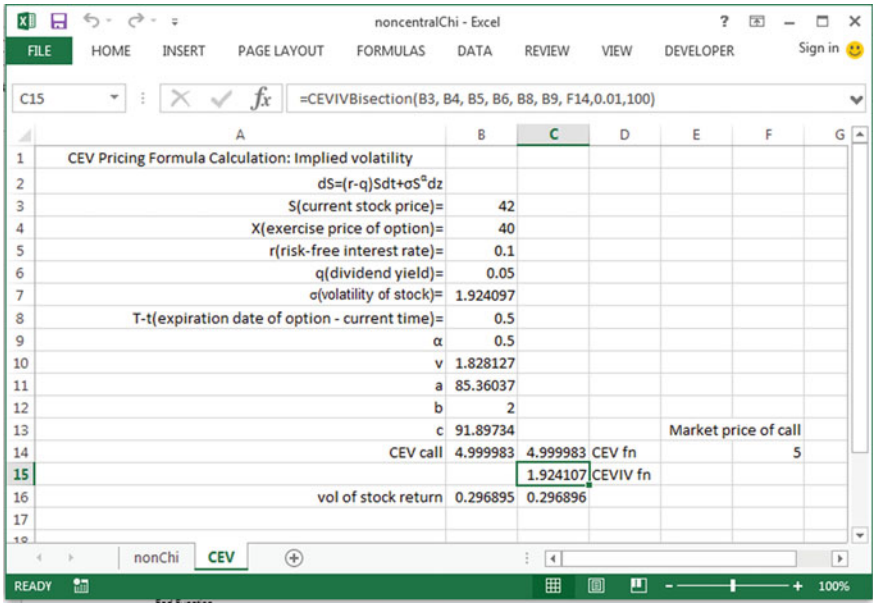
    If yb * ya > 0 Then

      CEVIVBisection = CVErr(xlErrValue)

    Else
```

```
        Do While Abs(a - b) > 0.000000001
          c = (a + b) / 2
          yc = CEVCall(S, X, r, q, T, c, alpha) - callprice
          ya = CEVCall(S, X, r, q, T, a, alpha) - callprice
          If ya * yc < 0 Then
             b = c
          Else
             a = c
          End If
         Loop
        CEVIVBisection = (a + b) / 2
    End If
End Function
```

After typing the parameters in the above function, we can get the sigma and implied volatility of stock return. The result is shown below:



The formula of sigma in CEV model in C15 is

$$= \text{CEVIVBisection}(B3, B4, B5, B6, B8, B9, F14, 0.01, 100).$$

The value of sigma in C15 is similar to B7 calculated by Goal Seek procedure. In the same way, we can calculate volatility of stock return in C16. The value of volatility of stock return in C16 is also near B16.

7.5 WEBSERVICE Function

An URL is a request and response Internet convention between two computers. A user would request a URL by typing the URL in the Internet browser, and the browser will respond to the request. For example, the user would request the USA Today website by typing in http://www.usatoday.com/ in the browser, and the browser would return the USA Today website. A lot of information is returned to the user. The browser would return a lot of text and graphical information, and the browser will format text and graphical information.

There are URLs that are constructed to return only data. One popular thing to do is retrieve stock prices from Yahoo.com. The following URL will return the stock price Microsoft for July 27, 2021,

https://query1.finance.yahoo.com/v7/finance/download/MSFT?period1=1627344000&period2=1627430400&interval=1d&events=history&includeAdjustedClose=true

The following URL will return the last stock price of IBM:

https://query1.finance.yahoo.com/v7/finance/download/IBM?period1=1627344000&period2=1627430400&interval=1d&events=history&includeAdjustedClose=true

The following URL will return the last stock price of GM:

https://query1.finance.yahoo.com/v7/finance/download/GM?period1=1627344000&period2=1627430400&interval=1d&events=history&includeAdjustedClose=true

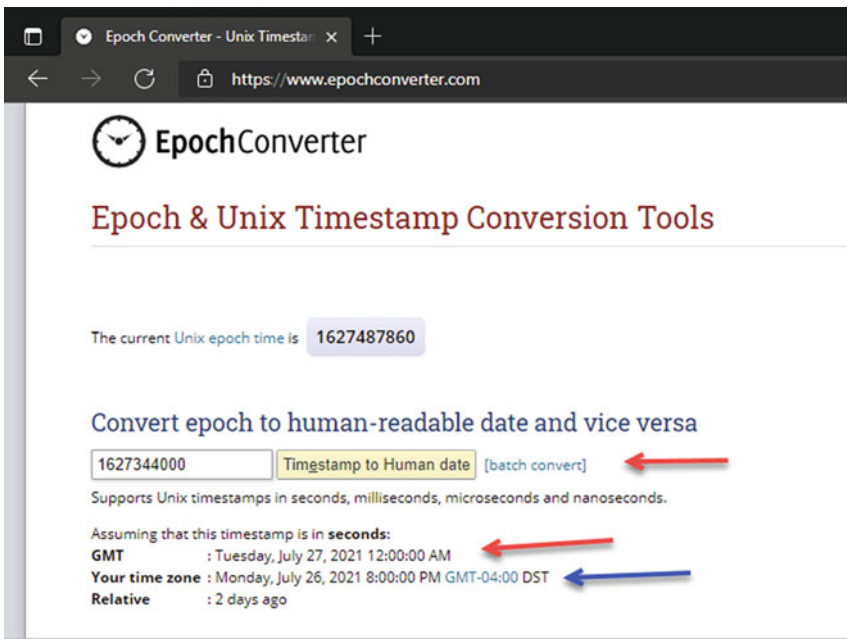
The following URL will return the last stock price of Ford:

https://query1.finance.yahoo.com/v7/finance/download/F?period1=1627344000&period2=1627430400&interval=1d&events=history&includeAdjustedClose=true

For periods the URL uses EPOCH time. The URL https://www.epochconverter.com/ defines EPOCH time as

> the number of seconds that have elapsed since January 1, 1970 (midnight UTC/GMT), not counting leap seconds (in ISO 8601: 1970-01-01T00:00:00Z). Literally speaking the epoch is Unix time 0 (midnight 1/1/1970), but 'epoch' is often used as a synonym for Unix time

The URL https://www.epochconverter.com/ has a converter to convert EPOCH to regular time.

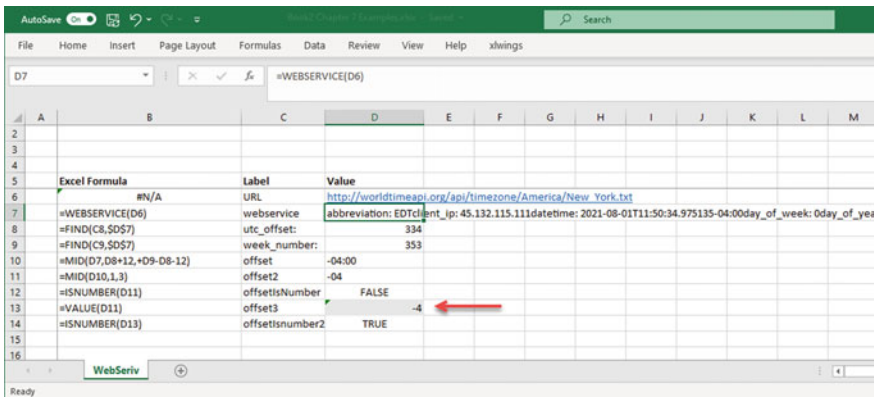


It is important to note that GMT is London time. As shown above, to get New York City time you would need to subtract GMT by 4 h during daylight savings time. During standard time you would subtract GMT by 5 h

The URL http://worldtimeapi.org/api/timezone/America/New_York.txt indicates if the offset should be 4 h or 5 h

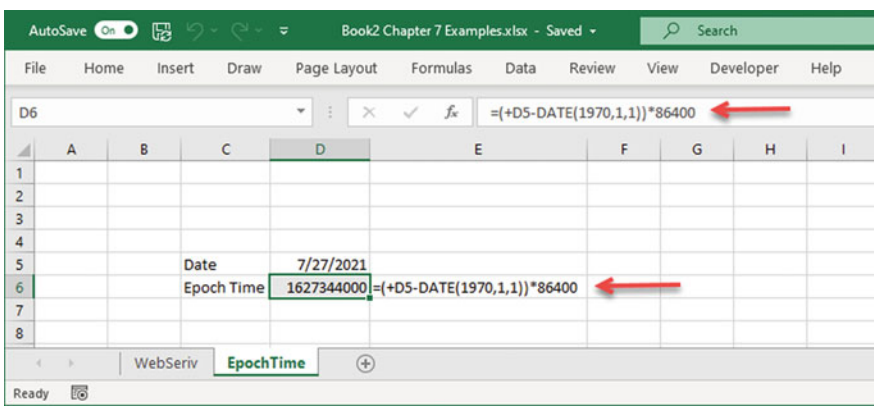


A person could use the Excel *WEBSERVICE* to retrieve or use this URL or API.



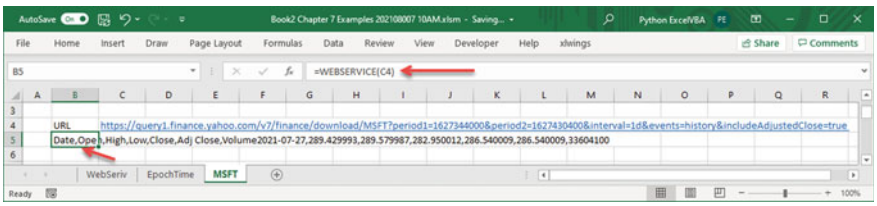
After using the *WEBSERVICE* function to retrieve the result, the steps in cells D8 to D11 are required to get the GMT offset number. Cell D4 shows the offset number.

The Excel formula to convert a date to *Epoch Time* is shown below:

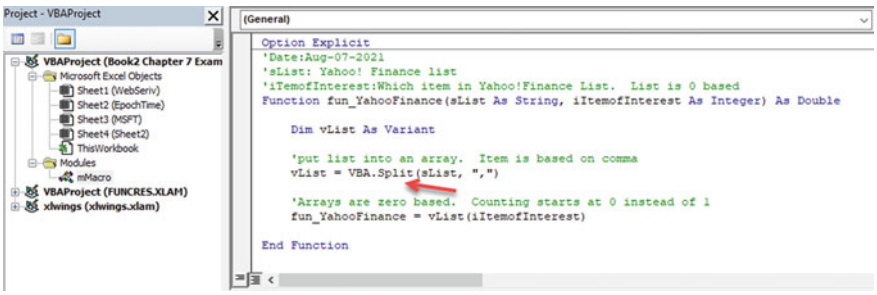


7.6 Retrieving a Stock Price for a Specific Date

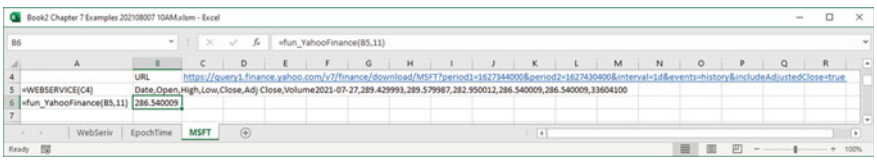
MSFT's Yahoo! Finance URL returns data as a comma-delimited list. The price of MSFT on July 27, 2021 is the second to last number, or 286.540009.



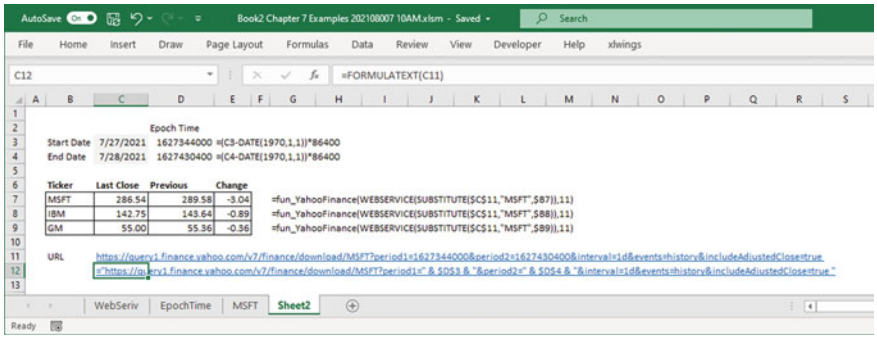
It would require a complicated Excel formula to retrieve this number. Instead, we will create a custom Excel VBA function to retrieve that number. Below is the custom VBA function to return a specific data item from a Yahoo! Finance list. One of the most important things to making this function work is the *SPLIT* command. This command transforms a delimited list into an array. In VBA, an array is 0 based, which means that the first element is considered a 0 instead of a 1.



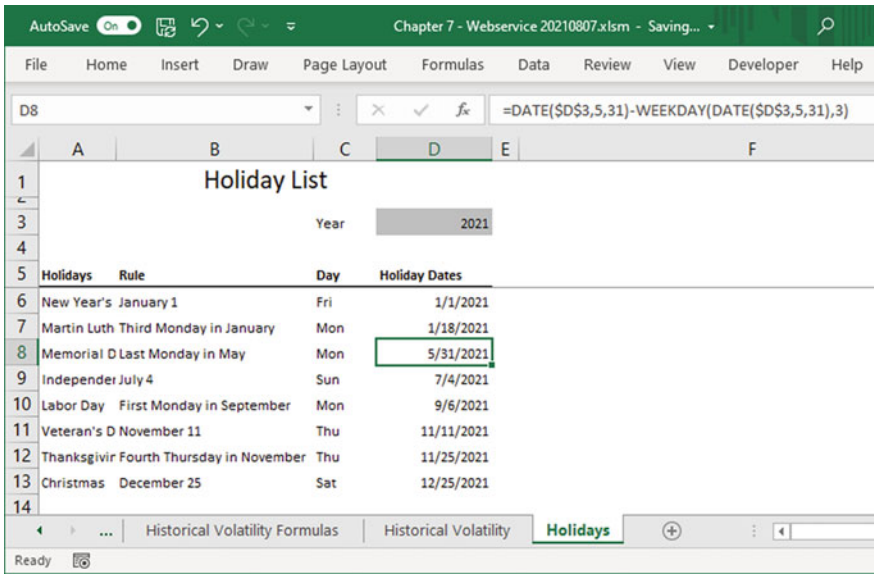
The use of the custom function is illustrated below:



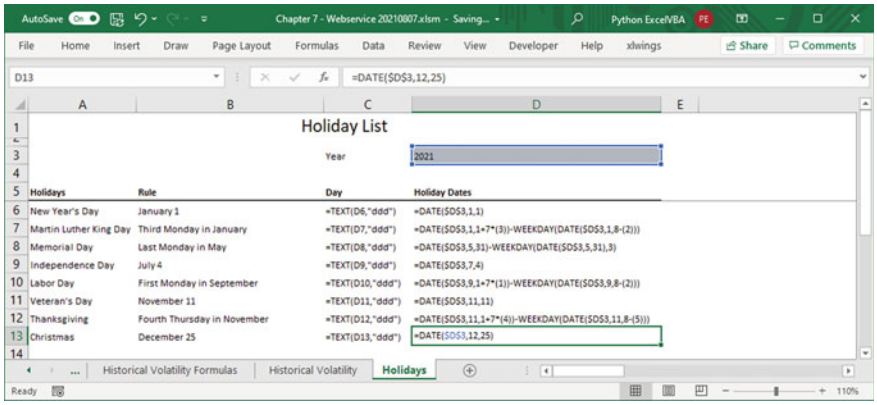
A more elaborate use of the *webservice* and *fun_YahooFinance* functions is given below. User would change the start and end dates in cells C3 and C4 to get the prices for a different date.



7.7 Calculated Holiday List

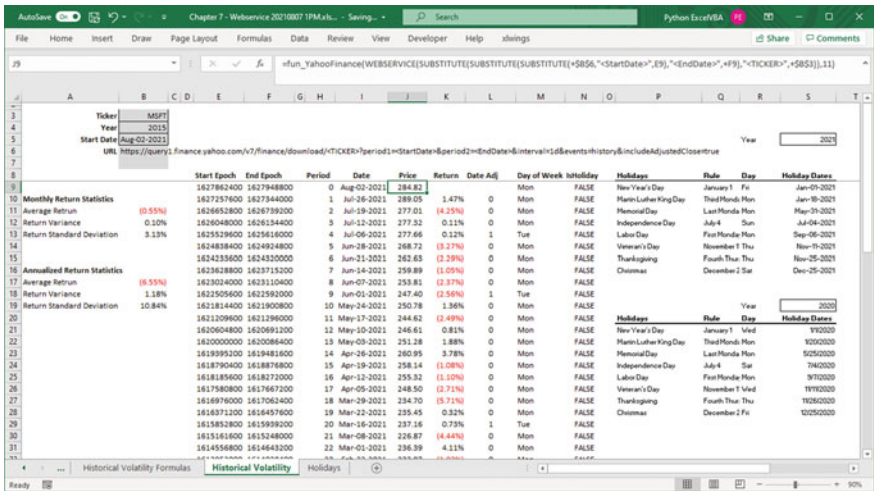


Financial calculation often needs to take into consideration the holidays. A list of holidays for 2021 is given above that is dynamically calculated using Excel functions. How each holiday is calculated is shown below:



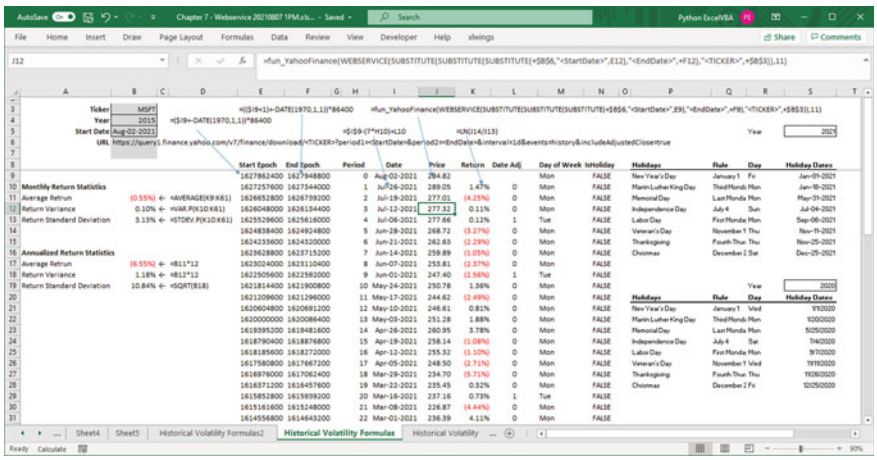
7.8 Calculating Historical Volatility

Another way to get the volatility value is to calculate historical volatility. It's a lot of effort to do this because it takes a lot of effort to get the historical price of a stock for each specific day. We will use our custom Excel function *fun_YahooFinance* and the concepts discussed above to solve this problem.

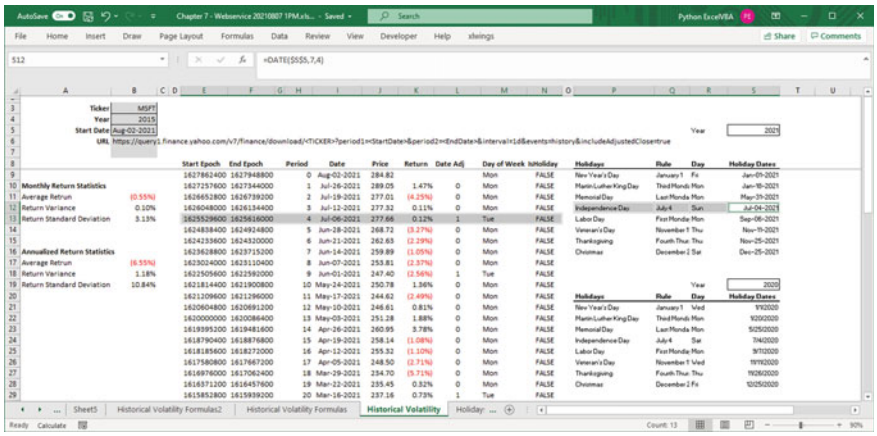


Above is a spreadsheet that calculates a 52-week historical variance for any stock. There are three input values to the spreadsheet. The three input values are "Ticker," "Year," and "Start Date." In calculating the historical variance, we have to be concerned about holidays because there are no stock prices on holidays. The "Year" input is used by the calculated calendars in columns P to S.

The formulas for the spreadsheet is shown below:



Every row in the date column is 7 days prior to the previous row. In cell H13, the date should have been September 07, 2015. In 2021, holiday calendar in column S shows that July 04, 2021 is a holiday on a Sunday. The holiday rule for trading is if a holiday lands on a Sunday, then the holiday is moved forward 1 day; this makes July 5, 2021, a trading holiday. Therefore, there is no stock price. Because of this, we have to push the date forward by 1 day to July 6, 2021. Pushing the day forward is done in column K.



7.9 Summary

In the inputs of Black and Scholes formula, only the volatility can't be measured directly. If we use the market price of an option, we can estimate the volatility implied by option market price. In this chapter, we introduce Corrado and Miller's approximation to estimate implied volatility. Next, we use the Goal Seek facility Excel to solve the root of nonlinear equation which is based on Newton–Raphson method. We apply a VBA function to calculate implied volatility by using bisection method.

We also calculated a 52-week volatility of a stock. This is a very difficult task because it is very labor intensive to get the stock price for all 52 weeks. To make it more difficult, we have to take into consideration the holidays. We demonstrate how to use the Excel *webservice* to retrieve stock prices from Yahoo! Finance.

We also showed the Excel equations to calculate holidays for any particular year dynamically.

Appendix 7.1: Application of CEV Model to Forecasting Implied Volatilities for Options on Index Futures

In this appendix, we use CEV model to forecast implied volatility (called IV hereafter) of options on index futures. Cox (1975) and Cox and Ross (1976) developed the "constant elasticity of variance (CEV) model" which incorporates an observed market phenomenon that the underlying asset variance tends to fall as the asset price increases (and vice versa). The advantage of CEV model is that it can describe the interrelationship between stock prices and its volatility. The constant elasticity of variance (CEV) model for a stock price, S, can be represented as follows:

$$dS = (r - q)Sdt + \delta S^{\alpha} dZ, \tag{7.1}$$

where r is the risk-free rate, q is the dividend yield, *dZ* is a Wiener process, δ is a volatility parameter, and α is a positive constant. The relationship between the instantaneous volatility of the asset return, $\sigma(S, t)$, and parameters in CEV model can be represented as

$$\sigma(S, t) = \delta S^{\alpha - 1}. \tag{7.2}$$

When $\alpha = 1$, the CEV model is the geometric Brownian motion model we have been using up to now. When $\alpha < 1$, the volatility increases as the stock price decreases. This creates a probability distribution similar to that observed for equities with a heavy left tail and a less heavy right tail. When $\alpha > 1$, the volatility increases as the stock price increases, giving a probability distribution with a heavy right tail and a less left tail. This corresponds to a volatility smile where the implied volatility is an increasing function of the strike price. This type of volatility smile is sometimes observed for options on futures.

The formula for pricing a European call option in CEV model is

$$C_t = \begin{cases} S_t e^{-q\tau}[1 - \chi^2(a,\ b+2,\ c)] - Ke^{-r\tau}\chi^2(c,\ b,\ a) \text{when} \alpha < 1 \\ S_t e^{-q\tau}[1 - \chi^2(c,\ -b,\ a)] - Ke^{-r\tau}\chi^2(a,\ 2-b,\ c) \text{when} \alpha > 1 \end{cases}, \tag{7.3}$$

where $a = \frac{\left[Ke^{-(r-q)\tau}\right]^{2(1-\alpha)}}{(1-\alpha)^2 \upsilon}$, $b = \frac{1}{1-\alpha}$, $c = \frac{S_t^{2(1-\alpha)}}{(1-\alpha)^2 \upsilon}$, $\upsilon = \frac{\delta^2}{2(r-q)(\alpha-1)}\left[e^{2(r-q)(\alpha-1)\tau} - 1\right]$, and $\chi^2(z, k, v)$ is the cumulative probability that a variable with a non-central χ^2 distribution with non-centrality parameter v and k degrees of freedom is less than Hsu et al. (2008) provided the detailed derivation of approximative formula for CEV model. Based on the approximated formula, CEV model can reduce computational and implementation costs rather than the complex models such as jump-diffusion stochastic volatility model. Therefore, CVE model with one more parameter than Black–Scholes–Merton Option Pricing Model (BSM) can be a better choice to improve the performance of predicting implied volatilities of index options (Singh and Ahmad 2011).

Beckers (1980) investigates the relationship between the stock price and its variance of returns by using an approximative closed-form formulas for CEV model based on two special cases of the constant elasticity class ($\alpha = 1\ or\ 0$). Based on the significant relationship between the stock price and its volatility in the empirical results, Beckers (1980) claimed that CEV model in terms of non-central Chi-square distribution performs better than BC model in terms of log-normal distribution in description of stock price behavior. MacBeth and Merville (1980) is the first paper to empirically test the performance of CEV model. Their empirical results show the negative relationship between stock prices and its volatility of returns, that is, the elasticity class is less than 2 (i.e., $\alpha < 2$). Jackwerth and Rubinstein (2001) and Lee et al. (2004) used S&P 500 index options to do empirical work and found that CEV model performed well because it took into account the negative correlation between the index level and volatility into model assumption. Pun and Wong (2013) combine asymptotics approach with CEV model to price American options. Larguinho et al. (2013) compute Greek letters under CEV model to measure

different dimension to the risk in option positions and investigate leverage effects in option markets.

Since the future price equals the expected future spot price in a risk-neutral measurement, the S&P 500 index futures prices have same distribution property of S&P 500 index prices. Therefore, for a call option on index futures can be given by Eq. (7.3) with S_t replaced by F_t and $q = r$ as Eq. (7.4)[1]:

$$C_t^F = \begin{cases} e^{-r\tau}(F_t[1-\chi^2(a,\ b+2,\ c)] - K\chi^2(c,\ b,\ a)) & \text{when} \alpha < 1 \\ e^{-r\tau}(F_t[1-\chi^2(c,\ -b,\ a)] - K\chi^2(a,\ 2-b,\ c)) & \text{when} \alpha > 1 \end{cases}, \tag{7.4}$$

where $a = \frac{K^{2(1-\alpha)}}{(1-\alpha)^2 \upsilon}$, $b = \frac{1}{1-\alpha}$, $c = \frac{F_t^{2(1-\alpha)}}{(1-\alpha)^2 \upsilon}$, $\upsilon = \delta^2 \tau$.

The MATLAB code to price European Call option on future price using CEV Model is given below:

```
function [ call ] = CevFCall(F,K,T,r,sigma,alpha)
% Compute European Call option on future price using CEV Model
% F is future price
% K is vector for options with different strike price on the same day
% Scaling S & K in the next tree line to enable
% APE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2))))
% PPE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2)))./data(:,4))
% SSE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2))).^2)
% [x,fval,exitflag, output] = fminsearch(SSE,[0.27,-1])
% Volatility = blsimpv(Price, Strike, Rate, Time, Value, Limit, Tolerance,
% Class)
% ctrl+c to stop Matlab when it is busy
KK = K;
F = F./K;
K = ones(size(K));

if (alpha ~= 1)

    v = (sigma^2)*T;

    a = K.^(2*(1-alpha))./(v*(1-alpha)^2);

    b = ones(size(K)).*(1/(1-alpha));

    c = (F.^ (2 *(1-alpha)))./(v*(1-alpha)^2);
    % Multiplying the call price by KK enable us to scale back
    % if (0 < alpha && alpha < 1)
    if (alpha < 1)
        call = KK.*( F.*( ones(size(K)) - ncx2cdf( a,b + 2,c)) -
K.*(ncx2cdf(c,b,a))).*exp(-r.*T);
    elseif (alpha > 1)
        call = KK.*( F.*ncx2cdf(c,-b,a) - K.*ncx2cdf(a,2-b,c)).*exp(-r.*T);
    end
else
    call = 0; % function not defined for alpa < 0 or = 1
end

end
```

The procedures to obtain estimated parameters of CEV model are given below:

(1) Let $C_{i,n,t}^F$ be market price of the *n*th option contract in category i, $\widehat{C_{i,n,t}^F}(\delta_0, \alpha_0)$ is the model option price determined by CEV model in Eq. (7.4) with the initial value of parameters, $\delta = \delta_0$ and $\alpha = \alpha_0$. For *n*th option

[1] When substituting $q = r$ into $\upsilon = \frac{\delta^2}{2(r-q)(\alpha-1)}\left[e^{2(r-q)(\alpha-1)\tau} - 1\right]$, we can use L'Hospital's rule to obtain υ. Let $x = r - q$, then $\lim_{x\to 0} \frac{\delta^2\left[e^{2x(\alpha-1)\tau} - 1\right]}{2x(\alpha-1)} = \lim_{x\to 0} \frac{\frac{\partial \delta^2\left[e^{2x(\alpha-1)\tau} - 1\right]}{\partial x}}{\frac{\partial 2x(\alpha-1)}{\partial x}} = \lim_{x\to 0} \frac{(2(\alpha-1)\tau)\delta^2\left[e^{2x(\alpha-1)\tau}\right]}{2(\alpha-1)} = \lim_{x\to 0} \frac{\tau\delta^2\left[e^{2x(\alpha-1)\tau}\right]}{1} = \tau\delta^2$.

contract in category i at date t, the difference between market price and model option price can be described as

$$\varepsilon^{F}_{i,n,t} = C^{F}_{i,n,t} - \widehat{C^{F}_{i,n,t}}(\delta_0, \alpha_0). \tag{7.5}$$

The Matlab code to find initial value of parameters in CEV model is given below:

```
function STradingTM=cevslpine(TradingTM,TM)
sigma=[0.1:0.05:0.7];
alpha=[-0.5; -0.3; -0.1; 0.1; 0.3; 0.5; 0.7; 0.9];
LA=length(alpha);
LB=length(sigma);

L=length(TradingTM);
Tn=ones(L,1);
Tr=ones(L,1);
y=ones(L,length(alpha),length(sigma));


a=ones(L,1);
b=ones(L,1);
iniError=ones(L,1);
inisigmaplace=ones(L,1);
iniaplhaplace=ones(L,1);
inisigma=ones(L,1);
inialpha=ones(L,1);

for i=1:L

    Tn(i)=Tr(i)+TradingTM(i,1)-1;
    if(i<L) Tr(i+1)=Tn(i)+1; end
end


for k=1:L
for i=1:LA
  for j=1:LB
    y(k,i,j)= sum(abs(TM(Tr(k):Tn(k),2)-CevFCall(TM(Tr(k):Tn(k),3), TM(Tr(k):Tn(k),1),
TM(Tr(k):Tn(k),4)/360.0, TM(Tr(k):Tn(k),5), sigma(j), alpha(i))));
  end
end
[~,b]=min(y(k,:,:));
  [iniError(k),inisigmaplace(k)]=min(min(y(k,:,:)));
  inialphaplace(k)=b(inisigmaplace(k));
  inisigma(k)=sigma(inisigmaplace(k));
  inialpha(k)=alpha(inialphaplace(k));
  disp(sprintf('iteration %d contract %d alpha and %d sigma', k, i,j));
end

STradingTM=[TradingTM Tr Tn inisigma inialpha];
end
[~,b]=min(y(k,:,:));
  [iniError(k),inisigmaplace(k)]=min(min(y(k,:,:)));
  inialphaplace(k)=b(inisigmaplace(k));
  inisigma(k)=sigma(inisigmaplace(k));
  inialpha(k)=alpha(inialphaplace(k));
  disp(sprintf('iteration %d contract %d alpha and %d sigma', k, i,j));
end

STradingTM=[TradingTM Tr Tn inisigma inialpha];
end
```

(2) For each date t, we can obtain the optimal parameters in each group by solving the minimum value of absolute pricing errors (minAPE) as

$$\text{minAPE}_{i,t} = \min_{\delta_0,\alpha_0} \sum_{n=1}^{N} \left| \varepsilon^{F}_{i,n,t} \right|, \tag{7.6}$$

where N is the total number of option contracts in group i at time t.

(3) We use optimization function in MATLAB to find a minimum value of the unconstrained multivariable function. The function code is given below:

$$[x,\ fval] = \text{fminunc}(\text{fun},\ x_0), \tag{7.7}$$

where x is the optimal parameters of CEV model, fval is the local minimum value of minAPE, fun is the specified MATLAB function of Eq. (7.4), and x_0 is the initial points of

parameters obtained in Step (1). The algorithm of fminunc function is based on quasi-Newton method. The MATLAB code is given below:

```
function [ call ] = CevFCalltr(F,K,T,r,sigma,alpha)
% Compute European Call option on future price using CEV Model
% F is future price
% K is vector for options with different strike price on the same day
% Scaling S & K in the next tree line to enable
% APE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2))))
% PPE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2)))./data(:,4))
% SSE=@(x) sum(abs(data(:,4)-CevFCall(data(:,7), data(:,1), data(:, 10)/360,
data(:,11), x(1),x(2))).^2)
% [x,fval,exitflag, output] = fminsearch(SSE,[0.27,-1])
% Volatility = blsimpv(Price, Strike, Rate, Time, Value, Limit, Tolerance,
% Class)
% ctrl+c to stop Matlab when it is busy

if (alpha ~= 1)



    v = (sigma^2)*T;

    a = K.^(2*(1-alpha))./(v*(1-alpha)^2);

    b = ones(size(K)).*(1/(1-alpha));

    c = (F.^ (2 *(1-alpha)))./(v*(1-alpha)^2);
    % Multiplying the call price by KK enable us to scale back
    % if (0 < alpha && alpha < 1)
    if (alpha < 1)
        call = ( F.*( ones(size(K)) - ncx2cdf( a,b + 2,c)) -
K.*(ncx2cdf(c,b,a))).*exp(-r.*T);
    elseif (alpha > 1)
        call =( F.*ncx2cdf(c,-b,a) - K.*ncx2cdf(a,2-b,c)).*exp(-r.*T);
    end
else
    call = 0; % function not defined for alpa < 0 or = 1
end

end

function EstCev=CevIVIA(Ini_id, Ini_ed, STradingTM,TM)

L=Ini_ed-Ini_id+1;
Tr=STradingTM(:,3);
Tn=STradingTM(:,4);
x_1=STradingTM(:,5);
x_2=STradingTM(:,6);

 EstCev=ones(L,9);
 CIVAPE=ones(L,1);
 CIAAPE=ones(L,1);
 CErrorAPE=ones(L,1);
 CIVPPE=ones(L,1);
 CIAPPE=ones(L,1);
 CErrorPPE=ones(L,1);
 CIVSSE=ones(L,1);
 CIASSE=ones(L,1);
 CErrorSSE=ones(L,1);

 %countforloop=0;

fileID=fopen('EstCev.txt', 'w');
%parfor i=1:L
parfor i=1:L

  Id_global=Ini_id+i-1;
     APE=@(x) sum(abs(TM(Tr(i):Tn(i),2)-CevFCall(TM(Tr(i):Tn(i),3), TM(Tr(i):Tn(i),1),
TM(Tr(i):Tn(i),4)/360.0, TM(Tr(i):Tn(i),5), x(1), x(2))));
```

```
   % [x,fval] =fminsearch(APE, [x0(i), 0.5], options);
    % using fmincon will cause error because the warning :Warning: Large-scale (trust
region) method does not currently solve this type of problem,
  %                                                          switching to medium-
scale (line search).
%  disp(sprintf('fminunc doing %d contract with initial sigma %d  and alpha %d sigma',
i, x_1(i), x_2(i)));
   options = psoptimset('UseParallel', 'always', 'CompletePoll', 'on', 'Vectorized',
'off', 'TimeLimit', 30, 'TolFun', 1e-2, 'TolX', 1e-4);
  [x,fval] =fminunc(APE, [x_1(i), x_2(i)],options);
    disp(sprintf('%d Id_global contract, %d contract local minimum IV is %d and aplha
is %d, minAPE is %d with initial sigma %d  and alpha %d',Id_global, i, x(1), x(2),
fval, x_1(i), x_2(i)));

    fprintf(fileID, '%d Id_global contract, %d contract local minimum IV is %d and
aplha is %d, minAPE is %d where initial sigma %d  and alpha %d',Id_global, i, x(1),
x(2), fval, x_1(i), x_2(i));

    CIVAPE(i)=x(1);
    CIAAPE(i)=x(2);
     CErrorAPE(i)=fval;
      CErrorPPE(i)=abs(sum((TM(Tr(i):Tn(i),2)-CevFCall(TM(Tr(i):Tn(i),3),
TM(Tr(i):Tn(i),1), TM(Tr(i):Tn(i),4)/360.0,TM(Tr(i):Tn(i),5), x(1),
x(2)))./TM(Tr(i):Tn(i),2)));
   CErrorSSE(i)=sum(abs(TM(Tr(i):Tn(i),2)-CevFCall(TM(Tr(i):Tn(i),3),
TM(Tr(i):Tn(i),1), TM(Tr(i):Tn(i),4)/360.0,TM(Tr(i):Tn(i),5), x(1), x(2))).^2);

end
disp(sprintf('farloop is over'));
fclose(fileID);
EstCev=[CIVAPE CIAAPE CErrorAPE CIVPPE CIAPPE CErrorPPE CIVSSE CIASSE CErrorSSE];

%matlabpool close
end
```

The data is the options on S&P 500 index futures which expired within January 1, 2010 to December 31, 2013which are traded at the Chicago Mercantile Exchange (CME).[2] The reason for using options on S&P 500 index futures instead of S&P 500 index is to eliminate from non-simultaneous price effects between options and its underlying assets (Harvey and Whaley 1991). The option and future markets are closed at 3:15 p.m. Central Time (CT), while stock market is closed at 3 p.m. CT. Therefore, using closing option prices to estimate the volatility of underlying stock return is problematic even though the correct option pricing model is used. In addition to no non-synchronous price issue, the underlying assets, S&P 500 index futures, do not need to be adjusted for discrete dividends. Therefore, we can reduce the pricing error in accordance with the needless dividend adjustment. According to the suggestions in Harvey and Whaley (1991, 1992a, 1992b), we select simultaneous index option prices and index future prices to do empirical analysis.

The risk-free rate is based on 1-year Treasury Bill from Federal Reserve Bank of ST. LOUIS.[3] Daily closing price and trading volumes of options on S&P 500 index futures and its underlying asset can be obtained from Datastream.

The futures options expired on March, June, and September in both 2010 and 2011 are selected because they have over 1-year trading date (above 252 observations) while other options only have more or less 100 observations. Studying futures option contracts with same expired months in 2010 and 2011 will allow the examination of IV characteristics and movements over time as well as the effects of different market climates.

In order to ensure reliable estimation of IV, we estimate market volatility by using multiple option transactions instead of a single contract. For comparing prediction power of Black model and CEV model, we use all futures options expired in 2010 and 2013 to generate implied volatility surface. Here we exclude the data based on the following criteria:

(1) IV cannot be computed by Black model.
(2) Trading volume is lower than 10 for excluding minuscule transactions.
(3) Time-to-maturity is less than 10 days for avoiding liquidity-related biases.
(4) Quotes not satisfying the arbitrage restriction: excluding option contact if its price larger than the difference between S&P500 index future and exercise price.
(5) Deep-in/out-of-money contacts where the ratio of S&P500 index future price to exercise price is either above 1.2 or below 0.8.

After arranging data based on these criteria, we still have 30,364 observations of future options which are expired within the period of 2010–2013. The period of option prices is from March 19, 2009 to November 5, 2013.

[2] Nowadays, Chicago Mercantile Exchange (CME), Chicago Board of Trade (CBOT), New York Mercantile Exchange (NYMEX), and Commodity Exchange (COMEX) are merged and operate as designated contract markets (DCM) of the CME Group which is the world's leading and most diverse derivatives marketplace. Website of CME group: http://www.cmegroup.com/.

[3] Website of Federal Reserve Bank of ST. LOUIS: http://research.stlouisfed.org/.

To deal with moneyness- and maturity-related biases, we use the "implied volatility matrix" to find proper parameters in CEV model. The option contracts are divided into nine categories by moneyness and time-to-maturity. Option contracts are classified by moneyness level as at-the-money (ATM), out-of-the-money (OTM), or in-the-money (ITM) based on the ratio of underlying asset price, S, to exercise price, K. If an option contract with S/K ratio is between 0.95 and 1.01, it belongs to ATM category. If its S/K ratio is higher (lower) than 1.01 (0.95), the option contract belongs to ITM (OTM) category.

According to the large observations in ATM and OTM, we divide moneyness-level group into five levels: ratio above 1.01, ratio between 0.98 and 1.01, ratio between 0.95 and 0.98, ratio between 0.90 and 0.95, and ratio below 0.90. By expiration day, we classified option contracts into short term (less than 30 trading days), medium term (between 30 and 60 trading days), and long term (more than 60 trading days).

In Fig. 7.1, we find that each option on index future contract's IV estimated by Black model varies across moneyness and time-to-maturity. This graph shows volatility skew (or smile) in options on S&P 500 index futures, i.e., the implied volatilities decrease as the strike price increases (the moneyness level decreases).

Even though everyday implied volatility surface changes, this characteristic still exists. Therefore, we divided future option contracts into a six by four matrix based on moneyness and time-to-maturity levels when we estimate implied volatilities of futures options in CEV model framework in accordance with this character. The whole option samples expired within the period of 2010–2013 contains 30,364 observations. The whole period of option prices is from March 19, 2009 to November 5, 2013. The observations for each group are presented in Table 7.1.

The whole period of option prices is from March 19, 2009 to November 5, 2013. Total observation is 30, 364. The lengths of period in groups are various. The range of lengths is from 260 (group with ratio below 0.90 and time-to-maturity within 30 days) to 1,100 (whole samples).

Since most trades are in the futures options with short time-to-maturity, the estimated implied volatility of the option samples in 2009 may be significantly biased because we didn't collect the futures options expired in 2009. Therefore, we only use option prices in the period between January 1, 2010 and November 5, 2013 to estimate parameters of CEV model. In order to find global optimization instead of local minimum of absolute pricing errors, the ranges for searching suitable δ_0 and α_0 are set as $\delta_0 \in [0.01, 0.81]$ with interval 0.05, and $\alpha_0 \in [-0.81,\ 1.39]$ with interval 0.1, respectively. We find the value of parameters, $\left(\widehat{\delta_0}, \widehat{\alpha_0}\right)$, within the ranges such that minimize value of absolute pricing errors in Eq. (7.5). Then we use this pair of parameters, $\left(\widehat{\delta_0}, \widehat{\alpha_0}\right)$, as optimal initial estimates in the procedure of estimating local minimum minAPE based on Steps (1)–(3). The initial parameter setting of CEV model is presented in Table 7.2.

The sample period of option prices is from January 1, 2010 to November 5, 2013. During the estimating procedure for initial parameters of CEV model, the volatility for S&P 500 index futures equals to $\delta_0 S^{\alpha_0 - 1}$.

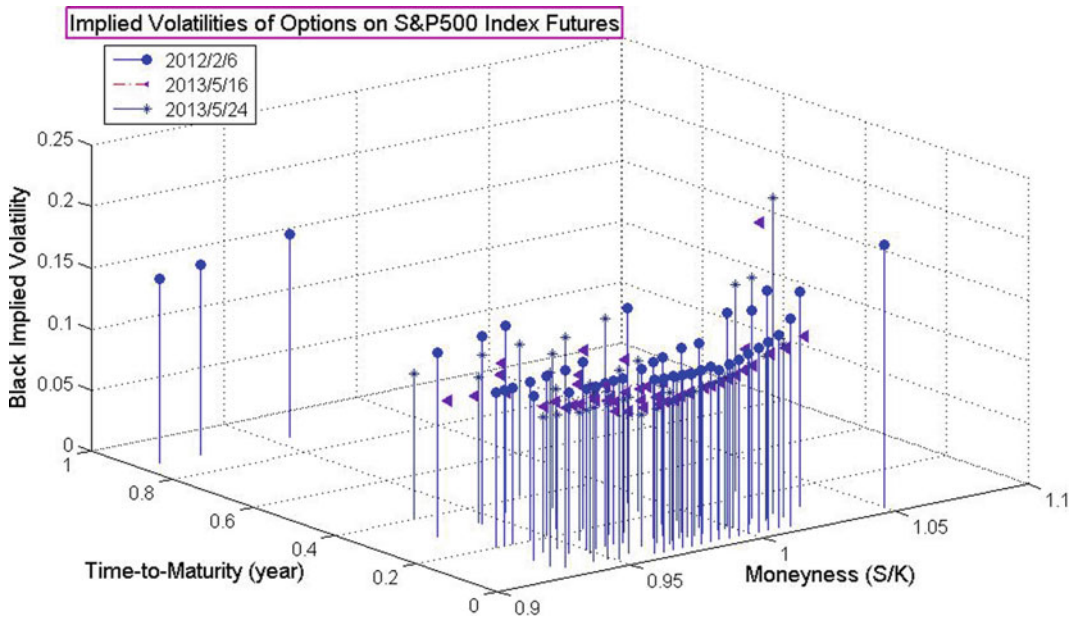


Fig. 7.1 Implied volatilities in Black model

Table 7.1 Average daily and total number of observations in each group

Time-to-maturity (TM)	TM < 30		30 ≦ TM ≦ 60		TM > 60		All TM	
Moneyness (S/K ratio)	Daily Obs	Total Obs	Daily Obs	Total Obs	Daily Obs	Total Obs	Daily Obs	Total Obs
S/K ratio > 1.01	1.91	844	1.64	499	1.53	462	2.61	1,805
0.98 ≦ S/K ratio ≦ 1.01	4.26	3,217	2.58	1,963	2.04	1,282	6.53	6,462
0.95 ≦ S/K ratio < 0.98	5.37	4,031	3.97	3,440	2.58	1,957	9.32	9,428
0.9 ≦ S/K ratio < 0.95	4.26	3,194	4.37	3,825	3.27	2,843	9.71	9,862
S/K ratio < 0.9	2.84	764	2.68	798	2.37	1,244	4.42	2,806
All Ratio	12.59	12,050	10.78	10,526	7.45	7,788	27.62	30,364

Table 7.2 Initial parameters of CEV model for estimation procedure

Time-to-maturity (TM)	TM < 30		30 ≦ TM ≦ 60		TM > 60		All TM	
Moneyness (S/K ratio)	α_0	δ_0	α_0	δ_0	α_0	δ_0	α_0	δ_0
S/K ratio > 1.01	0.677	0.400	0.690	0.433	0.814	0.448	0.692	0.429
0.98≦S/K ratio≦1.01	0.602	0.333	0.659	0.373	0.567	0.361	0.647	0.345
0.95≦S/K ratio < 0.98	0.513	0.331	0.555	0.321	0.545	0.349	0.586	0.343
0.9≦S/K ratio < 0.95	0.502	0.344	0.538	0.332	0.547	0.318	0.578	0.321
S/K ratio < 0.9	0.777	0.457	0.526	0.468	0.726	0.423	0.709	0.423
All ratio	0.854	0.517	0.846	0.512	0.847	0.534	0.835	0.504

In Table 7.2, the average sigma are almost the same while the average alpha value in either each group or whole sample is less than one. This evidence implies that the alpha of CEV model can capture the negative relationship between S&P 500 index future prices and its volatilities shown in Fig. 7.1. The instant volatility of S&P 500 index future prices equals to $\delta_0 S^{\alpha_0 - 1}$ where S is S&P 500 index future prices, δ_0 and α_0 , are the parameters in CEV model. The estimated parameters in Table 7.2 are similar across time-to-maturity level but volatile across moneyness.

Because of the implementation and computational costs, we select the sub-period from January 2012 to November 2013 to analyze the performance of CEV model. The total number of observations and the length of trading days in each group are presented in Table 7.3. The estimated parameters in Table 7.2 are similar across time-to-maturity level but volatile across moneyness. Therefore, we investigate the performance of all groups except the groups on the bottom row of Table 7.3. The performance of models can be measured by either the implied volatility graph or the average absolute pricing errors (AveAPE). The implied volatility graph should be flat across different moneyness level and time-to-maturity. We use subsample like Bakshi et al. (1997) and Chen et al. (2009) did to test implied volatility consistency among moneyness-maturity categories. Using the subsample data from January 2012 to May 2013 to test in-the-sample fitness, the average daily implied volatility of both CEV and Black models, and average alpha of CEV model are computed in Table 7.4. The fitness performance is shown in Table 7.5. The implied volatility graphs for both models are shown in Fig. 7.2. In Table 7.4, we estimate the optimal parameters of CEV model by using a more efficient program. In this efficient program, we scale the strike price and future price to speed up the program where the implied volatility of CEV model equals to $\delta(\text{ratio}^{\alpha-1})$, ratio is the moneyness level, and δ and α are the optimal parameters of program which are not the parameters of CEV model in Eq. (7.4). In Table 7.5, we found that CEV model performs well at in-the-money group.

The subsample period of option prices is from January 1, 2012 to November 5, 2013. Total observation is 13, 434. The lengths of period in groups are various. The range of lengths is from 47 (group with ratio below 0.90 and time-to-maturity within 30 days) to 1,100 (whole samples). The range of daily observations is from 1 to 30.

Figure 7.2 shows the IV computed by CEV and Black models. Although their implied volatility graphs are similar in each group, the reasons to cause volatility smile are totally different. In Black model, the constant volatility setting is misspecified. The volatility parameter of Black model in

Table 7.3 Total number of observations and trading days in each group

Time-to-maturity (TM)	TM < 30		30 ≦ TM ≦ 60		TM > 60		All TM	
Moneyness (S/K ratio)	Days	Total Obs	Days	Total Obs	Days	Total Obs	Days	Total Obs
S/K ratio > 1.01	172	272	104	163	81	122	249	557
0.98 ≦ S/K ratio≦ 1.01	377	1,695	354	984	268	592	448	3,271
0.95 ≦ S/K ratio < 0.98	362	1,958	405	1,828	349	1,074	457	4,860
0.9 ≦ S/K ratio < 0.95	315	919	380	1,399	375	1,318	440	3,636
S/K ratio < 0.9	32	35	40	73	105	173	134	281
All ratio	441	4,879	440	4,447	418	3,279	461	12,605

Table 7.4 Average daily parameters of in-sample

Time-to-maturity (TM)	TM < 30				30 ≦ TM ≦ 60				TM > 60				All TM			
Moneyness (S/K ratio)	CEV			Black	CEV			Black	CEV			Black	CEV			Black
Parameters	α	δ	IV	IV	α	δ	IV	IV	α	δ	IV	IV	α	δ	IV	IV
S/K ratio > 1.01	0.29	0.19	0.188	0.200	0.14	0.18	0.183	0.181	0.29	0.21	0.204	0.196	0.25	0.19	0.1890	0.1882
0.98≦S/K ratio≦1.01	0.34	0.16	0.162	0.1556	0.30	0.16	0.154	0.147	0.14	0.16	0.155	0.155	0.39	0.17	0.151	0.150
0.95≦S/K ratio < 0.98	0.22	0.13	0.137	0.135	0.30	0.13	0.134	0.131	0.24	0.14	0.141	0.139	0.37	0.14	0.136	0.132
0.9≦S/K ratio < 0.95	0.05	0.15	0.159	0.152	0.25	0.13	0.133	0.128	0.26	0.14	0.136	0.131	0.38	0.14	0.135	0.129
S/K ratio < 0.9	−0.23	0.22	0.252	0.243	−1.67	0.14	0.193	0.159	0.25	0.15	0.145	0.142	0.23	0.15	0.157	0.152

Table 7.5 AveAPE performance for in-sample fitness

Time-to-maturity (TM)	TM < 30			30 ≦ TM ≦ 60			TM > 60			All TM		
Moneyness (S/K ratio)	CEV	Black	Obs	CEV	Black	Obs	CEV	Black	Obs	CEV	Black	Obs
S/K ratio > 1.01	**1.65**	1.88	202	1.81	**1.77**	142	5.10	**5.08**	115	**5.80**	6.51	459
0.98 ≦ S/K ratio ≦ 1.01	**6.63**	7.02	1,290	**4.00**	4.28	801	4.59	**4.53**	529	**18.54**	18.90	2,620
0.95 ≦ S/K ratio < 0.98	2.38	**2.34**	1,560	4.25	**4.14**	1,469	3.96	**3.89**	913	14.25	**14.15**	3,942
0.9 ≦ S/K ratio < 0.95	0.69	**0.68**	710	1.44	**1.43**	1,094	3.68	**3.62**	1,131	**7.08**	7.10	2,935
S/K ratio < 0.9	0.01	0.01	33	**0.13**	0.18	72	0.61	**0.60**	171	0.69	**0.68**	276

Fig. 7.2b varies across moneyless and time-to-maturity levels while the IV in CEV model is a function of the underlying price and the elasticity of variance (alpha parameter). Therefore, we can image that the prediction power of CEV model will be better than Black model because of the explicit function of IV in CEV model. We can use alpha to measure the sensitivity of relationship between option price and its underlying asset. For example, in Fig. 7.2c, the in-the-money future options near expired date have significantly negative relationship between future price and its volatility.

The in-sample period of option prices is from January 1, 2012 to May 30, 2013. In the in-sample estimating procedure, CEV implied volatility for S&P 500 index futures

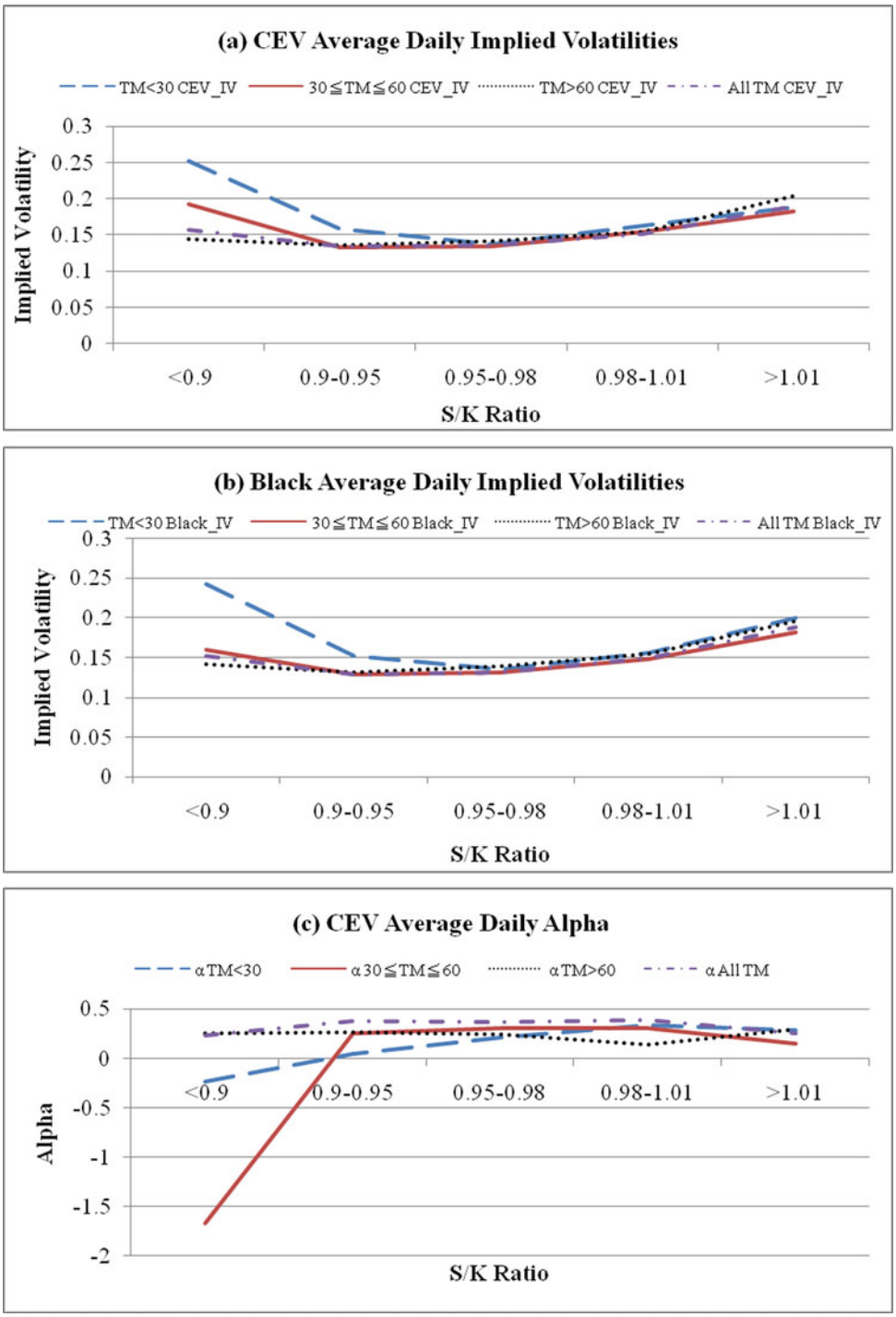


Fig. 7.2 Implied volatilities and CEV alpha graph

(CEV IV) equals to $\delta(\text{S /K ratio})^{\alpha-1}$ in accordance to reduce computational costs. The optimization setting of finding CEV IV and Black IV is under the same criteria.

The in-sample period of option prices is from January 1, 2012 to May 30, 2013.

The better performance of CEV model may result from the overfitting issue that will hurt the forecastability of CEV model. Therefore, we use out-of-sample data from June 2013 to November 2013 to compare the prediction power of Black and CEV models. We use the estimated parameters in previous day as the current day's input variables of model. Then, the theoretical option price computed by either Black or CEV model can calculate bias between theoretical price and market price. Thus, we can calculate the average absolute pricing errors (AveAPE) for both models. The lower the value of a model's AveAPE, the higher the pricing

Table 7.6 AveAPE performance for out-of-sample

Time-to-maturity(TM)	TM < 30		30 ≦ TM ≦ 60		TM > 60		All TM	
Moneyness (S/K ratio)	CEV	Black	CEV	Black	CEV	Black	CEV	Black
S/K ratio > 1.01	**3.22**	3.62	**3.38**	4.94	**8.96**	13.86	**4.25**	5.47
0.98 ≦ S/K ratio ≦ 1.01	**2.21**	2.35	2.63	**2.53**	**3.47**	3.56	**2.72**	2.75
0.95 ≦ S/K ratio < 0.98	**0.88**	1.04	**1.42**	1.46	1.97	**1.95**	**1.44**	1.45
0.9 ≦ S/K ratio < 0.95	**0.34**	0.53	**0.61**	0.62	1.40	1.40	**0.88**	0.90
S/K ratio < 0.9	**0.23**	0.79	**0.25**	0.30	1.28	**1.27**	**1.03**	1.66

prediction power of the model. The pricing errors of out-of-sample data are presented in Table 7.6. Here we find that CEV model can predict options on S&P 500 index futures more precisely than Black model. Based on the better performance in both in-sample and out-of-sample, we claim that CEV model can describe the options of S&P 500 index futures more precisely than Black model.

With regard to generate implied volatility surface to capture whole prediction of the future option market, the CEV model is the better choice than Black model because it not only captures the skewness and kurtosis effects of options on index futures but also has less computational costs than other jump-diffusion stochastic volatility models.

In sum, we show that CEV model performs better than Black model in aspects of either in-sample fitness or out-of-sample prediction. The setting of CEV model is more reasonable to depict the negative relationship between S&P 500 index future price and its volatilities. The elasticity of variance parameter in CEV model captures the level of this characteristic. The stable volatility parameter in CEV model in our empirical results implies that the instantaneous volatility of index future is mainly determined by current future price and the level of elasticity of variance parameter.

References

Bakshi, G, C Cao and Z Chen. 1997. "Empirical performance of alternative optionpricing models." *Journal of Finance*, 52, 2003–2049.

Beckers, S. 1980. "The constant elasticity of variance model and its implicationsfor option pricing." *Journal of Finance*, 35, 661–673.

Black, Fischer, and Myron Scholes. "The pricing of options and corporate liabilities." Journal of political economy 81.3 (1973): 637–654.

Chen, R., C.F. Lee. and H. Lee. 2009. "Empirical performance of the constant elasticity variance option pricing model." *Review of Pacific Basin Financial Markets and Policies*, 12(2), 177–217.

Cox, J. C. 1975. "Notes on option pricing I: constant elasticity of variance diffusions." *Working paper*, Stanford University.

Cox, J. C. and S. A. Ross. 1976. "The valuation of options for alternative stochastic processes." *Journal of Financial Economics* 3, 145–166.

Corrado, Charles J., and Thomas W. Miller Jr. "A note on a simple, accurate formula to compute implied standard deviations." Journal of Banking & Finance 20.3 (1996): 595–603.

Merton, Robert C. "Theory of rational option pricing." The Bell Journal of economics and management science (1973): 141–183.

Harvey, C. R. and R. E. Whaley. 1991. "S&P 100 index option volatility." *Journal of Finance*, 46, 1551–1561.

Harvey, C. R. and R. E. Whaley. 1992a. "Market volatility prediction and the efficiency of the S&P 100 index option market." *Journal of Financial Economics*, 31, 43–73.

Harvey, C. R. and R. E. Whaley. 1992b. "Dividends and S&P 100 index option valuation." *Journal of Futures Market*, 12, 123–137.

Jackwerth, JC and M Rubinstein. 2001. "Recovering stochastic processes fromoption prices." Working paper, London Business School.

Larguinho M., J.C.Dias, and C.A. Braumann. 2013. "On the computation of option prices and Greeks under the CEV model." *Quantitative Finance*, 13(6), 907–917.

Lee, C.F., T. Wu and R. Chen. 2004. "The constant elasticity of variance models:New evidence from S&P 500 index options." *Review of Pacific Basin Financial Markets and Policies*, 7(2), 173–190.

Lee, Cheng Few, and John C. Lee, eds. Handbook Of Financial Econometrics, Mathematics, Statistics, And Machine Learning (In 4 Volumes). World Scientific, 2020.

MacBeth, JD and LJ Merville. 1980. "Tests of the Black-Scholes and Cox Calloption valuation models." *Journal of Finance*, 35, 285–301.

Pun C. S. and H.Y. Wong. 2013. "CEV asymptotics of American options." *Journal of Mathematical Analysis and Applications*, 403 (2), 451–463.

Singh, V.K. and N. Ahmad. 2011. "Forecasting performance of constant elasticity of variance model: empirical evidence from India." *International Journal of Applied Economics and Finance*, 5, 87–96.

Check for updates

8 Greek Letters and Portfolio Insurance

8.1 Introduction

In Chapter 26, we have discussed how the call option value can be affected by the stock price per share, the exercise price per share, the contract period of the option, the risk-free rate, and the volatility of the stock return. In this chapter, we will mathematically analyze these kinds of relationships. Parts of these mathematical relationships are called "Greek letters" by finance professionals. Here, we specifically derive Greek letters for call (put) options on non-dividend stock and dividend-paying stock. Some examples will be provided to explain the applications of these Greek letters. Sections 8.1–8.5 discuss the formula, Excel function, and applications of delta, theta, gamma, vega, and rho, respectively. Section 8.6 derives the partial derivative of stock options with respect to their exercise prices. Section 8.7 describes the relationship between delta, theta, and gamma, and their implication in the delta-neutral portfolio. Section 8.8 presents a portfolio insurance example. Finally, in Sect. 8.9, we summarize and conclude this chapter.

8.2 Delta

The delta of an option, Δ, is defined as the rate of change of the option price respected to the rate of change of underlying asset price:

$$\Delta = \frac{\partial \Pi}{\partial S},$$

where Π is the option price and S is the underlying asset price. We next show the derivation of delta for various kinds of stock options.

8.2.1 Formula of Delta for Different Kinds of Stock Options

From Black and Scholes option pricing model, we know the price of call option on a non-dividend stock can be written as

$$C_t = S_t N(d_1) - Xe^{-r\tau} N(d_2),$$

and the price of put option on a non-dividend stock can be written as

$$P_t = Xe^{-r\tau} N(-d_2) - S_t N(-d_1),$$

where

$$d_1 = \frac{\ln\left(\frac{S_t}{X}\right) + \left(r + \frac{\sigma_s^2}{2}\right)\tau}{\sigma_s \sqrt{\tau}},$$

$$d_2 = \frac{\ln\left(\frac{S_t}{X}\right) + \left(r - \frac{\sigma_s^2}{2}\right)\tau}{\sigma_s \sqrt{\tau}} = d_1 - \sigma_s \sqrt{\tau},$$

$$\tau = T - t,$$

$N(\cdot)$ is the cumulative density function of normal distribution.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_8

$$N(d_1) = \int_{-\infty}^{d_1} f(u)du = \int_{-\infty}^{d_1} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} du$$

For a European call option on a non-dividend stock, delta can be shown as

$$\Delta = \mathrm{N}(\mathrm{d}_1)$$

For a European put option on a non-dividend stock, delta can be shown as

$$\Delta = \mathrm{N}(\mathrm{d}_1) - 1$$

If the underlying asset is a dividend-paying stock providing a dividend yield at rate q, Black and Scholes formulas for the prices of a European call option on a dividend-paying stock and a European put option on a dividend-paying stock are

$$\mathrm{C_t} = \mathrm{S_t e^{-q\tau} N(d_1)} - \mathrm{X e^{-r\tau} N(d_2)},$$

and

$$\mathrm{P_t} = \mathrm{X e^{-r\tau} N(-d_2)} - \mathrm{S_t e^{-q\tau} N(-d_1)},$$

where

$$\mathrm{d_1} = \frac{\ln\left(\frac{\mathrm{S_t}}{\mathrm{X}}\right) + \left(\mathrm{r} - \mathrm{q} + \frac{\sigma_s^2}{2}\right)\tau}{\sigma_s\sqrt{\tau}},$$

$$\mathrm{d_2} = \frac{\ln\left(\frac{\mathrm{S_t}}{\mathrm{X}}\right) + \left(\mathrm{r} - \mathrm{q} - \frac{\sigma_s^2}{2}\right)\tau}{\sigma_s\sqrt{\tau}} = \mathrm{d_1} - \sigma_s\sqrt{\tau},$$

For a European call option on a dividend-paying stock, delta can be shown as

$$\Delta = \mathrm{e^{-q\tau} N(d_1)}.$$

For a European put option on a dividend-paying stock, delta can be shown as

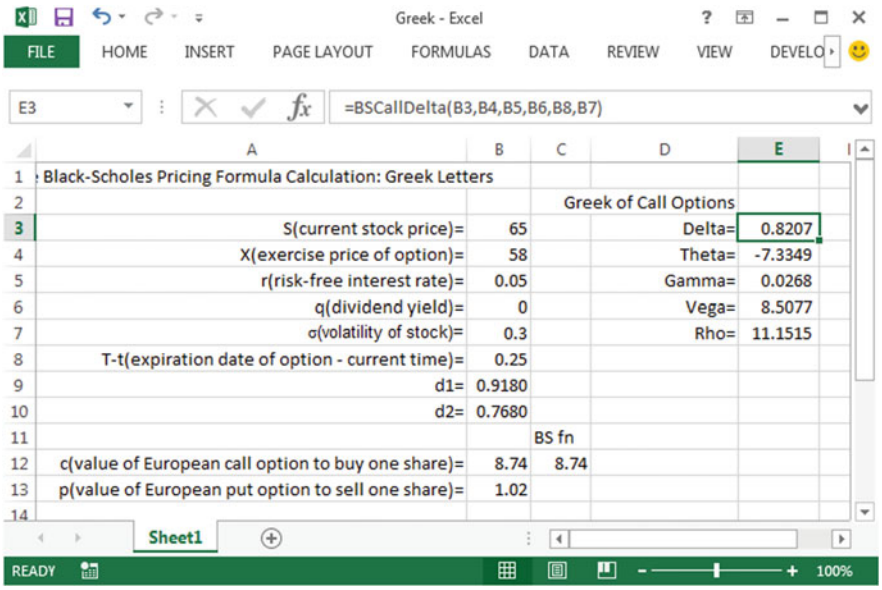
$$\Delta = \mathrm{e^{-q\tau}[N(d_1) - 1]}.$$

8.2.2 Excel Function of Delta for European Call Options

We can write a function to calculate the delta of call options. Below is the VBA function.

```
' BS Call Option Delta
Function BSCallDelta(S, X, r, q, T, sigma)
    Dim d1, Nd1
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    Nd1 = Application.NormSDist(d1)
    BSCallDelta = Exp(-q * T) * Nd1
End Function
```

With this function, we can use it in the Excel to calculate delta.



The formula for delta of a call option in Cell E3 is

$$= \text{BSCallDelta}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7})$$

8.2.3 Application of Delta

Figure 8.1 shows the relationship between the price of a call option and the price of its underlying asset. The delta of this call option is the slope of the line at the point of A corresponding to the current price of the underlying asset.

By calculating the delta ratio, a financial institution that sells option to a client can make a delta-neutral position to hedge the risk of changes in the underlying asset price. Suppose that the current stock price is \$100, the call option price on stock is \$10, and the current delta of the call option is 0.4. A financial institution sold 10 call options to its client, so the client has right to buy 1,000 shares at the time-to-maturity. To construct a delta hedge position, the financial institution should buy $0.4 \times 1{,}000 = 400$ shares of stock. If the stock price goes up to \$1, the option price will go up by \$0.40. In this situation, the financial institution has

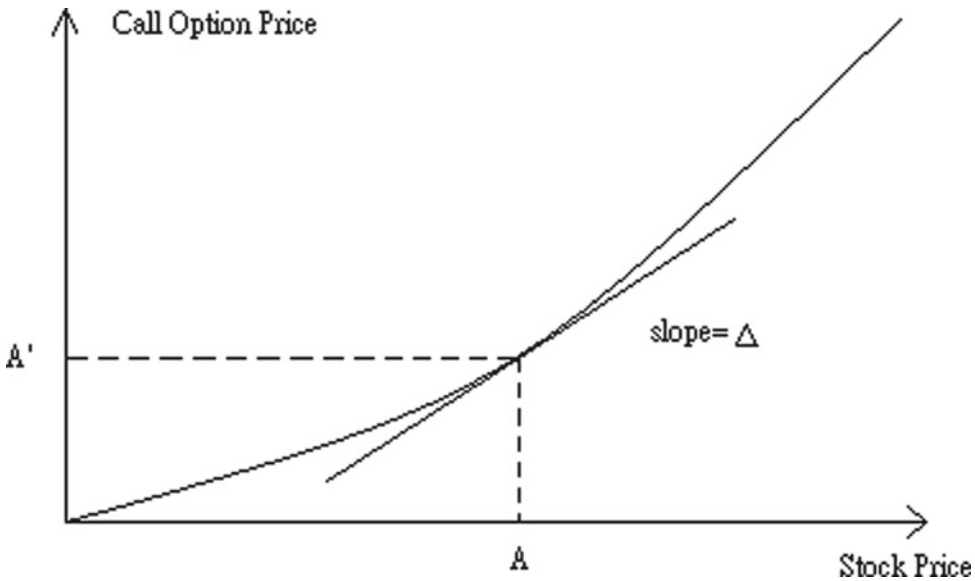


Fig. 8.1 The relationship between the price of a call option and the price of Its underlying asset

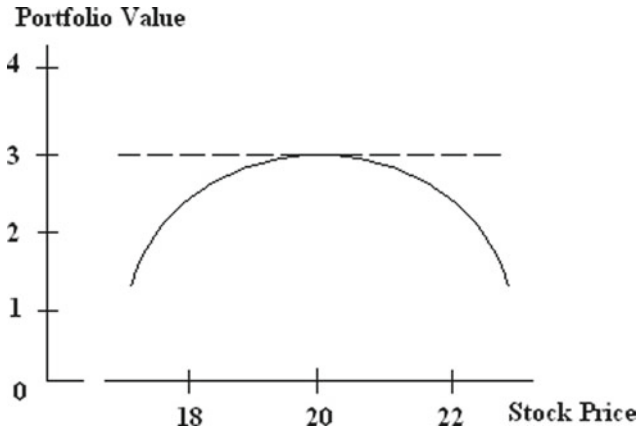


Fig. 8.2 Changes of Delta-Hedge

a \$400 (\$1 × 400 shares) gain in its stock position and a \$400 (\$0.40 × 1,000 shares) loss in its option position. The total payoff of the financial institution is zero. On the other hand, if the stock price goes down by \$1, the option price will go down by \$0.40. The total payoff of the financial institution is also zero.

However, the relationship between option price and stock price is not linear, so delta changes over different stock prices. If an investor wants to remain his portfolio delta-neutral, he should adjust his hedged ratio periodically. The more frequent adjustments he does, the better delta-hedging he gets.

Figure 8.2 exhibits the change in delta affecting the delta-hedges. If the underlying stock has a price equal to \$20, then the investor who uses only delta as risk measure will consider that his or her portfolio has no risk. However, as the underlying stock prices change, either up or down, the delta changes as well and thus he or she will have to use different delta-hedging. Delta measure can be combined with other risk measures to yield better risk measurement. We will discuss it further in the following sections.

8.3 Theta

The theta of an option, Θ, is defined as the rate of change of the option price with respect to the passage of time:

$$\Theta = \frac{\partial \Pi}{\partial t},$$

where Π is the option price and t is the passage of time.

If $\tau = T - t$, theta (Θ) can also be defined as minus one timing the rate of change of the option price is respected to the time–to-maturity. The derivation of such transformation is easy and straightforward:

$$\Theta = \frac{\partial \Pi}{\partial t} = \frac{\partial \Pi}{\partial \tau}\frac{\partial \tau}{\partial t} = (-1)\frac{\partial \Pi}{\partial \tau},$$

where $\tau = T - t$ is the time-to-maturity. For the derivation of theta for various kinds of stock options, we use the definition of negative differential on time-to-maturity.

8.3.1 Formula of Theta for Different Kinds of Stock Options

For a European call option on a non-dividend stock, theta can be written as

$$\Theta = -\frac{S_t \sigma_s}{2\sqrt{\tau}} \cdot N'(d_1) - rX \cdot e^{-r\tau} N(d_2).$$

For a European put option on a non-dividend stock, theta can be shown as

$$\Theta = -\frac{S_t \sigma_s}{2\sqrt{\tau}} \cdot N'(d_1) + rX \cdot e^{-r\tau} N(-d_2).$$

For a European call option on a dividend-paying stock, theta can be shown as

$$\Theta = q \cdot S_t e^{-q\tau} N(d_1) - \frac{S_t e^{-q\tau} \sigma_s}{2\sqrt{\tau}} \cdot N'(d_1) - rX \cdot e^{-r\tau} N(d_2).$$

For a European put option on a dividend-paying stock, theta can be shown as

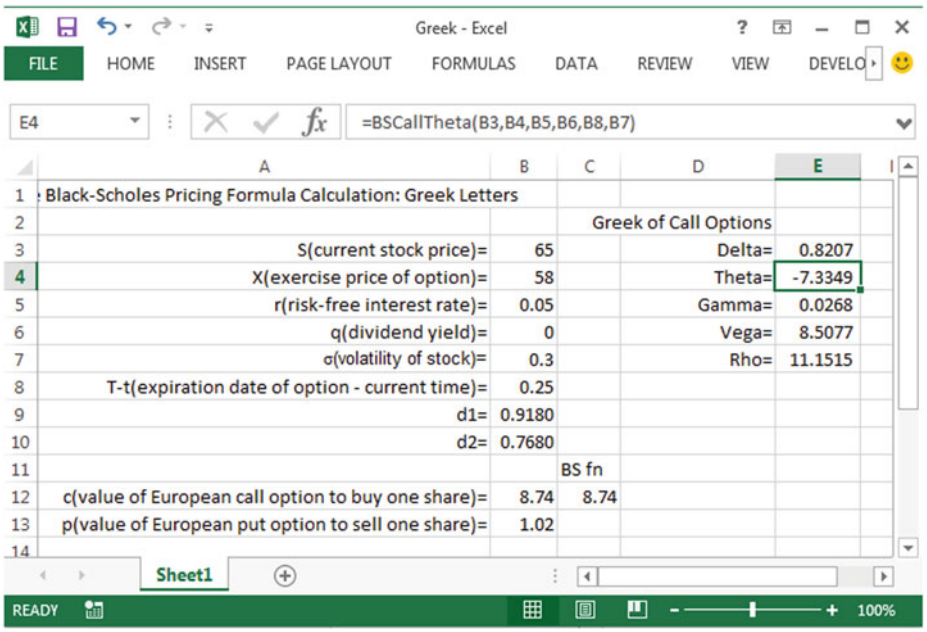
$$\Theta = rX \cdot e^{-r\tau} N(-d_2) - qS_t e^{-q\tau} N(-d_1) - \frac{S_t e^{-q\tau} \sigma_s}{2\sqrt{\tau}} \cdot N'(d_1).$$

8.3.2 Excel Function of Theta of the European Call Option

We also can write a function to calculate theta. The VBA function can be written as.

```
' BS Call Option Theta
Function BSCallTheta(S, X, r, q, T, sigma)
    Dim d1, d2, Nd1, Nd2, Ndash1
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    d2 = d1 - sigma * Sqr(T)
    Nd1 = Application.NormSDist(d1)
    Nd2 = Application.NormSDist(d2)
    Ndash1 = (1 / Sqr(2 * Application.Pi)) * Exp(-d1 ^ 2 / 2)
    BSCallTheta = q * Exp(-q * T) * S * Nd1 - S * Ndash1 * sigma * Exp(-
q * T) / (2 * Sqr(T)) - r * Exp(-r * T) * X * Nd2
End Function
```

Using this function, we can value the theta of a call option.



The function of theta for a European call option in Cell E4 is

$$= \text{BSCallTheta}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7})$$

8.3.3 Application of Theta

The value of option is the combination of time value and stock value. When time passes, the time value of the option decreases. Thus, the rate of change of the option price with respect to the passage of time, theta, is usually negative.

Because the passage of time on an option is not uncertain, we do not need to make a theta hedge portfolio against the effect of the passage of time. However, we still regard theta as a useful parameter, because it is a proxy of gamma in the delta-neutral portfolio. For the specific detail, we will discuss in the following sections.

8.4 Gamma

The gamma of an option, Γ, is defined as the rate of change of delta respective to the rate of change of underlying asset price:

$$\Gamma = \frac{\partial \Delta}{\partial S} = \frac{\partial^2 \Pi}{\partial S^2},$$

where Π is the option price and S is the underlying asset price.

Because the option is not linearly dependent on its underlying asset, delta-neutral hedge strategy is useful only when the movement of underlying asset price is small. Once the underlying asset price moves wider, gamma-neutral hedge is necessary. We next show the derivation of gamma for various kinds of stock options.

8.4.1 Formula of Gamma for Different Kinds of Stock Options

For a European call option on a non-dividend stock, gamma can be shown as

$$\Gamma = \frac{1}{S_t \sigma_s \sqrt{\tau}} N'(d_1).$$

For a European put option on a non-dividend stock, gamma can be shown as

$$\Gamma = \frac{1}{S_t \sigma_s \sqrt{\tau}} N'(d_1).$$

For a European call option on a dividend-paying stock, gamma can be shown as

$$\Gamma = \frac{e^{q\tau}}{S_t \sigma_s \sqrt{\tau}} N'(d_1).$$

For a European put option on a dividend-paying stock, gamma can be shown as

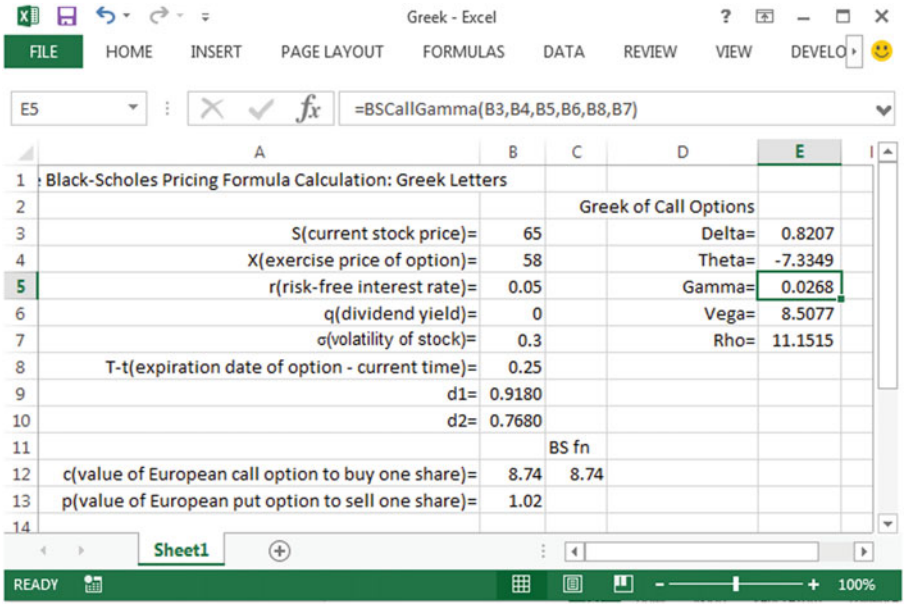
$$\Gamma = \frac{e^{-q\tau}}{S_t \sigma_s \sqrt{\tau}} N'(d_1).$$

8.4.2 Excel Function of Gamma for European Call Options

In addition, we can write a code to price gamma of a call option. Here is the VBA function to calculate gamma.

```
'   BS Call Option Gamma
Function BSCallGamma(S, X, r, q, T, sigma)
    Dim d1, Ndash1
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    Ndash1 = (1 / Sqr(2 * Application.Pi)) * Exp(-d1 ^ 2 / 2)
    BSCallGamma = Exp(-q * T) * Ndash1 / (S * sigma * Sqr(T))
End Function
```

We can use the function in Excel spreadsheet to calculate gamma.



	A	B	C	D	E
1	Black-Scholes Pricing Formula Calculation: Greek Letters				
2				Greek of Call Options	
3	S(current stock price)=	65		Delta=	0.8207
4	X(exercise price of option)=	58		Theta=	-7.3349
5	r(risk-free interest rate)=	0.05		Gamma=	0.0268
6	q(dividend yield)=	0		Vega=	8.5077
7	σ(volatility of stock)=	0.3		Rho=	11.1515
8	T-t(expiration date of option - current time)=	0.25			
9	d1=	0.9180			
10	d2=	0.7680			
11			BS fn		
12	c(value of European call option to buy one share)=	8.74	8.74		
13	p(value of European put option to sell one share)=	1.02			
14					

he function of gamma for a European call option in Cell E5 is

$$= \text{BSCallGamma}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}).$$

8.4.3 Application of Gamma

One can use delta and gamma together to calculate the changes in the option due to changes in the underlying stock price. This change can be approximated by the following relations:

$$\text{change in option value} \approx \Delta \times \text{change in stock price} + \frac{1}{2} \times \Gamma \times (\text{change in stock price})^2.$$

From the above relation, one can observe that the gamma makes the correction for the fact that the option value is not a linear function of underlying stock price. This approximation comes from the Taylor series expansion near the initial stock price. If we let V be the option value, S be the stock price, and S_0 be the initial stock price, then the Taylor series expansion around S_0 yields the following:

$$\begin{aligned} V(S) &\approx V(S_0) + \frac{\partial V(S_0)}{\partial S}(S - S_0) + \frac{1}{2!}\frac{\partial^2 V(S_0)}{\partial S^2}(S - S_0)^2 \\ &\quad + \cdots + \frac{1}{2!}\frac{\partial^n V(S_0)}{\partial S^n}(S - S_0)^n \\ &\approx V(S_0) + \frac{\partial V(S_0)}{\partial S}(S - S_0) + \frac{1}{2!}\frac{\partial^2 V(S_0)}{\partial S^2}(S - S_0)^2 + o(S) \end{aligned}$$

If we only consider the first three terms, the approximation is then.

$$\begin{aligned} V(S) - V(S_0) &\approx \frac{\partial V(S_0)}{\partial S}(S - S_0) + \frac{1}{2!}\frac{\partial^2 V(S_0)}{\partial S^2}(S - S_0)^2 \\ &\approx \Delta(S - S_0) + \frac{1}{2}\Gamma(S - S_0)^2 \end{aligned}$$

For example, if a portfolio of options has a delta equal to \$10,000 and a gamma equal to \$5,000, the change in the portfolio value if the stock price drop to \$34 from \$35 is approximately

$$\begin{aligned} \text{change in portfolio value} &\approx (\$10000) \times (\$\ 34 - \$\ 35) + \frac{1}{2} \\ &\quad \times (\$5000) \times (\$\ 34 - \$\ 35)^2 \\ &\approx -\$7500 \end{aligned}$$

The above analysis can also be applied to measure the price sensitivity of interest rate-related assets or portfolio to interest rate changes. Here, we introduce *Modified Duration* and *Convexity* as risk measure corresponding to the above delta and gamma. Modified duration measures the percentage change in asset or portfolio value resulting from a percentage change in interest rate.

$$\begin{aligned} \text{Modified Duration} &= \left(\frac{\text{Change in price}}{\text{Change in interest rate}}\right) \Big/ \text{Price} \\ &= -\Delta/P \end{aligned}$$

Using the modified duration.

$$\begin{aligned}\text{Change in Portfolio Value} &= \Delta \times \text{Change in interest rate} \\ &= (-\textit{Duration} \times \text{P}) \\ &\quad \times \text{Change in interest rate},\end{aligned}$$

we can calculate the value changes of the portfolio. The above relation corresponds to the previous discussion of delta measure. We want to know how the price of the portfolio changes given a change in interest rate. Similar to delta, modified duration only shows the first-order approximation of the changes in value. In order to account for the nonlinear relation between the interest rate and portfolio value, we need a second-order approximation similar to the gamma measure before, this is then the convexity measure. Convexity is the interest rate gamma divided by price as given below:

$$\textit{Convexity} = \Gamma/\text{P},$$

and this measure captures the nonlinear part of the price changes due to interest rate changes. Using the modified duration and convexity together allows us to develop first- as well as second-order approximation of the price changes similar to the previous discussion.

$$\begin{aligned}\text{Change in Portfolio Value} \approx &-\textit{Duration} \times \text{P} \times (\text{change in rate}) \\ &+ \frac{1}{2} \times \textit{Convexity} \times \text{P} \times (\text{change in rate})^2\end{aligned}$$

As a result, (−Duration × P) and (Convexity × P) act like the delta and gamma measures, respectively, in the previous discussion. This shows that these Greeks can also be applied in measuring risk in interest rate-related assets or portfolio.

Next, we discuss how to make a portfolio gamma-neutral. Suppose the gamma of a delta-neutral portfolio is Γ, the gamma of the option in this portfolio is Γ_o, and ω_o is the number of options added to the delta-neutral portfolio. Then, the gamma of this new portfolio is

$$\omega_o \Gamma_o + \Gamma.$$

To make a gamma-neutral portfolio, we should trade $\omega_o^* = -\Gamma/\Gamma_o$ options. Because the position of option changes, the new portfolio is not delta-neutral. We should change the position of the underlying asset to maintain delta-neutral.

For example, the delta and gamma of a particular call option are 0.7 and 1.2. A delta-neutral portfolio has a gamma of − 2,400. To make a delta-neutral and gamma-neutral portfolio, we should add a long position of 2,400/1.2 = 2,000 shares and a short position of 2,000 × 0.7 = 1,400 shares in the original portfolio.

8.5 Vega

The vega of an option, v, is defined as the rate of change of the option price respective to the volatility of the underlying asset:

$$\text{v} = \frac{\partial \Pi}{\partial \sigma}$$

where Π is the option price and σ is the volatility of the stock price. We next show the derivation of vega for various kinds of stock options.

8.5.1 Formula of Vega for Different Kinds of Stock Options

For a European call option on a non-dividend stock, vega can be shown as

$$\text{v} = \text{S}_\text{t}\sqrt{\tau} \cdot \text{N}'(\text{d}_1).$$

For a European put option on a non-dividend stock, vega can be shown as

$$\text{v} = \text{S}_\text{t}\sqrt{\tau} \cdot \text{N}'(\text{d}_1).$$

For a European call option on a dividend-paying stock, vega can be shown as

$$\text{v} = \text{S}_\text{t}\text{e}^{-\text{q}\tau}\sqrt{\tau} \cdot \text{N}'(\text{d}_1).$$

For a European put option on a dividend-paying stock, vega can be shown as

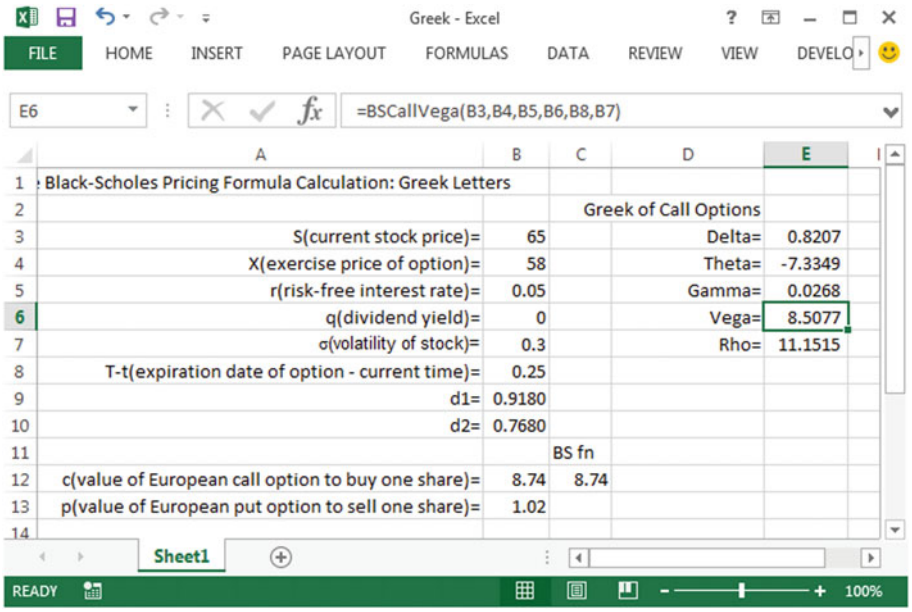
$$\text{v} = \text{S}_\text{t}\text{e}^{-\text{q}\tau}\sqrt{\tau} \cdot \text{N}'(\text{d}_1).$$

8.5.2 Excel Function of Vega for European Call Options

We can write a function to calculate vega. Below is the VBA function of vega for European call options.

```
'   BS Call Option Vega
Function BSCallVega(S, X, r, q, T, sigma)
    Dim d1, Ndash1
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    Ndash1 = (1 / Sqr(2 * Application.Pi)) * Exp(-d1 ^ 2 / 2)
    BSCallVega = Exp(-q * T) * S * Sqr(T) * Ndash1
End Function
```

Using this function, we can calculate vega for a European call option in the Excel spreadsheet.



The function of vega for a European call option in Cell E5 is

$$= \text{BSCallVega}(B3, B4, B5, B6, B8, B7)$$

8.5.3 Application of Vega

Suppose a delta-neutral and gamma-neutral portfolio has a vega equal to v and the vega of a particular option is v_o. Similar to gamma, we can add a position of $-v/v_o$ in option to make a vega-neutral portfolio. To maintain delta-neutral, we should change the underlying asset position. However, when we change the option position, the new portfolio is not gamma-neutral. Generally, a portfolio with one option cannot maintain its gamma-neutral and vega-neutral at the same time. If we want a portfolio to be both gamma-neutral and vega-neutral, we should include at least two kinds of options on the same underlying asset in our portfolio.

For example, a delta-neutral and gamma-neutral portfolio contains option A, option B, and underlying asset. The gamma and vega of this portfolio are − 3,200 and − 2,500, respectively. Option A has a delta of 0.3, gamma of 1.2, and vega of 1.5. Option B has a delta of 0.4, gamma of 1.6, and vega of 0.8. The new portfolio will be both gamma-neutral and vega-neutral when adding ω_A of option A and ω_B of option B into the original portfolio.

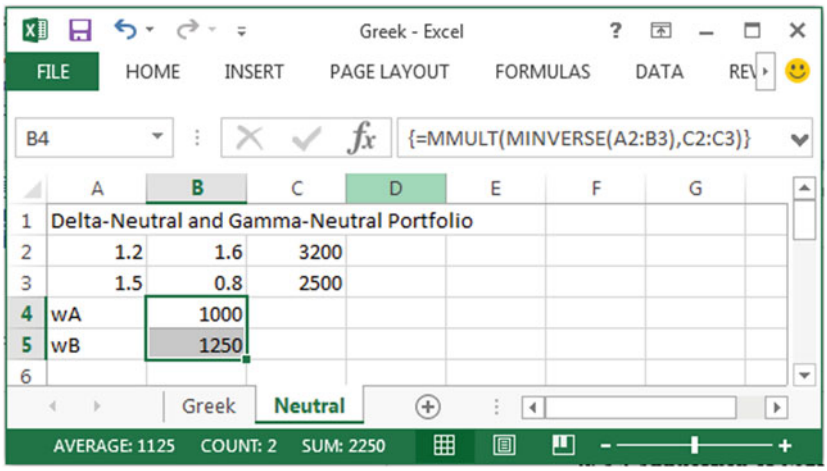
$$\textit{Gamma}\,\text{Neutral:}\ -3200 + 1.2\omega_A + 1.6\omega_B = 0.$$

$$\textit{Vega}\,\text{Neutral:}\ -2500 + 1.5\,\omega_A + 0.8\omega_B = 0.$$

From the two equations shown above, we can get the solution that ω_A = 1000 and ω_B = 1250. The delta of new portfolio is 1000 × 0.3 + 1250 × 0.4 = 800. To maintain

delta-neutral, we need to short 800 shares of the underlying asset.

We can use the Excel matrix function to solve these linear equations.



The function in Cell B4:B5 is

$$= \text{MMULT}(\text{MINVERSE}(\text{A2} : \text{B3}), \text{C2} : \text{C3})$$

Because this is matrix function, we need to use [ctrl] + [shift] + [enter] to get our result.

8.6 Rho

The rho of an option is defined as the rate of change of the option price respected to the interest rate:

$$\text{rho} = \frac{\partial \Pi}{\partial \text{r}},$$

where Π is the option price and r is the interest rate. The rho for an ordinary stock call option should be positive because higher interest rate reduces the present value of the strike price which in turn increases the value of the call option. Similarly, the rho of an ordinary put option should be negative by the same reasoning. We next show the derivation of rho for various kinds of stock options.

8.6.1 Formula of Rho for Different Kinds of Stock Options

For a European call option on a non-dividend stock, rho can be shown as

$$\text{rho} = \text{X}\tau \cdot \text{e}^{-\text{r}\tau}\text{N}(\text{d}_2).$$

For a European put option on a non-dividend stock, rho can be shown as

$$\text{rho} = -\text{X}\tau \cdot \text{e}^{-\text{r}\tau}\text{N}(-\text{d}_2).$$

For a European call option on a dividend-paying stock, rho can be shown as

$$\text{rho} = \text{X}\tau \cdot \text{e}^{-\text{r}\tau}\text{N}(\text{d}_2).$$

For a European put option on a dividend-paying stock, rho can be shown as

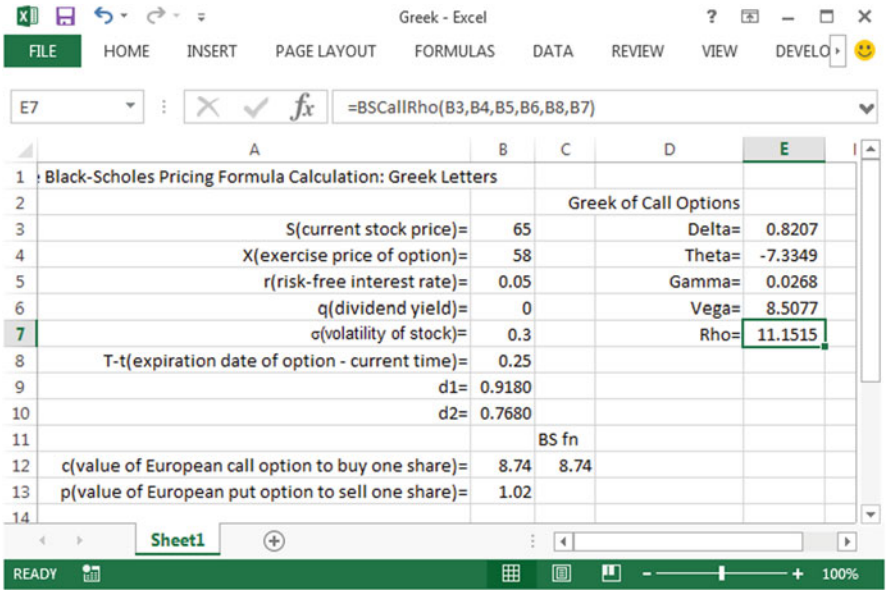
$$\text{rho} = -\text{X}\tau \cdot \text{e}^{-\text{r}\tau}\text{N}(-\text{d}_2).$$

8.6.2 Excel Function of Rho for European Call Options

We can write a function to calculate rho. Here is the VBA function to calculate rho for European call options.

```
'   BS Call Option Rho
Function BSCallRho(S, X, r, q, T, sigma)
    Dim d1, d2, Nd2
    d1 = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma * Sqr(T))
    d2 = d1 - sigma * Sqr(T)
    Nd2 = Application.NormSDist(d2)
    BSCallRho = T * Exp(-r * T) * X * Nd2
End Function
```

Then we can use this function to calculate rho in the Excel worksheet.



The function of rho in Cell E7 is

$$= \text{BSCallRho}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7})$$

8.6.3 Application of Rho

Assume that an investor would like to see how interest rate changes affect the value of a 3-month European call option she holds with the following information. The current stock price is \$65 and the strike price is \$58. The interest rate and the volatility of the stock are 5% and 30% per annum, respectively. The rho of this European call can be calculated as follows:

$$Rho_{put} = X\tau e^{-r\tau}N(d_2) = 11.1515$$

This calculation indicates that given a 1% change increase in interest rate, say from 5 to 6%, the value of this European call option will decrease by 0.111515 (0.01×11.1515). This simple example can be further applied to stocks that pay dividends using the derivation results shown previously.

8.7 Formula of Sensitivity for Stock Options with Respect to Exercise Price

For a European call option on a non-dividend stock, the sensitivity can be shown as

$$\frac{\partial C_t}{\partial X} = -e^{-r\tau}N(d_2).$$

For a European put option on a non-dividend stock, the sensitivity can be shown as

$$\frac{\partial P_t}{\partial X} = e^{-r\tau}N(-d_2)$$

For a European call option on a dividend-paying stock, the sensitivity can be shown as

$$\frac{\partial C_t}{\partial X} = -e^{-r\tau}N(d_2).$$

For a European put option on a dividend-paying stock, the sensitivity can be shown as

$$\frac{\partial P_t}{\partial X} = e^{-r\tau}N(-d_2).$$

8.8 Relationship Between Delta, Theta, and Gamma

So far, the discussion has introduced the derivation and application of individual Greeks and how they can be applied in portfolio management. In practice, the interaction or trade-off between these parameters is of concern as well. For example, recall the Black–Scholes–Merton differential equation with non-dividend paying stock can be written as

$$\frac{\partial \Pi}{\partial t} + rS\frac{\partial \Pi}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 \Pi}{\partial S^2} = r\Pi,$$

where Π is the value of the derivative security contingent on stock price, S is the price of stock, r is the risk-free rate, σ is the volatility of the stock price, and t is the time to expiration of the derivative. Given the earlier derivation, we can rewrite the Black–Scholes partial differential equation (PDE) as

$$\Theta + rS\Delta + \frac{1}{2}\sigma^2 S^2 \Gamma = r\Pi.$$

This relation gives us the trade-off between delta, gamma, and theta. For example, suppose there are two delta-neutral ($\Delta = 0$) portfolios, one with positive gamma ($\Gamma > 0$) and the other one with negative gamma ($\Gamma < 0$), and they both have a value of \$1 ($\Pi = 1$). The trade-off can be written as

$$\Theta + \frac{1}{2}\sigma^2 S^2 \Gamma = r.$$

For the first portfolio, if gamma is positive and large, then theta is negative and large. When gamma is positive, changes in stock prices result in higher value of the option. This means that when there is no change in stock prices, the value of the option declines as we approach the expiration date. As a result, the theta is negative. On the other hand, when gamma is negative and large, changes in stock prices result in lower option value. This means that when there is no stock price change, the value of the option increases as we approach the expiration and theta is positive. This gives us a trade-off between gamma and theta and they can be used as proxy for each other in a delta-neutral portfolio.

8.9 Portfolio Insurance

Portfolio insurance is a strategy of hedging a portfolio of stocks against the market risk by using a synthetic put option. What is a synthetic put option? A synthetic put option is like to buy a put option to hedge a portfolio. That is a protective put strategy. Although this strategy uses short stocks or futures to construct a delta which is like to buy a put option, the risk of this strategy is not the same as to buy a put option.

Consider two strategies. The first one is long 1 index portfolio and long 1 put, then the delta in this strategy is 1 + Δp, where Δp is the delta of put and the value is negative. The second one is long 1 index portfolio, short $-\Delta$p amount of index, and invest the money that short index to riskless asset, then the delta of this strategy is 1 $-(-\Delta p^*1) = 1 + \Delta p$, which is equal to the first strategy.

The second strategy is so-called portfolio insurance. The dynamic adjustment in this strategy is like below. As the value of the index portfolio increase, the Δp become less negative and some of the index portfolios are repurchased. As the value of the index portfolio decreases, Δp becomes more negative and more of the index portfolio have to be sold.

However, the portfolio insurance strategy did not work well on October 19, 1987. That day stock market declines very quickly. The managers using portfolio insurance strategy should short index portfolio. This action increased the pace of decline in the stock market. Therefore, synthetic put cannot create the same payoff like buying a put option. There is no effect of insurance in the crash market.

8.10 Summary

In this chapter, we have shown the partial derivatives of stock option with respect to five variables. Delta (Δ), the rate of change of option price to change in the price of underlying asset, is first derived. After delta is obtained, gamma (Γ) can be derived as the rate of change of delta with respect to the underlying asset price. Another two risk measures are theta (Θ) and rho (ρ); they measure the change in option value with respect to passing time and interest rate, respectively. Finally, one can also measure the change in option value with respect to the volatility of the underlying asset and this gives us the vega (v). The applications of these Greek letters in the portfolio management have also been discussed. In addition, we use the Black and Scholes PDE to show the relationship between these risk measures. In sum, risk management is one of the important topics in finance for both academics and practitioners. Given the recent credit crisis, one can observe that it is crucial to properly measure the risk related to the even more complicated financial assets. The comparative static analysis of option pricing models gives an introduction to the portfolio risk management.

References

Bjork, T. Arbitrage Theory in Continuous Time. New York: Oxford University Press, 1998.

Boyle, P. P. and D. Emanuel. "Discretely Adjusted Option Hedges." Journal of Financial Economics, v. 8(3) (1980), pp. 259–282.

Duffie, D. Dynamic Asset Pricing Theory. Princeton, NJ: Princeton University Press, 2001.

Fabozzi, F. J. Fixed Income Analysis, 2nd Edn. New York: Wiley, 2007.

Figlewski, S. "Options Arbitrage in Imperfect Markets." Journal of Finance, v. 44(5) (1989), pp. 1289–1311.

Galai, D. "The Components of the Return from Hedging Options against Stocks." Journal of Business, v. 56(1) (1983), pp. 45–54.

Hull, J. Options, Futures, and Other Derivatives, 8th Edn. Upper Saddle River, NJ: Pearson, 2011.

Hull, J. and A. White. "Hedging the Risks from Writing Foreign Currency Options." Journal of International Money and Finance, v. 6(2) (1987), pp. 131–152.

Karatzas, I. and S. E. Shreve. Brownian Motion and Stochastic Calculus. Berlin: Springer, 2000.

Klebaner, F. C. Introduction to Stochastic Calculus with Applications. London: Imperial College Press, 2005.

McDonald, R. L. Derivatives Markets, 2nd Edn. Boston, MA: Addison-Wesley, 2005.

Shreve, S. E. Stochastic Calculus for Finance II: Continuous Time Model. New York: Springer, 2004.

Tuckman, B. Fixed Income Securities: Tools for Today's Markets, 2nd Edn. New York: Wiley, 2002.

Check for updates

Portfolio Analysis and Option Strategies

9

9.1 Introduction

The main purposes of this chapter are to show how excel programs can be used to perform portfolio selection decisions and to construct option strategies. In Sect. 9.2, we demonstrate how Microsoft Excel can be used to inverse the matrix. In Sect. 9.3, we discuss how Excel Programs can be used to estimate the Markowitz portfolio models. In Sect. 9.4, we discuss option strategies. Finally, in Sect. 9.5, we summarize the chapter.

9.2 Three Alternative Methods to Solve the Simultaneous Equation

In this section, we discuss four alternative methods to solve the system of linear equations including 9.2.1 Substitution Method, 9.2.2 Cramer's Rule, 9.2.3 Matrix Method, and 9.2.4 Excel Matrix Inversion and Multiplication.

9.2.1 Substitution Method (Reference: Wikipedia)

The simplest method for solving a system of linear equations is to repeatedly eliminate variables. This method can be described as follows:

1. In the first equation, solve for one of the variables in terms of the others.
2. Substitute this expression into the remaining equations. This yields a system of equations with one fewer equation and one fewer unknown.
3. Continue until you have reduced the system to a single linear equation.
4. Solve this equation and then back-substitute until the entire solution is found.

For example, consider the following system:

$$\begin{aligned} x+3y-2z &= 5 \\ 3x+5y+6z &= 7 \\ 2x+4y+3z &= 8 \end{aligned}$$

Solving the first equation for x gives x = 5 + 2z − 3y, and plugging this into the second and third equations yields

$$\begin{aligned} -4y+12z &= -8 \\ -2y+7z &= -2 \end{aligned}$$

Solving the first of these equations for y yields y = 2 + 3z, and plugging this into the second equation yields z = 2. We now have

$$\begin{aligned} x &= 5+2z-3y \\ y &= 2+3z \\ z &= 2 \end{aligned}$$

Substituting z = 2 into the second equation gives y = 8, and substituting z = 2 and y = 8 into the first equation yields x = −15. Therefore, the solution set is the single point (x, y, z) = (−15, 8, 2).

9.2.2 Cramer's Rule

Explicit formulas for small systems (Reference: Wikipedia).

Consider the linear system $\begin{cases} a_1x+b_1y=c_1 \\ a_2x+b_2y=c_2 \end{cases}$ which in matrix format is $\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$.

Assume $a_1b_2 - b_1a_2$ is nonzero. Then, x and y can be found with Cramer's rule as

$$x = \begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix} \Big/ \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = \frac{c_1b_2 - b_1c_2}{a_1b_2 - b_1a_2}$$

and

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_9

$$y = \begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix} \Big/ \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = \frac{a_1c_2 - c_1a_2}{a_1b_2 - b_1a_2}.$$

The rules for 3×3 matrices are similar. Given $\begin{cases} a_1x + b_1y + c_1z = d_1 \\ a_2x + b_2y + c_2z = d_2 \\ a_3x + b_3y + c_3z = d_3 \end{cases}$ which in matrix format is

$$\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}.$$

Then the values of x, y, and z can be found as follows:

$$x = \frac{\begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}, \quad y = \frac{\begin{vmatrix} a_1 & d_1 & c_1 \\ a_2 & d_2 & c_2 \\ a_3 & d_3 & c_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}, \text{ and } \quad z = \frac{\begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}.$$

And then you need to use determinant calculation, and the calculation for the determinant is as follows:

For example, for 3×3 matrices, the determinant of a 3×3 matrix is defined by

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix}$$
$$= a(ei - fh) - b(di - fg) + c(dh - eg)$$
$$= aei + bfg + cdh - ceg - bdi - afh.$$

We use the same example as we did in the first method:

$$\mathrm{x} = \frac{\begin{bmatrix} 5 & 3 & -2 \\ 7 & 5 & 6 \\ 8 & 4 & 3 \end{bmatrix}}{\begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix}}, y = \frac{\begin{bmatrix} 1 & 5 & -2 \\ 3 & 7 & 6 \\ 2 & 8 & 3 \end{bmatrix}}{\begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix}}, z = \frac{\begin{bmatrix} 1 & 3 & 5 \\ 3 & 5 & 7 \\ 2 & 4 & 8 \end{bmatrix}}{\begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix}}$$

$$\mathrm{x} = \frac{5*5*3+3*6*8+(-2)*7*4-(-2)*5*8-3*7*3-5*6*4}{1*5*3+3*6*2+(-2)*3*4-(-2)*5*2-3*3*3-1*6*4}$$
$$= \frac{75+144-28+80-63-120}{15+36-24+20-27-24} = \frac{60}{-4} = -15$$

$$\mathrm{y} = \frac{1*7*3+5*6*2+(-2)*3*8-(-2)*7*2-5*3*3-1*6*8}{1*5*3+3*6*2+(-2)*3*4-(-2)*5*2-3*3*3-1*6*4}$$
$$= \frac{21+60-48+28-45-48}{15+36-24+20-27-24} = \frac{-32}{-4} = 8$$

$$\mathrm{z} = \frac{1*5*8+3*7*2+5*3*4-5*5*2-3*3*8-1*7*4}{1*5*3+3*6*2+(-2)*3*4-(-2)*5*2-3*3*3-1*6*4}$$
$$= \frac{40+42+60-50-72-28}{15+36-24+20-27-24} = \frac{-8}{-4} = 2$$

9.2.3 Matrix Method

Using the example in the last two sections above, we can derive the following matrix equation:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix}^{-1} * \begin{bmatrix} 5 \\ 7 \\ 8 \end{bmatrix}$$

The inversion of matrix A is by the definition

$$A^{-1} = \frac{1}{\det A} * (AdjA);$$

The Adjoint A is defined by the transpose of the cofactor matrix. First we need to calculate the cofactor matrix of A. Suppose the cofactor matrix is:

$$\text{cofactor matrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix},$$

$$A_{11} = \begin{bmatrix} 5 & 6 \\ 4 & 3 \end{bmatrix} = -9, A_{12} = -\begin{bmatrix} 3 & 6 \\ 2 & 3 \end{bmatrix} = 3, A_{13} = \begin{bmatrix} 3 & 5 \\ 2 & 4 \end{bmatrix} = 2,$$

$$A_{21} = -\begin{bmatrix} 3 & -2 \\ 4 & 3 \end{bmatrix} = -17, A_{22} = \begin{bmatrix} 1 & -2 \\ 2 & 3 \end{bmatrix} = 7,$$
$$A_{23} = -\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} = 2,$$

$$A_{31} = \begin{bmatrix} 3 & -2 \\ 5 & 6 \end{bmatrix} = 28, A_{32} = -\begin{bmatrix} 1 & -2 \\ 3 & 6 \end{bmatrix} = -12,$$
$$A_{33} = \begin{bmatrix} 1 & 3 \\ 3 & 5 \end{bmatrix} = -4,$$

Therefore,

$$\text{Cofactor matrix} = \begin{bmatrix} -9 & 3 & 2 \\ -17 & 7 & 2 \\ 28 & -12 & -4 \end{bmatrix},$$

Then, we can get Adjoint A:

$$\text{Adj A} = \begin{bmatrix} -9 & -17 & 28 \\ 3 & 7 & -12 \\ 2 & 2 & -4 \end{bmatrix},$$

The determinant of A we have calculated in Cramer's rule:

$$\text{Det A} = \begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix} = -4,$$

$$A^{-1} = \frac{1}{(-4)} * \begin{bmatrix} -9 & -17 & 28 \\ 3 & 7 & -12 \\ 2 & 2 & -4 \end{bmatrix} = \begin{bmatrix} \frac{9}{4} & \frac{17}{4} & -\frac{28}{4} \\ -\frac{3}{4} & -\frac{7}{4} & 3 \\ -\frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix},$$

Therefore,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{9}{4} & \frac{17}{4} & -\frac{28}{4} \\ -\frac{3}{4} & -\frac{7}{4} & 3 \\ -\frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix} * \begin{bmatrix} 5 \\ 7 \\ 8 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{9}{4} * 5 + \frac{17}{4} * 7 + \left(-\frac{28}{4}\right) * 8 \\ \left(-\frac{3}{4}\right) * 5 + \left(-\frac{7}{4}\right) * 7 + 3 * 8 \\ \left(-\frac{1}{2}\right) * 5 + \left(-\frac{1}{2}\right) * 7 + 1 * 8 \end{bmatrix} = \begin{bmatrix} -15 \\ 8 \\ 2 \end{bmatrix}.$$

9.2.4 Excel Matrix Inversion and Multiplication

1. Using minverse () function to get the A inverse. Type "Ctrl + Shift + Enter" together you will get the inverse of A.
2. Using mmult () function to do the matrix multiplication and type "Ctrl + Shift + Enter" together, you will get the answers for x, y, and z.

Excel matrix inversion and multiplication method discussed in this section is identical to the method discussed in a previous section.

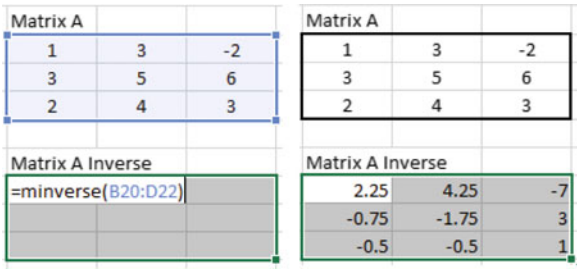
9.3 Markowitz Model for Portfolio Selection

The Markowitz model of portfolio selection is a mathematical approach for deriving optimal portfolios. There are two methods to obtain optimal weights for portfolio selection, these two methods are as follows: (a) The least risk for a given level of expected return and (b) The greatest expected return for a given level of risk.

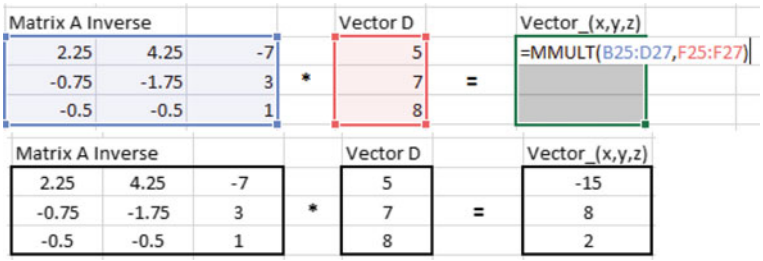
How does a portfolio manager apply these techniques in the real world?

The process would normally begin with a universe of securities available to the fund manager. These securities would be determined by the goals and objectives of the mutual fund. For example, a portfolio manager who runs a mutual fund specializing in health-care stocks would be required to select securities from the universe of health-care stocks. This would greatly reduce the analysis of the fund manager by limiting the number of securities available.

The next step in the process would be to determine the proportions of each security to be included in the portfolio. To do this, the fund manager would begin by setting a target



rate of return for the portfolio. After determining the target rate of return, the fund manager can determine the different proportions of each security that will allow the portfolio to reach this target rate of return.



The final step in the process would be for the fund manager to find the portfolio with the lowest variance given the target rate of return.

The optimal portfolio can be obtained mathematically through the use of the Lagrangian multipliers. The Lagrangian method allows the minimization or maximization of an objective function when the objective function is subject to some constraints. One of the goals of portfolio analysis is minimizing the risk or variance of the portfolio, subject to the portfolio's attaining some target expected rate of return, and also subject to the portfolio weights' summing to one. The problem can be stated mathematically as follows:

$$\text{Min } \sigma_p^2 = \sum_{i=1}^{n}\sum_{j=1}^{n} W_i W_j \sigma_{ij} \tag{9.1}$$

Subject to

(i) $\sum_{i=1}^{n} W_i E(R_i) = E^*$,

where E^* is the target expected return and

(ii) $\sum_{i=1}^{n} W_i = 1.0$.

The first constraint simply says that the expected return on the portfolio should equal the target return determined by the portfolio manager. The second constraint says that the weights of the securities invested in the portfolio must sum to one.

The Lagrangian objective function can be written as follows:

$$C = \sum_{i=1}^{n}\sum_{j=1}^{n} W_i W_j \text{Cov}(R_i R_j) + \lambda_1\left(1 - \sum_{i=1}^{n} W_i\right) + \lambda_2\left[E^* - \sum_{i=1}^{n} W_i E(R_i)\right]. \tag{9.2}$$

For three securities case, the Lagrangian objective function is as follows:

$$C = W_1^2\sigma_1^2 + W_2^2\sigma_2^2 + W_3^2\sigma_3^2 + 2W_1W_2\sigma_{12} + 2W_1W_3\sigma_{13} + 2W_2W_3\sigma_{23} + \lambda_1(1 - W_1 - W_2 - W_3) + \lambda_2 E^* - W_1E(R_1) - W_2E(R_2) - W_3E(R_3).$$

Taking the partial derivatives of (9.3) with respect to each of the variables, W_1, W_2, W_3, λ_1, λ_2 and setting the resulting five equations equal to zero yield the minimization of risk subject to the Lagrangian constraints. We can obtain the following equations.

$$\begin{aligned}
\frac{\partial C}{\partial W_1} &= 2W_1\sigma_1^2 + 2W_2\sigma_{12} + 2W_3\sigma_{13} - \lambda_1 - \lambda_2 E(R_1) = 0\\
\frac{\partial C}{\partial W_2} &= 2W_2\sigma_2^2 + 2W_1\sigma_{12} + 2W_3\sigma_{23} - \lambda_1 - \lambda_2 E(R_2) = 0\\
\frac{\partial C}{\partial W_3} &= 2W_3\sigma_3^2 + 2W_1\sigma_{13} + 2W_2\sigma_{23} - \lambda_1 - \lambda_2 E(R_3) = 0
\end{aligned} \tag{9.3}$$

$$\frac{\partial C}{\partial \lambda_1} = 1 - W_1 - W_2 - W_3 = 0$$

$$\frac{\partial C}{\partial \lambda_2} = E^* - W_1E(R_1) - W_2E(R_2) - W_3E(R_3) = 0$$

This system of five equations and five unknowns can be solved by the use of matrix algebra. Briefly, the Jacobian matrix of these equations is

$$\begin{bmatrix} 2\sigma_{11} & 2\sigma_{12} & 2\sigma_{13} & -1 & -E(R_1)\\ 2\sigma_{21} & 2\sigma_{22} & 2\sigma_{23} & -1 & -E(R_2)\\ 2\sigma_{31} & 2\sigma_{32} & 2\sigma_{33} & -1 & -E(R_3)\\ 1 & 1 & 1 & 0 & 0\\ E(R_1) & E(R_2) & E(R_3) & 0 & 0 \end{bmatrix} \times \begin{bmatrix} W_1\\ W_2\\ W_3\\ \lambda_1\\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0\\ 0\\ 0\\ 1\\ E^* \end{bmatrix} \tag{9.4}$$

Equation 9.4 can be redefined as

$$\text{AW} = \text{K} \tag{9.4a}$$

To solve for the unknown W of Eq. (9.4a), we can premultiply both sides of the Eq. (9.4a) by the inverse of A (denoted A^{-1}) and solve for the W column. This procedure can be found in Sect. 9.2.3.

Following the example from Lee et al. (2013), this example uses the information of returns and risk of Johnson & Johnson (JNJ), International Business Machines Corp. (IBM), and Boeing Co. (BA), for the period from April 2001 to April 2010. The data used are tabulated in Table 9.1.

Plugging the data listed in Table 9.1 and E^* = 0.00106 into the matrix-defined Eq. 9.4 above yields:

Table 9.1 Data for three securities

Company	E(R_i)	σ_i^2	Cov(R_i, R_j)
JNJ	0.0080	0.0025	$\sigma_{12} = 0.0007$
IBM	0.0050	0.0071	$\sigma_{23} = 0.0006$
BA	0.0113	0.0083	$\sigma_{13} = 0.0007$

$$\begin{bmatrix} 0.0910 & 0.0018 & 0.0008 & -1 & -0.0053 \\ 0.0036 & 0.1228 & 0.0020 & -1 & -0.0055 \\ 0.0008 & 0.0020 & 0.1050 & -1 & -0.0126 \\ 1 & 1 & 1 & 0 & 0 \\ 0.0053 & 0.0055 & 0.0126 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0.00106 \end{bmatrix} \tag{9.5}$$

When matrix A is properly inverted and post-multiplied by K, the solution vector $A^{-1}K$ is derived:

$$\begin{matrix} W \\ \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} \end{matrix} = \begin{matrix} A^{-1}K \\ \begin{bmatrix} 0.9442 \\ 0.6546 \\ -0.5988 \\ 0.1937 \\ -20.1953 \end{bmatrix} \end{matrix} \tag{9.6}$$

With the knowledge of the efficient-portfolio weights given that $E(R_p)$ is equal to 0.00106, 0.00212, and 0.00318.

Now we use data of IBM, Microsoft, and S&P500 as an example to calculate the optimal weights of the Markowitz model. The monthly rates of return for these three companies from 2016 to 2020 for all three stocks can be found in Appendix 9.1. The means, variances, and variance–covariance matrices for these three companies are presented in Fig. 9.1. By using the excel program, we can calculate the optimal Markowitz portfolio model, and its results are present in Fig. 9.2.

In Fig. 9.2, the top portion is the equation system used to calculate optimal weights, which was discussed previously. Then we use the input data and calculate related information for the equation system as presented in Step 1. Step 2 presents the procedure for calculating optimal weights. Finally, in the lower portion of this figure, we present the expected rate of return and the variance for this optimal portfolio.

There is a special case in terms of the Markowitz model. This case is the Minimum Variance Model. The only difference between these two models is that we exclude the expected return constraint that is

$$\sum_{i=1}^{n} W_i E(R_i) = E^*$$

For calculating the optimal expected return of the specific portfolio, we need first to calculate the mean, standard deviation, and variance–covariance matrix for companies. In this chapter, we use Fig. 9.1 to calculate the information.

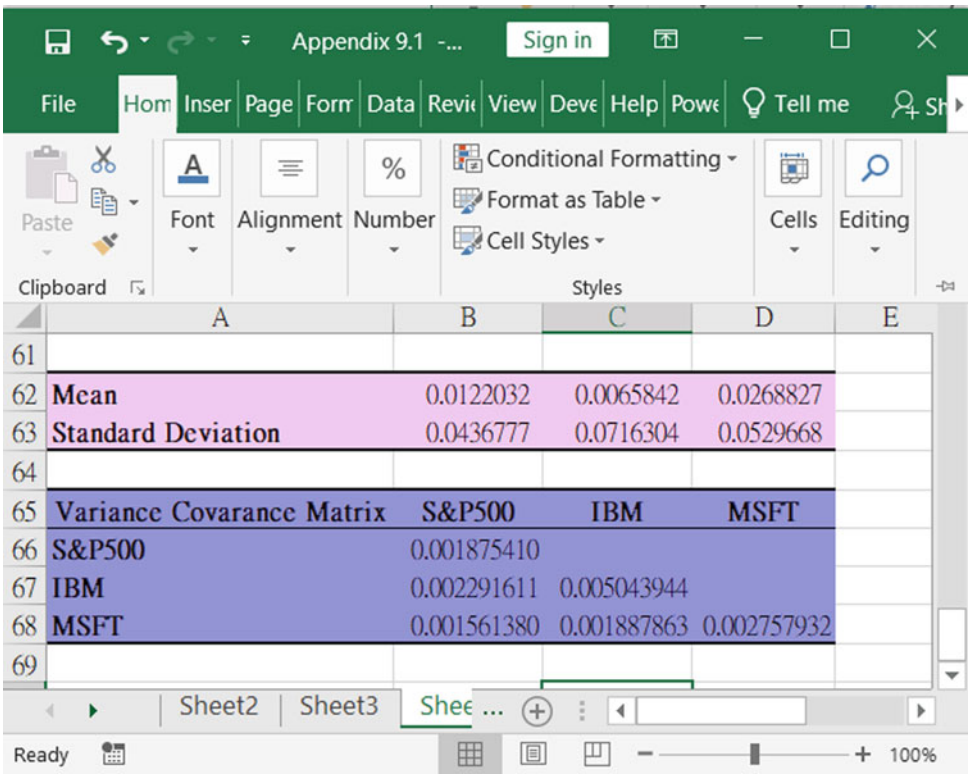


Fig. 9.1 The mean, standard deviation, and variance–covariance matrix for companies S&P500, IBM, and MSFT

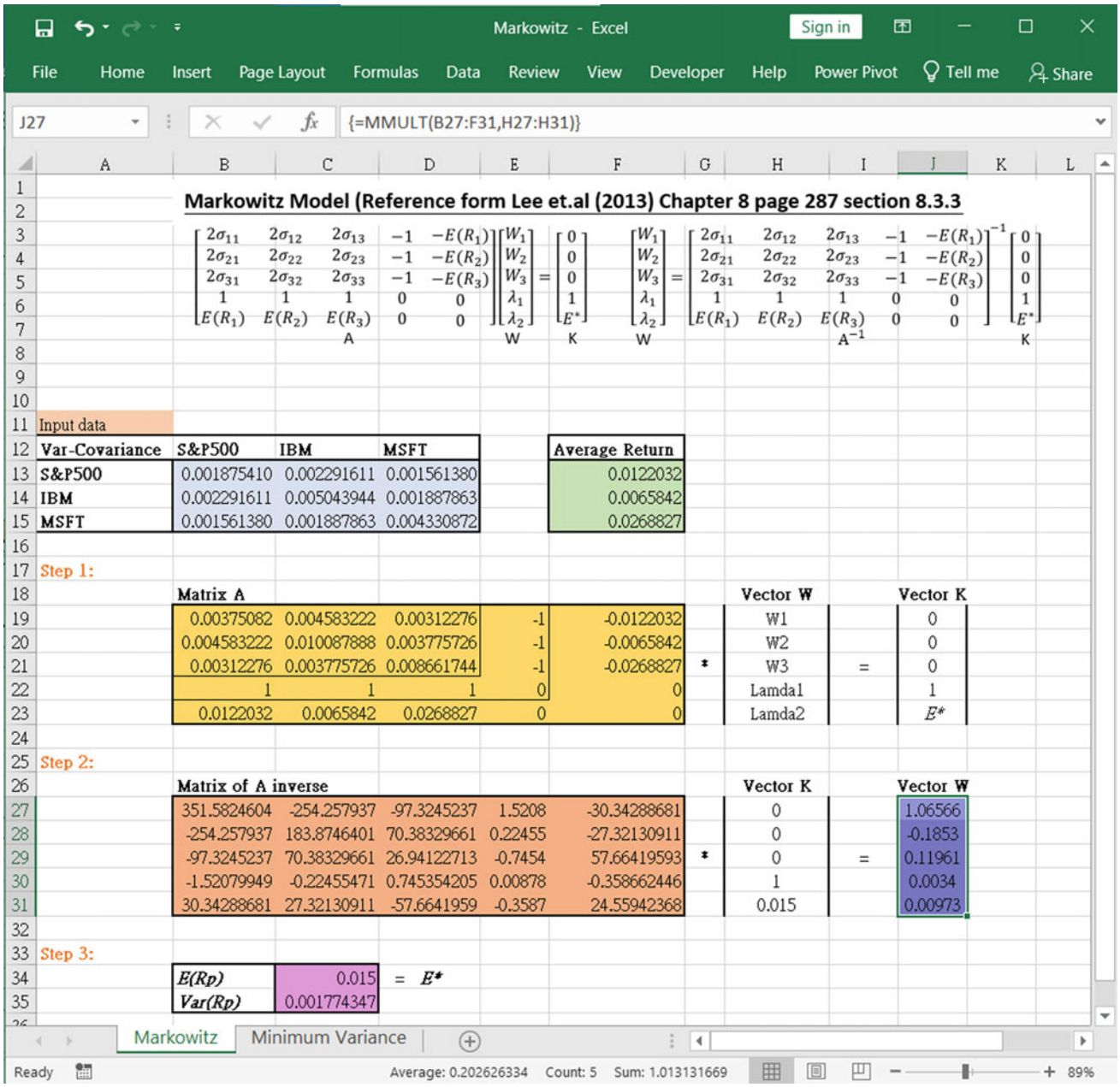


Fig. 9.2 Excel application of Markowitz model

9.4 Option Strategies

In this section, we will discuss how Excel can be used to calculate seven different option strategies. The seven strategies will include a long straddle, a short straddle, a long vertical spread, a short vertical spread, a protective put, a covered call, and a collar. The IBM options data on July 23, 2021, as presented in Appendix 9.2 is used to do the following seven options strategies.

9.4.1 Long Straddle

Assume that an investor expects the volatility of IBM stock to increase in the future and then can use a long straddle to profit. The investor can purchase a call option and a put option with the same exercise price of $150. The investor will profit from this type of position as long as the price of the underlying asset moves sufficiently up or down to more than cover the original cost of the option premiums. Let S_T and X denote the stock purchase price, future stock

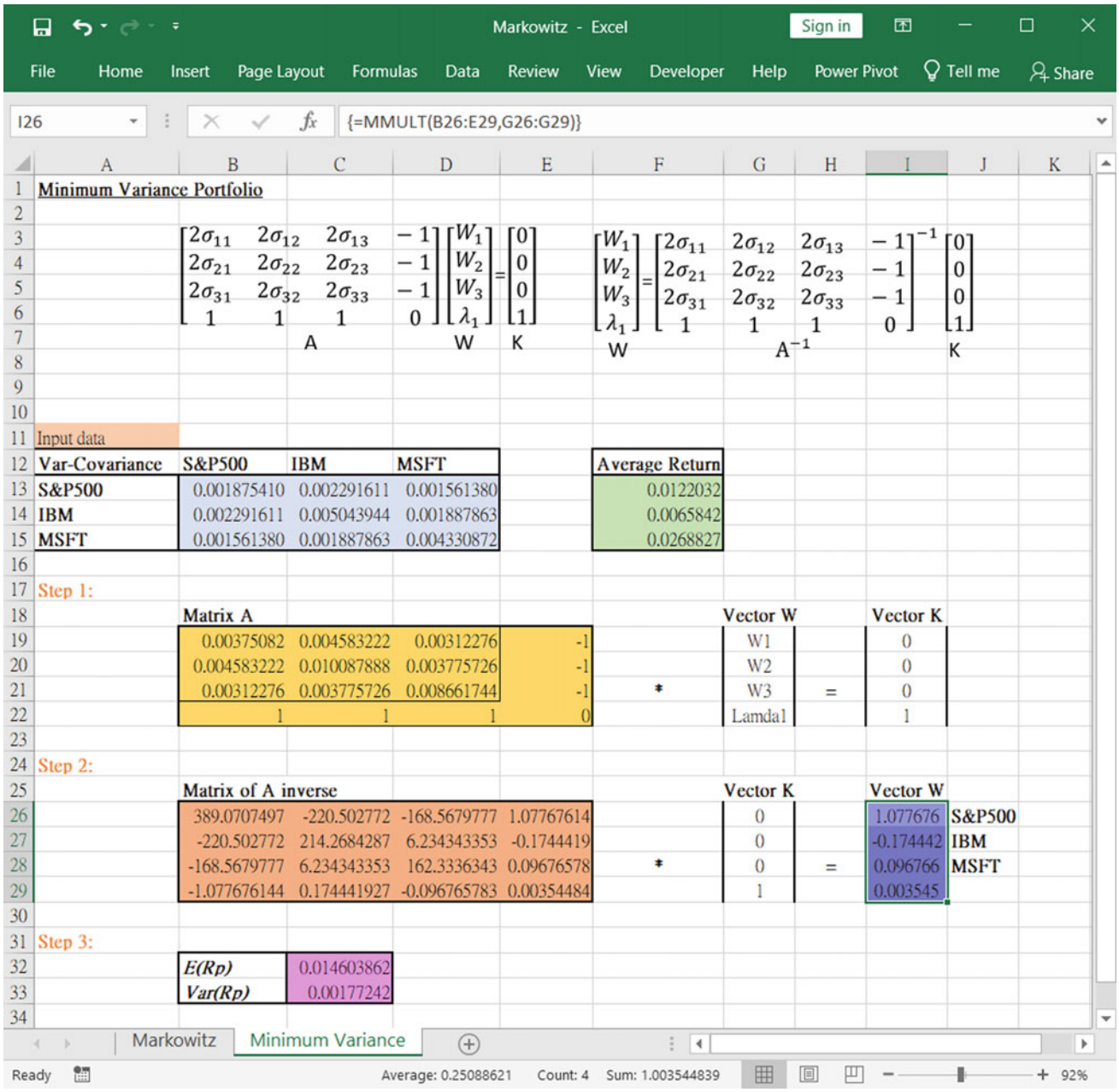
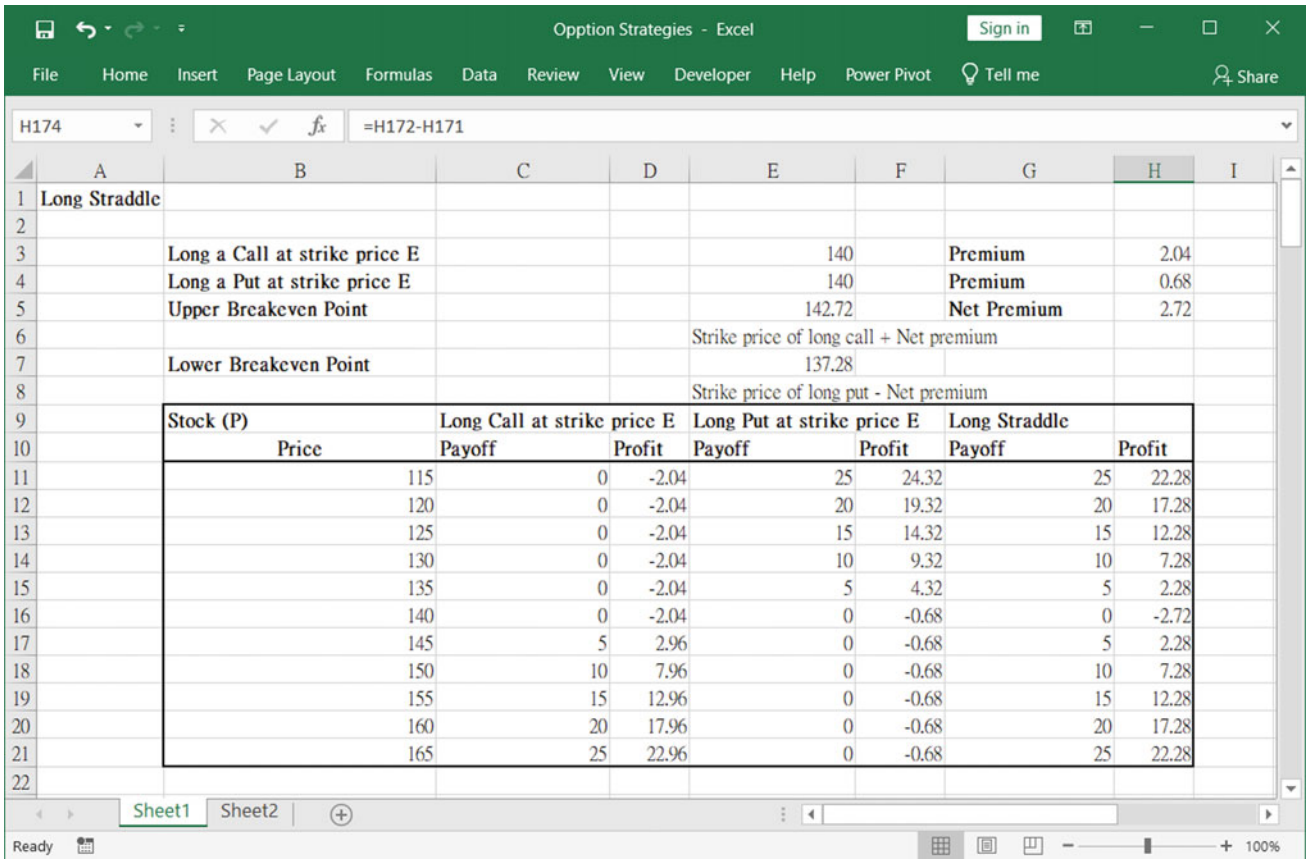


Fig. 9.3 Excel application for minimum variance model

price at the expiration time T, and the strike price, respectively. Given $X(E) = \$140$, S_T (you can find the value for S_T in the first column of the table in Fig. 9.4), and the premiums for the call option \$2.04 and put option \$0.68, Fig. 9.4 shows the values for long straddle at different stock prices at time T. For information in detail, you can find the excel function in Fig. 9.5 for calculations of the numbers in Fig. 9.4. The profit profile of the long straddle position is constructed in Fig. 9.6. The Break-even point means when the profit equals to zero. The formula for calculating the Upper Break-even point is (Strike Price of Long Call + Net Premium Paid) and the Lower Break-even point can be calculated as (Strike Price of Long Put − Net Premium Paid). For this example, the upper break-even point is \$142.72 and the lower break-even point is \$137.28.

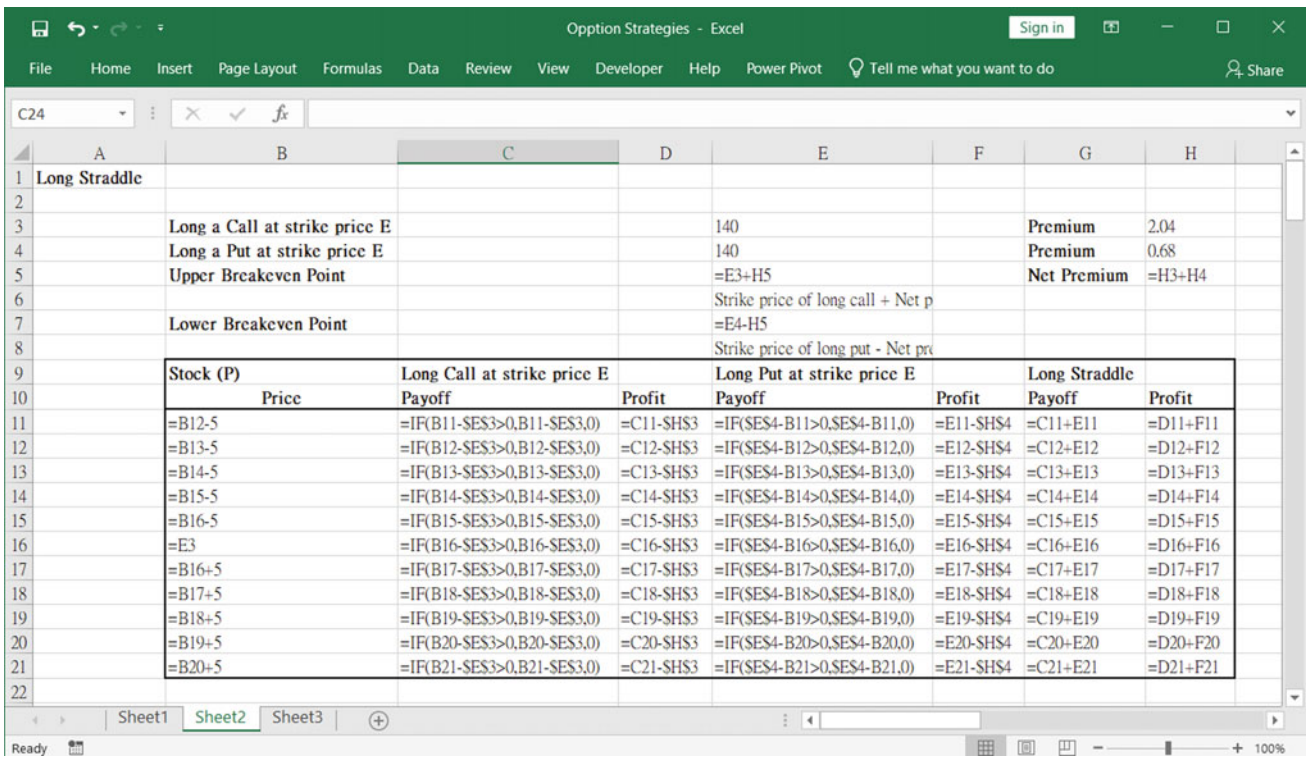
9.4.2 Short Straddle

Contrary to the long straddle strategy, an investor will use a short straddle via a short call and a short put on IBM stock with the same exercise price of \$150 when he or she expects



	A	B	C	D	E	F	G	H	I
1	Long Straddle								
2									
3		Long a Call at strike price E			140		Premium	2.04	
4		Long a Put at strike price E			140		Premium	0.68	
5		Upper Breakeven Point			142.72		Net Premium	2.72	
6					Strike price of long call + Net premium				
7		Lower Breakeven Point			137.28				
8					Strike price of long put - Net premium				
9		Stock (P)	Long Call at strike price E		Long Put at strike price E		Long Straddle		
10		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	
11		115	0	-2.04	25	24.32	25	22.28	
12		120	0	-2.04	20	19.32	20	17.28	
13		125	0	-2.04	15	14.32	15	12.28	
14		130	0	-2.04	10	9.32	10	7.28	
15		135	0	-2.04	5	4.32	5	2.28	
16		140	0	-2.04	0	-0.68	0	-2.72	
17		145	5	2.96	0	-0.68	5	2.28	
18		150	10	7.96	0	-0.68	10	7.28	
19		155	15	12.96	0	-0.68	15	12.28	
20		160	20	17.96	0	-0.68	20	17.28	
21		165	25	22.96	0	-0.68	25	22.28	
22									

Fig. 9.4 Value of a long straddle position at option expiration



	A	B	C	D	E	F	G	H
1	Long Straddle							
2								
3		Long a Call at strike price E			140		Premium	2.04
4		Long a Put at strike price E			140		Premium	0.68
5		Upper Breakeven Point			=E3+H5		Net Premium	=H3+H4
6					Strike price of long call + Net p			
7		Lower Breakeven Point			=E4-H5			
8					Strike price of long put - Net pr			
9		Stock (P)	Long Call at strike price E		Long Put at strike price E		Long Straddle	
10		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
11		=B12-5	=IF(B11-$E$3>0,B11-$E$3,0)	=C11-$H$3	=IF($E$4-B11>0,$E$4-B11,0)	=E11-$H$4	=C11+E11	=D11+F11
12		=B13-5	=IF(B12-$E$3>0,B12-$E$3,0)	=C12-$H$3	=IF($E$4-B12>0,$E$4-B12,0)	=E12-$H$4	=C12+E12	=D12+F12
13		=B14-5	=IF(B13-$E$3>0,B13-$E$3,0)	=C13-$H$3	=IF($E$4-B13>0,$E$4-B13,0)	=E13-$H$4	=C13+E13	=D13+F13
14		=B15-5	=IF(B14-$E$3>0,B14-$E$3,0)	=C14-$H$3	=IF($E$4-B14>0,$E$4-B14,0)	=E14-$H$4	=C14+E14	=D14+F14
15		=B16-5	=IF(B15-$E$3>0,B15-$E$3,0)	=C15-$H$3	=IF($E$4-B15>0,$E$4-B15,0)	=E15-$H$4	=C15+E15	=D15+F15
16		=E3	=IF(B16-$E$3>0,B16-$E$3,0)	=C16-$H$3	=IF($E$4-B16>0,$E$4-B16,0)	=E16-$H$4	=C16+E16	=D16+F16
17		=B16+5	=IF(B17-$E$3>0,B17-$E$3,0)	=C17-$H$3	=IF($E$4-B17>0,$E$4-B17,0)	=E17-$H$4	=C17+E17	=D17+F17
18		=B17+5	=IF(B18-$E$3>0,B18-$E$3,0)	=C18-$H$3	=IF($E$4-B18>0,$E$4-B18,0)	=E18-$H$4	=C18+E18	=D18+F18
19		=B18+5	=IF(B19-$E$3>0,B19-$E$3,0)	=C19-$H$3	=IF($E$4-B19>0,$E$4-B19,0)	=E19-$H$4	=C19+E19	=D19+F19
20		=B19+5	=IF(B20-$E$3>0,B20-$E$3,0)	=C20-$H$3	=IF($E$4-B20>0,$E$4-B20,0)	=E20-$H$4	=C20+E20	=D20+F20
21		=B20+5	=IF(B21-$E$3>0,B21-$E$3,0)	=C21-$H$3	=IF($E$4-B21>0,$E$4-B21,0)	=E21-$H$4	=C21+E21	=D21+F21
22								

Fig. 9.5 Excel formula for calculating the value of a long straddle position at option expiration

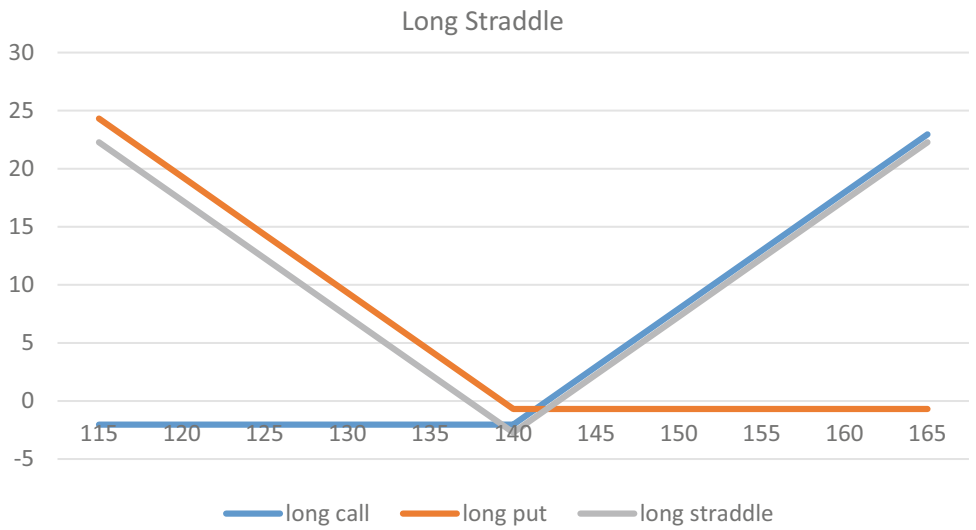


Fig. 9.6 Profit profile for long straddle

little or no movement in the price of IBM stock. Given X (E) = \$150, S_T (you can find the value for S_T in the first column of the table in Fig. 9.7) and the premiums for the call option \$4.35 and put option \$4.15, Fig. 9.7 shows the values for short straddle at different stock prices at time T. For information in detail, you can find the excel function in Fig. 9.8 for calculations of the numbers in Fig. 9.7. The profit profile of the short straddle position is constructed in Fig. 9.9. The Break-even point means when the profit equals to zero. The Upper Break-even point for Short Straddle can be calculated as (Strike Price of Short Call + Net Premium Received) and the Lower Break-even point can be calculated as (Strike Price of Short Put − Net Premium Received). For this example, the upper break-even point is \$158.50 and the lower break-even point is \$141.50.

9.4.3 Long Vertical Spread

This strategy combines a long call (or put) with a low strike price and a short call (or put) with a high strike price. For example, an investor purchases a call with the exercise price of \$155 and sells a call with the exercise price of \$150. Given X_1(E1) = \$155, X_2(E2) = \$150, S_T (you can find the value for S_T in the first column of the table in Fig. 9.10), and the premiums for the long call option is \$1.97 and the short call option is \$4.60, Fig. 9.10 shows the values for Long Vertical Spread at different stock prices at time T. For information in detail, you can find the excel function in Fig. 9.11 for calculations of the numbers in Fig. 9.10. The profit profile of the Long Vertical Spread is constructed in Fig. 9.12. The Break-even point means when the profit equals to zero. The Break-even point for Long Vertical Spread can be calculated as (Strike Price of Long Call + Net Premium Paid).

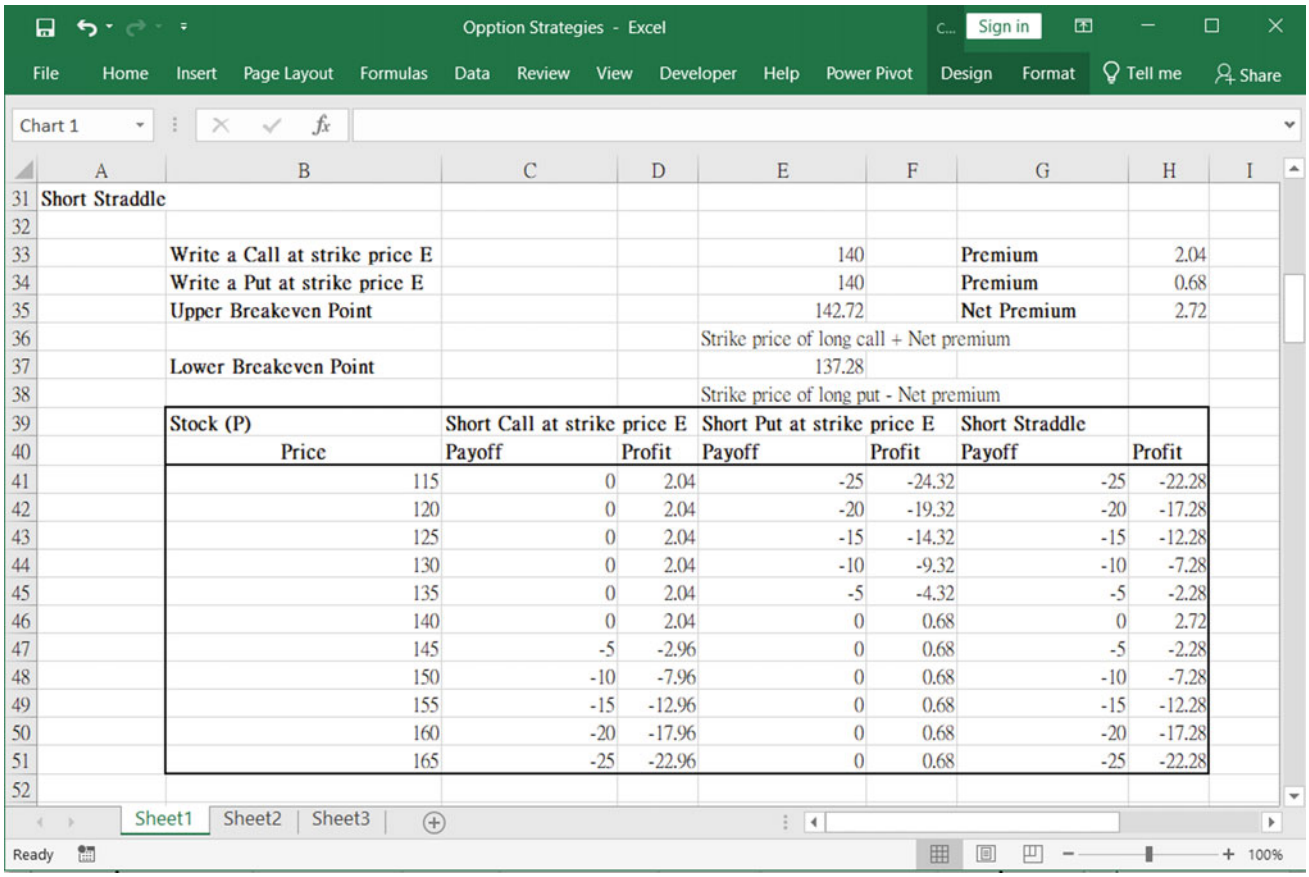
For this example, the break-even point is \$152.63.

9.4.4 Short Vertical Spread

Contrary to a long vertical spread, this strategy combines a long call (or put) with a high strike price and a short call (or put) with a low strike price. For example, an investor purchases a call with the exercise price of \$150 and sells a call with the exercise price of \$155. Given X_1 (E1) = \$150, X_2 (E2) = \$155, S_T (you can find the value for S_T in the first column of the table in Fig. 9.13), and the premiums for the long call option is \$4.35 and the short call option is \$2.13, Fig. 9.13 shows the values for the short vertical spread at different stock prices at time T. For information in detail, you can find the excel function in Fig. 9.14 for calculations of the numbers in Fig. 9.13. The profit profile of the short vertical spread is constructed in Fig. 9.15. The Break-even point means when the profit equals to zero. The Break-even point for Short Vertical Spread can be calculated as (Strike Price of Short Call + Net Premium Received). For this example, the break-even point is \$152.22.

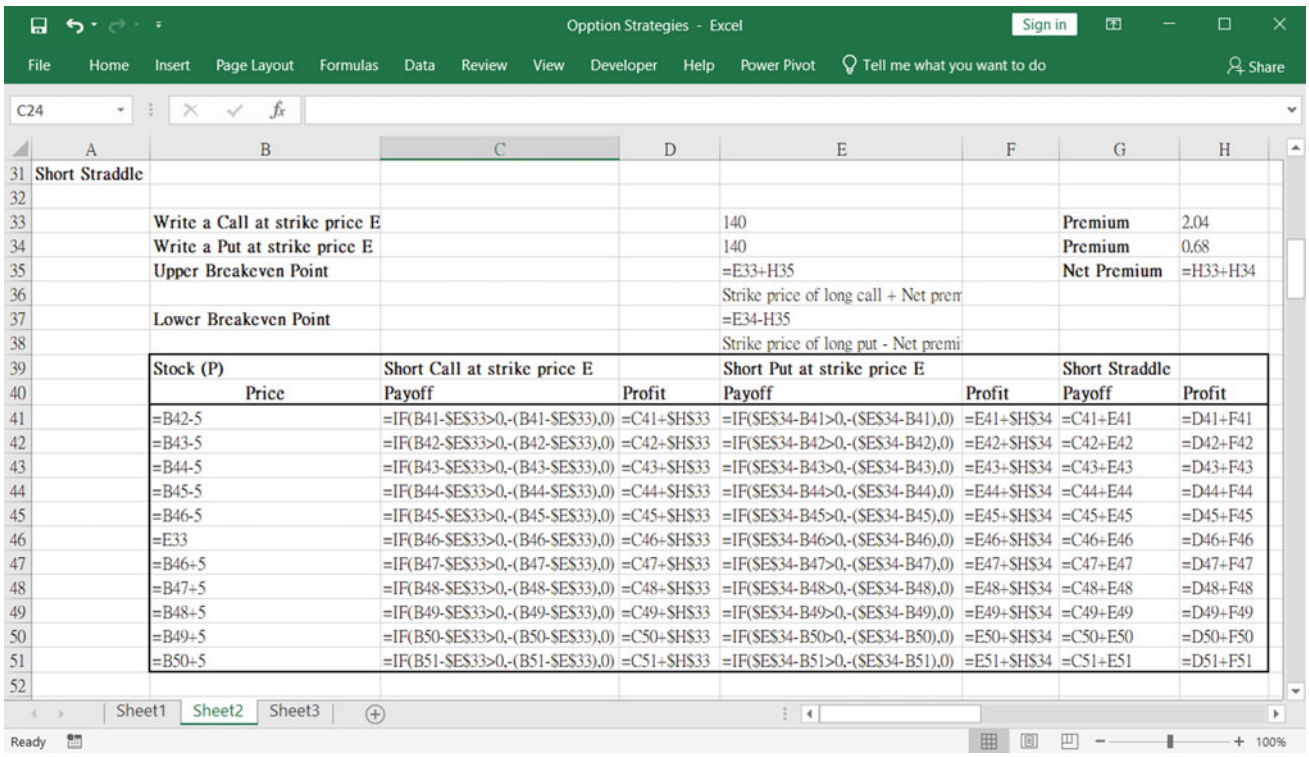
9.4.5 Protective Put

Assume that an investor wants to invest in the IBM stock on March 9, 2011, but does not desire to bear any potential loss for prices below \$150. The investor can purchase IBM stock and at the same time buy the put option with a strike price of



	A	B	C	D	E	F	G	H	I
31	Short Straddle								
32									
33		Write a Call at strike price E			140		Premium	2.04	
34		Write a Put at strike price E			140		Premium	0.68	
35		Upper Breakeven Point			142.72		Net Premium	2.72	
36					Strike price of long call + Net premium				
37		Lower Breakeven Point			137.28				
38					Strike price of long put - Net premium				
39		Stock (P)	Short Call at strike price E		Short Put at strike price E		Short Straddle		
40		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	
41		115	0	2.04	-25	-24.32	-25	-22.28	
42		120	0	2.04	-20	-19.32	-20	-17.28	
43		125	0	2.04	-15	-14.32	-15	-12.28	
44		130	0	2.04	-10	-9.32	-10	-7.28	
45		135	0	2.04	-5	-4.32	-5	-2.28	
46		140	0	2.04	0	0.68	0	2.72	
47		145	-5	-2.96	0	0.68	-5	-2.28	
48		150	-10	-7.96	0	0.68	-10	-7.28	
49		155	-15	-12.96	0	0.68	-15	-12.28	
50		160	-20	-17.96	0	0.68	-20	-17.28	
51		165	-25	-22.96	0	0.68	-25	-22.28	
52									

Fig. 9.7 Value of a short straddle position at option expiration



	A	B	C	D	E	F	G	H
31	Short Straddle							
32								
33		Write a Call at strike price E			140		Premium	2.04
34		Write a Put at strike price E			140		Premium	0.68
35		Upper Breakeven Point			=E33+H35		Net Premium	=H33+H34
36					Strike price of long call + Net prem			
37		Lower Breakeven Point			=E34-H35			
38					Strike price of long put - Net premi			
39		Stock (P)	Short Call at strike price E		Short Put at strike price E		Short Straddle	
40		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
41		=B42-5	=IF(B41-$E$33>0,-(B41-$E$33),0)	=C41+$H$33	=IF($E$34-B41>0,-($E$34-B41),0)	=E41+$H$34	=C41+E41	=D41+F41
42		=B43-5	=IF(B42-$E$33>0,-(B42-$E$33),0)	=C42+$H$33	=IF($E$34-B42>0,-($E$34-B42),0)	=E42+$H$34	=C42+E42	=D42+F42
43		=B44-5	=IF(B43-$E$33>0,-(B43-$E$33),0)	=C43+$H$33	=IF($E$34-B43>0,-($E$34-B43),0)	=E43+$H$34	=C43+E43	=D43+F43
44		=B45-5	=IF(B44-$E$33>0,-(B44-$E$33),0)	=C44+$H$33	=IF($E$34-B44>0,-($E$34-B44),0)	=E44+$H$34	=C44+E44	=D44+F44
45		=B46-5	=IF(B45-$E$33>0,-(B45-$E$33),0)	=C45+$H$33	=IF($E$34-B45>0,-($E$34-B45),0)	=E45+$H$34	=C45+E45	=D45+F45
46		=E33	=IF(B46-$E$33>0,-(B46-$E$33),0)	=C46+$H$33	=IF($E$34-B46>0,-($E$34-B46),0)	=E46+$H$34	=C46+E46	=D46+F46
47		=B46+5	=IF(B47-$E$33>0,-(B47-$E$33),0)	=C47+$H$33	=IF($E$34-B47>0,-($E$34-B47),0)	=E47+$H$34	=C47+E47	=D47+F47
48		=B47+5	=IF(B48-$E$33>0,-(B48-$E$33),0)	=C48+$H$33	=IF($E$34-B48>0,-($E$34-B48),0)	=E48+$H$34	=C48+E48	=D48+F48
49		=B48+5	=IF(B49-$E$33>0,-(B49-$E$33),0)	=C49+$H$33	=IF($E$34-B49>0,-($E$34-B49),0)	=E49+$H$34	=C49+E49	=D49+F49
50		=B49+5	=IF(B50-$E$33>0,-(B50-$E$33),0)	=C50+$H$33	=IF($E$34-B50>0,-($E$34-B50),0)	=E50+$H$34	=C50+E50	=D50+F50
51		=B50+5	=IF(B51-$E$33>0,-(B51-$E$33),0)	=C51+$H$33	=IF($E$34-B51>0,-($E$34-B51),0)	=E51+$H$34	=C51+E51	=D51+F51
52								

Fig. 9.8 Excel formula for calculating the value of a short straddle position at option expiration

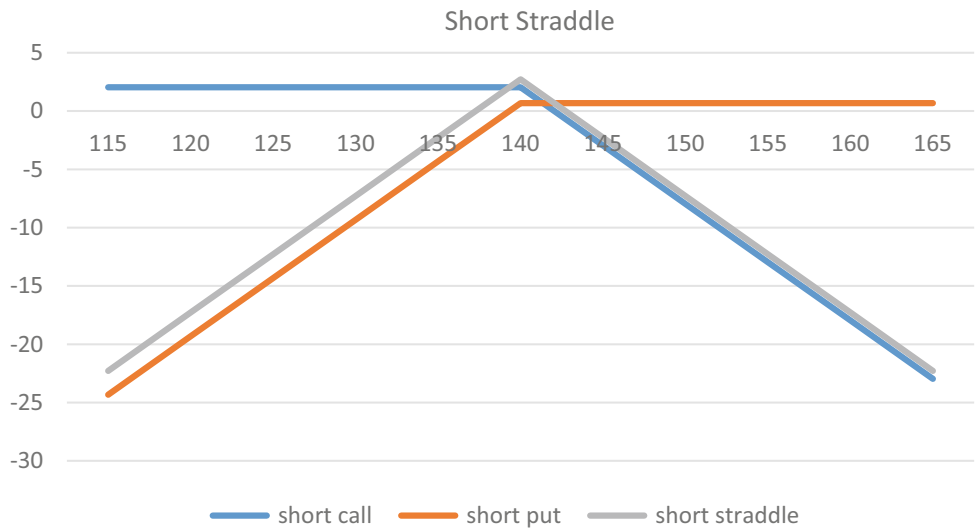
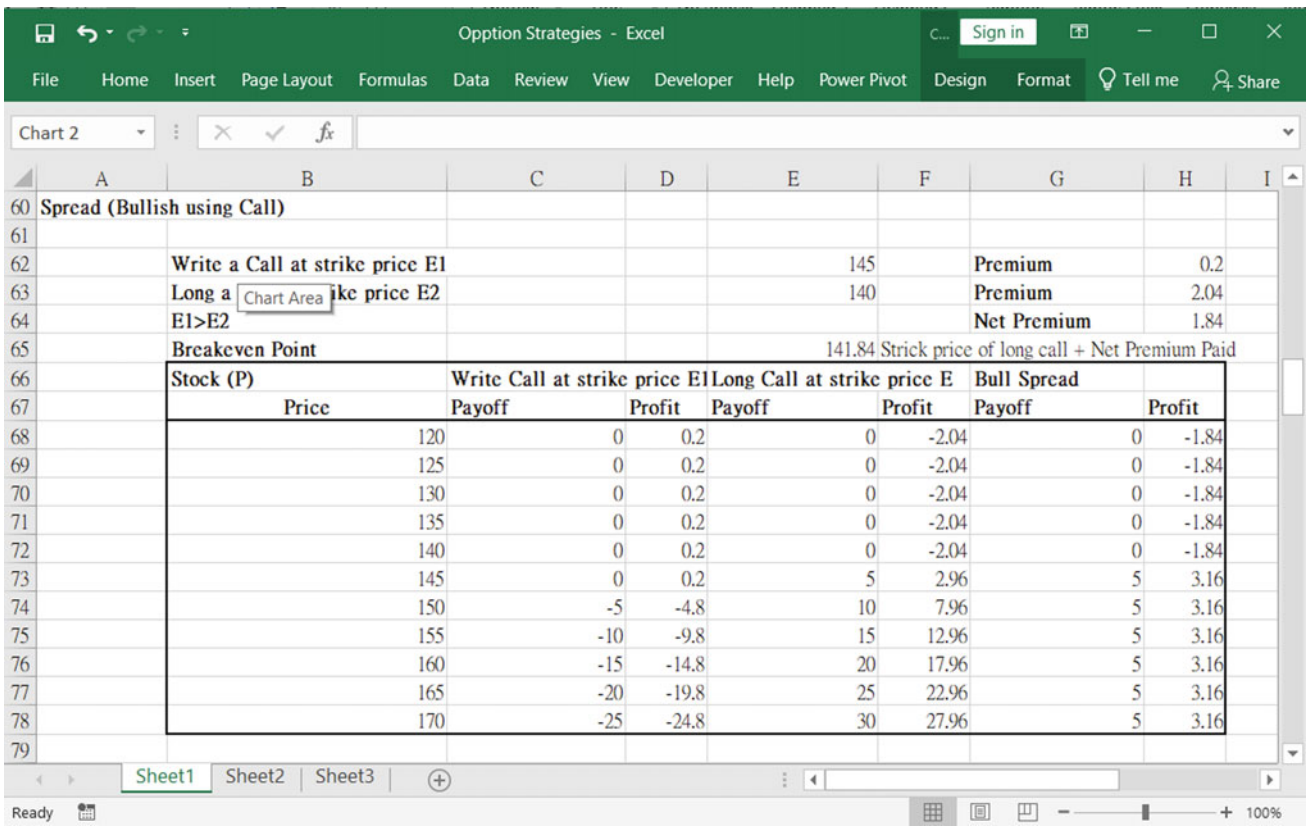


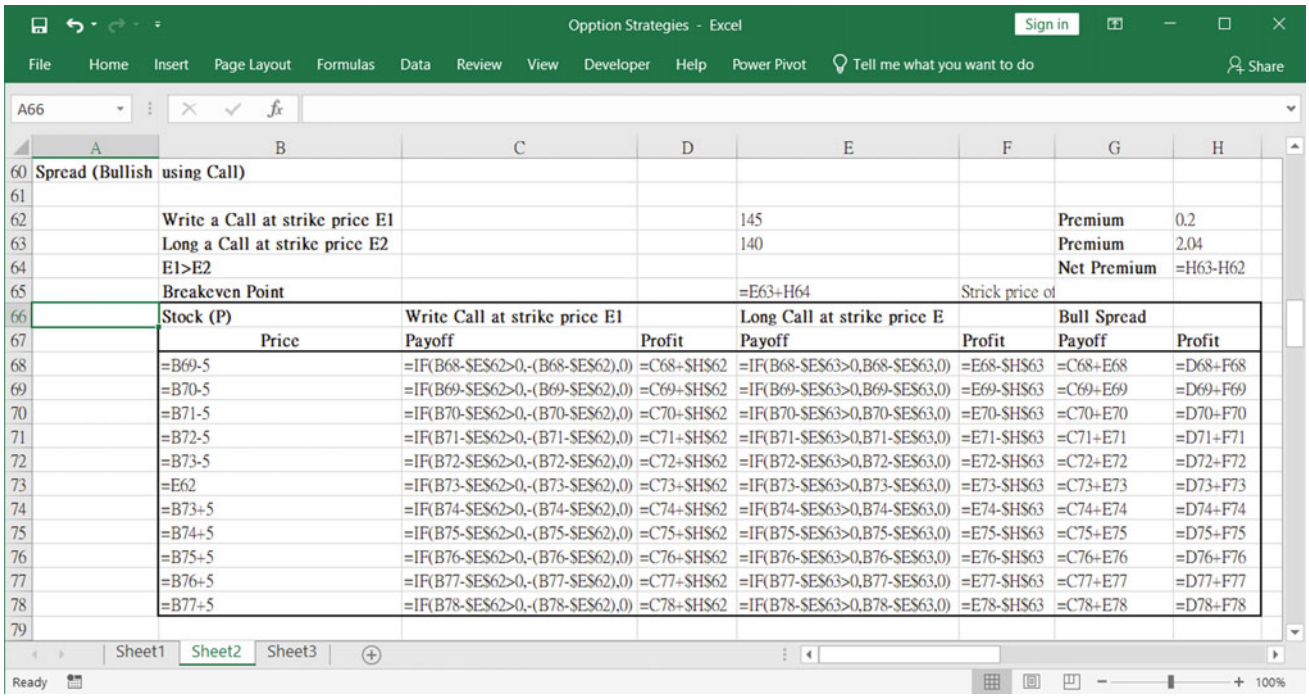
Fig. 9.9 Profit profile for short straddle



	A	B	C	D	E	F	G	H	I
60	Spread (Bullish using Call)								
61									
62		Write a Call at strike price E1			145		Premium	0.2	
63		Long a ike price E2			140		Premium	2.04	
64		E1>E2					Net Premium	1.84	
65		Breakeven Point			141.84 Strick price of long call + Net Premium Paid				
66		Stock (P)	Write Call at strike price E1		Long Call at strike price E		Bull Spread		
67		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	
68		120	0	0.2	0	-2.04	0	-1.84	
69		125	0	0.2	0	-2.04	0	-1.84	
70		130	0	0.2	0	-2.04	0	-1.84	
71		135	0	0.2	0	-2.04	0	-1.84	
72		140	0	0.2	0	-2.04	0	-1.84	
73		145	0	0.2	5	2.96	5	3.16	
74		150	-5	-4.8	10	7.96	5	3.16	
75		155	-10	-9.8	15	12.96	5	3.16	
76		160	-15	-14.8	20	17.96	5	3.16	
77		165	-20	-19.8	25	22.96	5	3.16	
78		170	-25	-24.8	30	27.96	5	3.16	
79									

Fig. 9.10 Value of a long vertical spread position at option expiration

$150. Given current stock S_0 = $155.54, exercise price X (E) = $150, S_T (you can find the value for S_T in the first column of the table in Fig. 9.16), and the premium for the put option $4.40 (the ask price), Fig. 9.16 shows the values for Protective Put at different stock prices at time T. For information in detail, you can find the excel function in Fig. 9.17 for calculations of the numbers in Fig. 9.16. The profit profile of the Protective Put position is constructed in Fig. 9.18. The Break-even point means when the profit equals to zero. The Break-even point for Protective Put can



Opption Strategies - Excel

	A	B	C	D	E	F	G	H
60	Spread (Bullish	using Call)						
61								
62		Write a Call at strike price E1			145		Premium	0.2
63		Long a Call at strike price E2			140		Premium	2.04
64		E1>E2					Net Premium	=H63-H62
65		Breakeven Point			=E63+H64	Strick price of		
66		Stock (P)	Write Call at strike price E1		Long Call at strike price E		Bull Spread	
67		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
68		=B69-5	=IF(B68-$E$62>0,-(B68-$E$62),0)	=C68+$H$62	=IF(B68-$E$63>0,B68-$E$63,0)	=E68-$H$63	=C68+E68	=D68+F68
69		=B70-5	=IF(B69-$E$62>0,-(B69-$E$62),0)	=C69+$H$62	=IF(B69-$E$63>0,B69-$E$63,0)	=E69-$H$63	=C69+E69	=D69+F69
70		=B71-5	=IF(B70-$E$62>0,-(B70-$E$62),0)	=C70+$H$62	=IF(B70-$E$63>0,B70-$E$63,0)	=E70-$H$63	=C70+E70	=D70+F70
71		=B72-5	=IF(B71-$E$62>0,-(B71-$E$62),0)	=C71+$H$62	=IF(B71-$E$63>0,B71-$E$63,0)	=E71-$H$63	=C71+E71	=D71+F71
72		=B73-5	=IF(B72-$E$62>0,-(B72-$E$62),0)	=C72+$H$62	=IF(B72-$E$63>0,B72-$E$63,0)	=E72-$H$63	=C72+E72	=D72+F72
73		=E62	=IF(B73-$E$62>0,-(B73-$E$62),0)	=C73+$H$62	=IF(B73-$E$63>0,B73-$E$63,0)	=E73-$H$63	=C73+E73	=D73+F73
74		=B73+5	=IF(B74-$E$62>0,-(B74-$E$62),0)	=C74+$H$62	=IF(B74-$E$63>0,B74-$E$63,0)	=E74-$H$63	=C74+E74	=D74+F74
75		=B74+5	=IF(B75-$E$62>0,-(B75-$E$62),0)	=C75+$H$62	=IF(B75-$E$63>0,B75-$E$63,0)	=E75-$H$63	=C75+E75	=D75+F75
76		=B75+5	=IF(B76-$E$62>0,-(B76-$E$62),0)	=C76+$H$62	=IF(B76-$E$63>0,B76-$E$63,0)	=E76-$H$63	=C76+E76	=D76+F76
77		=B76+5	=IF(B77-$E$62>0,-(B77-$E$62),0)	=C77+$H$62	=IF(B77-$E$63>0,B77-$E$63,0)	=E77-$H$63	=C77+E77	=D77+F77
78		=B77+5	=IF(B78-$E$62>0,-(B78-$E$62),0)	=C78+$H$62	=IF(B78-$E$63>0,B78-$E$63,0)	=E78-$H$63	=C78+E78	=D78+F78
79								

Fig. 9.11 Excel formula for calculating the value of a long vertical spread position at option expiration

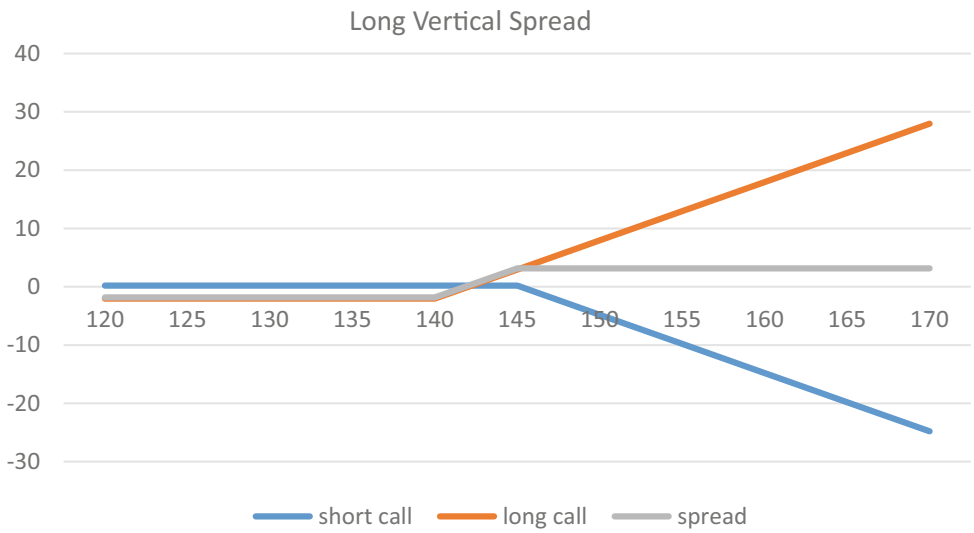
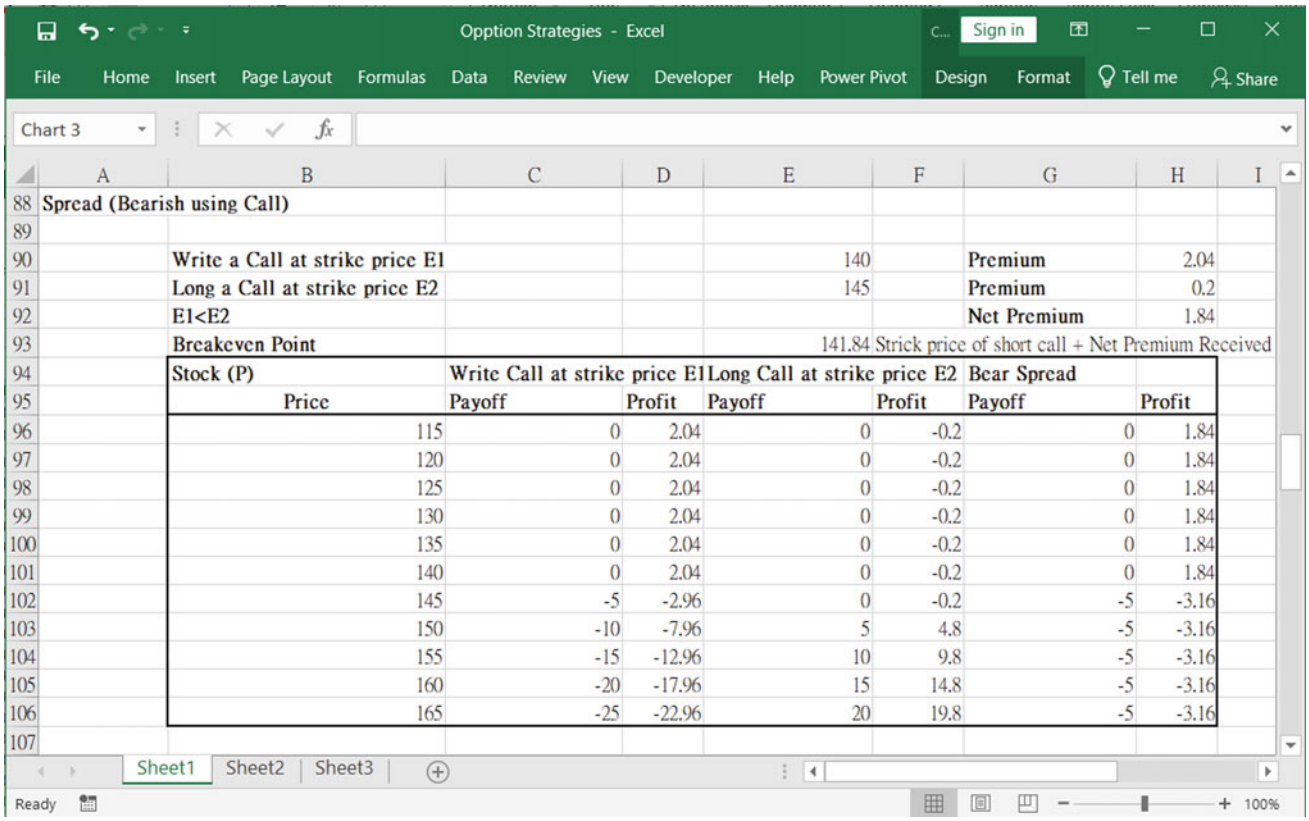


Fig. 9.12 Profit profile for long vertical spread

be calculated as (Purchase Price of underlying + Premium Paid). For this example, the break-even point is $155.54.

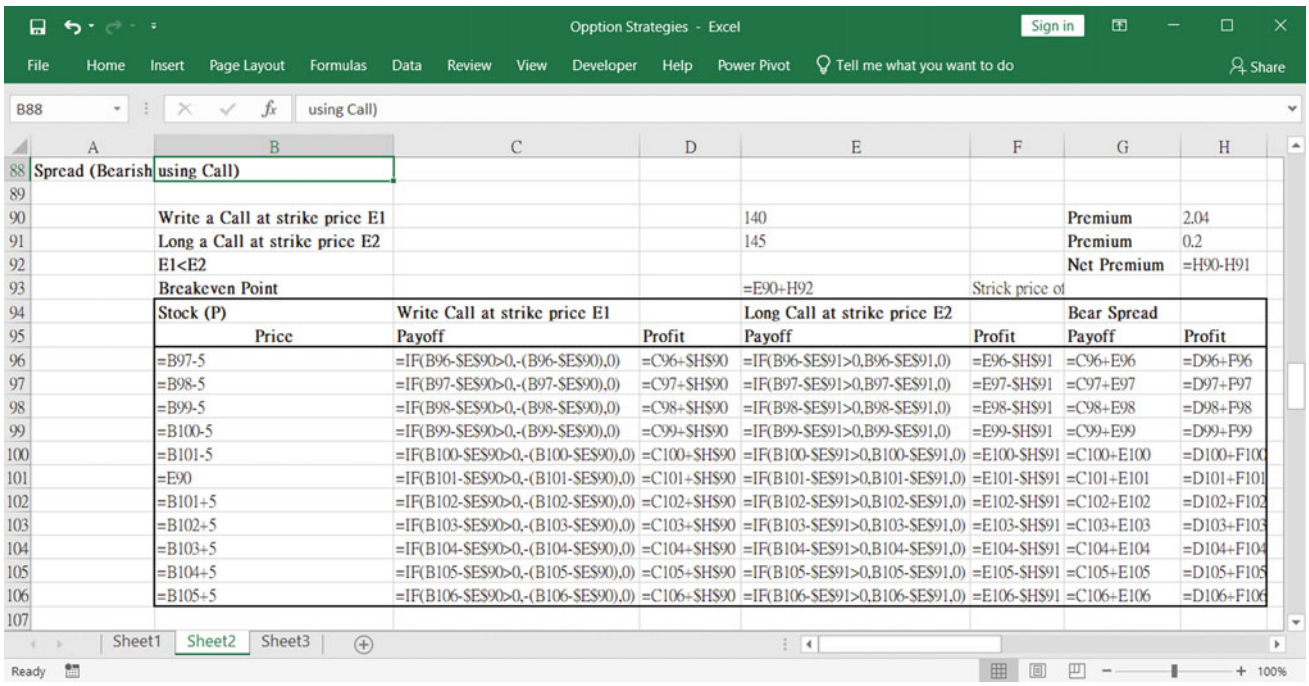
9.4.6 Covered Call

This strategy involves investing in a stock and selling a call option on the stock at the same time. The value at the expiration of the call will be the stock value minus the value of the call. The call is "covered" because the potential obligation of delivering the stock is covered by the stock held in the portfolio. In essence, the sale of the call sold the claim to any stock value above the strike price in return for the initial premium. Suppose a manager of a stock fund holds a share of IBM stock on October 12, 2015, and she plans to sell the IBM stock if its price hits $155. Then she can write a share of a call option with a strike price of $155 to establish the position. She shorts the call and collects premiums. Given that current stock price S_0 = $151.14, X (E) = $155, S_T (you can find the value for S_T in the first



	A	B	C	D	E	F	G	H	I
88	Spread (Bearish using Call)								
89									
90		Write a Call at strike price E1			140		Premium	2.04	
91		Long a Call at strike price E2			145		Premium	0.2	
92		E1<E2					Net Premium	1.84	
93		Breakeven Point				141.84	Strick price of short call + Net Premium Received		
94		Stock (P)	Write Call at strike price E1		Long Call at strike price E2		Bear Spread		
95		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	
96		115	0	2.04	0	-0.2	0	1.84	
97		120	0	2.04	0	-0.2	0	1.84	
98		125	0	2.04	0	-0.2	0	1.84	
99		130	0	2.04	0	-0.2	0	1.84	
100		135	0	2.04	0	-0.2	0	1.84	
101		140	0	2.04	0	-0.2	0	1.84	
102		145	-5	-2.96	0	-0.2	-5	-3.16	
103		150	-10	-7.96	5	4.8	-5	-3.16	
104		155	-15	-12.96	10	9.8	-5	-3.16	
105		160	-20	-17.96	15	14.8	-5	-3.16	
106		165	-25	-22.96	20	19.8	-5	-3.16	
107									

Fig. 9.13 Value of a short vertical spread position at option expiration



	A	B	C	D	E	F	G	H
88	Spread (Bearish	using Call)						
89								
90		Write a Call at strike price E1			140		Premium	2.04
91		Long a Call at strike price E2			145		Premium	0.2
92		E1<E2					Net Premium	=H90-H91
93		Breakeven Point			=E90+H92	Strick price of		
94		Stock (P)	Write Call at strike price E1		Long Call at strike price E2		Bear Spread	
95		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
96		=B97-5	=IF(B96-$E$90>0,-(B96-$E$90),0)	=C96+$H$90	=IF(B96-$E$91>0,B96-$E$91,0)	=E96-$H$91	=C96+E96	=D96+F96
97		=B98-5	=IF(B97-$E$90>0,-(B97-$E$90),0)	=C97+$H$90	=IF(B97-$E$91>0,B97-$E$91,0)	=E97-$H$91	=C97+E97	=D97+F97
98		=B99-5	=IF(B98-$E$90>0,-(B98-$E$90),0)	=C98+$H$90	=IF(B98-$E$91>0,B98-$E$91,0)	=E98-$H$91	=C98+E98	=D98+F98
99		=B100-5	=IF(B99-$E$90>0,-(B99-$E$90),0)	=C99+$H$90	=IF(B99-$E$91>0,B99-$E$91,0)	=E99-$H$91	=C99+E99	=D99+F99
100		=B101-5	=IF(B100-$E$90>0,-(B100-$E$90),0)	=C100+$H$90	=IF(B100-$E$91>0,B100-$E$91,0)	=E100-$H$91	=C100+E100	=D100+F100
101		=E90	=IF(B101-$E$90>0,-(B101-$E$90),0)	=C101+$H$90	=IF(B101-$E$91>0,B101-$E$91,0)	=E101-$H$91	=C101+E101	=D101+F101
102		=B101+5	=IF(B102-$E$90>0,-(B102-$E$90),0)	=C102+$H$90	=IF(B102-$E$91>0,B102-$E$91,0)	=E102-$H$91	=C102+E102	=D102+F102
103		=B102+5	=IF(B103-$E$90>0,-(B103-$E$90),0)	=C103+$H$90	=IF(B103-$E$91>0,B103-$E$91,0)	=E103-$H$91	=C103+E103	=D103+F103
104		=B103+5	=IF(B104-$E$90>0,-(B104-$E$90),0)	=C104+$H$90	=IF(B104-$E$91>0,B104-$E$91,0)	=E104-$H$91	=C104+E104	=D104+F104
105		=B104+5	=IF(B105-$E$90>0,-(B105-$E$90),0)	=C105+$H$90	=IF(B105-$E$91>0,B105-$E$91,0)	=E105-$H$91	=C105+E105	=D105+F105
106		=B105+5	=IF(B106-$E$90>0,-(B106-$E$90),0)	=C106+$H$90	=IF(B106-$E$91>0,B106-$E$91,0)	=E106-$H$91	=C106+E106	=D106+F106
107								

Fig. 9.14 Excel formula for calculating the value of a short vertical spread position at option expiration

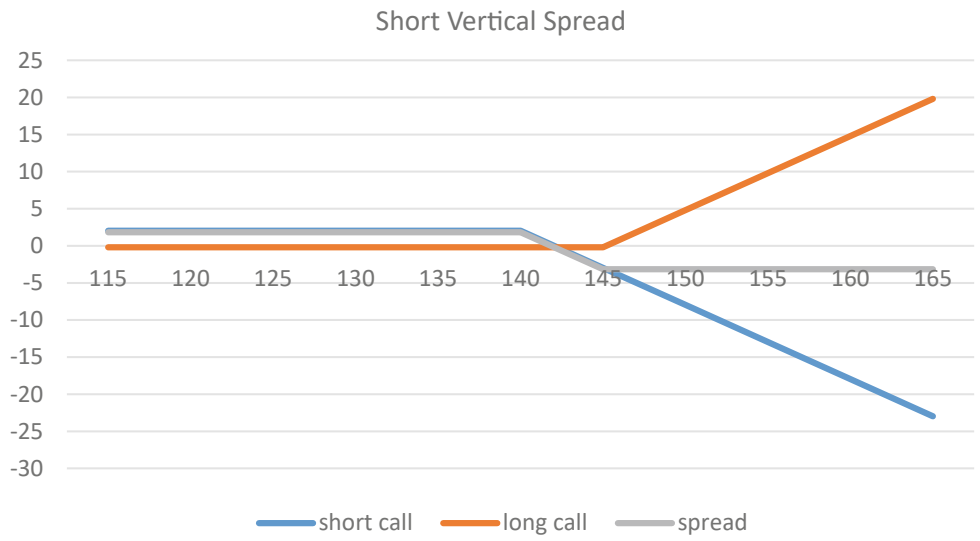


Fig. 9.15 Profit profile for short vertical spread

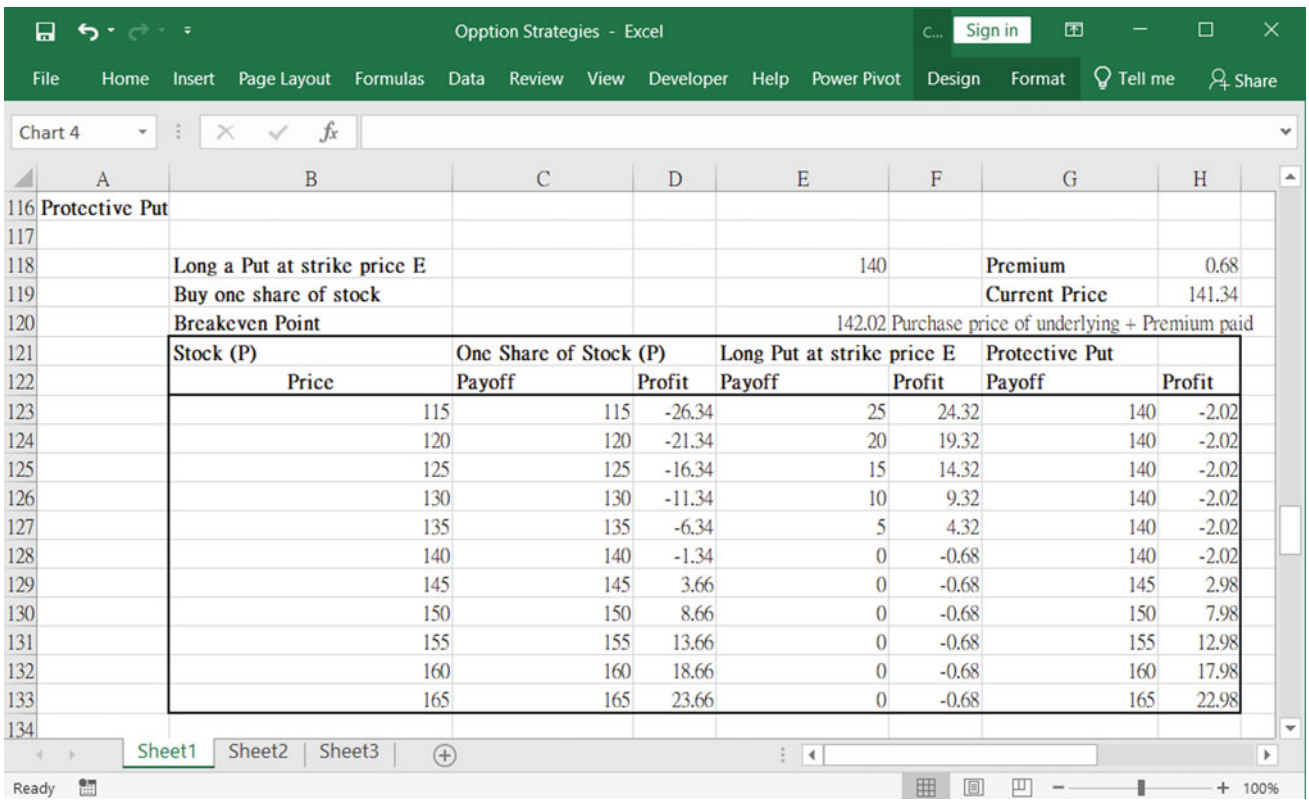
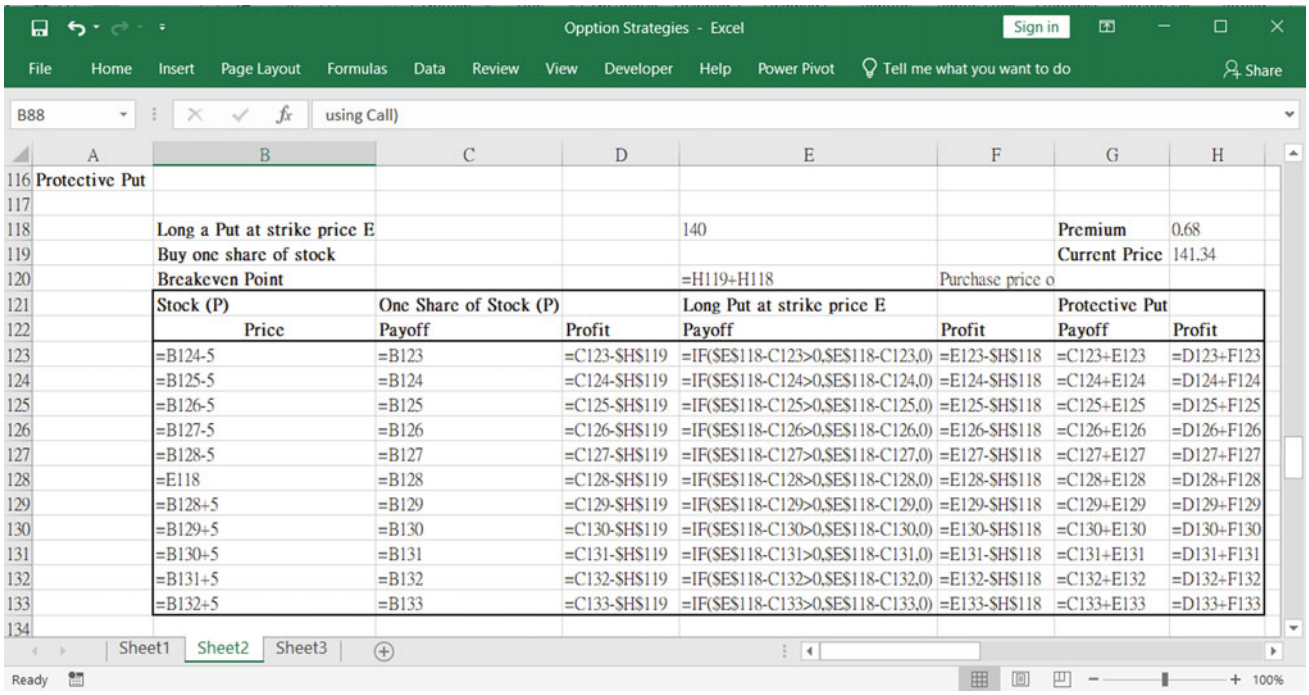


Fig. 9.16 Value of a protective put position at option expiration

column of the table in Fig. 9.19), and the premium for the call option $1.97(the bid price), Fig. 9.19 shows the values for the covered call at different stock prices at time *T*. For information in detail, you can find the excel function in Fig. 9.20 for calculations of the numbers in Fig. 9.19. The profit profile of the covered call position is constructed in Fig. 9.21. It can be shown that the payoff pattern of a covered call is exactly equal to shorting a put. Therefore, the covered call has frequently been used to replace shorting a put in dynamic hedging practice. The Break-even point means when the profit equals to zero. The Break-even point for a Covered Call can be calculated as (Purchase price of



	A	B	C	D	E	F	G	H
116	Protective Put							
117								
118		Long a Put at strike price E			140		Premium	0.68
119		Buy one share of stock					Current Price	141.34
120		Breakeven Point			=H119+H118	Purchase price o		
121		Stock (P)	One Share of Stock (P)		Long Put at strike price E		Protective Put	
122		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
123		=B124-5	=B123	=C123-$H$119	=IF($E$118-C123>0,$E$118-C123,0)	=E123-$H$118	=C123+E123	=D123+F123
124		=B125-5	=B124	=C124-$H$119	=IF($E$118-C124>0,$E$118-C124,0)	=E124-$H$118	=C124+E124	=D124+F124
125		=B126-5	=B125	=C125-$H$119	=IF($E$118-C125>0,$E$118-C125,0)	=E125-$H$118	=C125+E125	=D125+F125
126		=B127-5	=B126	=C126-$H$119	=IF($E$118-C126>0,$E$118-C126,0)	=E126-$H$118	=C126+E126	=D126+F126
127		=B128-5	=B127	=C127-$H$119	=IF($E$118-C127>0,$E$118-C127,0)	=E127-$H$118	=C127+E127	=D127+F127
128		=E118	=B128	=C128-$H$119	=IF($E$118-C128>0,$E$118-C128,0)	=E128-$H$118	=C128+E128	=D128+F128
129		=B128+5	=B129	=C129-$H$119	=IF($E$118-C129>0,$E$118-C129,0)	=E129-$H$118	=C129+E129	=D129+F129
130		=B129+5	=B130	=C130-$H$119	=IF($E$118-C130>0,$E$118-C130,0)	=E130-$H$118	=C130+E130	=D130+F130
131		=B130+5	=B131	=C131-$H$119	=IF($E$118-C131>0,$E$118-C131,0)	=E131-$H$118	=C131+E131	=D131+F131
132		=B131+5	=B132	=C132-$H$119	=IF($E$118-C132>0,$E$118-C132,0)	=E132-$H$118	=C132+E132	=D132+F132
133		=B132+5	=B133	=C133-$H$119	=IF($E$118-C133>0,$E$118-C133,0)	=E133-$H$118	=C133+E133	=D133+F133
134								

Fig. 9.17 Excel formula for calculating the value of a protective put position at option expiration

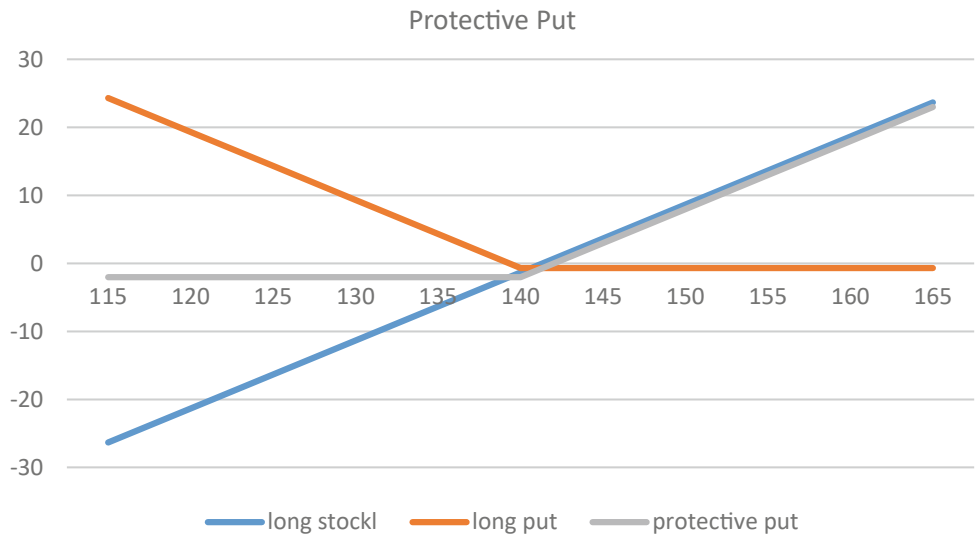
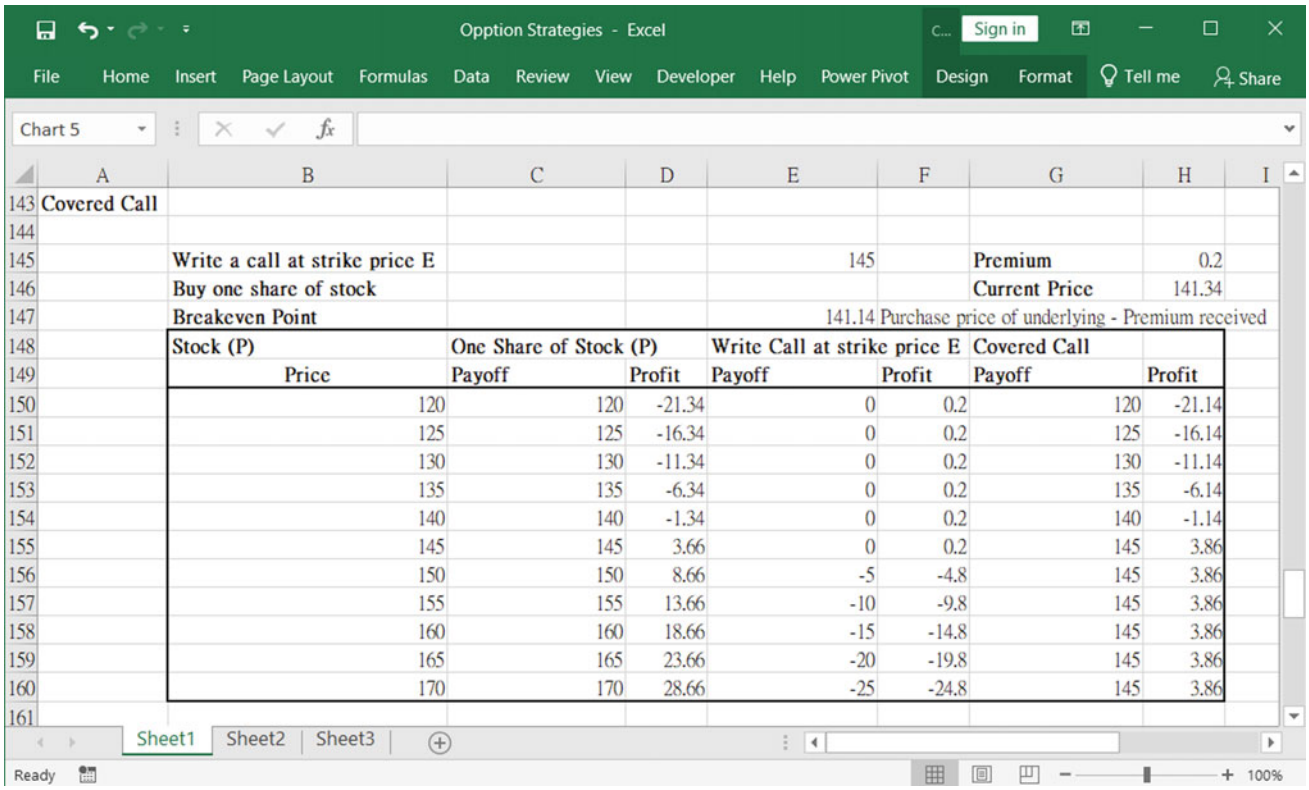


Fig. 9.18 Profit profile for protective put

underlying + Premium Received). For this example, the break-even point is $149.17.

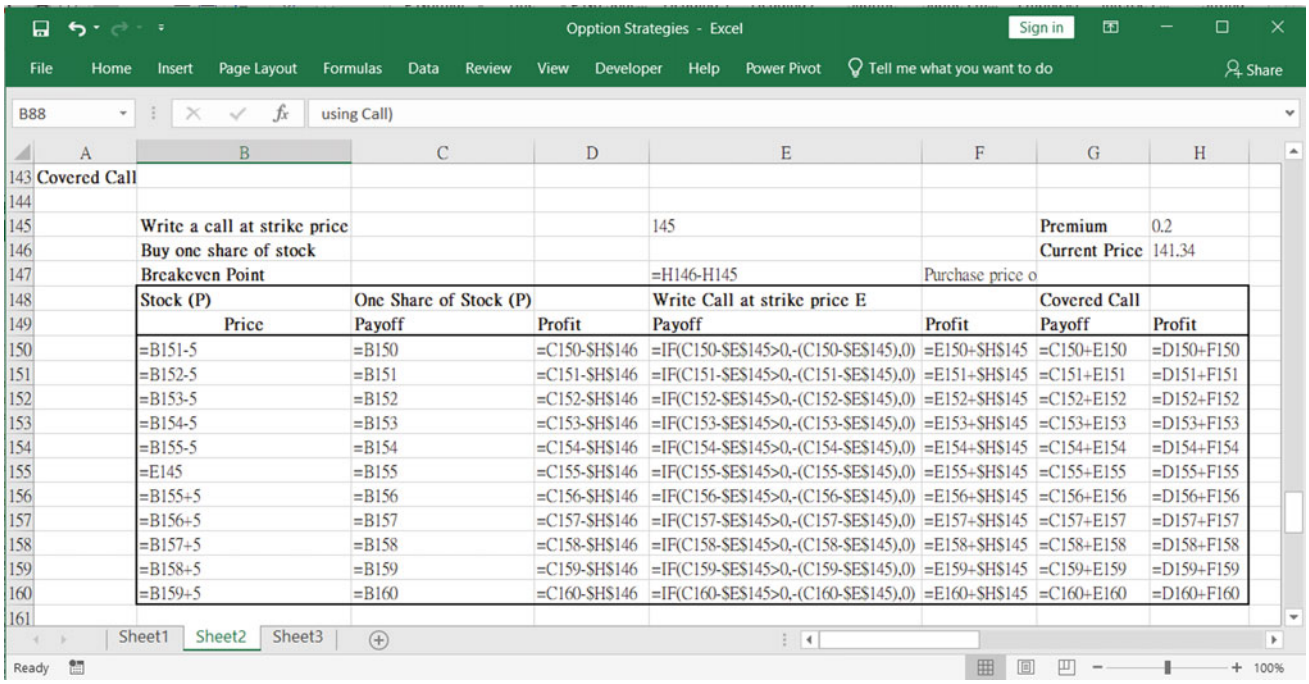
9.4.7 Collar

A collar combines a protective put and a short call option to bracket the value of a portfolio between two bounds. For example, an investor holds the IBM stock selling at $151.10. Buying a protective put using the put option with an exercise price of $150 places a lower bound of $150 on the value of the portfolio. At the same time, the investor can write a call option with an exercise price of $155. You can find the S_T, which is the value for S_T in the first column of the table in Fig. 9.22. The call and the put sell at $1.97 (the bid price) and $4.40 (the ask price), respectively, making the net outlay for the two options to be only $2.43. Figure 9.22 shows the values of the collar position at different stock prices at time



	A	B	C	D	E	F	G	H	I
143	Covered Call								
144									
145		Write a call at strike price E			145		Premium	0.2	
146		Buy one share of stock					Current Price	141.34	
147		Breakeven Point			141.14	Purchase price of underlying - Premium received			
148		Stock (P)	One Share of Stock (P)		Write Call at strike price E		Covered Call		
149		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	
150		120	120	-21.34	0	0.2	120	-21.14	
151		125	125	-16.34	0	0.2	125	-16.14	
152		130	130	-11.34	0	0.2	130	-11.14	
153		135	135	-6.34	0	0.2	135	-6.14	
154		140	140	-1.34	0	0.2	140	-1.14	
155		145	145	3.66	0	0.2	145	3.86	
156		150	150	8.66	-5	-4.8	145	3.86	
157		155	155	13.66	-10	-9.8	145	3.86	
158		160	160	18.66	-15	-14.8	145	3.86	
159		165	165	23.66	-20	-19.8	145	3.86	
160		170	170	28.66	-25	-24.8	145	3.86	
161									

Fig. 9.19 Value of a covered call position at option expiration



	A	B	C	D	E	F	G	H
143	Covered Call							
144								
145		Write a call at strike price			145		Premium	0.2
146		Buy one share of stock					Current Price	141.34
147		Breakeven Point			=H146-H145	Purchase price o		
148		Stock (P)	One Share of Stock (P)		Write Call at strike price E		Covered Call	
149		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit
150		=B151-5	=B150	=C150-$H$146	=IF(C150-$E$145>0,-(C150-$E$145),0)	=E150+$H$145	=C150+E150	=D150+F150
151		=B152-5	=B151	=C151-$H$146	=IF(C151-$E$145>0,-(C151-$E$145),0)	=E151+$H$145	=C151+E151	=D151+F151
152		=B153-5	=B152	=C152-$H$146	=IF(C152-$E$145>0,-(C152-$E$145),0)	=E152+$H$145	=C152+E152	=D152+F152
153		=B154-5	=B153	=C153-$H$146	=IF(C153-$E$145>0,-(C153-$E$145),0)	=E153+$H$145	=C153+E153	=D153+F153
154		=B155-5	=B154	=C154-$H$146	=IF(C154-$E$145>0,-(C154-$E$145),0)	=E154+$H$145	=C154+E154	=D154+F154
155		=E145	=B155	=C155-$H$146	=IF(C155-$E$145>0,-(C155-$E$145),0)	=E155+$H$145	=C155+E155	=D155+F155
156		=B155+5	=B156	=C156-$H$146	=IF(C156-$E$145>0,-(C156-$E$145),0)	=E156+$H$145	=C156+E156	=D156+F156
157		=B156+5	=B157	=C157-$H$146	=IF(C157-$E$145>0,-(C157-$E$145),0)	=E157+$H$145	=C157+E157	=D157+F157
158		=B157+5	=B158	=C158-$H$146	=IF(C158-$E$145>0,-(C158-$E$145),0)	=E158+$H$145	=C158+E158	=D158+F158
159		=B158+5	=B159	=C159-$H$146	=IF(C159-$E$145>0,-(C159-$E$145),0)	=E159+$H$145	=C159+E159	=D159+F159
160		=B159+5	=B160	=C160-$H$146	=IF(C160-$E$145>0,-(C160-$E$145),0)	=E160+$H$145	=C160+E160	=D160+F160
161								

Fig. 9.20 Excel formula for calculating the value of a covered call position at option expiration

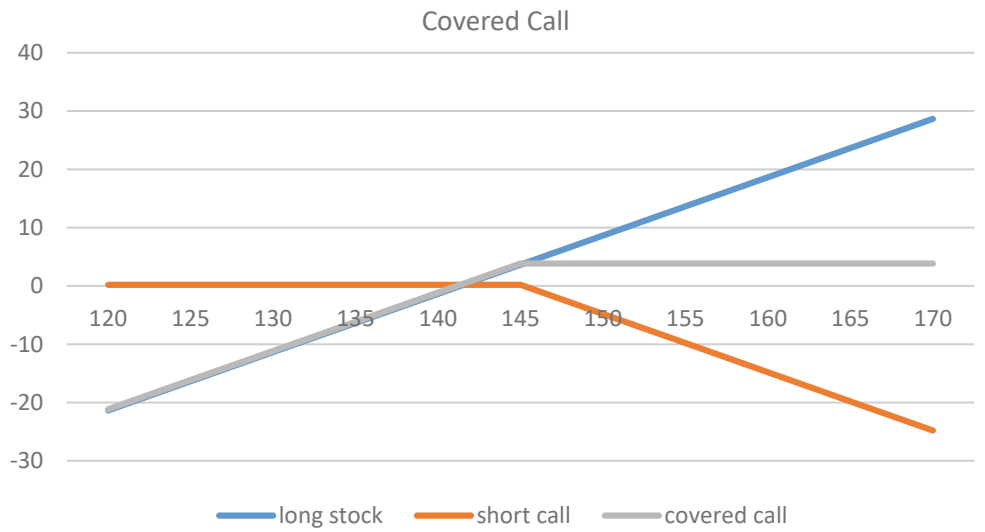
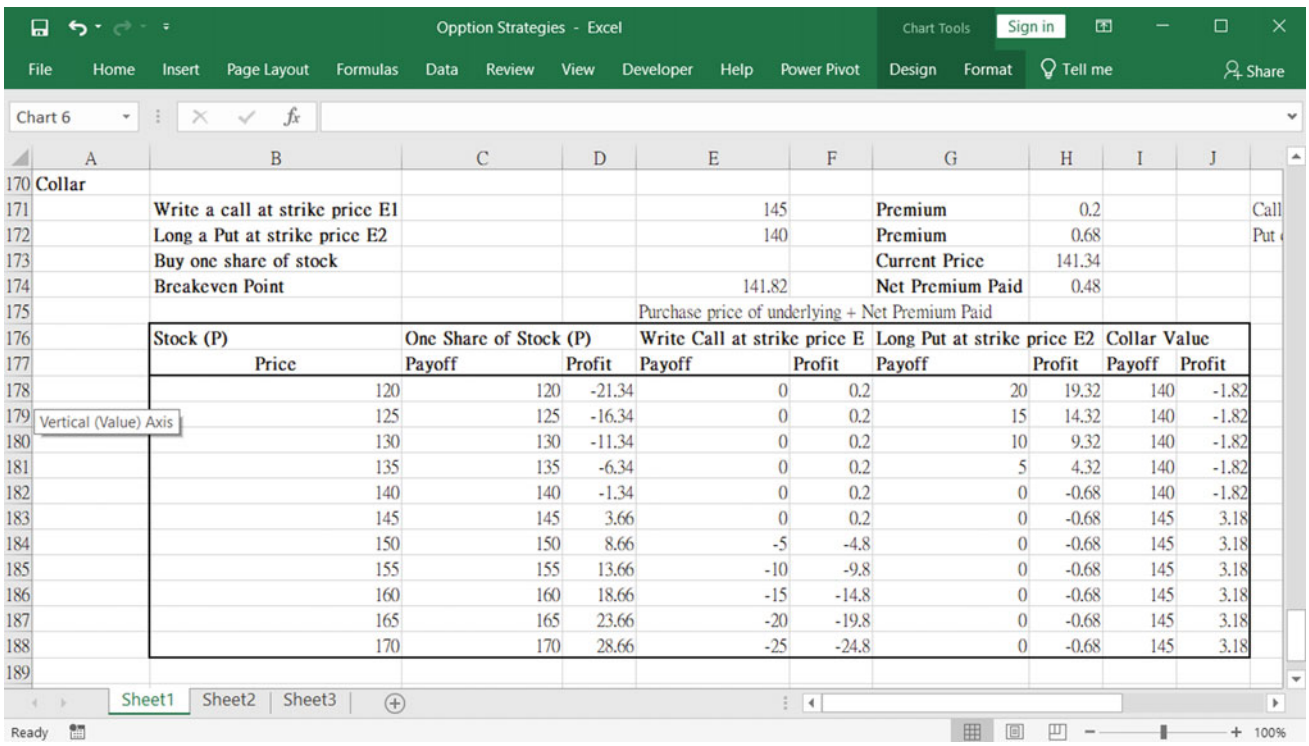
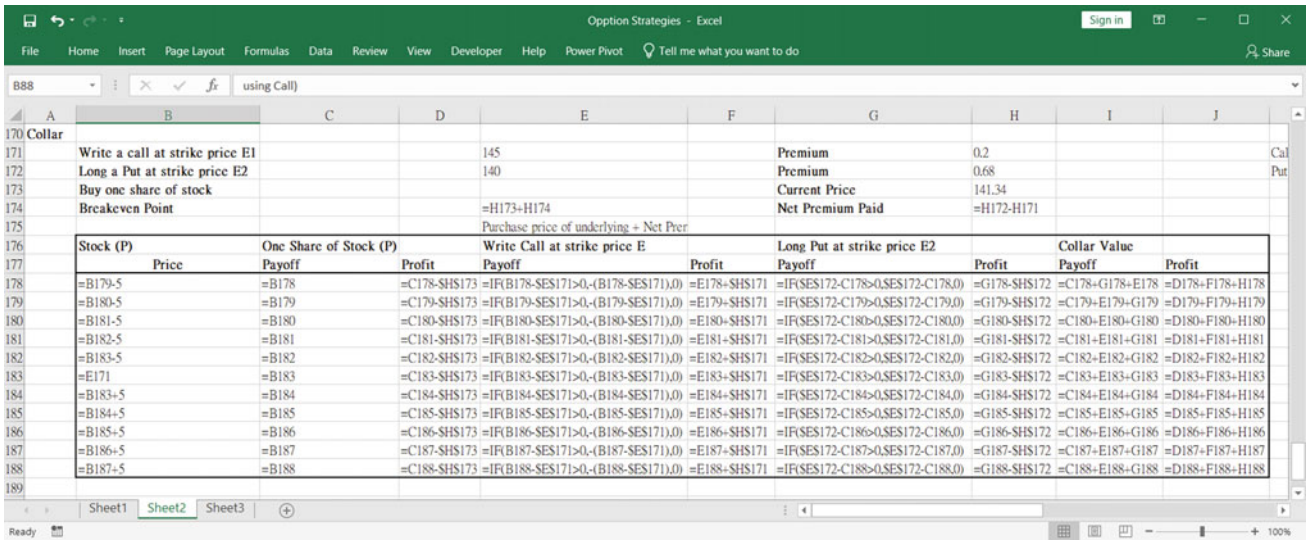


Fig. 9.21 Profit profile for covered call



	A	B	C	D	E	F	G	H	I	J
170	Collar									
171		Write a call at strike price E1			145		Premium	0.2		Call
172		Long a Put at strike price E2			140		Premium	0.68		Put
173		Buy one share of stock					Current Price	141.34		
174		Breakeven Point			141.82		Net Premium Paid	0.48		
175					Purchase price of underlying + Net Premium Paid					
176		Stock (P)	One Share of Stock (P)		Write Call at strike price E		Long Put at strike price E2		Collar Value	
177		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	Payoff	Profit
178		120	120	-21.34	0	0.2	20	19.32	140	-1.82
179		125	125	-16.34	0	0.2	15	14.32	140	-1.82
180		130	130	-11.34	0	0.2	10	9.32	140	-1.82
181		135	135	-6.34	0	0.2	5	4.32	140	-1.82
182		140	140	-1.34	0	0.2	0	-0.68	140	-1.82
183		145	145	3.66	0	0.2	0	-0.68	145	3.18
184		150	150	8.66	-5	-4.8	0	-0.68	145	3.18
185		155	155	13.66	-10	-9.8	0	-0.68	145	3.18
186		160	160	18.66	-15	-14.8	0	-0.68	145	3.18
187		165	165	23.66	-20	-19.8	0	-0.68	145	3.18
188		170	170	28.66	-25	-24.8	0	-0.68	145	3.18
189										

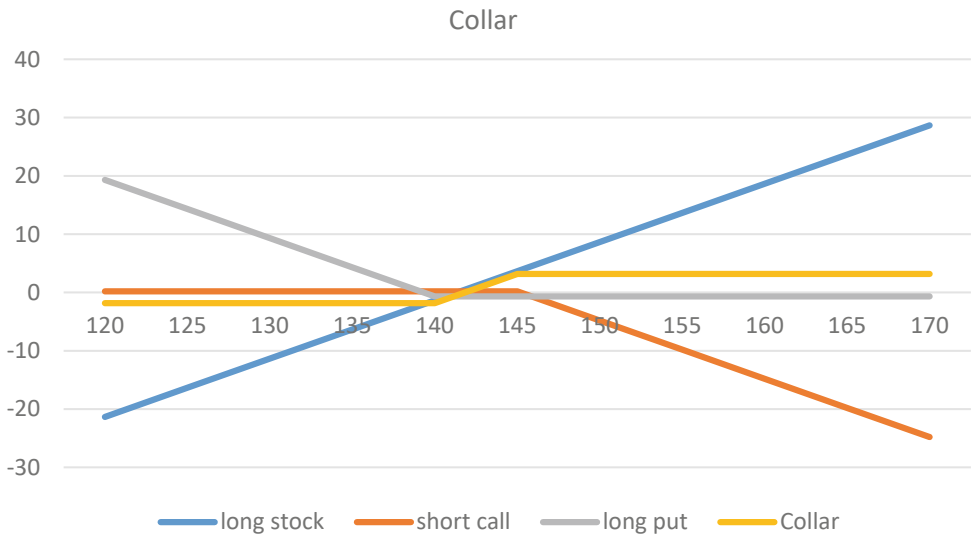
Fig. 9.22 Value of a collar position at option expiration



	A	B	C	D	E	F	G	H	I	J
170	Collar									
171		Write a call at strike price E1			145		Premium	0.2		Cal
172		Long a Put at strike price E2			140		Premium	0.68		Put
173		Buy one share of stock					Current Price	141.34		
174		Breakeven Point			=H173+H174		Net Premium Paid	=H172-H171		
175					Purchase price of underlying + Net Prem					
176		Stock (P)	One Share of Stock (P)		Write Call at strike price E		Long Put at strike price E2		Collar Value	
177		Price	Payoff	Profit	Payoff	Profit	Payoff	Profit	Payoff	Profit
178		=B179-5	=B178	=C178-$H$173	=IF(B178-$E$171>0,-(B178-$E$171),0)	=E178+$H$171	=IF($E$172-C178>0,$E$172-C178,0)	=G178-$H$172	=C178+G178+E178	=D178+F178+H178
179		=B180-5	=B179	=C179-$H$173	=IF(B179-$E$171>0,-(B179-$E$171),0)	=E179+$H$171	=IF($E$172-C179>0,$E$172-C179,0)	=G179-$H$172	=C179+E179+G179	=D179+F179+H179
180		=B181-5	=B180	=C180-$H$173	=IF(B180-$E$171>0,-(B180-$E$171),0)	=E180+$H$171	=IF($E$172-C180>0,$E$172-C180,0)	=G180-$H$172	=C180+E180+G180	=D180+F180+H180
181		=B182-5	=B181	=C181-$H$173	=IF(B181-$E$171>0,-(B181-$E$171),0)	=E181+$H$171	=IF($E$172-C181>0,$E$172-C181,0)	=G181-$H$172	=C181+E181+G181	=D181+F181+H181
182		=B183-5	=B182	=C182-$H$173	=IF(B182-$E$171>0,-(B182-$E$171),0)	=E182+$H$171	=IF($E$172-C182>0,$E$172-C182,0)	=G182-$H$172	=C182+E182+G182	=D182+F182+H182
183		=E171	=B183	=C183-$H$173	=IF(B183-$E$171>0,-(B183-$E$171),0)	=E183+$H$171	=IF($E$172-C183>0,$E$172-C183,0)	=G183-$H$172	=C183+E183+G183	=D183+F183+H183
184		=B183+5	=B184	=C184-$H$173	=IF(B184-$E$171>0,-(B184-$E$171),0)	=E184+$H$171	=IF($E$172-C184>0,$E$172-C184,0)	=G184-$H$172	=C184+E184+G184	=D184+F184+H184
185		=B184+5	=B185	=C185-$H$173	=IF(B185-$E$171>0,-(B185-$E$171),0)	=E185+$H$171	=IF($E$172-C185>0,$E$172-C185,0)	=G185-$H$172	=C185+E185+G185	=D185+F185+H185
186		=B185+5	=B186	=C186-$H$173	=IF(B186-$E$171>0,-(B186-$E$171),0)	=E186+$H$171	=IF($E$172-C186>0,$E$172-C186,0)	=G186-$H$172	=C186+E186+G186	=D186+F186+H186
187		=B186+5	=B187	=C187-$H$173	=IF(B187-$E$171>0,-(B187-$E$171),0)	=E187+$H$171	=IF($E$172-C187>0,$E$172-C187,0)	=G187-$H$172	=C187+E187+G187	=D187+F187+H187
188		=B187+5	=B188	=C188-$H$173	=IF(B188-$E$171>0,-(B188-$E$171),0)	=E188+$H$171	=IF($E$172-C188>0,$E$172-C188,0)	=G188-$H$172	=C188+E188+G188	=D188+F188+H188
189										

Fig. 9.23 Excel formula for calculating the value of a collar position at option expiration

Fig. 9.24 Profit profile for collar



T. For information in detail, you can find the excel function in Fig. 9.23 for calculations of the numbers in Fig. 9.22. The profit profile of the collar position is shown in Fig. 9.24. The Break-even point means when the profit equals to zero. The Break-even point for Collar can be calculated as (Purchase Price of Underlying + Net Premium Paid). For this example, the break-even point is $153.57.

9.5 Summary

In this chapter, we have shown how excel programs can be used to calculate the optimal weights in terms of the Markowitz portfolio model. In addition, we also show how excel programs can use to do alternative options strategies.

Appendix 9.1: Monthly Rates of Returns for S&P500, IBM, and MSFT

Date	S&P500 (%)	IBM (%)	MSFT (%)
2016/2/1	−0.41	5.00	−7.64
2016/3/1	6.60	16.76	9.33
2016/3/31	0.27	−3.64	−9.70
2016/4/30	1.53	5.34	6.28
2016/5/31	0.09	0.63	−2.78
2016/6/30	3.56	5.82	10.77
2016/7/31	−0.12	−1.08	1.38
2016/8/31	−0.12	0.84	0.87
2016/9/30	−1.94	−3.25	4.03
2016/10/31	3.42	5.55	0.57
2016/12/1	1.82	3.25	3.82
2017/1/1	1.79	5.14	4.04
2017/2/1	3.72	3.04	−1.04
2017/3/1	−0.04	−2.39	3.56
2017/3/31	0.91	−7.95	3.95
2017/4/30	1.16	−4.78	2.02
2017/5/31	0.48	1.77	−0.74
2017/6/30	1.93	−5.95	5.47
2017/7/31	0.05	−1.13	2.85
2017/8/31	1.93	2.50	0.16
2017/9/30	2.22	6.19	11.67
2017/10/31	0.37	−3.21	1.19
2017/12/1	3.43	3.91	2.14
2018/1/1	5.62	6.70	11.07
2018/2/1	−3.89	−4.81	−1.31
2018/3/1	−2.69	−0.57	−2.21
2018/3/31	0.27	−5.52	2.47
2018/4/30	2.16	−2.52	5.69
2018/5/31	0.48	−0.04	0.20
2018/6/30	3.60	3.74	7.58
2018/7/31	3.03	1.07	5.89
2018/8/31	0.43	4.34	2.21
2018/9/30	−6.94	−23.66	−6.61
2018/10/31	1.79	7.66	3.82
2018/12/1	−9.18	−7.36	−8.01
2019/1/1	7.87	18.25	2.82
2019/2/1	2.97	2.76	7.28
2019/3/1	1.79	3.34	5.72
2019/3/31	3.93	−0.59	10.73
2019/4/30	−6.58	−9.47	−5.30
2019/5/31	6.89	9.88	8.71
2019/6/30	1.31	7.50	1.72
2019/7/31	−1.81	−8.57	1.17
2019/8/31	1.72	8.56	1.18
2019/9/30	2.04	−8.04	3.12
2019/10/31	3.40	0.54	5.59
2019/12/1	2.86	0.87	4.53
2020/1/1	−0.16	7.23	7.95
2020/2/1	−8.41	−9.45	−4.83
2020/3/1	−12.51	−13.88	−2.39
2020/3/31	12.68	13.19	13.63
2020/4/30	4.53	−0.53	2.25
2020/5/31	1.84	−2.01	11.37
2020/6/30	5.51	1.80	0.74
2020/7/31	7.01	0.30	10.01
2020/8/31	−3.92	−0.04	−6.51
2020/9/30	−2.77	−8.23	−3.74
2020/10/31	10.75	10.62	5.73
2020/12/1	3.71	3.39	4.17

Appendix 9.2: Options Data for IBM (Stock Price = 141.34) on July 23, 2021

Contract name	Strike	Last price	Bid	Ask	Change	% Change (%)	Volume	Open interest	Implied volatility
IBM210730C00139000	139	2.79	2.64	2.94	0.06	+2.20	10	242	0.2073
IBM210730C00140000	140	2.04	1.98	2.16	0.39	+23.64	601	777	0.1929
IBM210730C00141000	141	1.44	1.39	1.47	0.26	+22.03	1,199	477	0.179
IBM210730C00142000	142	0.94	0.89	1.07	0.14	+17.50	997	601	0.1897
IBM210730C00143000	143	0.61	0.54	0.59	0.13	+27.08	291	437	0.1716
IBM210730C00144000	144	0.32	0.32	0.37	0.05	+18.52	437	739	0.1763
IBM210730C00145000	145	0.2	0.17	0.2	0.03	+17.65	616	1066	0.1738
IBM210730C00146000	146	0.11	0.1	0.12	0.02	+22.22	254	585	0.1797
IBM210730C00147000	147	0.07	0.06	0.08	−0.02	−22.22	65	252	0.1904
IBM210730C00148000	148	0.05	0.04	0.06	0	–	40	515	0.2041
IBM210730C00149000	149	0.05	0.03	0.05	0	–	9	132	0.2207
IBM210730C00150000	150	0.04	0.03	0.04	0.01	+33.33	82	1161	0.2344
IBM210730C00152500	152.5	0.03	0.02	0.03	−0.01	−25.00	34	690	0.2774
IBM210730C00155000	155	0.02	0.02	0.03	0	–	25	328	0.3262
IBM210730C00157500	157.5	0.02	0.02	0.03	−0.01	−33.33	2	961	0.375
IBM210730C00160000	160	0.02	0.01	0.03	0	–	66	138	0.4219
IBM210730C00162500	162.5	0.01	0.01	0.16	−0.04	−80.00	3	75	0.5391
IBM210730C00165000	165	0.01	0	0.02	−0.02	−66.67	6	50	0.4844
IBM210730P00125000	125	0.02	0	0	0	–	18	0	0.25
IBM210730P00128000	128	0.02	0	0	0	–	39	0	0.25
IBM210730P00129000	129	0.06	0	0	0	–	6	0	0.25
IBM210730P00130000	130	0.03	0	0	0	–	74	0	0.125
IBM210730P00131000	131	0.04	0	0	0	–	17	0	0.125
IBM210730P00132000	132	0.05	0	0	0	–	17	0	0.125
IBM210730P00133000	133	0.06	0	0	0	–	88	0	0.125
IBM210730P00134000	134	0.07	0	0	0	–	11	0	0.125
IBM210730P00135000	135	0.09	0	0	0	–	95	0	0.125
IBM210730P00136000	136	0.12	0	0	0	–	89	0	0.0625
IBM210730P00137000	137	0.14	0	0	0	–	70	0	0.0625
IBM210730P00138000	138	0.25	0	0	0	–	390	0	0.0625
IBM210730P00139000	139	0.41	0	0	0	–	193	0	0.0313
IBM210730P00140000	140	0.68	0	0	0	–	431	0	0.0313
IBM210730P00141000	141	0.97	0	0	0	–	284	0	0.0078
IBM210730P00142000	142	1.64	0	0	0	–	85	0	0
IBM210730P00143000	143	2.12	0	0	0	–	37	0	0
IBM210730P00144000	144	2.87	0	0	0	–	207	0	0
IBM210730P00145000	145	3.87	0	0	0	–	17	0	0
IBM210730P00146000	146	4.73	0	0	0	–	33	0	0
IBM210730P00147000	147	6.13	0	0	0	–	2	0	0
IBM210730P00148000	148	6.75	0	0	0	–	2	0	0

(continued)

(continued)

Contract name	Strike	Last price	Bid	Ask	Change	% Change (%)	Volume	Open interest	Implied volatility
IBM210730P00149000	149	8.14	0	0	0	–	1	0	0
IBM210730P00150000	150	8.68	0	0	0	–	10	0	0
IBM210730P00152500	152.5	11.25	0	0	0	–	10	0	0

References

Alexander, G. J. and J. C. Francis. *Portfolio Analysis.* New York: Prentice-Hall, Inc., 1986.

Amram, M. and N. Kulatilaka. *Real Options*. New York: Oxford University Press, 2001.

Ball, C. and W. Torous. "Bond Prices Dynamics and Options." *Journal of Financial and Quantitative Analysis*, v. 18 (December 1983), pp. 517–532.

Baumol, W. J. "An Expected Gain-Confidence Limit Criterion for Portfolio Selection." *Management Science*, v. 10 (October 1963), pp. 171–182.

Bertsekas, D. "Necessary and Sufficient Conditions for Existence of an Optimal Portfolio." *Journal of Economic Theory*, v. 8 (June 1974), pp. 235–247.

Bhattacharya, M. "Empirical Properties of the Black–Scholes Formula under Ideal Conditions." *Journal of Financial and Quantitative Analysis*, v. 15 (December 1980), pp. 1081–1106.

Black, F. "Capital Market Equilibrium with Restricted Borrowing." *Journal of Business*, v. 45 (July 1972a), pp. 444–455.

Black, F. "Capital Market Equilibrium with Restricted Borrowing." *Journal of Business*, v. 45 (July 1972b), pp. 444–445.

Black, F. "Fact and Fantasy in the Use of Options." *Financial Analysts Journal*, v. 31 (July/August 1985), pp. 36–72.

Black, F. and M. Scholes. "The Pricing of Options and Corporate Liabilities." *Journal of Political Economy*, v. 31 (May/June 1973), pp. 637–654.

Blume, M. "Portfolio Theory: A Step toward Its Practical Application." *Journal of Business*, v. 43 (April 1970), pp. 152–173.

Bodhurta, J. and G. Courtadon. "Efficiency Tests of the Foreign Currency Options Market." *Journal of Finance*, v. 41 (March 1986), pp. 151–162.

Bodie, Z., A. Kane and A. Marcus. *Investments*, 9th ed. New York: McGraw-Hill Book Company, 2010.

Bookstaber, R. M. *Option Pricing and Strategies in Investing*. Reading, MA: Addison-Wesley Publishing Company, 1981.

Bookstaber, R. M., and R. Clarke. *Option Strategies for Institutional Investment Management.* Reading, MA: Addison-Wesley Publishing Company, 1983.

Brealey, R. A. and S. D. Hodges. "Playing with Portfolios." *Journal of Finance*, v. 30 (March 1975), pp. 125–134.

Breen, W. and R. Jackson. "An Efficient Algorithm for Solving Large-Scale Portfolio Problems." *Journal of Financial and Quantitative Analysis*, v. 6 (January 1971), pp. 627–637.

Brennan, M. and E. Schwartz. "The Valuation of American Put Options." *Journal of Finance*, v. 32 (May 1977), pp. 449–462.

Brennan, M. J. "The Optimal Number of Securities in a Risky Asset Portfolio Where There are Fixed Costs of Transaction: Theory and Some Empirical Results." *Journal of Financial and Quantitative Analysis*, v. 10 (September 1975), pp. 483–496.

Cohen, K. and J. Pogue. "An Empirical Evaluation of Alter native Portfolio-Selection Models." *Journal off Business*, v. 46 (April 1967), pp. 166–193.

Cox, J. C. "Option Pricing: A Simplified Approach." *Journal of Financial Economics*, v. 8 (September 1979), pp. 229–263.

Cox, J. C. and M. Rubinstein. *Option Markets.* Englewood Cliffs, NJ: Prentice-Hall, 1985.

Dyl, E. A. "Negative Betas: The Attractions of Selling Short." *Journal of Portfolio Management*, v. I (Spring 1975), pp. 74–76.

Eckardt, W. and S. Williams. "The Complete Options Indexes." *Financial Analysts Journal*, v. 40 (July/August 1984), pp. 48–57.

Elton, E. J. and M. E. Padberg. "Simple Criteria for Optimal Portfolio Selection." *Journal of Finance*, v.11 (December 1976), pp. 1341–1357.

Elton, E. J. and M. E. Padberg. "Simple Criteria for Optimal Portfolio Selection: Tracing Out the Efficient Frontier." *Journal of Finance*, v. 13 (March 1978), pp. 296–302.

Elton, E. J. and Martin Gruber. "Portfolio Theory When Investment Relatives are Log Normally Distributed." *Journal of Finance*, v. 29 (September 1974), pp. 1265–1273.

Elton, E. J., M. J. Gruber, S. J. Brown and W. N. Goetzmann. *Modern Portfolio Theory and Investment Analysis*, 7th ed. New York: John Wiley & Sons, 2006.

Ervine, J. and A. Rudd. "Index Options: The Early Evidence." *Journal of Finance*, v. 40 (June 1985), pp. 743–756.

Evans, J. and S. Archer. "Diversification and the Reduction of Dispersion: An Empirical Analysis." *Journal of Finance*, v. 3 (December 1968), pp. 761–767.

Fama, E. F. "Efficient Capital Markets: A Review of Theory and Empirical Work." *Journal of Finance*, v. 25 (May 1970), pp. 383–417.

Feller, W. *An Introduction to Probability Theory and Its Application*, Vol. 1. New York: John Wiley and Sons, Inc., 1968.

Finnerty, J. "The Chicago Board Options Exchange and Market Efficiency." *Journal of Financial and Quantitative Analysis*, v. 13 (March 1978), pp. 28–38.

Francis, J. C. and S. H. Archer. *Portfolio Analysis.* New York: Prentice-Hall, Inc., 1979.

Galai, D. and R. W. Masulis. "The Option Pricing Model and the Risk Factor of Stock." *Journal of Financial Economics*, v. 3 (March 1976), pp. 53–81.

Galai, D., R. Geske and S. Givots. *Option Markets.* Reading, MA: Addison-Wesley Publishing Company, 1988.

Gastineau, G. *The Stock Options Manual.* New York: McGraw-Hill, 1979.

Geske, R. and K. Shastri. "Valuation by Approximation: A Comparison of Alternative Option Valuation Techniques." *Journal of Financial and Quantitative Analysis*, v. 20 (March 1985), pp. 45–72.

Gressis, N., G. Philiippatos and J. Hayya. "Multiperiod Portfolio Analysis and the Inefficiencies of the Market Portfolio." *Journal of Finance*, v. 31 (September 1976), pp. 1115–1126.

Guerard, J. B. Handbook of Portfolio and Construction: Contemporary Applications of Markowitz Techniques. New York: Springer, 2010.

Henderson, J. and R. Quandt. *Microeconomic Theory: A Mathematical Approach,* 3rd ed. New York: McGraw-Hill, 1980.

Hull, J. *Options, Futures, and Other Derivatives*, 6th ed. Upper Saddle. River, New Jersey: Prentice Hall, 2005.

Jarrow R. and S. Turnbull. *Derivatives Securities,* 2nd ed. Cincinnati, OH: South-Western College Pub, 1999.

Jarrow, R. A. and A. Rudd. *Option Pricing*. Homewood, IL: Richard D. Irwin, 1983.

Lee, C. F. and A. C. Lee. *Encyclopedia of Finance*. New York: Springer, 2006.

Lee, C. F. and Alice C. Lee, *Encyclopedia of Finance.* New York, NY: Springer, 2006.

Lee, C. F. *Handbook of Quantitative Finance and Risk Management.* New York, NY: Springer, 2009.

Lee, C. F., A. C. Lee and J. C. Lee . *Handbook of Quantitative Finance and Risk Management*. New York: Springer, 2010.

Lee, C. F., J. C. Lee and A. C. Lee. *Statistics for Business and Financial Economics*. Singapore: World Scientific Publishing Co., 2013.

Levy, H. and M. Sarnat. "A Note on Portfolio Selection and Investors' Wealth." *Journal of Financial and Quantitative Analysis*, v. 6 (January 1971), pp. 639–642.

Lewis, A. L. "A Simple Algorithm for the Portfolio Selection Problem." *Journal of Finance*, v. 43 (March 1988), pp. 71–82.

Liaw, K. T. and R. L. Moy, *The Irwin Guide to Stocks, Bonds, Futures, and Options*,.New York: McGraw-Hill Co., 2000.

Lintner, J. "The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolio and Capital Budgets." *Review of Economics and Statistics*, v. 47 (February 1965), pp. 13–27.

Macbeth, J. and L. Merville. "An Empirical Examination of the Black–Scholes Call Option Pricing Model." *Journal of Finance*, v. 34 (December 1979), pp. J173–J186.

Maginn, J. L., D. L. Tuttle, J. E. Pinto and D. W. McLeavey. *Managing Investment Portfolios: A Dynamic Process*, CFA Institute Investment Series, 3rd ed. New York: John Wiley & Sons, 2007.

Mao, J. C. F. *Quantitative Analysis of Financial Decisions.* New York: Macmillan, 1969.

Markowitz, H. M. "Markowitz Revisited." *Financial Analysts Journal*, v. 32 (September/October 1976), pp. 47–52.

Markowitz, H. M. "Portfolio Selection." *Journal of Finance*, v. 1 (December 1952), pp. 77–91.

Markowitz, H. M. Mean-Variance Analysis in Portfolio Choice and Capital Markets. New York: Blackwell, 1987.

Markowitz, H. M. *Portfolio Selection.* Cowles Foundation Monograph 16. New York: John Wiley and Sons, Inc., 1959.

Martin, A. D., Jr. "Mathematical Programming of Portfolio Selections." Management Science, v. 1 (1955), pp. 152–166.

McDonald, R. L. *Derivatives Markets*, 2nd ed. Boston, MA: Addison Wesley, 2005.

Merton, R. "An Analytical Derivation of Efficient Portfolio Frontier." *Journal of Financial and Quantitative Analysis*, v. 7 (September 1972), pp. 1851–1872.

Merton, R. "Theory of Rational Option Pricing." *Bell Journal of Economics and Management Science*, v. 4 (Spring 1973), pp. 141–183.

Mossin, J. "Optimal Multiperiod Portfolio Policies." *Journal of Business*, v.41 (April 1968), pp. 215–229.

Rendleman, R. J. Jr. and B. J. Barter. "Two-State Option Pricing." *Journal of Finance*, v. 34 (September 1979), pp. 1093–1110.

Ritchken, P. *Options: Theory, Strategy and Applications.* Glenview, IL: Scott, Foresman, 1987.

Ross, S. A. "On the General Validity of the Mean-Variance Approach in Large Markets," in W. F. Sharpe and C. M. Cootner, *Financial Economics: Essays in Honor of Paul Cootner*, pp. 52–84. New York: PrenticeHall, Inc., 1982.

Rubinstein, M. and H. Leland. "Replicating Options with Positions in Stock and Cash." *Financial Analysts Journal*, v. 37 (July/August 1981), pp.63–72.

Rubinstein, M. and J. Cox. *Option Markets.* Englewood Cliffs, NJ: Prentice-Hall, 1985.

Sears, S. and G. Trennepohl. "Measuring Portfolio Risk in Options." *Journal of Financial and Quantitative Analysis*, v. 17 (September 1982), pp.391–410.

Sharpe, W. F. *Portfolio Theory and Capital Markets.* New York: McGraw-Hill, 1970.

Simkowitz, M. A. and W. L. Beedles. "Diversitifcation in a Three-Moment World." *Journal of Finance and Quantitative Analysis*, v. 13 (1978), pp. 927–941.

Smith, C. "Option Pricing: A Review." *Journal of Financial Economics*, v. 3 (January 1976), pp. 3–51.

Stoll, H. "The Relationships between Put and Call Option Prices." *Journal of Finance*, v. 24 (December 1969), pp. 801–824.

Summa, J. F. and J. W. Lubow, *Options on Futures.* New York: John Wiley & Sons, 2001.

Trennepohl, G. "A Comparison of Listed Option Premium and Black–Scholes Model Prices: 1973–1979." *Journal of Financial Research*, v. 4 (Spring 1981), pp. 11–20.

Von Neumann, J. and O. Morgenstern. *Theory of Games and Economic Behavior*, 2nd ed. Princeton, NJ: Princeton University Press, 1947.

Wackerly, D., W. Mendenhall and R. L. Scheaffer. *Mathematical Statistics with Applications*, 7th ed. California: Duxbury Press, 2007.

Weinstein, M. "Bond Systematic Risk and the Options Pricing Model." *Journal of Finance*, v. 38 (December 1983), pp. 1415–1430.

Welch, W. *Strategies for Put and Call Option Trading*. Cambridge, MA: Winthrop, 1982.

Whaley, R. "Valuation of American Call Options on Dividend Paying Stocks: Empirical Tests." *Journal of Financial Economics*, v. 10 (March 1982), pp. 29–58.

Zhang, P. G., *Exotic Options: A Guide to Second Generation Options*, 2nd ed. Singapore: World Scientific, 1998.

Simulation and Its Application

10

10.1 Introduction

In this chapter, we will introduce Monte Carlo simulation which is a problem-solving technique. This technique can approximate the probability of certain outcomes by using random variables, called simulations. Monte Carlo simulation is named after the city in Monaco. The primary attractions in this place are casinos that have gambling games, like dice, roulette, and slot machines. These games of chance exist in random behavior.

In option pricing methods, we can use Monte Carlo simulation to generate the underlying asset price process, then to value today's option price. At first, we will introduce how to use excel to simulate stock price and get the option price. Next, we also introduce different methods to improve the efficiency of the simulation. These include antithetic variates and Quasi-Monte Carlo simulation. Finally, we apply Monte Carlo simulation to the path-depend option.

This chapter can be broken down into the following sections. In Sect. 10.2, we discuss Monte Carlo simulation; in Sect. 10.3, we discuss antithetic variates; and in Sect. 10.4, we discuss Quasi-Monte Carlo simulation. In Sect. 10.5, we discuss the applications, and finally, in Sect. 10.6, we summarize the chapter.

10.2 Monte Carlo Simulation

The advantages of Monte Carlo simulation are its generality and relative ease to use. For instance, it may take many complicating features of exotic options into account and it lends itself to treating high-dimensional problems. However, it is difficult to apply simulation to American options. Simulation goes forward in time, but establishing an optimal exercise policy requires going backward in time.

At first, we generate asset price paths by Monte Carlo simulation. For convenience, we recall the geometric Brownian motion for the asset price. Geometric Brownian motion is the standard assumption for the stock price process. This stock process is plausibly explained in John Hull textbook. Mathematically speaking, the asset price S(t), with drift μ and volatility σ:

$$\mathrm{dS} = \mu \mathrm{Sdt} + \sigma \mathrm{Sdz},$$

where dz is Brownian motion, ε is the standard normal random variable, dt is in a very short time, and δt can be any time period.

Using Ito's lemma, we can get the stochastic process under a logarithm stock price:

$$\mathrm{dlnS} = \left(\mu - 0.5\sigma^2\right)dt + \sigma dz.$$

Because there is no stock price in the drift and diffusion term, we can discretize the time period and get the stock price process like this:

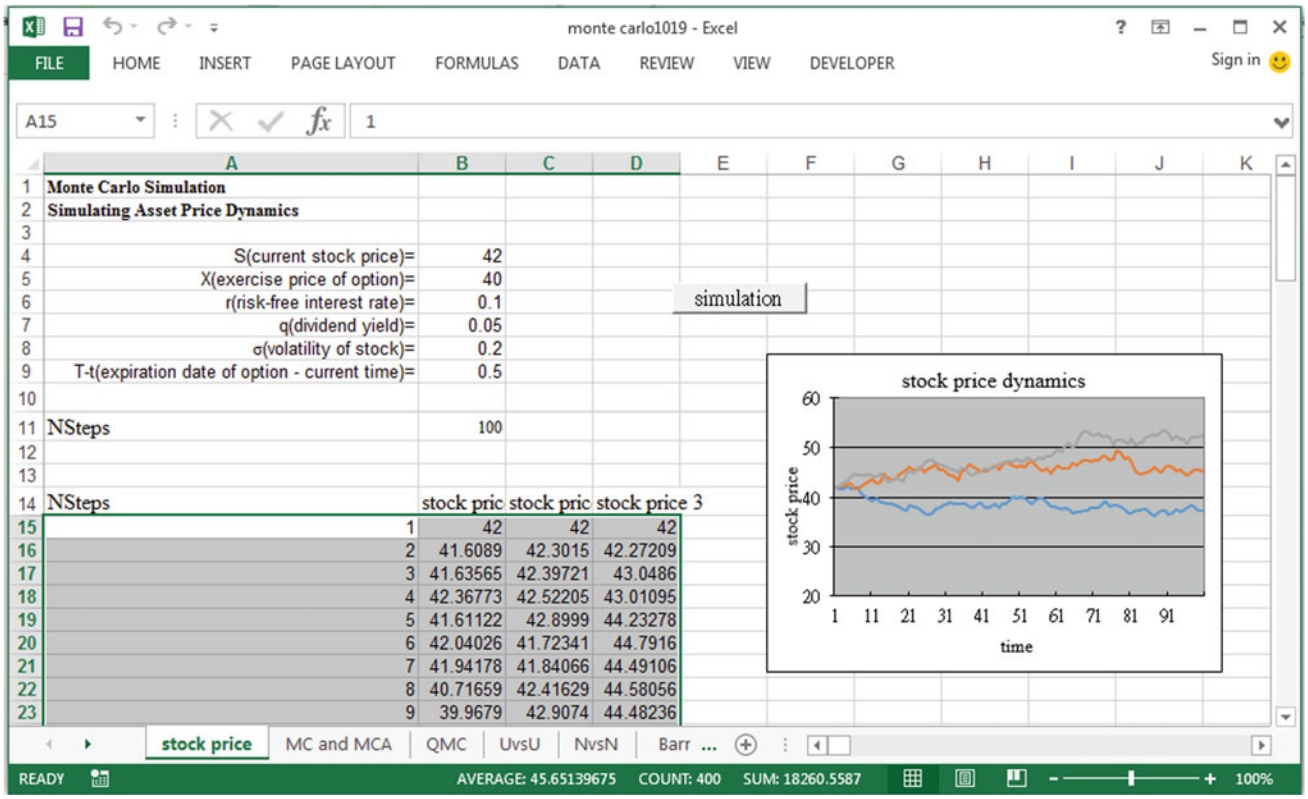
$$\mathrm{lnS}(\mathrm{t} + \delta \mathrm{t}) - \mathrm{lnS}(\mathrm{t}) = (\mu - 0.5\sigma^2)\delta \mathrm{t} + \sigma\sqrt{\delta t}\varepsilon.$$

We also can use another form to represent the stock price process:

$$\mathrm{S}(\mathrm{t} + \delta \mathrm{t}) = \mathrm{S}(\mathrm{t})\exp[\left(\mu - 0.5\sigma^2\right)\delta t + \sigma\sqrt{\delta t}\varepsilon].$$

In order to generate a stock price process, we can use the below subroutine:

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_10



At the heart of the Monte Carlo simulation for option valuation is the stochastic process that generates the share price. The stochastic equation for the underlying share price at time T when the option on the share expires was given as follows:

$$S_T = S_0 exp[(\mu - 0.5\sigma^2)T + \sigma\sqrt{T}\epsilon].$$

The associated European option payoff depends on the expectation of S_T in the risk-neutral world. Thus, the stochastic equation for S_T for risk-neutral valuation takes the following form:

$$S_T = S_0 exp[(r - q - 0.5\sigma^2)T + \sigma\sqrt{T}\epsilon].$$

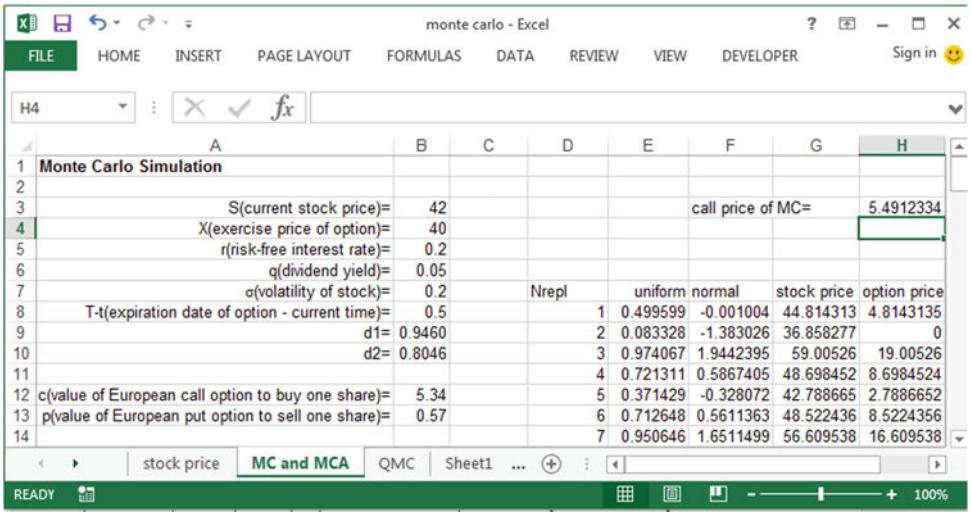
The share price process outlined above is the same as that assumed for binomial tree valuation. RAND gives random numbers uniformly distributed in the range [0, 1]. Regarding its outputs as cumulative probabilities, the NORMSINV function converts them into standard normal variate values, mostly between −3 and 3. The random normal samples (value of ε) are then used to generate share prices and the corresponding option payoff.

In European option pricing, we need to estimate the expected value of the discounted payoff of the option:

$$
\begin{aligned}
f &= \mathrm{e}^{-\mathrm{rT}}\mathrm{E}(f_T) \\
&= \mathrm{e}^{-\mathrm{rT}}\mathrm{E}[\max(S_T - X, 0)] \\
&= \mathrm{e}^{-\mathrm{rT}}\mathrm{E}[\max(S_0 exp[(r - q - 0.5\sigma^2)T + \sigma\sqrt{T}\varepsilon] - X, 0)].
\end{aligned}
$$

The standard deviation of the simulated payoffs divided by the square root of the number of trials is relatively large. To improve the precision of the Monte Carlo value estimate, the number of simulation trials must be increased.

We can replicate many stock prices at option maturity date in the Excel worksheet.



	A	B	C	D	E	F	G	H
1	**Monte Carlo Simulation**							
2								
3	S(current stock price)=	42				call price of MC=		5.4912334
4	X(exercise price of option)=	40						
5	r(risk-free interest rate)=	0.2						
6	q(dividend yield)=	0.05						
7	σ(volatility of stock)=	0.2		Nrepl	uniform	normal	stock price	option price
8	T-t(expiration date of option - current time)=	0.5		1	0.499599	-0.001004	44.814313	4.8143135
9	d1=	0.9460		2	0.083328	-1.383026	36.858277	0
10	d2=	0.8046		3	0.974067	1.9442395	59.00526	19.00526
11				4	0.721311	0.5867405	48.698452	8.6984524
12	c(value of European call option to buy one share)=	5.34		5	0.371429	-0.328072	42.788665	2.7886652
13	p(value of European put option to sell one share)=	0.57		6	0.712648	0.5611363	48.522436	8.5224356
14				7	0.950646	1.6511499	56.609538	16.609538

Using Excel RAND() function, we can generate a uniform random number, like cell E8. We simulate 100 random numbers in the sheet. Next, we use the NORMSINV function to transfer a uniform random number to a standard normal random number in cell F8. Then the random normal samples are used to generate stock prices from the formula, like G8. The stock price formula in G8 is

$$
\begin{aligned}
&= \$B\$3 * \mathrm{EXP}\big((\$B\$5 - \$B\$6 - 0.5 * \$B\$7^2) \\
&\quad * \$B\$8 + \$B\$7 * \mathrm{SQRT}(\$B\$8) * F8).
\end{aligned}
$$

Finally, the corresponding call option payoff in cell H8 is

$$= \mathrm{MAX}(G8 - \$B\$4, 0).$$

The discount value of the average of the 100 simulated option payoffs is the estimate call value from the Monte Carlo simulation. Pressing the F9 in Excel, we can generate a further 100 trials and another Monte Carlo simulation. The formula for the call option estimated by Monte Carlo simulation in H3 is

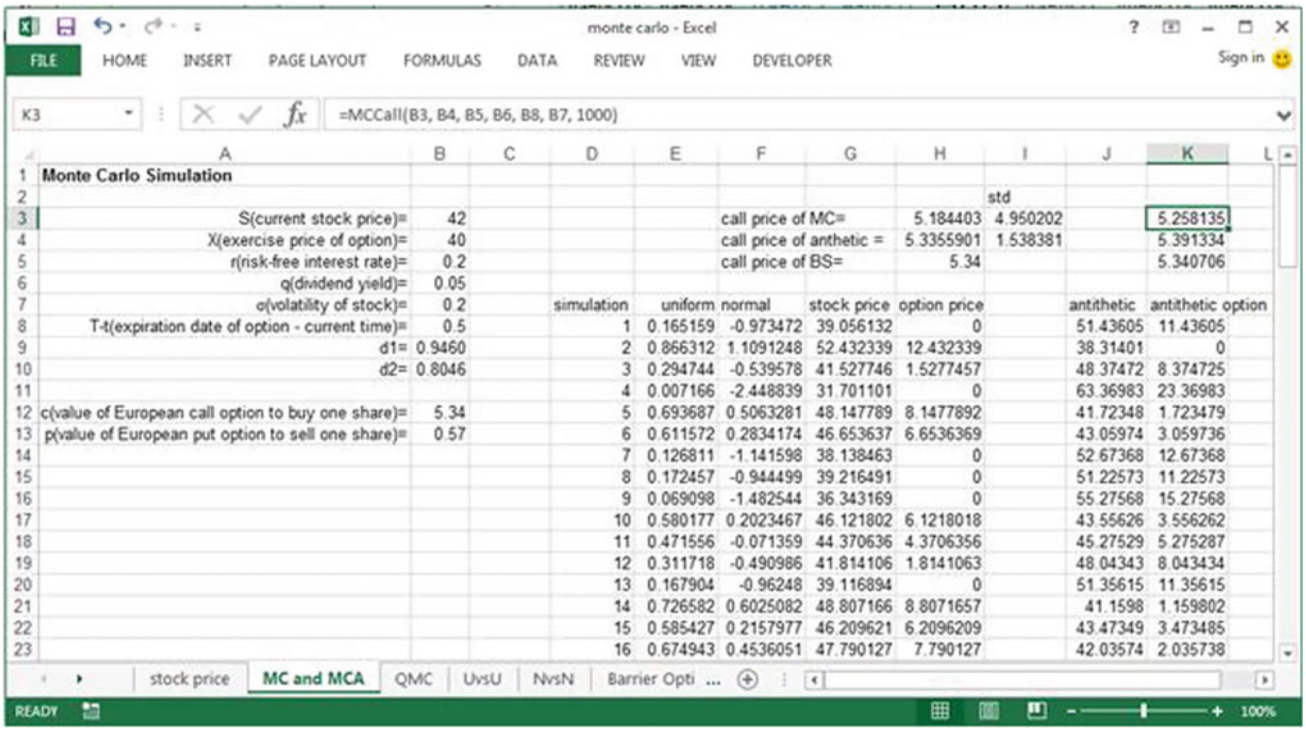
$$= \mathrm{EXP}(-\$B\$5 * \$B\$8) * \mathrm{AVERAGE}(H8 : H107).$$

The value in H3 is 5.49. Compare with the true Black and Scholes call value, there exist some differences. To improve the precision of the Monte Carlo estimate, the number of simulation trials has to be increased.

We can write a function for crude Monte Carlo simulation.

```
'   Monte-Carlo simulation for Call option
Function MCCall(S, X, r, q, T, sigma, NRepl)
    Dim nuT, siT, Sum, randns, ST, payoff
    Dim i As Integer
    Randomize
    nuT = (r - q - 0.5 * sigma ^ 2) * T
    siT = sigma * Sqr(T)
    Sum = 0
    For i = 1 To NRepl
        randns = Application.NormSInv(Rnd)
        ST = S * Exp(nuT + randns * siT)
        payoff = Application.Max((ST - X), 0)
        Sum = Sum + payoff
    Next i
    MCCall = Exp(-r * T) * Sum / NRepl
End Function
```

Using this function we can get option price calculated by Monte Carlo simulation. We can also change different numbers of replication to get a more efficient option price. This is shown below.



	A	B	C	D	E	F	G	H	I	J	K	L
1	**Monte Carlo Simulation**											
2									std			
3	S(current stock price)=	42				call price of MC=		5.184403	4.950202		5.258135	
4	X(exercise price of option)=	40				call price of anthetic =		5.3355901	1.538381		5.391334	
5	r(risk-free interest rate)=	0.2				call price of BS=		5.34			5.340706	
6	q(dividend yield)=	0.05										
7	σ(volatility of stock)=	0.2		simulation	uniform	normal	stock price	option price		antithetic	antithetic option	
8	T-t(expiration date of option - current time)=	0.5		1	0.165159	-0.973472	39.056132	0		51.43605	11.43605	
9	d1=	0.9460		2	0.866312	1.1091248	52.432339	12.432339		38.31401	0	
10	d2=	0.8046		3	0.294744	-0.539578	41.527746	1.5277457		48.37472	8.374725	
11				4	0.007166	-2.448839	31.701101	0		63.36983	23.36983	
12	c(value of European call option to buy one share)=	5.34		5	0.693687	0.5063281	48.147789	8.1477892		41.72348	1.723479	
13	p(value of European put option to sell one share)=	0.57		6	0.611572	0.2834174	46.653637	6.6536369		43.05974	3.059736	
14				7	0.126811	-1.141598	38.138463	0		52.67368	12.67368	
15				8	0.172457	-0.944499	39.216491	0		51.22573	11.22573	
16				9	0.069098	-1.482544	36.343169	0		55.27568	15.27568	
17				10	0.580177	0.2023467	46.121802	6.1218018		43.55626	3.556262	
18				11	0.471556	-0.071359	44.370636	4.3706356		45.27529	5.275287	
19				12	0.311718	-0.490986	41.814106	1.8141063		48.04343	8.043434	
20				13	0.167904	-0.96248	39.116894	0		51.35615	11.35615	
21				14	0.726582	0.6025082	48.807166	8.8071657		41.1598	1.159802	
22				15	0.585427	0.2157977	46.209621	6.2096209		43.47349	3.473485	
23				16	0.674943	0.4536051	47.790127	7.790127		42.03574	2.035738	

The Monte Carlo simulation for the European call option in K3 is

$$= \text{MCCall}(\text{B3},\ \text{B4},\ \text{B5}, \text{B6},\ \text{B8},\ \text{B7}, 1000).$$

In this case, we replicate 1000 times to get the call option value. The value in K3 is equal to 5.2581 which is a little more near the value of Black–Scholes, 5.34.

10.3 Antithetic Variables

In addition to increasing the number of trials, we have another way of improving the precision of the Monte Carlo estimate antithetic variables. The antithetic variates method is a variance reduction technique used in Monte Carlo methods. The standard error of the Monte Carlo estimate (with antithetic variables) is substantially lower than that for the uncontrolled sampling approach. Therefore, the antithetic variates method reduces the variance of the simulation results and improves the efficiency of the simulation.

The antithetic variates technique consists, for every sample path obtained, in taking its antithetic path. Suppose that we have two random samples X_1 and X_2:

$$X_1(1), X_1(2), \ldots, X_1(n)$$

$$X_2(1), X_2(2), \ldots, X_2(n).$$

We would like to estimate

$$\theta = \mathrm{E}[\mathrm{h}(\mathrm{X})].$$

An unbiased estimator is given by

$$\mathrm{X(i)} = \frac{X_1(i) + X_2(i)}{2}.$$

Therefore,

$$\mathrm{var}\left[\frac{\sum X(i)}{n}\right]$$

$$= \frac{var[X(i)]}{n}$$

$$= \frac{var[X_1(i)] + var[X_2(i) + 2cov[X_1(i), X_2(i)]}{4n} < \frac{var[X_1(i)]}{n}.$$

In order to reduce the sample mean variance, we should take $\mathrm{cov}[X_1(i), X_2(i)] < var[X_i(i)]$. In the antithetic method, we will choose the second sample in such a way that X_1 and X_2 are not i.i.d., but cov(X_1, X_2) is negative. As a result, variance is reduced.

There are two advantages in the antithetic method. First, it reduces the number of normal samples to be taken to generate N paths. Second, it reduces the variance of the sample paths, improving the accuracy. An important point to bear in mind is that antithetic sampling may not yield a variance reduction when some monotonicity condition is not satisfied.

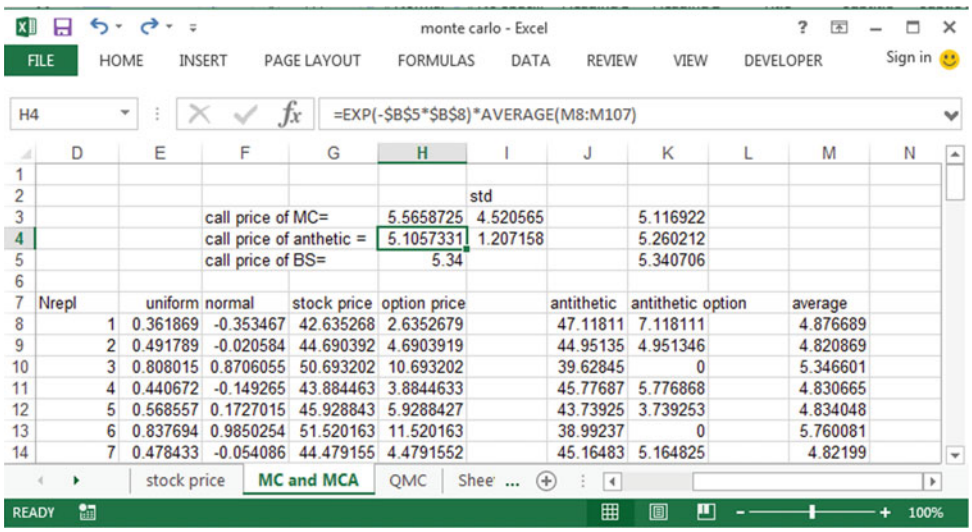
We use a spreadsheet to implement the antithetic method which is shown below. The stock price in G8 is

$$= \$B\$3 * \mathrm{EXP}\left(\left(\$B\$5 - \$B\$6 - 0.5 * \$B\$7^2\right) * \$B\$8 + \$B\$7 * \mathrm{SQRT}(\$B\$8) * F8\right).$$

The antithetic variable method generates the other stock price in J8 which is equal to

$$= \$B\$3 * \mathrm{EXP}\left(\left(\$B\$5 - \$B\$6 - 0.5 * \$B\$7^2\right) * \$B\$8 + \$B\$7 * \mathrm{SQRT}(\$B\$8) * (-F8)\right).$$

The most important in these two formulas are random variables. The first one uses F8 and the other one use –F8.



The call option value estimated by the antithetic method in H4 is

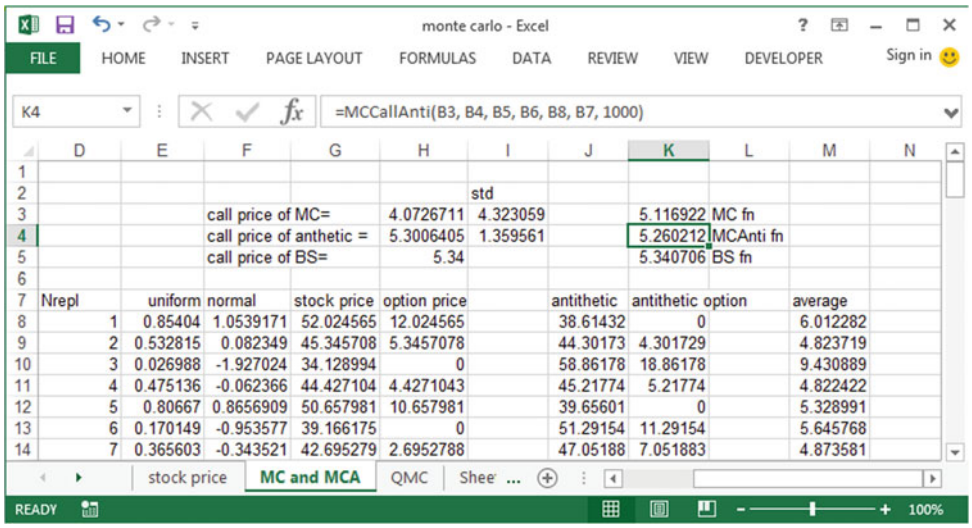
$$= \text{EXP}(-\$B\$5 * \$B\$8) * \text{AVERAGE}(M8 : M107).$$

We also calculate the standard deviations of Monte Carlo simulation and antithetic variates method in I3 and I4. The standard deviation of the antithetic variates method is smaller than the one of the Monte Carlo simulation.

In addition, we can write a function for the antithetic method to improve the precision of the Monte Carlo estimate. Below is the code:

```
'   Monte-Carlo simulation and antithetic variates for Call option
Function MCCallAnti(S, X, r, q, T, sigma, NRepl)
    Dim nuT, siT, Sum, randns, ST1, ST2, payoff1, payoff2
    Dim i As Integer
    Randomize
    nuT = (r - q - 0.5 * sigma ^ 2) * T
    siT = sigma * Sqr(T)
    Sum = 0
    For i = 1 To NRepl
        randns = Application.NormSInv(Rnd)
        ST1 = S * Exp(nuT + randns * siT)
        ST2 = S * Exp(nuT - randns * siT)
        payoff1 = Application.Max((ST1 - X), 0)
        payoff2 = Application.Max((ST2 - X), 0)
        Sum = Sum + 0.5 * (payoff1 + payoff2)
    Next i
    MCCallAnti = Exp(-r * T) * Sum / NRepl
End Function
```

We can directly use this function in the worksheet to get the estimate of the antithetic method. After changing the numbers of replication, we can get the option prices in different numbers of replication.



	D	E	F	G	H	I	J	K	L	M	N
1											
2						std					
3			call price of MC=		4.0726711	4.323059		5.116922	MC fn		
4			call price of anthetic =		5.3006405	1.359561		5.260212	MCAnti fn		
5			call price of BS=		5.34			5.340706	BS fn		
6											
7	Nrepl	uniform	normal	stock price	option price		antithetic	antithetic option		average	
8	1	0.85404	1.0539171	52.024565	12.024565		38.61432	0		6.012282	
9	2	0.532815	0.082349	45.345708	5.3457078		44.30173	4.301729		4.823719	
10	3	0.026988	-1.927024	34.128994	0		58.86178	18.86178		9.430889	
11	4	0.475136	-0.062366	44.427104	4.4271043		45.21774	5.21774		4.822422	
12	5	0.80667	0.8656909	50.657981	10.657981		39.65601	0		5.328991	
13	6	0.170149	-0.953577	39.166175	0		51.29154	11.29154		5.645768	
14	7	0.365603	-0.343521	42.695279	2.6952788		47.05188	7.051883		4.873581	

The formula for the call value of the antithetic variates method in K4 is

$$= \text{MCCallAnti}(\text{B3}, \text{B4}, \ \text{B5}, \text{B6}, \ \text{B8}, \text{B7}, 1000).$$

The value in K4 is closer to Black–Scholes, K5, than K3 estimated by Monte Carlo in 100 times replication.

10.4 Quasi-Monte Carlo Simulation

Quasi-Monte Carlo simulation is another way to improve the efficiency of Monte Carlo. This simulation method is a method for solving some other problems using low-discrepancy sequences (also called quasi-random sequences or sub-random sequences). This is in contrast to the regular Monte Carlo simulation, which is based on sequences of pseudorandom numbers. To generate U(0,1) variables, the standard method is based on linear congruential generators (LCGs). LCG is a process that gives an initial z_0 and through a formula to generate the next number. The formula is

$$z_i = (a * z_{i-1} + c)(modm).$$

For example, 15 mod 6 = 3 (remainder of integer division). Then the uniform random number is

$$U_i = \frac{z_i}{m}.$$

There is nothing random in this sequence. First, it must start from an initial number z_0, seed. Secondly, the generator is periodic.

The inverse transform is a general approach to transform uniform variates into normal variates. Since no analytical form for it is known, we cannot invert the normal distribution function efficiently. One old-fashioned possibility, which is still suggested in some textbooks is to exploit the central limit theorem to generate a normal random number by summing a suitable number of uniform variates. Computational efficiency would restrict the number of uniform variates. An alternative method is the Box–Muller approach. Consider two independent variables X,Y $\sim$ N(0,1), and let (R,θ) be the polar coordinates of the point of Cartesian coordinates (X,Y) in the planes, so that

$$\text{d} = R^2 = X^2 + Y^2$$

$$\theta = \tan^{-1}\frac{Y}{X}$$

The Box–Muller algorithm can be represented as follows:

1. Generate two independent uniform random variates U1 and U2 $\sim$ U(0,1).
2. Set R^2 = −2*log(U1) and θ = 2π*U2.
3. Set X = R*cosθ and Y = R*sinθ,
   then X $\sim$ N(0,1) and Y $\sim$ N(0,1) are independent standard normal variates.

Here is the VBA function to generate a Box–Muller normal random numbers:

```
'   Box Muller transformation 1
Function BMNormSInv1(x1 As Double, x2 As Double) As Double
    Dim vlog, norm1
    vlog = Sqr(-2 * Log(x1))
    norm1 = vlog * Cos(2 * Application.Pi() * x2)
    BMNormSInv1 = norm1
End Function
```

The random numbers produced by a LCG or by more sophisticated algorithms are not random at all. So one could try to devise alternative deterministic sequences of numbers that are in some sense evenly distributed. This idea may be made more precise by defining the discrepancy of a sequence of numbers. The only trick in the selection process is to remember the values of all the previous numbers chosen as each new number is selected. Using quasi-random sampling means that the error in any estimate based on the samples is proportional to 1/n rather than 1/sqrt(n).

There are many quasi-random sequences (Low-discrepancy sequences), like Halton's sequence, Sobol's sequence, Faure's sequence, and Niederreiter's sequence. For instance, the Halton sequence is constructed according to a deterministic method that uses a prime number as its base. Here is a simple example to create Halton's Sequence which base is 2:

1. Representing an integer number n in a base b, where b is a prime number:

$$\begin{aligned} \mathrm{n} &= (\ldots d_4\, d_3\, d_2\, d_1 d_0)_b \\ &= \sum_{k=0}^{m} d_k b^k. \end{aligned}$$

For example, $4 = (100)_2 = 1 * 2^2 + 0 * 2^1 + 0 * 2^0$.

2. Reflecting the digits and adding a radix point to obtain a number with the unit interval:

$$\begin{aligned} \mathrm{h}(\mathrm{n}, \mathrm{b}) &= (0.d_0\, d_1\, d_2\, d_3 d_4 \ldots)_b \\ &= \sum_{k=0}^{m} d_k b^{-k+1}. \end{aligned}$$

For example, $(0.001)_2 = 1 * \frac{1}{2^3} + 0 * \frac{1}{2^2} + 0 * \frac{1}{2} = \frac{1}{8}$.

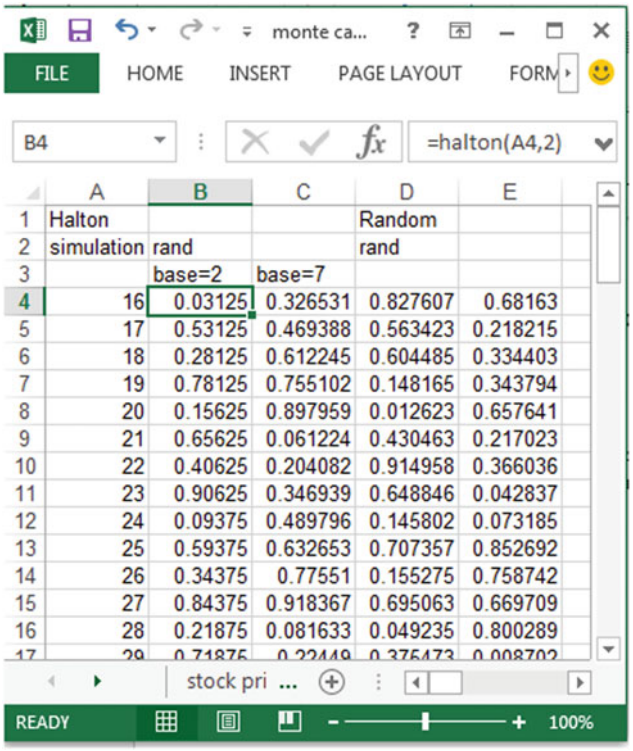
Therefore, we get Halton's sequence:

n :	1	2	3	4	5	6	7	...
h(n, 2) :	1/2	1/4	3/4	1/8	5/8	3/8	7/8	...

Below is a function to generate Halton's sequence:

```
' Helton's sequence
Function Halton(n, b) As Double
  Dim h As Double, f As Double
  Dim n1 As Integer, n0 As Integer, r As Integer
  n0 = n
  h = 0
  f = 1 / b
  Do While n0 > 0
     n1 = Int(n0 / b)
     r = n0 - n1 * b
     h = h + f * r
     f = f / b
     n0 = n1
  Loop
  Halton = h
End Function
```

Using this function in the worksheet, we can get a sequence number generated by the Halton function. In addition, we can change the prime number to get a Halton number from a different base.



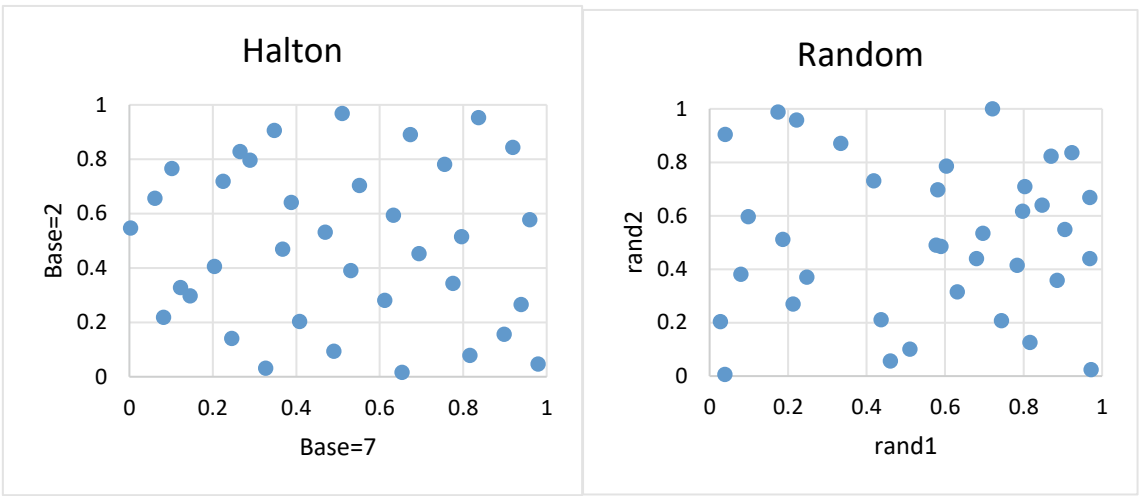
	A	B	C	D	E
1	Halton			Random	
2	simulation	rand		rand	
3		base=2	base=7		
4	16	0.03125	0.326531	0.827607	0.68163
5	17	0.53125	0.469388	0.563423	0.218215
6	18	0.28125	0.612245	0.604485	0.334403
7	19	0.78125	0.755102	0.148165	0.343794
8	20	0.15625	0.897959	0.012623	0.657641
9	21	0.65625	0.061224	0.430463	0.217023
10	22	0.40625	0.204082	0.914958	0.366036
11	23	0.90625	0.346939	0.648846	0.042837
12	24	0.09375	0.489796	0.145802	0.073185
13	25	0.59375	0.632653	0.707357	0.852692
14	26	0.34375	0.77551	0.155275	0.758742
15	27	0.84375	0.918367	0.695063	0.669709
16	28	0.21875	0.081633	0.049235	0.800289
17	29	0.71875	0.22449	0.375473	0.008702

The formula for the Halton number in B4 is

$$= \text{halton}(\text{A4}, 2)$$

which is the 16th number under the base is equal to 2. We can change the base to 7 as shown in C4.

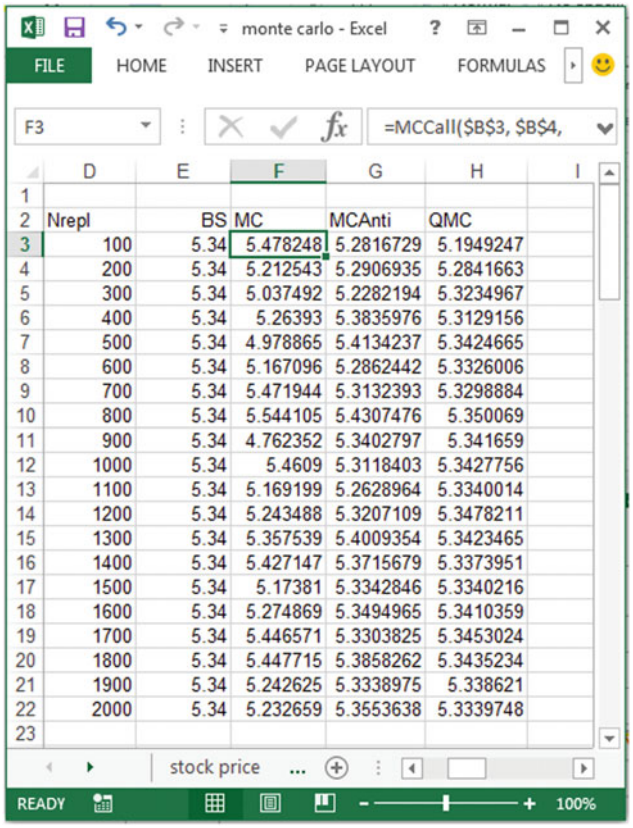
Two independent numbers generated by Halton or random generator can construct a join distribution. The results are shown in the below figures. We can see that the numbers generated from Halton's sequence are more discrepant than the numbers generated from a random generator in Excel.



We can use Halton's sequences and the Box–Muller approach to generate a normal random number. And create stock prices at the maturity of the option. Then we can estimate the option price today. This estimating process is called the Quasi-Monte Carlo simulation. The following function can accomplish this task:

```
'   Quasi Monte-Carlo simulation for Call option
Function QMCCallBM(S, X, r, q, T, sigma, NRepl)
    Dim nuT, siT, sum, ST1, qrandns1, ST2, qrandns2, NRepl1
    Dim i As Integer, iskip As Integer
    nuT = (r - q - 0.5 * sigma ^ 2) * T
    siT = sigma * Sqr(T)
    iskip = (2 ^ 4) - 1
    sum = 0
    NRepl1 = Application.Ceiling(NRepl / 2, 1)
   For i = 1 To NRepl1
        qrandns1 = BMNormSInv1(Halton(i + iskip, 2), Halton(i + iskip,
3))
        ST1 = S * Exp(nuT + qrandns1 * siT)
        qrandns2 = BMNormSInv2(Halton(i + iskip, 2), Halton(i + iskip,
3))
        ST2 = S * Exp(nuT + qrandns2 * siT)
        sum = sum + 0.5 * (Application.Max((ST1 - X), 0) +
Application.Max((ST2 - X), 0))
    Next i
    QMCCallBM = Exp(-r * T) * sum / NRepl1
End Function
```

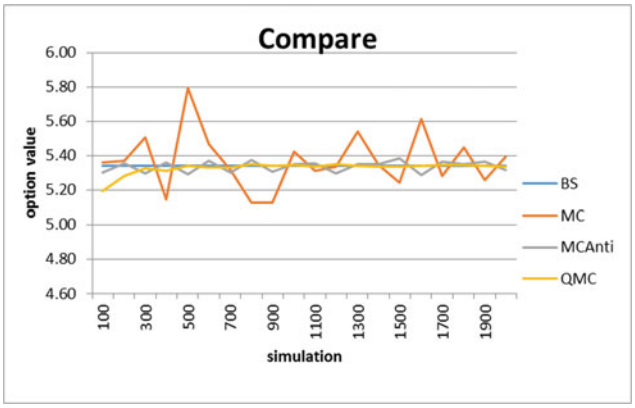
The Halton sequence can have the desirable property. The error in any estimate based on the samples is proportional to 1/M rather than $1/\sqrt{M}$, where M is the number of samples. We compare Monte Carlo estimates, Antithetic variates, and Quasi-Monte Carlo estimates in different simulation numbers. In the table below, we represent different replication numbers, 100, 200, … 2000, to price option. The following figure is the result.



	D	E	F	G	H
1					
2	Nrepl	BS	MC	MCAnti	QMC
3	100	5.34	5.478248	5.2816729	5.1949247
4	200	5.34	5.212543	5.2906935	5.2841663
5	300	5.34	5.037492	5.2282194	5.3234967
6	400	5.34	5.26393	5.3835976	5.3129156
7	500	5.34	4.978865	5.4134237	5.3424665
8	600	5.34	5.167096	5.2862442	5.3326006
9	700	5.34	5.471944	5.3132393	5.3298884
10	800	5.34	5.544105	5.4307476	5.350069
11	900	5.34	4.762352	5.3402797	5.341659
12	1000	5.34	5.4609	5.3118403	5.3427756
13	1100	5.34	5.169199	5.2628964	5.3340014
14	1200	5.34	5.243488	5.3207109	5.3478211
15	1300	5.34	5.357539	5.4009354	5.3423465
16	1400	5.34	5.427147	5.3715679	5.3373951
17	1500	5.34	5.17381	5.3342846	5.3340216
18	1600	5.34	5.274869	5.3494965	5.3410359
19	1700	5.34	5.446571	5.3303825	5.3453024
20	1800	5.34	5.447715	5.3858262	5.3435234
21	1900	5.34	5.242625	5.3338975	5.338621
22	2000	5.34	5.232659	5.3553638	5.3339748
23					

In column E, we use the Black–Scholes function BSCall (S, X, r, q, T, sigma) which is used as a benchmark. The Monte Carlo simulation function, MCCall(S, X, r, q, T, sigma, NRepl), is used in column F. In column G, the call value is evaluated by the antithetic variates function, MCCallAnti(S, X, r, q, T, sigma, NRepl). Quasi-Monte Carlo simulation function, QMCCallBM(S, X, r, q, T, sigma, NRepl) is used in column H.

The relative convergence of different Monte Carlo simulations can be compared. The data in range E3:H22 can be used to chart. The result is shown below.



In the figure above, we can see that the Monte Carlo estimate is more volatile than Antithetic variates and Quasi-Monte Carlo estimates.

10.5 Application

The binomial tree method is well suited to the price American option. However, Monte Carlo simulation is suitable to value path-dependent options. In this section, we introduce the application of Monte Carlo simulation in the path-depend option.

Barrier options are one kind of path-depend options where the payoff depends on whether the price of the underlying reaches a certain level of price during a certain period of time. There are a number of different types of barrier options. They can be classified as knock-out or knock-in options. Here we give a down-and-out put option as an example.

A down-and-out put option is a put option that becomes void if the asset price falls below the barrier S_b ($S_b < S_0$ and $S_b < X$)

$$P = P_{di} + P_{do}.$$

In principle, the barrier might be monitored continuously; in practice, periodic monitoring may be applied. If the barrier can be monitored continuously, analytical pricing formulas are available for certain barrier option

$$P_{do} = Xe^{-rT}\{N(d_4) - N(d_2) - a[N(d_7) - N(d_5)]\}$$

$$-S_0 e^{-qT}\{N(d_3) - N(d_1) - b[N(d_8) - N(d_6)]\}$$

$$a = (S_b/S_0)^{-1+2r/\sigma^2}$$

$$b = (S_b/S_0)^{1+2r/\sigma^2}$$

$$d_1 = \left[\ln(S_0/X) + \left(r - q + \sigma^2/2\right)T\right]/(\sigma\sqrt{T})$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

$$d_3 = \left[\ln(S_0/S_b) + \left(r - q + \sigma^2/2\right)T\right]/(\sigma\sqrt{T})$$

$$d_4 = d_3 - \sigma\sqrt{T}$$

$$d_5 = \left[\ln(S_0/S_b) - \left(r - q - \sigma^2/2\right)T\right]/(\sigma\sqrt{T})$$

$$d_6 = d_5 - \sigma\sqrt{T}$$

$$d_7 = \left[\ln\left(S_0X/S_b^2\right) - \left(r - q - \sigma^2/2\right)T\right]/(\sigma\sqrt{T})$$

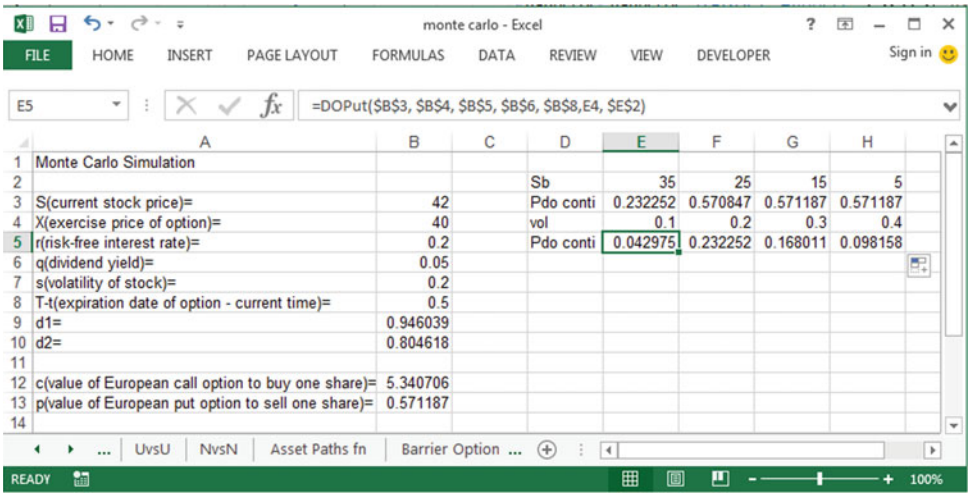
$$d_8 = d_7 - \sigma\sqrt{T}$$

As an example, a down-and-out put option with strike price X, expiring in T time units, with a barrier set to S_b. S_0, r, q, σ have the usual meaning.

To accomplish this, we can use the below code to generate a function:

```
'Down-and-out put option
Function DOPut(S, X, r, q, T, sigma, Sb)
    Dim NDOne, NDTwo, NDThree, NDFour, NDFive, NDSix, NDSeven, NDEight,
a, b, DOne, DTwo, DThree, DFour, DFive, DSix, DSeven, DEight
    a = (Sb / S) ^ (-1 + (2 * r / sigma ^ 2))
    b = (Sb / S) ^ (1 + (2 * r / sigma ^ 2))
    DOne = (Log(S / X) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DTwo = (Log(S / X) + (r - q - 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DThree = (Log(S / Sb) + (r - q + 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DFour = (Log(S / Sb) + (r - q - 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DFive = (Log(S / Sb) - (r - q - 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DSix = (Log(S / Sb) - (r - q + 0.5 * sigma ^ 2) * T) / (sigma *
Sqr(T))
    DSeven = (Log(S * X / Sb ^ 2) - (r - q - 0.5 * sigma ^ 2) *T) /
(sigma * Sqr(T))
    DEight = (Log(S * X / Sb ^ 2) - (r - q + 0.5 * sigma ^ 2) *T) /
(sigma * Sqr(T))
    NDOne = Application.NormSDist(DOne)
    NDTwo = Application.NormSDist(DTwo)
    NDThree = Application.NormSDist(DThree)
    NDFour = Application.NormSDist(DFour)
    NDFive = Application.NormSDist(DFive)
    NDSix = Application.NormSDist(DSix)
    NDSeven = Application.NormSDist(DSeven)
    NDEight = Application.NormSDist(DEight)
    DOPut = X * Exp(-r * T) * (NDFour - NDTwo - a * (NDSeven - NDFive))
- S * Exp(-q * T) * (NDThree - NDOne - b * (NDEight - NDSix))
End Function
```

Barrier options often have very different properties from plain vanilla options. For instance, sometimes the Greek letter, vega, is negative. Below is the spreadsheet to show this phenomenon.



The formula for down-and-out put option in cell E5 is

$$= \text{DOPut}(\$B\$3, \$B\$4, \$B\$5, \$B\$6, \$B\$8, E4, \$E\$2)$$

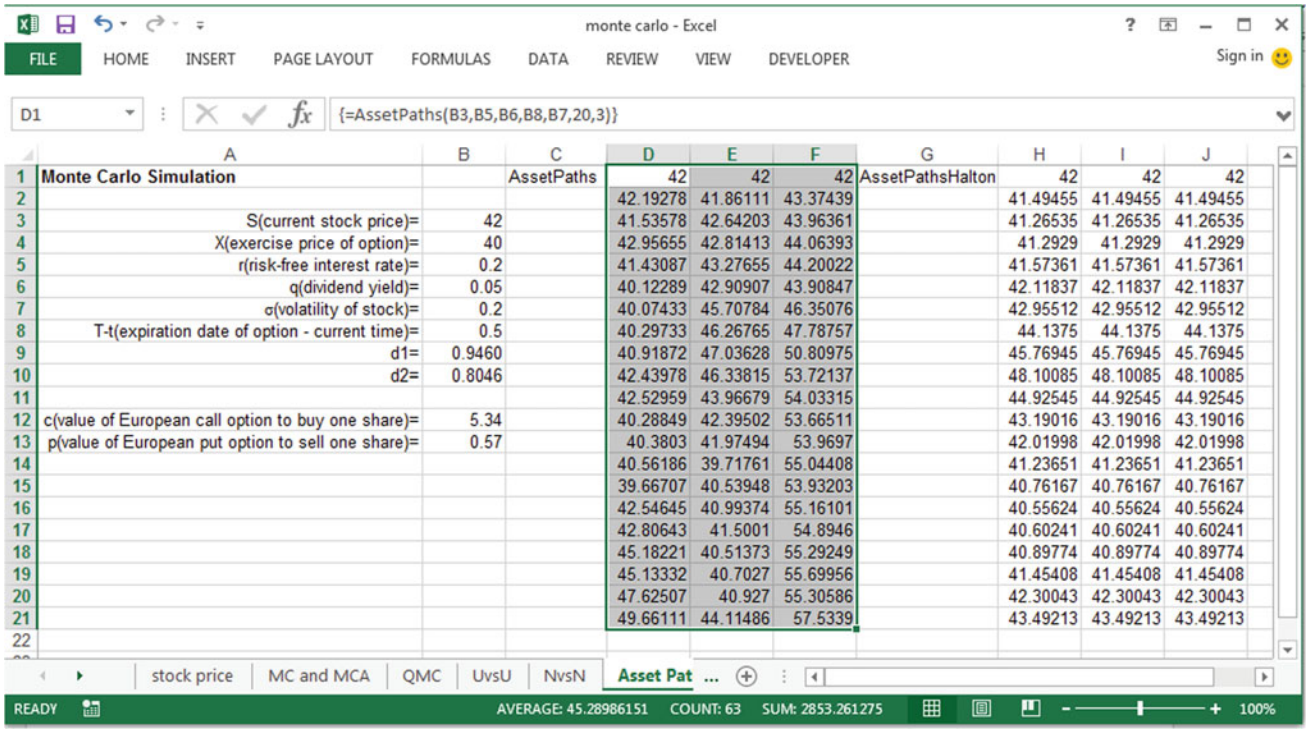
As volatility increases, the price of down-and-out put option may decrease because the stock is easy to drop down across the barrier. We can see this effect in the below figure. As the volatility increases from 0.1 to 0.2, the barrier option price increases. However, as the volatility increases from 0.2 to 0.3, the barrier option price decreases.

However, the monitored continuous barrier option is theoretical. In practice, we can only consider a down-and-out put option periodically, under the assumption that the barrier is checked at the end of each trading day. In order to price the barrier option, we have to generate a stock price process and not only the maturity price. Below are the functions to generate two asset price processes under random number and Halton's sequence:

```
'Random Asset Paths
Function AssetPaths(S, r, q, T, sigma, NSteps, NRepl)
     Dim dt, nut, sit
     Dim i, j As Integer
     Dim spath()
     Randomize
     dt = T / NSteps
     nut = (r - q - 0.5 * sigma ^ 2) * dt
     sit = sigma * Sqr(dt)
     ReDim spath(NSteps, 1 To NRepl)
     For j = 1 To NRepl
        spath(0, j) = S
        For i = 1 To NSteps
            randns = Application.NormSInv(Rnd)
            spath(i, j) = spath(i - 1, j) * Exp(nut + randns * sit)
        Next i
     Next j
     AssetPaths = spath
     End Function
```

```
' Halton Asset Paths
Function AssetPathsHalton(S, r, q, T, sigma, NSteps, NRepl)
     Dim dt, nut, sit
     Dim i, j As Integer
     Dim spath()
     Randomize
     dt = T / NSteps
     nut = (r - q - 0.5 * sigma ^ 2) * dt
     sit = sigma * Sqr(dt)
     ReDim spath(NSteps, 1 To NRepl)
     For j = 1 To NRepl
        spath(0, j) = S
        For i = 1 To NSteps
            randns = Application.NormSInv(Halton((j - 1) * NStpes + i +
16, 13))
            spath(i, j) = spath(i - 1, j) * Exp(nut + randns * sit)
        Next i
     Next j
     AssetPathsHalton = spath
 End Function
```

Where NSteps is the number of time intervals from now to option maturity and NRepl is how many replications to simulate. After we input the parameters, we can get the stock price process. Below we replicate three stock price processes for each method. Each process with 20 time intervals.



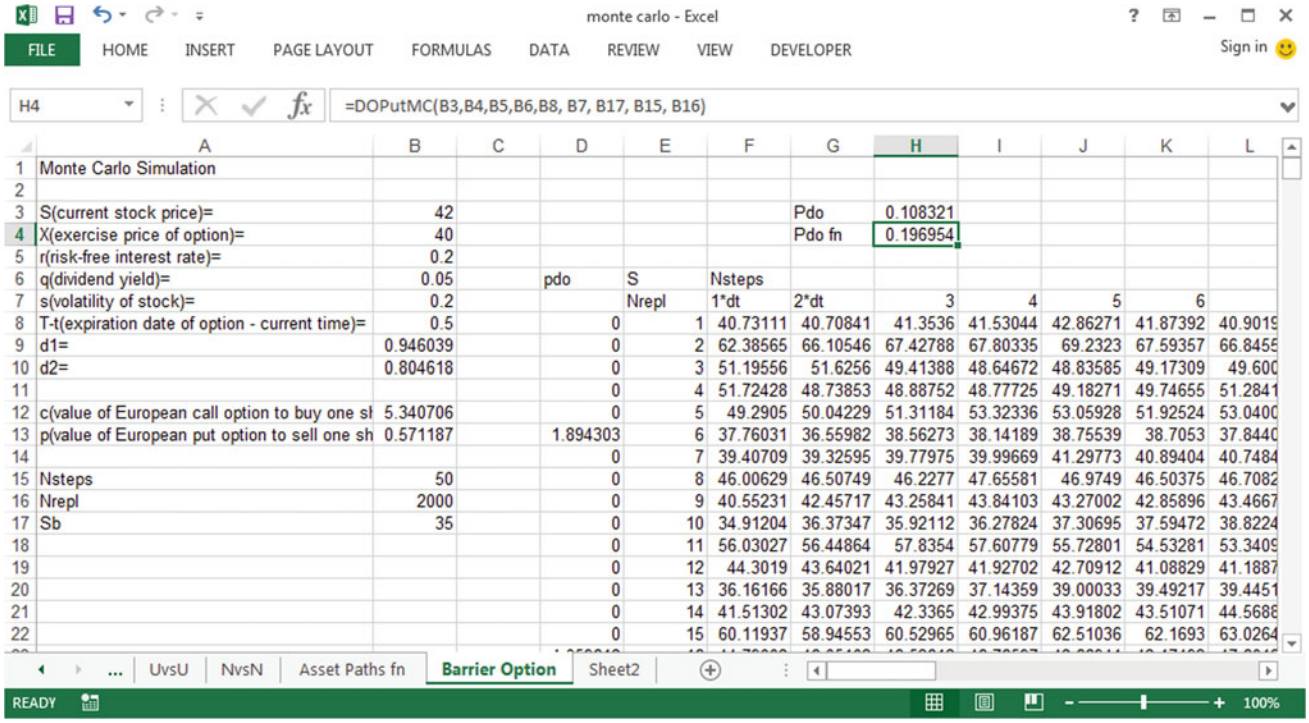
	A	B	C	D	E	F	G	H	I	J
1	**Monte Carlo Simulation**		AssetPaths	42	42	42	AssetPathsHalton	42	42	42
2				42.19278	41.86111	43.37439		41.49455	41.49455	41.49455
3	S(current stock price)=	42		41.53578	42.64203	43.96361		41.26535	41.26535	41.26535
4	X(exercise price of option)=	40		42.95655	42.81413	44.06393		41.2929	41.2929	41.2929
5	r(risk-free interest rate)=	0.2		41.43087	43.27655	44.20022		41.57361	41.57361	41.57361
6	q(dividend yield)=	0.05		40.12289	42.90907	43.90847		42.11837	42.11837	42.11837
7	σ(volatility of stock)=	0.2		40.07433	45.70784	46.35076		42.95512	42.95512	42.95512
8	T-t(expiration date of option - current time)=	0.5		40.29733	46.26765	47.78757		44.1375	44.1375	44.1375
9	d1=	0.9460		40.91872	47.03628	50.80975		45.76945	45.76945	45.76945
10	d2=	0.8046		42.43978	46.33815	53.72137		48.10085	48.10085	48.10085
11				42.52959	43.96679	54.03315		44.92545	44.92545	44.92545
12	c(value of European call option to buy one share)=	5.34		40.28849	42.39502	53.66511		43.19016	43.19016	43.19016
13	p(value of European put option to sell one share)=	0.57		40.3803	41.97494	53.9697		42.01998	42.01998	42.01998
14				40.56186	39.71761	55.04408		41.23651	41.23651	41.23651
15				39.66707	40.53948	53.93203		40.76167	40.76167	40.76167
16				42.54645	40.99374	55.16101		40.55624	40.55624	40.55624
17				42.80643	41.5001	54.8946		40.60241	40.60241	40.60241
18				45.18221	40.51373	55.29249		40.89774	40.89774	40.89774
19				45.13332	40.7027	55.69956		41.45408	41.45408	41.45408
20				47.62507	40.927	55.30586		42.30043	42.30043	42.30043
21				49.66111	44.11486	57.5339		43.49213	43.49213	43.49213
22										

Because the output of this function is a matrix, we should follow the step below to generate the outcome. First, select the range of cells in which you want to enter the array formula, in this example, D1:F21. Second, enter the formula that you want to use, in this example, AssetPaths(B3,B5,B6, B8,B7,20,3). Finally, press Ctrl + Shift + Enter.

Now, we can use a Monte Carlo simulation to compute the price of the down-and-out put option. Following the function can help us accomplish this task:

```
'Down-and-out put Monte Carlo Simulation
Function DOPutMC(S, X, r, q, T, sigma, Sb, NSteps, NRepl)
    Dim payoff, sum
    Dim spath()
    ReDim spath(NSteps, 1 To NRepl)
    sum = 0
    spath = AssetPaths(S, r, q, T, sigma, NSteps, NRepl)
    For j = 1 To NRepl
        payoff = Application.Max(X - spath(NSteps, j), 0)
        For i = 1 To NSteps
            If spath(i, j) <= Sb Then
            payoff = 0
            i = NSteps
            End If
        Next i
        sum = sum + payoff
    Next j
    DOPutMC = Exp(-r * tyr) * sum / NRepl
End Function
```

Using this function, we can enter parameters into the function at the worksheet. Or we can generate the stock price process in the worksheet directly. Below is the figure to show these two results.



	A	B	C	D	E	F	G	H	I	J	K	L
1	Monte Carlo Simulation											
2												
3	S(current stock price)=	42					Pdo	0.108321				
4	X(exercise price of option)=	40					Pdo fn	0.196954				
5	r(risk-free interest rate)=	0.2										
6	q(dividend yield)=	0.05		pdo	S	Nsteps						
7	s(volatility of stock)=	0.2			Nrepl	1*dt	2*dt	3	4	5	6	
8	T-t(expiration date of option - current time)=	0.5		0	1	40.73111	40.70841	41.3536	41.53044	42.86271	41.87392	40.9019
9	d1=	0.946039		0	2	62.38565	66.10546	67.42788	67.80335	69.2323	67.59357	66.8455
10	d2=	0.804618		0	3	51.19556	51.6256	49.41388	48.64672	48.83585	49.17309	49.600
11				0	4	51.72428	48.73853	48.88752	48.77725	49.18271	49.74655	51.2841
12	c(value of European call option to buy one sl	5.340706		0	5	49.2905	50.04229	51.31184	53.32336	53.05928	51.92524	53.040
13	p(value of European put option to sell one sh	0.571187		1.894303	6	37.76031	36.55982	38.56273	38.14189	38.75539	38.7053	37.844
14				0	7	39.40709	39.32595	39.77975	39.99669	41.29773	40.89404	40.7484
15	Nsteps	50		0	8	46.00629	46.50749	46.2277	47.65581	46.9749	46.50375	46.7082
16	Nrepl	2000		0	9	40.55231	42.45717	43.25841	43.84103	43.27002	42.85896	43.4667
17	Sb	35		0	10	34.91204	36.37347	35.92112	36.27824	37.30695	37.59472	38.8224
18				0	11	56.03027	56.44864	57.8354	57.60779	55.72801	54.53281	53.3409
19				0	12	44.3019	43.64021	41.97927	41.92702	42.70912	41.08829	41.1887
20				0	13	36.16166	35.88017	36.37269	37.14359	39.00033	39.49217	39.4451
21				0	14	41.51302	43.07393	42.3365	42.99375	43.91802	43.51071	44.5688
22				0	15	60.11937	58.94553	60.52965	60.96187	62.51036	62.1693	63.0264

The formula in cell H3 estimated by the worksheet is

$$= \text{AVERAGE}(\text{D13} : \text{D1012}) * \text{EXP}(-\$\text{B}\$5 * \$\text{B}\$8).$$

The formula in cell H4 estimated by user-defined VBA function is

$$= \text{DOPutMC}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}, \ \text{B17}, \text{B15}, \ \text{B16}).$$

If you want to know how many replications the stock price crosses the barrier, below is the function to complete this job:

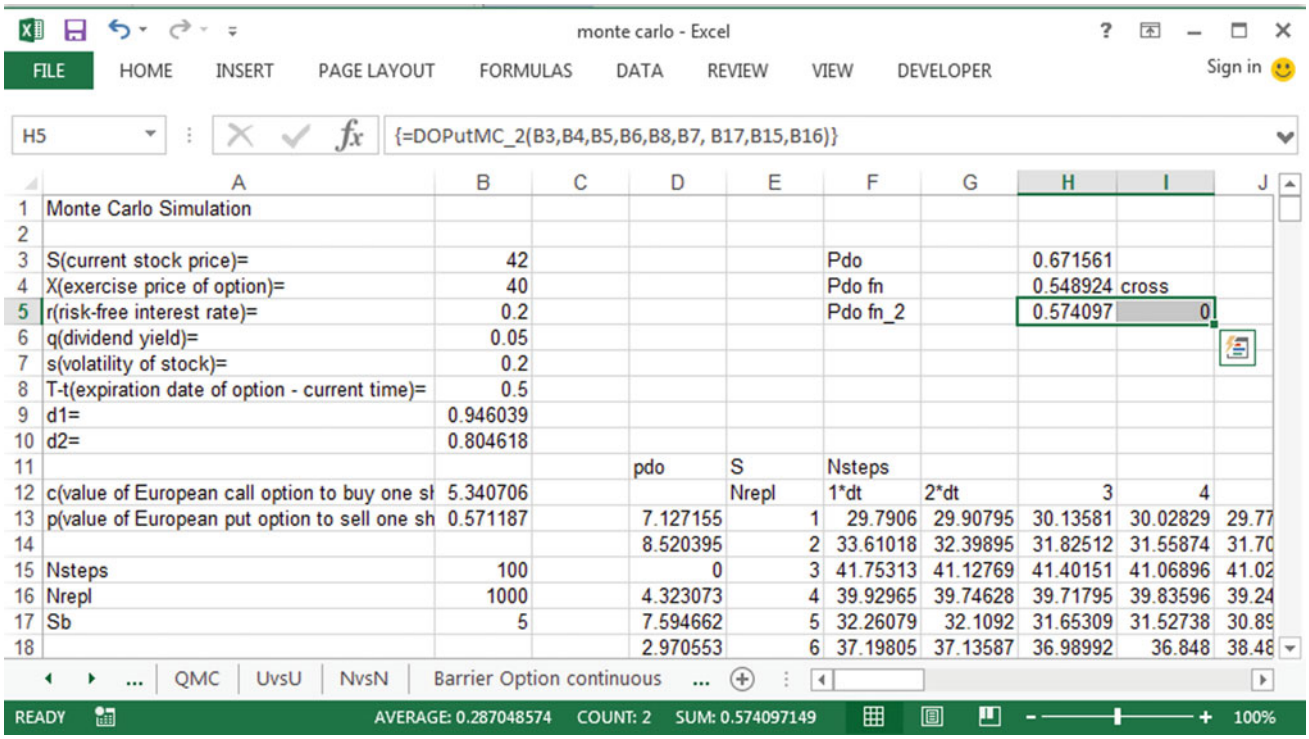
```
' Down-and-out put Monte Carlo Simulation and cross times
Function DOPutMC_2(S, X, r, q, T, sigma, Sb, NSteps, NRepl)
    Dim payoff, Sum, cross
    Dim temp(1)
    Dim spath()
    ReDim spath(NSteps, 1 To NRepl)
    Sum = 0
    cross = 0
    spath = AssetPaths(S, r, q, T, sigma, NSteps, NRepl)
    For j = 1 To NRepl
        payoff = Application.Max(X - spath(NSteps, j), 0)
        For i = 1 To NSteps
            If spath(i, j) <= Sb Then
            payoff = 0
            i = NSteps
            cross = cross + 1
            End If
        Next i
        Sum = Sum + payoff
    Next j
    temp(0) = Exp(-r * T) * Sum / NRepl
    temp(1) = cross
    DOPutMC_2 = temp
End Function
```

Using the above function, we can get two outcomes in the cells, H5:I5. H5 is down-and-out put option value and I5 is the times that the price crosses the barrier. The formula for option price and crossed number in cells, H5:I5 is

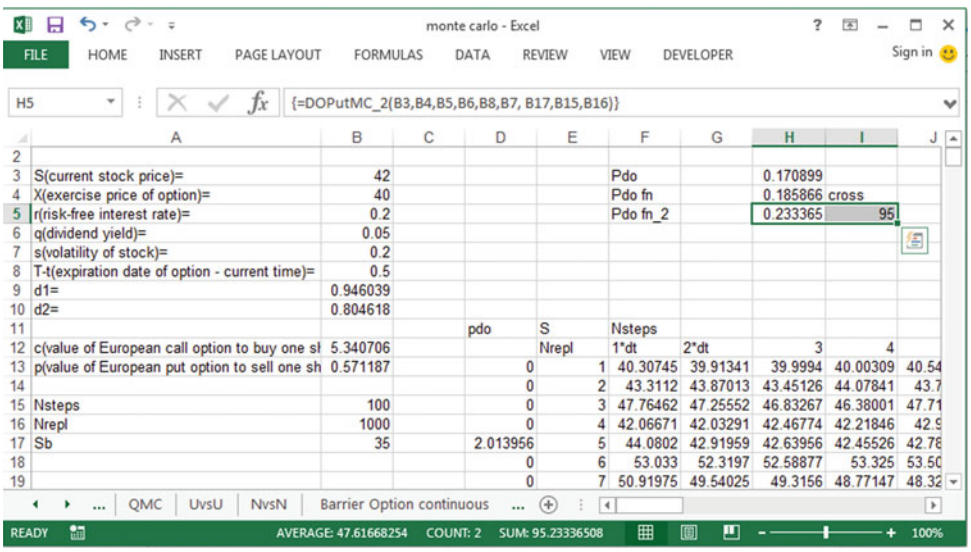
$$= \text{DOPutMC_2}(\text{B3}, \text{B4}, \text{B5}, \text{B6}, \text{B8}, \text{B7}, \ \text{B17}, \text{B15}, \text{B16}).$$

We should mark the range H5:I5, then type the formula. Finally, press the [ctrl] + [shift] + [enter]. Then we can get the result.

In order to see the different crossed numbers, we set two barriers, Sb. In the first case, Sb is equal to 5. Because barrier Sb in this case is 5, much below exercise and stock price, there is no price that crosses the barrier.



In the second case, Sb is equal to 35. We can see in this case, Sb = 35 is near the strike price of 40. Hence, there are 95 times that stock price crosses the barrier.



10.6 Summary

Monte Carlo Simulation consists of using random numbers to generate a stochastic stock price. Traditionally, we use the random generator in the Excel, rand(). However, it takes a lot of time to run a Monte Carlo simulation. In this chapter, we introduce antithetic variates to improve the efficiency of the simulation. In addition, owing to random number generate from random generator is not discrepancy. We generate Halton's sequence, a non-random number, and use Box–Muller to generate normal samples. Then we can run a Quasi-Monte Carlo simulation, which produces a smaller error of estimation. In the application, we apply Monte Carlo

simulation to the path-depend option. We simulate all the underlying asset price processes to the price barrier option which is one kind of path-depend option.

Appendix 10.1: EXCEL CODE—Share Price Paths

```
'Native code to generate share price paths by Monte Carlo simulation
Sub shareprice()
    Dim nudt, sidt, Sum, randns
    Dim i As Integer
    Randomize
    Range("A15:d200").Select
    Selection.ClearContents
    S = Cells(4, 2)
    X = Cells(5, 2)
    r = Cells(6, 2)
    q = Cells(7, 2)
    T = Cells(9, 2)
    sigma = Cells(8, 2)
    NSteps = Cells(11, 2)
    nudt = (r - q - 0.5 * sigma ^ 2) * (T / NSteps)
    sidt = sigma * Sqr(T / NSteps)
    Sum = 0
    Cells(14, 1) = Cells(11, 1)
    Cells(14, 2) = "stock price 1"
    Cells(14, 3) = "stock price 2"
    Cells(14, 4) = "stock price 3"
    Cells(15, 1) = 1
    Cells(15, 2) = Cells(4, 2)
    Cells(15, 3) = Cells(4, 2)
    Cells(15, 4) = Cells(4, 2)
    For i = 2 To NSteps
        randns = Application.NormSInv(Rnd)
        Cells(14 + i, 2) = Cells(14 + i - 1, 2) * Exp(nudt + randns *
sidt)
        randns = Application.NormSInv(Rnd)
        Cells(14 + i, 3) = Cells(14 + i - 1, 3) * Exp(nudt + randns *
sidt)
        randns = Application.NormSInv(Rnd)
        Cells(14 + i, 4) = Cells(14 + i - 1, 4) * Exp(nudt + randns *
sidt)
        Cells(14 + i, 1) = i
    Next i
End Sub
```

References

Boyle, Phelim P. "Options: A monte carlo approach." Journal of financial economics 4.3 (1977): 323-338

Boyle, Phelim, Mark Broadie, and Paul Glasserman. "Monte Carlo methods for security pricing." Journal of economic dynamics and control 21.8 (1997): 1267-1321.

Joy, Corwin, Phelim P. Boyle, and Ken Seng Tan. "Quasi-Monte Carlo methods in numerical finance." Management Science 42.6 (1996): 926–938.

Wilmott, Paul. Paul Wilmott on quantitative finance. John Wiley & Sons, 2013.

Hull, John C. Options, Futures, and Other Derivatives. Prentice Hall, 2015

On the Web

http://roth.cs.kuleuven.be/wiki/Main_Page

Part III

Applications of Python, Machine Learning for Financial Derivatives and Risk Management

Linear Models for Regression

11

11.1 Introduction

The goal of regression is to predict the target value y as a function $f(\boldsymbol{x})$ of the d-dimensional input variables $\boldsymbol{x}$, where the underlying function f is unknown (Altman and Krzywinski 2015). Examples of regression include predicting the GDP using the inflation x, to predict cancer or not ($y = 0,1$) using a patient's X-ray image $\boldsymbol{x}$. The former example is the case of a regression problem with continuous target variable y, while the second example is a classification problem. In either case, our objective will choose a specific function $f(\boldsymbol{x})$ for each input $\boldsymbol{x}$. A polynomial is a specific example of a broad class of the functions to proxy the underlying function f. A more useful class of functions known as linear combinations of a set of basis functions, which are linear in the parameters but nonlinear with respect to the input variables, gives simple analytical properties for the estimation and prediction purpose.

To choose $f(\boldsymbol{x})$ for the underlying function, we incur a loss $L[y, f(\boldsymbol{x})]$ and the optimal function $f(\boldsymbol{x})$ is the one that minimizes the loss function. However, the loss function L depends on whether the problem is a regression with a continuous target variable or classification (Altman and Krzywinski 2015). In the following section, we start with the former case. In the following, we will start from a regression problem with a continuous target variable y, in which the underlying function f is modeled as a linear combination of a set of basis functions.

This chapter is broken down into the following sections. Section 11.2 discusses loss functions and least squares, Sect. 11.3 discusses regularized least squares—Ridge and Lasso regression, and Sect. 11.4 discusses a logistic regression for classification: a discriminative model. Section 11.5 talks about K-fold cross-validation, and Sect. 11.6 discusses the types of basis functions. Section 11.7 looks at the accuracy of measures in classification, and Sect. 11.8 is a Python programming example. Finally, Sect. 11.9 summarizes the chapter.

11.2 Loss Functions and Least Squares

Consider a training dataset of N examples with the inputs $\{\boldsymbol{x}_i | i = 1, \ldots, N\} \subset R^D$, the target is the sum of the model function $f(\boldsymbol{x}_i)$ and the noise ε_i, i.e.,

$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i \tag{11.1}$$

where $1 \leq i \leq N$, $\varepsilon_1, \ldots, \varepsilon_N$ are *i.i.d.* Gaussian noises with means zeros and variance γ^{-1}. In many practical applications, the d-dimensional $\boldsymbol{x}$ is preprocessed to result in the features expressed in terms of a set of basis functions $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_0(\boldsymbol{x}), \ldots, \phi_M(\boldsymbol{x})]'$, and the model output is

$$f(\boldsymbol{x}_i) = \sum_{j=0}^{M} \phi_j(\boldsymbol{x}_i) w_j = \phi(\boldsymbol{x}_i)' \boldsymbol{w} \tag{11.2}$$

where $\boldsymbol{\phi}(\boldsymbol{x}_i) = [\phi_0(\boldsymbol{x}_i), \ldots, \phi_M(\boldsymbol{x}_i)]'$ is a set of M basis functions $\{\phi_j(\boldsymbol{x}_i) | j = 0, \ldots, M\}$, and $\boldsymbol{w} = [w_0, \ldots, w_M]'$ are the corresponding weight parameters. Typically, $\phi_0(\boldsymbol{x}) = 1$, so that w_0 acts as a bias. Popular basis functions are given in Sect. 31–3. To find an estimator $\hat{y}$ of the target variable y, one often considers the squared-error loss function

$$L[y, \hat{y}(\boldsymbol{x})] = (y - \hat{y}(\boldsymbol{x}))^2$$

Suppose the estimator $\hat{y}$ is the one that minimizes the expected loss function given by

$$E(L) = \iint L[y, \hat{y}(\boldsymbol{x})] p(\boldsymbol{x}, y) d\boldsymbol{x} dy \tag{11.3}$$

where $p(\boldsymbol{x}, y)$ is the joint probability function of $\boldsymbol{x}$ and y. As the noises $\varepsilon_1, \ldots, \varepsilon_N$ in (11.1) are *i.i.d.* Gaussian with means zeros and variance γ^{-1}, it can be shown the estimator $\hat{y}(\boldsymbol{x})$ that minimizes the expected squared-error loss function $E(L)$ in (11.3) is simply the conditional mean

$$\hat{y}(\boldsymbol{x}) = E(y|\boldsymbol{x}) = f(\boldsymbol{x}).$$

Therefore, like all forms of regression analysis, the focus is on the conditional probability distribution $p(y|\boldsymbol{x})$ rather

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_11

than on the joint probability distribution $p(\boldsymbol{x}, y)$. In the following section, if the model function $f(\boldsymbol{x})$ is given in the form of (11.2), we discuss the procedure to obtain the estimates of the weight parameters $\boldsymbol{w}$, and thus an estimate of the model on $f(\boldsymbol{x})$.

11.3 Regularized Least Squares—Ridge and Lasso Regression

Suppose we want to estimate the model function $f(\boldsymbol{x})$ in (11.1) given a training dataset $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)$. Recall in (11.1), the noises $\varepsilon_1, \ldots, \varepsilon_N$ are *i.i.d.* Gaussian with means zeros and variance γ^{-1}, thus the conditional probability distribution $p(y_i|\boldsymbol{x}_i)$, $1 \leq i \leq N$, is Gaussian with mean $f(\boldsymbol{x}_i)$ and variance γ^{-1}. Suppose the model function $f(\boldsymbol{x}_i)$ is given by (11.2), then the joint likelihood is

$$exp\left\{\vartheta - \frac{\sqrt{\gamma}}{2}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})^2\right\}$$

The estimates of $\boldsymbol{w}$ that minimizes the expected loss function (11.3) are the ones that maximize the log-likelihood function

$$l = \vartheta - \frac{\sqrt{\beta}}{2}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})^2$$

Maximizing the log-likelihood function l is equivalent to minimizing the sum-of-squares error function

$$Er_0(\boldsymbol{w}) = \sum_{i=1}^{N}(y_i - \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})^2 \tag{11.4}$$

The estimates $\widehat{\boldsymbol{w}}$ of the weight parameters $\boldsymbol{w}$ that minimize the sum-of-squares error function are called the least-squared estimates.

One rough heuristic is that increing the dimension M of the features $\boldsymbol{\phi}(\boldsymbol{x})$ decreases the sum-of-squares error Er_0 and therefore increases the fit of the model. However, it will increase the model complexity and result in the overfitting problem. The overfitting problem becomes more prevalent as the number of training data points is limited (Gruber 1998; Kaufman and Rosset 2014). One solution to control the overfitting phenomenon is to add a penalty term to the error function to discourage the weight parameters $\boldsymbol{w}$ from reaching larger values. This technique to resolve the overfitting phenomenon is called regularization (Friedman et al. 2010). There are two types of penalty terms often used that lead to two different regression cases (Coad and Srhoj 2020; Tibshirani 1997):

(1) **Ridge Regression**: The modified sum-of-squares error function is

$$Er_1(\boldsymbol{w}) = \sum_{i=1}^{N}(y_i - \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})^2 + \lambda\sum_{j=0}^{M}w_j^2 \tag{11.5a}$$

(2) **Lasso Regression**: The modified sum-of-squares error function is

$$Er_2(\boldsymbol{w}) = \sum_{i=1}^{N}(y_i - \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})^2 + \lambda\sum_{j=0}^{M}\left|w_j^2\right| \tag{11.5b}$$

where the coefficient λ governs the relative importance of the regularization term compared with the sum-of-squares error term.

11.4 Logistic Regression for Classification: A Discriminative Model

Consider first the case of two classes C_1 and C_2, and we want to classify between classes C_1 and C_2 (i.e., the target $y = 0,1$) based on the model function $f(\boldsymbol{x})$. There are two approaches to choose the model function $f(\boldsymbol{x})$ (Hosmer 1997). The first approach, or the generative model approach, models the joint probability density function $p(\boldsymbol{x}, y)$ directly. The second approach, or the discriminative model approach, models the posterior class probability.

$$p(C_k|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|C_k)p(C_k)}{p(\boldsymbol{x}|C_1)p(C_1) + p(\boldsymbol{x}|C_2)p(C_2)}$$

where $p(\boldsymbol{x}|C_k)$ and $p(C_k)$ are the class-conditional density function and the prior, respectively, $k = 1,2$. The logistic regression approach is a discriminative model approach, in which the posterior probability $p(C_1|\boldsymbol{x})$ is modeled by an *S*-shaped *logistic sigmoid* function $\sigma(\cdot)$ on a linear function of the features or of a set of basis functions $\boldsymbol{\phi}(\boldsymbol{x}) = [\boldsymbol{\phi}_0(\boldsymbol{x}), \ldots, \boldsymbol{\phi}_M(\boldsymbol{x})]'$, i.e.,

$$p(C_1|\boldsymbol{x}) = \sigma(f(\boldsymbol{x})) \tag{11.6}$$

where $f(\boldsymbol{x}) = \phi(\boldsymbol{x}_i)\boldsymbol{w}$ and σ is the *logistic sigmoid* function

$$\sigma(a) = \frac{1}{1 + e^{-a}}$$

As the inverse of the logistic sigmoid is the *logit* function given by

$$a = \ln\left(\frac{\sigma}{1 - \sigma}\right)$$

Thus, one has $f(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i)\boldsymbol{w}$, $1 \leq i \leq N$, as the odds ratio

$$\phi(\boldsymbol{x}_i)\boldsymbol{w} = \ln\left(\frac{\pi_i}{1-\pi_i}\right)$$

where $\pi_i = p(C_1|\boldsymbol{x}_i)$, $1 \leq i \leq N$. For this reason, (11.6) is termed logistic regression. For a training dataset $(\boldsymbol{x}_1, y_1)$, …, $(\boldsymbol{x}_N, y_N)$, the likelihood function is

$$l = \prod_{i=1}^{N} \pi_i^{y_i}(1-\pi_i)^{N-y_i}$$

By taking the negative logarithm of the likelihood l, we obtain the error function in the terms of the cross-entropy form

$$E(l) = -\sum_{i=1}^{N}\{y_i ln(\pi_i) + (N-y_i)ln(1-\pi_i)\} \qquad (11.7)$$

There is no closed-form solution for the cross-entropy error function in (11.7) due to the nonlinearity of the logistic sigmoid function σ in (11.6). However, as the cross-entropy error function (11.7) is concave, thus a unique minimum exists and an efficient iterative technique by taking the gradient of the error function in (11.7) with respect to $\boldsymbol{w}$ based on the Newton–Raphson iterative optimization scheme can be applied.

Extension of the two-class classifier for classification to $K > 2$ classes, we can use either of the following algorithms:

(1) One-versus-the-rest classifier: Using $(K-1)$ of two-class classifiers, each of the two-class classifiers solves a two-class classification problem of separating class C_k from other classes, $1 \leq k \leq K$.
(2) One-versus-one classifier: Using $K(K-1)/2$ of two-class classifiers, one for every possible pair of classes.

11.5 *K*-fold Cross-Validation

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample in order to estimate how the model is expected to perform in general when used to make predictions on data not used during the training of the model (Kohavi 1995). This approach involves randomly dividing the set of observations into K groups, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K-1$ folds. As such, the procedure is often called K-fold cross-validation. When a specific value for K is chosen, it may be used in place of K in the reference to the model, such as $K = 10$ becoming tenfold cross-validation.

Cross-validation is a popular method because it is simple to understand and it generally results in less biased than other methods, such as a simple train/test split. The general procedure is as follows:

1. Shuffle the dataset randomly;
2. Split the dataset into K groups;
3. For each group
   (a) Take the group as a hold-out or test dataset, and the remaining groups as a training dataset;
   (b) Fit a model on the training set and evaluate it on the test set;
   (c) Retain the evaluation score and discard the model;
   (d) Summarize the result of the model using the sample of model evaluation scores.

The K value must be chosen carefully for your data sample. A poorly chosen value for K may result in a misrepresentative idea of the model, such as a score with a high variance or a high bias. Three common tactics for choosing a value for K are as follows:

- **Representative**: The value for K is chosen such that each train/test group of data samples is large enough to be statistically representative of the broader dataset.
- $K = \mathbf{10}$: The value for K is fixed to 10, a value that has been found through experimentation to generally result in a model estimate with low bias and a modest variance.
- $K = \boldsymbol{n}$: The value for K is fixed to the size of the dataset n to give each test sample an opportunity to be used in the hold-out dataset. This approach is called leave-one-out cross-validation.

The results of a K-fold cross-validation run are often summarized with the mean of the model skill scores. It is also good practice to include a measure of the variance of the scores, such as the standard deviation or standard error.

11.6 Types of Basis Function

The world is complicated that most regression problems don't really map linear to real-valued vectors in the d-dim vector space. To overcome this problem, features or basis functions that turn various kinds of inputs into numerical vectors are introduced. Three types of basis functions are given as follows:

1. Polynomial basis functions:

$$\phi_j(x) = x^j$$

Global: a small change in x affects all basis functions.

2. Gaussian Basis Functions:

$$\phi_j(x) = exp\left\{\frac{(x-\mu_j)^2}{2s^2}\right\}$$

Local: a small change in x only affects nearby basis functions.
μ_j and s control location and scale (width).

3. Logistic sigmoidal basis function:

$$\phi_j(x) = \sigma\left\{\frac{x-\mu_j}{s}\right\}$$

where $\sigma(a) = \frac{1}{1+e^{-a}}$

11.7 Accuracy Measures in Classification

Let us assume for simplicity to have a two-class problem, in which a diagnostic test discriminates between subjects affected by a disease (patients) and healthy subjects (controls). Accuracy measures for binary classification can be described in terms of four values as follows:

- TP or true positives, the number of correctly classified patients;
- TN or true negatives, the number of correctly classified controls;
- FP or false positives, the number of controls classified as patients;
- FN or false negatives, the number of patients classified as controls.

Note TP + TN + FP + FN = n, where n is the number of examples in the dataset. These values can be arranged in a 2×2 matrix called contingency matrix in the following:

Predicted	Positive	Negative
Actual Positive	TP	FN
Negative	FP	TN

Four error measures are associated with the contingency matrix, which are given as follows:

1. Sensitivity (also known as recall) is defined as the proportion of true positives on the total number of positive examples:
   Sensitivity = TP/(TP + FN).
2. Specificity (also referred to as precision) is defined as the proportion of true positives on the total number of examples classified as positive:
   Specificity = TP/(TP + FP).
3. The percentage of correctly classified positive instances:
   Accuracy =(TP + TN)/n.
4. F-score has been introduced to balance between sensitivity and specificity. It is defined as the harmonic mean of the sensitivity and specificity, multiplied by 2.

Since the choice of the accuracy measure to optimize greatly affects the selection of the best model, then the proper score should be determined taking into account the goal of the analysis. When performing model selection in a binary classification problem, e.g., when selecting the best threshold for a classifier with a continuous output, a reasonable criterion is to find a compromise between the amount of false positives and the amount of false negatives. The receiver operating characteristic (ROC) curve is a graphical representation of the true positive rate (the sensitivity) as a function of the false positive rate (the so-called false alarm rate, computed as FP/(FP + TN)). A good classifier would be represented by a point near the upper left corner of the graph and far from the diagonal. An indicator related to the ROC curve is the area under the curve (AUC), which is equal to 1 for a perfect classifier and to 0.5 for a random guess.

11.8 Python Programming Example

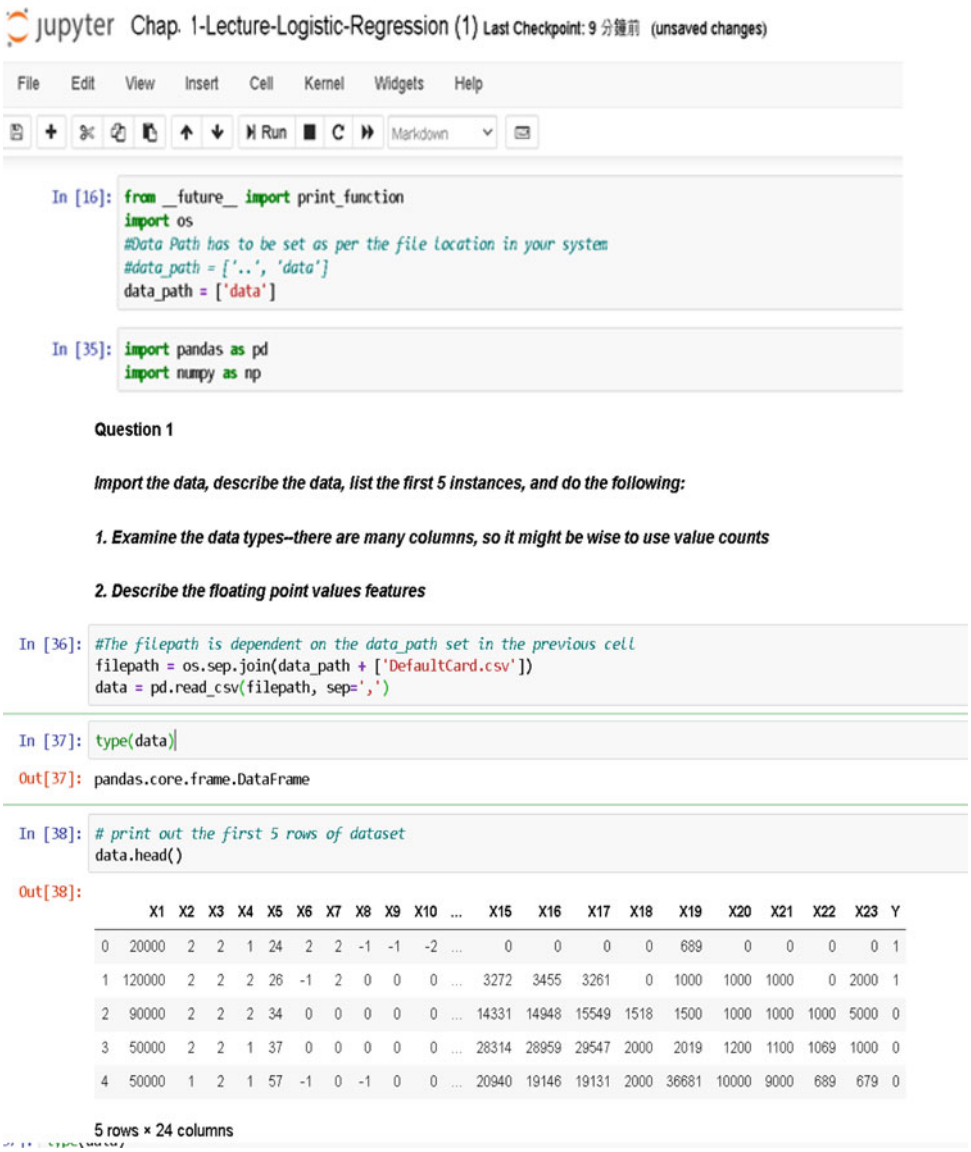
Consider the dataset of credit card holders' payment data in October, 2005, from a bank (a cash and credit card issuer) in Taiwan. Among the total 25,000 observations, 5529 observations (22.12%) are cardholders with default payments. Thus, the target variable y is the default payment (Yes = 1, No = 0), and the explanatory variables are the following 23 variables:

- X_1: Amount of the given credit (NT dollar): It includes both the individual consumer credit and his/her family (supplementary) credit.
- X_2: Gender (1 = male; 2 = female).
- X_3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
- X_4: Marital status (1 = married; 2 = single; 3 = others).
- X_5: Age (year).
- X_6-X_{11}: History of past payments from September to April, 2005;
  (The measurement scale for the repayment status is as follows: −1 = pay duly; 1 = payment delay for one

month; 2 = payment delay for two months;...; 8 = payment delay for eight months; 9 = payment delay for nine months and above).

- X_{12}–X_{17}: Amount of bill statement from September to April, 2005.
- X_{18}–X_{23}: Amount of previous payment (NT dollar) from September to April, 2005.

Questions and Problems for Coding



jupyter Chap. 1-Lecture-Logistic-Regression (1) Last Checkpoint: 9 分鐘前 (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Run C Markdown

In [16]:
```
from __future__ import print_function
import os
#Data Path has to be set as per the file location in your system
#data_path = ['..', 'data']
data_path = ['data']
```

In [35]:
```
import pandas as pd
import numpy as np
```

Question 1

Import the data, describe the data, list the first 5 instances, and do the following:

1. Examine the data types--there are many columns, so it might be wise to use value counts

2. Describe the floating point values features

In [36]:
```
#The filepath is dependent on the data_path set in the previous cell
filepath = os.sep.join(data_path + ['DefaultCard.csv'])
data = pd.read_csv(filepath, sep=',')
```

In [37]:
```
type(data)
```

Out[37]: pandas.core.frame.DataFrame

In [38]:
```
# print out the first 5 rows of dataset
data.head()
```

Out[38]:

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	...	X15	X16	X17	X18	X19	X20	X21	X22	X23	Y
0	20000	2	2	1	24	2	2	-1	-1	-2	...	0	0	0	0	689	0	0	0	0	1
1	120000	2	2	2	26	-1	2	0	0	0	...	3272	3455	3261	0	1000	1000	1000	0	2000	1
2	90000	2	2	2	34	0	0	0	0	0	...	14331	14948	15549	1518	1500	1000	1000	1000	5000	0
3	50000	2	2	1	37	0	0	0	0	0	...	28314	28959	29547	2000	2019	1200	1100	1069	1000	0
4	50000	1	2	1	57	-1	0	-1	0	0	...	20940	19146	19131	2000	36681	10000	9000	689	679	0

5 rows × 24 columns

In [39]:
```
print(data.shape)
print(data.shape[0])   # print Number of rows
print(data.shape[1])   # print Number of columns
```

```
(30000, 24)
30000
24
```

In [21]:
```
data.describe()
```

Out[21]:

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	...
count	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	...
mean	167484.322667	1.603733	1.853133	1.551867	35.485500	-0.016700	-0.133767	-0.166200	-0.220667	-0.266200	...
std	129747.661567	0.489129	0.790349	0.521970	9.217904	1.123802	1.197186	1.196868	1.169139	1.133187	...
min	10000.000000	1.000000	0.000000	0.000000	21.000000	-2.000000	-2.000000	-2.000000	-2.000000	-2.000000	...
25%	50000.000000	1.000000	1.000000	1.000000	28.000000	-1.000000	-1.000000	-1.000000	-1.000000	-1.000000	...
50%	140000.000000	2.000000	2.000000	2.000000	34.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...
75%	240000.000000	2.000000	2.000000	2.000000	41.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...
max	1000000.000000	2.000000	6.000000	3.000000	79.000000	8.000000	8.000000	8.000000	8.000000	8.000000	...

8 rows × 24 columns

In [22]:
```
print(data.columns.tolist())   # print Column (feature) names
```

```
['X1', 'X2', 'X3', 'X4', 'X5', 'X6', 'X7', 'X8', 'X9', 'X10', 'X11', 'X12', 'X13', 'X14', 'X15', 'X16', 'X17', 'X18', 'X19', 'X20', 'X21', 'X22', 'X23', 'Y']
```

In [24]:
```
data.dtypes.value_counts()        # summary of Data types of the 23 columns
```

Out[24]:
```
int64    24
dtype: int64
```

In [26]:
```
## Frequency of the feature y: the default payment (Yes = 1, No = 0),
data["Y"].value_counts()
```

Out[26]:
```
0    23364
1     6636
Name: Y, dtype: int64
```

In [29]:
```
# print the first 20 rows of columns 1:5
print(data.iloc[1:20, :5])
```

```
       X1  X2  X3  X4  X5
1   120000   2   2   2  26
2    90000   2   2   2  34
3    50000   2   2   1  37
4    50000   1   2   1  57
5    50000   1   1   2  37
6   500000   1   1   2  29
7   100000   2   2   2  23
8   140000   2   3   1  28
9    20000   1   3   2  35
10  200000   2   3   2  34
11  260000   2   1   2  51
12  630000   2   2   2  41
13   70000   1   2   2  30
14  250000   1   1   2  29
15   50000   2   3   3  23
16   20000   1   1   2  24
17  320000   1   1   1  49
18  360000   2   1   1  49
19  180000   2   1   2  29
```

Question 3

1. Split the data into train and test data sets.

Using Scikit-learn's `StratifiedShuffleSplit` to maintain the same ratio of predictor classes.

2. Compare the ratio of classes in both the train and test splits.

In [61]:
```
from sklearn.model_selection import StratifiedShuffleSplit

# StratifiedShuffleSplit uses stratified sampling to choose training and test dataset according to "Y"

# Get the split indexes
strat_shuf_split = StratifiedShuffleSplit(n_splits=1,test_size=0.3, random_state=42)
train_idx, test_idx = next(strat_shuf_split.split(data[feature_cols], data.Y))   # Recall "feature_cols" has the names of all the featur


# Create the training dataframes
X_train = data.loc[train_idx, feature_cols]
y_train = data.loc[train_idx, 'Y']


# Create the testing dataframes
X_test  = data.loc[test_idx, feature_cols]
y_test  = data.loc[test_idx, 'Y']
```

```
In [62]: y_train.value_counts(normalize=True)
Out[62]: 0    0.77881
         1    0.22119
         Name: Y, dtype: float64

In [63]: y_test.value_counts(normalize=True)
Out[63]: 0    0.778778
         1    0.221222
         Name: Y, dtype: float64
```

Question 4

1. Fit a logistic regression model without any regularization using all of the features.

2. Using cross validation to determine the hyperparameters,

3. Fit models using L1, and L2 regularization. Store each of these models as well.

4. Note the limitations on multi-class models, solvers, and regularizations. The regularized models, in particular the L1 model, will probably take a while to fit.

```
In [65]: from sklearn.linear_model import LogisticRegression

In [77]: # Standard logistic regression without regularization

         lr = LogisticRegression().fit(X_train, y_train)   # binary-class logistic regression without any regularization using all of the features
                                                           # lr.coef_:1x23 contains the estimated coeffs of the linear logistic model
         print(lr.coef_)

C:\Users\USER\Anaconda3\lib\site-packages\sklearn\linear_model\logistic.py:432: FutureWarning: Default solver will be changed to 'lbfg
s' in 0.22. Specify a solver to silence this warning.
  FutureWarning)

[[-3.22981110e-06 -1.33732146e-03 -1.49979011e-03 -1.45479663e-03
  -1.63484987e-02  3.16086549e-03  2.48869450e-03  2.15974595e-03
   2.01793206e-03  1.80709418e-03  1.70818626e-03 -7.23917410e-06
   2.61712397e-06  2.37540428e-06  1.35531177e-06  2.59979022e-06
   2.61746483e-06 -2.25317492e-05 -2.37187436e-05 -6.94151228e-06
  -7.57835620e-06 -4.03149788e-06 -2.19894724e-06]]

In [78]: from sklearn.linear_model import LogisticRegressionCV

In [79]: # L1 regularized logistic regression
         # Cs is the inverse of regularization strength

         lr_l1 = LogisticRegressionCV(Cs=10, cv=4, penalty='l1', solver='liblinear').fit(X_train, y_train)

         print(lr_l1.coef_)       # lr_l1.coef_:1x23 contains the estimated coeffs of the linear logistic model

[[-8.99341776e-07 -1.05927859e-01 -1.21467728e-01 -1.51363801e-01
   6.32776003e-03  5.82118947e-01  8.65953610e-02  7.17824731e-02
   5.01064988e-02  1.30068620e-02  2.02303565e-02 -4.36167997e-06
   2.43344924e-07  1.44589414e-06  6.26410658e-07 -1.95515099e-08
   7.65885549e-07 -9.36189977e-06 -1.23018023e-05 -1.38215954e-06
  -2.93152636e-06 -1.71257243e-06 -2.16107336e-06]]
```

Try with different solvers: 'newton-cg', 'lbfgs', 'sag', 'saga'

ps.'newton-cg': the conjugate gradient method to unconstrained nonlinear optimization.

ps. 'lbfgs': Limited-memory BFGS in a family of quasi-Newton methods

```
n [80]: # l2 regularized logistic regression
        lr_l2 = LogisticRegressionCV(Cs=10, cv=4, penalty='l2',solver='lbfgs').fit(X_train, y_train)
        print(lr_l2.coef_)      # lr_l2.coef_:1x23 contains the estimated coeffs of the linear logistic model
```

Question 5

1. Compare the magnitudes of the coefficients for each set under 3 models, each set of coefficients can be plotted separately

```
In [128]: # Combine all the coefficients in the three models lr, l1, and l2 into a dataframe

          coefficients = list()

          coeff_labels = ['lr', 'l1', 'l2']
          coeff_models = [lr, lr_l1, lr_l2]

          for lab,mod in zip(coeff_labels, coeff_models):
              coeffs = mod.coef_
              coeff_label = pd.MultiIndex(levels=[[lab],['coef']],labels=[[0],[0]])
              coefficients.append(pd.DataFrame(coeffs.T, columns=coeff_label))

          coefficients = pd.concat(coefficients, axis=1)
          print(coefficients)
```

```
           lr            l1            l2
         coef          coef          coef
0   -0.000003 -8.993418e-07 -4.655806e-06
1   -0.001337 -1.059279e-01 -3.311126e-04
2   -0.001500 -1.214677e-01 -4.159627e-04
3   -0.001455 -1.513638e-01 -3.415699e-04
4   -0.016348  6.327760e-03 -5.846043e-03
5    0.003161  5.821189e-01  4.996547e-04
6    0.002489  8.659536e-02  3.913155e-04
7    0.002160  7.178247e-02  3.316393e-04
8    0.002018  5.010650e-02  3.226453e-04
9    0.001807  1.300686e-02  2.935676e-04
10   0.001708  2.023036e-02  2.817792e-04
11  -0.000007 -4.361680e-06 -8.056214e-06
12   0.000003  2.433449e-07  2.341164e-06
13   0.000002  1.445894e-06  9.979900e-07
14   0.000001  6.264107e-07  2.445751e-06
15   0.000003 -1.955151e-08  3.836797e-06
16   0.000003  7.658855e-07  2.540217e-06
17  -0.000023 -9.361900e-06 -2.485904e-05
18  -0.000024 -1.230180e-05 -2.709697e-05
19  -0.000007 -1.382160e-06 -1.099968e-05
20  -0.000008 -2.931526e-06 -1.011589e-05
21  -0.000004 -1.712572e-06 -3.948574e-06
22  -0.000002 -2.161073e-06 -1.730953e-06
```

```
C:\Users\USER\Anaconda3\lib\site-packages\ipykernel_launcher.py:10: FutureWarning:
ead
  # Remove the CWD from sys.path while we load stuff.
```

Question 6

- Predict and store the class for each model.
- Also store the probability for the predicted class for each model.

In [146]:
```
# Predict the class and the probability for each

y_pred = list()
y_prob = list()

coeff_labels = ['lr', 'l1', 'l2']
coeff_models = [lr, lr_l1, lr_l2]

for lab,mod in zip(coeff_labels, coeff_models):

    pd.Series(mod.predict(X_test), name=lab)

    y_pred.append(pd.Series(mod.predict(X_test), name=lab))    # Store the predictions in a column array
    y_prob.append(pd.Series(mod.predict_proba(X_test).max(axis=1), name=lab))

y_pred = pd.concat(y_pred, axis=1)
y_prob = pd.concat(y_prob, axis=1)
```

In [144]:
```
y_pred.head()
```

In [144]:
```
y_pred.head()
```

Out[144]:

	lr	l1	l2
0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0

In [145]:
```
print(y_prob)
```

```
            lr        l1        l2
0     0.814773  0.834836  0.837927
1     0.691520  0.769400  0.647028
2     0.952052  0.918416  0.958365
3     0.650627  0.700208  0.566158
4     0.712751  0.764933  0.674664
...        ...       ...       ...
8995  0.703503  0.860681  0.732891
8996  0.646134  0.767643  0.603840
8997  0.711908  0.823060  0.695245
8998  0.889412  0.812139  0.910230
8999  0.711367  0.740820  0.655438

[9000 rows x 3 columns]
```

Question 7

For each model, calculate the following error metrics:

- accuracy
- precision
- recall
- fscore
- confusion matrix

Decide how to combine the multi-class metrics into a single value for each model.

In [148]:
```
from sklearn.metrics import precision_recall_fscore_support as score
from sklearn.metrics import confusion_matrix, accuracy_score, roc_auc_score
from sklearn.preprocessing import label_binarize    #Binarize labels in a one-vs-all fashion

metrics = list()
cm = dict()

for lab in coeff_labels:

    # Preciision, recall, f-score from the multi-class support function
    precision, recall, fscore, _ = score(y_test, y_pred[lab], average='weighted')

    # The usual way to calculate accuracy
    accuracy = accuracy_score(y_test, y_pred[lab])

    # ROC-AUC scores can be calculated by binarizing the data
```

References

Altman, Naomi; Krzywinski, Martin (2015). Simple linear regression. *Nature Methods*. **12** (11): 999–1000.

Coad, Alex; Srhoj, Stjepan (2020). Catching Gazelles with a Lasso: Big data techniques for the prediction of high-growth firms. *Small Business Economics*. **55** (1): 541–565.

Fu, Wenjiang J. 1998. The Bridge versus the Lasso. Journal of Computational and Graphical Statistics 7 (3). Taylor & Francis: 397–416.

Gruber, Marvin (1998). Improving Efficiency by Shrinkage: The James–Stein and Ridge Regression Estimators, CRC Press.

Hosmer, D.W. (1997). A comparison of goodness-of-fit tests for the logistic regression model. *Stat Med*. **16** (9): 965–980.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software 33 (1): 1–21.

Kaufman, S.; Rosset, S. (2014). When does more regularization imply fewer degrees of freedom? Sufficient conditions and counterexamples. *Biometrika*. **101** (4): 771–784.

Kohavi, Ron (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, **2** (12): 1137–1143.

Tibshirani, Robert (1997). The lasso Method for Variable Selection in the Cox Model. *Statistics in Medicine*. **16** (4): 385–395.

Kernel Linear Model

12

12.1 Introduction

The kernel concept was introduced into the field of pattern recognition by Aizerman et al. (1964). It was re-introduced into machine learning in the context of large margin classifiers by Boser et al. (1992). The kernel concept allows us to build interesting extensions of many well-known algorithms. These well-known algorithms require the raw data to be explicitly transformed into representations via a user-specified feature map. Instead, kernel methods, require only a user-specified similarity function over pairs of data points in raw representation. This dual representation of raw data arises the kernel trick, which enables them to operate in a high-dimensional, implicit feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. Any linear model can be turned into a nonlinear model by applying the kernel trick to the model: replacing its features (predictors) by a kernel function.

Algorithms capable of operating with kernels include the kernel regression, Gaussian process regression, support vector machines, principal components analysis (PCA), spectral clustering, linear adaptive filters, and many others. In the following, the ideas of kernel approach and its applications will be given.

The sections of this chapter are as follows. Section 12.2 discusses constructing kernels. Section 12.3 discusses the Nadaraya–Watson model of kernel regression, Sect. 12.4 talks about relevant vector machines, and Sect. 12.5 talks about the Gaussian process for regression. Section 12.6 discusses support vector machines, and Sect. 12.7 talks about Python programming.

12.2 Constructing Kernels

A kernel function corresponds to a scalar product in some feature space. For models based on a fixed nonlinear feature space mapping $\phi(\boldsymbol{x})$, the corresponding kernel function is the inner product

$$k(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^{\mathrm{T}} \cdot \phi(\boldsymbol{x}').$$

It is obvious a kernel function is a symmetric of its arguments, i.e., $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}', \boldsymbol{x})$. Some examples include.

1. **Liner Kernel**—$k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^{\mathrm{T}} \cdot \boldsymbol{x}'$.
2. **Polynomial Kernel**—$k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^{\mathrm{T}} \cdot \boldsymbol{x}' + 1)^d$, d is the degree of the polynomial.

There are many other forms of kernel functions in common use. One type of kernel functions is known as stationary kernels, which satisfy $k(\boldsymbol{x}, \cdot \boldsymbol{x}') = \phi(\boldsymbol{x} - \boldsymbol{x}')$. In other words, stationary kernels are functions of the difference between the arguments only and thus are invariant to translations in input space. Another type involves dial basis functions, which depend only on the magnitude of the distance (typically Euclidean) between the arguments so that $k(\boldsymbol{x}, \boldsymbol{x}') = \varphi(\|\boldsymbol{x} - \boldsymbol{x}'\|)$. The most well-known example is the Gaussian kernel:

3. **Gaussian Kernel**—$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2\right)$.

12.3 Kernel Regression (Nadaraya–Watson Model)

Radial basis functions, which depend only on the radial distance (typically Euclidean) from a center point, were introduced for the purpose of exact function interpolation (Powell 1987). Consider a set of training dataset of

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_12

N examples, $\{\boldsymbol{x}_i | i = 1, \ldots, N\}$ are the inputs; $\{y_i | i = 1, \ldots, N\}$ are the corresponding target values. The goal is to find a smooth function $f(\boldsymbol{x})$ that fits every target value as close as possible, which can be achieved by expressing $f(\boldsymbol{x})$ as a linear combination of radial basis functions, one centered on every data point

$$f(\boldsymbol{x}) = \sum_{j=1}^{N} \phi(\boldsymbol{x} - \boldsymbol{x}_i) y_j \tag{12.1}$$

where ϕ is a radial basis function.

As the inputs $\{\boldsymbol{x}_i | i = 1, \ldots, N\}$ are noisy, the kernel regression model (12.1) can be handled from a different perspective: starting with kernel density estimation in which the joint density function is given by

$$p(\boldsymbol{x}, y) = \frac{1}{N} \sum_{j=1}^{N} h(\boldsymbol{x} - \boldsymbol{x}_i, y - y_j)$$

where h is the component density function. By assuming for all $\boldsymbol{x}$, one has

$$\int h(\boldsymbol{x}, y) y dy = 0$$

The regression function $f(\boldsymbol{x})$ is now the conditional mean of the target variable y conditioned on the input variable $\boldsymbol{x}$ is zero

$$f(\boldsymbol{x}) = E[y|\boldsymbol{x}] = \frac{\int p(\boldsymbol{x}, y) y dy}{\int h(\boldsymbol{x}, y) dy} = \sum_{j=1}^{N} k(\boldsymbol{x}, \boldsymbol{x}_i) y_j \tag{12.2}$$

where the kernel function

$$k(\boldsymbol{x}, \boldsymbol{x}_i) = \frac{g(\boldsymbol{x} - \boldsymbol{x}_i)}{\sum_{j=1}^{N} g(\boldsymbol{x} - \boldsymbol{x}_j)} \tag{12.3}$$

and $g(\boldsymbol{x}) = \int h(\boldsymbol{x}, y) dy$. (12.2) is known as the Nadaraya–Watson model, or kernel regression (Nadaraya 1964; Watson 1964). For a localized kernel function, it has the property of giving more weight to the data points that are close to $\boldsymbol{x}$.

An example of the component density $h(\boldsymbol{x}, y)$ is the standard normal density. More general joint density $p(\boldsymbol{x}, y)$ involves a Gaussian mixture model, in which the number of components in the mixture model can be smaller than the number of training set points, resulting in a model that is faster to evaluate for test data points.

12.4 Relevance Vector Machines

The Relevance Vector Machine (RVM), a Bayesian sparse kernel technique for regression and classification, is introduced by Tipping (2001). As a Bayesian approach, it produces sparse solutions using an improper hierarchical prior and optimizing over hyper-parameters. In more specific, given a training dataset of N examples with the inputs $\{\boldsymbol{x}_i | i = 1_{z_{\text{mimi}}} N\} \subset R^D$, the target is the sum of the model output $f(\boldsymbol{x}_i)$ and the noise ε_i, i.e.,

$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i$$

where $1 \leq i \leq N$, the model output is

$$f(\boldsymbol{x}_i) = \sum_{j=1}^{N} \phi_j(\boldsymbol{x}_i) w_j = \phi(\boldsymbol{x}_i) \boldsymbol{w} \tag{12.4}$$

Here $\phi(\boldsymbol{x}_i)' = [\phi_1(\boldsymbol{x}_i), \ldots, \phi_N(\boldsymbol{x}_i)]$ is a set of N basis functions $\{\phi_j(\boldsymbol{x}_i) | j = 1, \ldots, N\}$, and $\boldsymbol{w} = [w_1, \ldots, w_N]'$ are the corresponding weights, $\varepsilon_1, \ldots, \varepsilon_N$ are *i.i.d.* Gaussian noises with means zeros and variance β^{-1}. Here the basis functions are given by kernels, with one kernel associated with each of the data points from the training set. It is assumed the prior on the weights $\boldsymbol{w}$ is Gaussian

$$p(\boldsymbol{w}|A) \sim N(\boldsymbol{0}, A^{-1}) \tag{12.5}$$

where $A = diag[\alpha_1, \ldots, \alpha_N]$ is a diagonal matrix with precision hyper-parameters $\alpha_1, \ldots, \alpha_N$. The N model outputs can be formulated as $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]' = \Phi \boldsymbol{w}$, where Φ is the $N \times N$ matrix with the (i, j)th entry $\Phi_{ij} = \phi_j(\boldsymbol{x}_i), 1 \leq i, j \leq N$. The likelihood is

$$p(\boldsymbol{y}|\boldsymbol{w}) \sim N(\Phi \boldsymbol{w}, \beta^1 I_N) \tag{12.6}$$

where $\boldsymbol{y} = [y_1, \ldots, y_N]'$ are the targets. The values of $\alpha_1, \ldots, \alpha_N$ and β are estimated using the evidence approximation, in which we maximize the marginal likelihood function

$$\int p(\boldsymbol{w}|A) p(\boldsymbol{y}|\boldsymbol{w}) d\boldsymbol{w}. \tag{12.7}$$

The posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$, which is proportional to the product of the prior $p(\boldsymbol{w}|A)$ and the likelihood (12.6), is given by

$$p(\boldsymbol{w}|\mathbf{y}) \sim N(\boldsymbol{m}, S_N) \tag{12.8}$$

where $\boldsymbol{m} = \beta S_N \Phi' \boldsymbol{y}$ and $S_N = [A + \beta \Phi' \Phi]^{-1}$ are the posterior mean and covariance of $\boldsymbol{m}$, respectively.

In the process of estimating $\alpha_1, \ldots, \alpha_N$ and β, a proportion of the hyper-parameters $\{\alpha_i\}$ are driven to large values, and so the weight parameters w_i, $1 \leq i \leq N$, corresponding to the large α_i has posterior distribution with mean and variance both zero. Thus the parameter w_i and the corresponding basis functions $\phi_i(\mathbf{x}), 1 \leq i \leq N$, are removed from the model and play no role in making predictions for new inputs, and are ultimately responsible for the sparsity property. On the other hand, the example $\boldsymbol{x}_i$ associated with the nonzero weight w_i

are termed "relevance" vectors. In another word, RVM satisfies the principle of *automatic relevance determination* (ARD) via the hyper-parameters α_i, $1 \le i \le N$ (Tipping 2001).

With the posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$, the predictive distribution $p(y_*|\boldsymbol{x}_*, \boldsymbol{y})$ of y_* at a new test input $\boldsymbol{x}_*$, obtained as the integration of the likelihood $p(y_*|\boldsymbol{x}_*)$ over the posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$, can be formulated as

$$p(y_*|\boldsymbol{x}_*, \boldsymbol{y}) \sim N(\boldsymbol{m}'\phi(\mathbf{x}_*), \sigma_*^2) \tag{12.9}$$

where the variance of the predictive distribution

$$\sigma_*^2 = \frac{1}{\beta} + \phi'(\boldsymbol{x}_*) S_N \phi(\boldsymbol{x}_*). \tag{12.10}$$

Here S_N is the posterior covariance given in (12.7).

If the N basis functions $\phi(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \ldots, \phi_N(\boldsymbol{x})]$ are localized with centers the inputs $\{\boldsymbol{x}_i | i = 1, \ldots, N\}$ of the training dataset of N examples, then as the test input $\boldsymbol{x}_*$ is located in region away from the N centers, the contribution from the second term in (12.10) will get smaller and leave only the noise contribution $1/\beta$. In another word, the model becomes very confident in its predictions when extrapolating outside the region occupied by the N centers of the training dataset, which is generally an undesirable behavior. For this reason, we consider a more appropriate model, namely, the Gaussian process regression, that avoids this undesirable behavior of RVM in the following section.

12.5 Gaussian Process for Regression

The Gaussian process regression, based on a non-degenerate kernel function, is a non-parametric approach so the parametric model $f(\boldsymbol{x}) = w'\varphi(\boldsymbol{x})$ in (12.4) is dispensed. Instead of imposing a prior distribution over $\boldsymbol{w}$, a prior distribution is imposing directly on the model outputs $\boldsymbol{f} = [f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]'$, namely,

$$p(\boldsymbol{f}) \sim N(\mathbf{0}_N, K) \tag{12.11}$$

where the covariance matrix K is a Gram matrix with the entry $K_{ij} = k[f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)]$, $1 \le i \le j \le N$, where k is a kernel function. Recall the target $y_i = f(\boldsymbol{x}_i) + \varepsilon_i$, $1 \le i \le N$, where $\varepsilon_1, \ldots, \varepsilon_N$ are *i.i.d.* Gaussian with means zeros and variance β^{-1}. Thus the predictive distribution can be formulated as $p(y_*|\boldsymbol{x}_*, \boldsymbol{y}) \sim N(m_G, \quad \sigma_{**}^2)$, where

$$m_G = \boldsymbol{k}^*[K + \beta I_N]^{-1}\boldsymbol{y} \tag{12.12}$$

$$\sigma_{**}^2 = k[\boldsymbol{x}_*, \boldsymbol{x}_*] - \boldsymbol{k}^{*\prime}[K + \beta I_N]^{-1}\boldsymbol{k}^* \tag{12.13}$$

Here $\boldsymbol{k}^{*\prime}$ is the row vector $\boldsymbol{k}^{*\prime} = (k[\boldsymbol{x}_1, \boldsymbol{x}_*], \ldots, k[\boldsymbol{x}_N, \boldsymbol{x}_*])$ I_N is the $N \times N$ identity matrix. If the $N \times N$ covariance matrix K is degenerate, i.e., K can be expanded by a set of finite basis functions, namely,

$$K = \Phi\Sigma\Phi'$$

where Φ is the $N \times M$ matrix with the $(i, j)^{\text{th}}$ entry $\Phi_{ij} = \phi_j(\boldsymbol{x}_i), 1 \le i \le N, 1 \le j \le M; \{\phi_1(\boldsymbol{x}), \phi_M(\boldsymbol{x})\}$ is a set of M basis functions; Σ is a $M \times M$ diagonal matrix. It can be shown that the predictive variance (12.13) is lesser as $\boldsymbol{k}^{*\prime}$ is in the direction of the eigenvectors corresponding to zero eigenvalues of the covariance matrix K, that is, the predictive variance (12.13) is lesser as $\Phi'\boldsymbol{k}^* = 0$. If the basis functions in Φ are localized basis functions, the same problem is met as in the RVM that the model becomes very confident in its predictions when extrapolating outside the region occupied by the basis functions. For the above reason, when adopting Gaussian process regression, covariance matrix K based on non-degenerate kernel function is considered.

Without the mechanism of *automatic relevance determination* (ARD), however, the main limitation of Gaussian process regression is memory requirements and computational demands grow as the square and cube, respectively, of the number of training examples N. To overcome the computational limitations, numerous authors have recently suggested a wealth of sparse approximations (Csat´o and Opper 2002; Seeger et al. 2003; Qui˜nonero-Candela and Rasmussen 2005; Snelson and Ghahramani 2006).

12.6 Support Vector Machines

Support-vector machines (SVMs), one of the most widely used classification algorithms in industrial applications developed by Vapnik (1997), are supervised machine learning models that analyze data for classification and regression analysis. As a non-probabilistic binary linear classifier, a set of training examples is given, each marked as belonging to one of two categories. And an SVM learning algorithm maps training examples to points in space so as to maximize the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In more specific, a data point $\boldsymbol{x}$ is viewed as a p-dimensional vector, and suppose we have N data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$,

$$f(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i)\boldsymbol{w} + b$$

where $\boldsymbol{\phi}(\boldsymbol{x})$ denotes a fixed feature-space transformation, b is the bias parameter. The N data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ are labeled

with their class y_i, where $y_i \in \{-1, 1\}$, $1 \leq i \leq N$. We want to find a $(p\text{-}1)$-dimensional hyperplane to separate these N data points according to their classes. There are many hyperplanes that might classify the two classes of the N datapoints. The best is the one that represents the largest separation, or margin, between the two classes of data points. If such a hyperplane exists, it is known as the *maximum-margin hyperplane* and the linear classifier it defines is known as a *maximum-margin classifier*; or equivalently, the *perceptron of optimal stability*. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin),

More formally, suppose the hyperplane that separate the two classes of data points is given by $f(\boldsymbol{x}) = 0$, then the perpendicular distance of a data point $\boldsymbol{x}$ from the hyperplane $f(\boldsymbol{x}) = 0$ takes the form

$$|f(\boldsymbol{x})|/\|\boldsymbol{w}\| = y[\phi(\boldsymbol{x})\boldsymbol{w} + b]/\|\boldsymbol{w}\| \tag{12.14}$$

where y is the label of the data point $\boldsymbol{x}$. Now the margin is defined as the perpendicular distance to the closest data point from the data set, say, $\boldsymbol{x}_n$, $1 \leq n \leq N$. The parameters $\boldsymbol{w}$ and b are those that maximize the margin in (12.14). The optimization problem is equivalent to minimize $\|\boldsymbol{w}\|^2$, subject to the constraint that

$$y_i\left[\phi(\boldsymbol{x_i})'\boldsymbol{w} + b\right] \geq 1 \tag{12.15}$$

for all $1 \leq i \leq N$. In the case the equality holds, the constraints are said to be active, whereas for the remainder they are said to be inactive. Any data point for which the equality holds is called a support vector and the remaining data points play no role in making predictions for new data points. By definition, there will always be at least one active constraint, because there will always be a closest point, and once the margin has been maximized there will be at least two active constraints. The dual representation of the maximum margin problem in (12.15) is to maximize

$$\sum_{i=1}^{N} a_i - \sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j y_i y_j k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right). \tag{12.16}$$

Subject to the constraint $a_i \geq 0$ for all $1 \leq i \leq N$, and

$$\sum_{i=1}^{N} a_i y_i = 0 \tag{12.17}$$

where the kernel function $k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \phi(\boldsymbol{x_i})\phi\left(\boldsymbol{x_j}\right)$. To solve the maximization problem (12.16)–(12.17), quadratic programming technique is required.

Once the maximization problem (12.16)–(12.17) is solved, the weight parameters are

$$\boldsymbol{w} = \sum_{i=1}^{N} a_i y_i \phi(\boldsymbol{x_i}).$$

In order to classify new data point $\boldsymbol{x}$ using the trained model, we evaluate the sign of $\boldsymbol{w}\phi(\boldsymbol{x}) + b$. As $\hat{y}[\boldsymbol{w}\phi(\boldsymbol{x}) + b]0$, thus $\hat{y} \geq 0$ if $[\boldsymbol{w}\phi(\boldsymbol{x}) + b]0$, otherwise $\hat{y} \leq 0$.

Whereas the above we consider a linear hyperplane, it often happens that the sets to discriminate are not linearly separable in that space. In addition to linear classification, the formulation of the objective function (12.16) allows SVMs to efficiently perform a nonlinear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. It was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space. To keep the computational load reasonable, the mappings are designed so that the dot products of pairs of input data points are defined by a kernel function to suit the problem.

12.7 Python Programming

Consider the dataset of credit card holders' payment data in October 2005, from a bank (a cash and credit card issuer) in Taiwan. Among the total 25,000 observations, 5529 observations (22.12%) are the cardholders with default payment. Thus the target variable y is the default payment (Yes = 1, No = 0), and the explanatory variables are the following 23 variables:

- X_1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
- X_2: Gender (1 = male; 2 = female).
- X_3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
- X_4: Marital status (1 = married; 2 = single; 3 = others).
- X_5: Age (year).
- X_6-X_{11}: History of past payment from September to April 2005.
  (The measurement scale for the repayment status is: −1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; ...; 8 = payment delay for eight months; 9 = payment delay for nine months and above).
- X_{12}-X_{17}: Amount of bill statement from September to April 2005.
- X_{18}-X_{23}: Amount of previous payment (NT dollar) from September to April 2005.

12.8 Kernel Linear Model and Support Vector Machines

We will be using "DefaultCard.csv" dataset. This data set contains 23 features. It also contains a binary category y ("Default") (yes = 1 or no = 0).

```
from __future__ import print_function
import os
#Please set the path below as per your system data folder location
#data_path = ['..', 'data']
data_path = [ 'data']
import pandas as pd
import numpy as np
filepath = os.sep.join(data_path + [ 'DefaultCard.csv'])
data = pd.read_csv(filepath, sep =',')
```

Question 1

- Create a pairplot for the dataset.
- Create a bar plot showing the correlations between each column and y
- Pick the most 2 correlated fields (using the absolute value of correlations) and create X
- Use MinMaxScaler to scale X. Note that this will output a np.array.
- Make it a DataFrame again and rename the columns appropriately.
- Create a pairplot for X8–X9 colored by "Default"

```
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

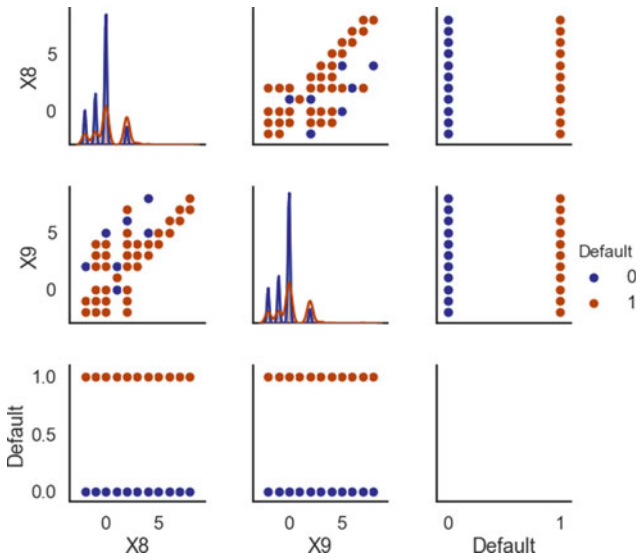
sns.set_context('talk')
sns.set_palette('dark')
sns.set_style('white')

fields = list(data.columns[7:9])

X=data[fields]

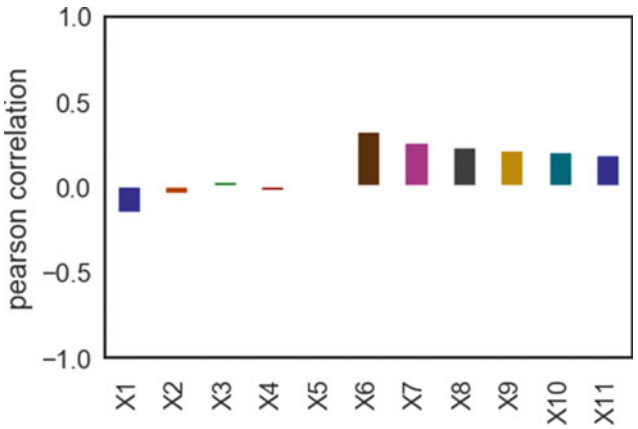
X['Default']=data["Y"]   # Add the last column "Default"

sns.pairplot(X, hue='Default')
```



Question 2a. Get the "correlations" between X1–X11 and y; and plot the bar plot

```
fields = list(data.columns[0:11])
y=data.Y
correlations = data[fields].corrwith(y)
ax = correlations.plot(kind='bar')
ax.set(ylim=[-1, 1],   ylabel='pearson correlation');
```



Question 2b. Sort "correlations" with/without absolute values

```
correlations.sort_values(inplace=True)

correlationsAbs=correlations.map(abs).sort_values()
```

Question 2c. Find the two x features with the largest absolute correlations with y; and obtain the feature matrix X

```
fields =correlationsAbs.iloc[-2:].index

X = data[fields]
```

Question 2D. Re-scale the two features using MinMaxScaler. Change X to a DataFrame, and change the titles of the two features as "xxx_scaled"

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

X = scaler.fit_transform(X)

X = pd.DataFrame(X, columns=['%s_scaled' % fld for fld in fields])
```

Question 3. Find the decision boundary of a Linear SVC classifier on this dataset.

- Fit a Linear Support Vector Machine Classifier to X, y.
- Pick 900 samples from X. Get the corresponding y value. Store them in variables X_default and y_default. This is because original dataset is too large and it produces a crowded plot.
- Modify y_defaultand get the new y_color so that it has the value "red" instead of 1 and 'yellow' instead of 0.
- Scatter plot X_default columns. Use the keyword argument "color = y_default" to color code samples.

```
from sklearn.svm import LinearSVC

LSVC = LinearSVC()

LSVC.fit(X, y)

X_default = X.sample(900, random_state=45)

y_default = y.loc[X_default.index]

y_color = y_default.map(lambda r: 'red' if r == 1 else 'blue')

ax = plt.axes()

ax.scatter(

   X_default.iloc[:, 0], X_default.iloc[:, 1],

   color=y_color, alpha=1)

# ------------------------------------------------------------------

x_axis, y_axis = np.arange(0, 1.00, .005), np.arange(0, 1.00, .005)

xx, yy = np.meshgrid(x_axis, y_axis)

xx_ravel = xx.ravel()

yy_ravel = yy.ravel()

X_grid = pd.DataFrame([xx_ravel, yy_ravel]).T

y_grid_predictions = LSVC.predict(X_grid)

y_grid_predictions = y_grid_predictions.reshape(xx.shape)

ax.contourf(xx, yy, y_grid_predictions, cmap=plt.cm.autumn_r, alpha=.3)

# ------------------------------------------------------------------

ax.set(

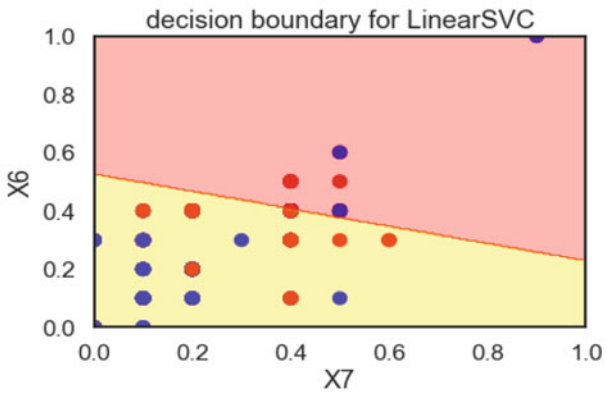
   xlabel=fields[0],

   ylabel=fields[1],
```

```
xlim=[0, 1],

ylim=[0, 1],

title='decision boundary for LinearSVC');
```



Question 4. Fit a Gaussian kernel SVC and see how the decision boundary changes

- Consolidate the code snippets in Question 3 into one function which takes in an estimator, X and y, and produces the final plot with decision boundary. The steps are
  1. fit model
  2. get sample 900 records from X and the corresponding y's
  3. create grid, predict, plot using ax.contourf
  4. add on the scatter plot

- After copying and pasting code make sure the finished function uses your input estimator and not the LinearSVC model you built.
- For the following values of gamma, create a Gaussian Kernel SVC and plot the decision boundary.
- gammas = [10, 20, 100, 200]
- Holding gamma constant, for various values of C, plot the decision boundary. You may try
- Cs = [0.1, 1, 10, 50]

```
def plot_decision_boundary(estimator, X, y):

  estimator.fit(X, y)


  X_default = X.sample(900, random_state=45)

  y_default = y.loc[X_default.index]


y_color = y_default.map(lambda r: 'red' if r == 1 else 'blue')

x_axis, y_axis = np.arange(0, 1, .005), np.arange(0, 1, .005)

xx, yy = np.meshgrid(x_axis, y_axis)

xx_ravel = xx.ravel()

yy_ravel = yy.ravel()

X_grid = pd.DataFrame([xx_ravel, yy_ravel]).T

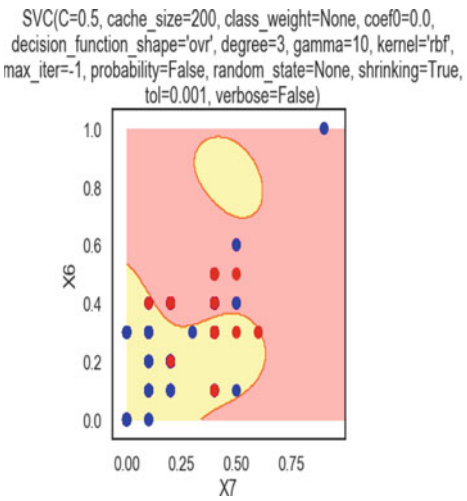
y_grid_predictions = estimator.predict(X_grid)

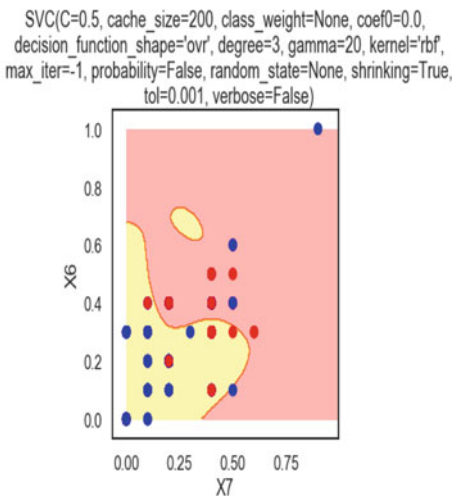
y_grid_predictions = y_grid_predictions.reshape(xx.shape)
```

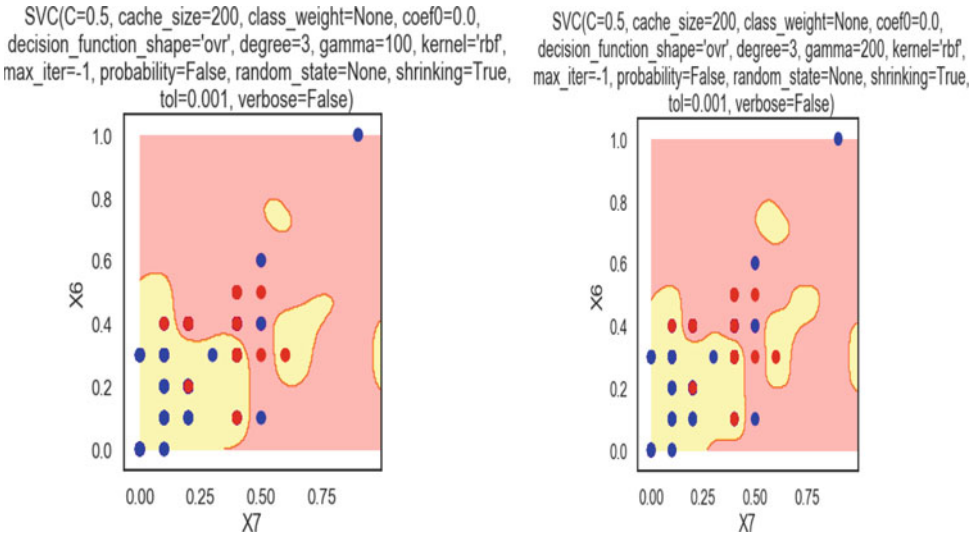
```
    fig, ax = plt.subplots(figsize=(5, 5))
    ax.contourf(xx, yy, y_grid_predictions, cmap=plt.cm.autumn_r, alpha=.3)
    ax.scatter(X_default.iloc[:, 0], X_default.iloc[:, 1], color=y_color, alpha=1)

    ax.set(
        xlabel=fields[0],
        ylabel=fields[1],
        title=str(estimator))

from sklearn.svm import SVC
gammas = [10, 20, 100, 200]
for gamma in gammas:
    SVC_Gaussian = SVC(kernel='rbf', C=0.5, gamma=gamma)
    plot_decision_boundary(SVC_Gaussian, X, y)
```





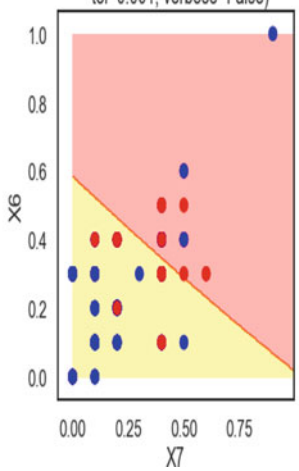


Question 5 Fit a Polynomial kernel SVC with degree 5 and see how the decision boundary changes

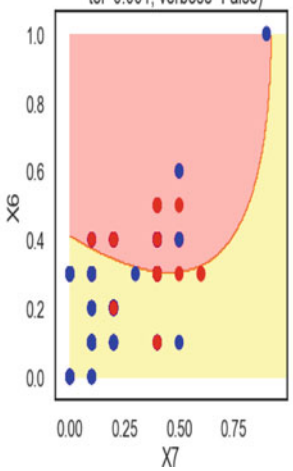
- Use the plot decision boundary function from the previous question and try the Polynomial Kernel SVC
- For various values of C, plot the decision boundary. You may try Cs = [0.1, 1, 10, 50]
- Try to find out a C value that gives the best possible decision boundary

```
from sklearn.svm import SVC
Cs = [.1, 1, 10, 100]
for C in Cs:
    SVC_Polynomial = SVC(kernel='poly', degree=5, coef0=1, C=C)
    plot_decision_boundary(SVC_Polynomial, X, y)
```

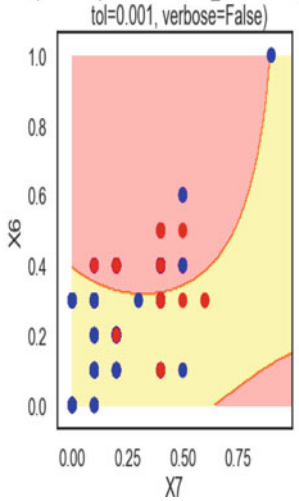
SVC(C=0.1, cache_size=200, class_weight=None, coef0=1, decision_function_shape='ovr', degree=5, gamma='auto', kernel='poly', max_iter=-1, probability=False, random_state=None, shrinking=True, tol=0.001, verbose=False)



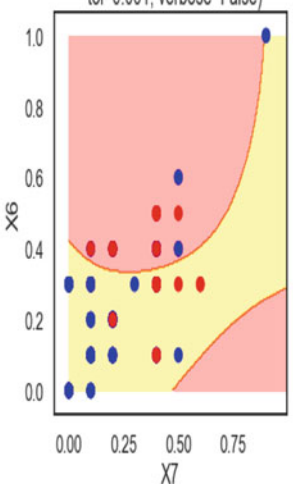
SVC(C=1, cache_size=200, class_weight=None, coef0=1, decision_function_shape='ovr', degree=5, gamma='auto', kernel='poly', max_iter=-1, probability=False, random_state=None, shrinking=True, tol=0.001, verbose=False)



SVC(C=10, cache_size=200, class_weight=None, coef0=1, decision_function_shape='ovr', degree=5, gamma='auto', kernel='poly', max_iter=-1, probability=False, random_state=None, shrinking=True, tol=0.001, verbose=False)



SVC(C=100, cache_size=200, class_weight=None, coef0=1, decision_function_shape='ovr', degree=5, gamma='auto', kernel='poly', max_iter=-1, probability=False, random_state=None, shrinking=True, tol=0.001, verbose=False)



Question 6a. Try tuning hyper-parameters for the svm kernal

- Take the complete dataset. Do a test and train split. For various values of Cs = [0.1, 1, 10, 100], compare the precision, recall, fscore, accuracy, and cm For various values of gammas = [10, 20, 100, 200], compare the precision, recall, fscore, accuracy, and cm

Question 6b. Do cross-validation with 5 folds

Question 6c. Using gridsearchcv to run through the data using the various parameters values

- Get the mean and standard deviation on the set for the various combination of gammas = [10, 20, 100, 200] and Cs = [0.1, 1, 10, 100]
- print the best parameters in the training set

```
from sklearn import svm

from sklearn.svm import SVC

from sklearn.model_selection import GridSearchCV

from sklearn.model_selection import cross_val_score

from sklearn.model_selection import train_test_split


X_train, X_test, y_train, y_test = train_test_split(X, y,

                    test_size=0.3, random_state=42)

gammas = [10, 20, 100, 200]

coeff_labels_gamma = ['gamma=10','gamma=20','gamma=100','gamma=200']

y_pred = list()

for gam,lab in zip(gammas,coeff_labels_gamma):

  clf = svm.SVC(kernel='rbf', C=1, gamma=gam)

  lr=clf.fit(X_train,y_train)

  y_pred.append(pd.Series(lr.predict(X_test), name=lab))
```

```
y_pred = pd.concat(y_pred, axis=1)


from sklearn.metrics import precision_recall_fscore_support as score
from sklearn.metrics import confusion_matrix, accuracy_score, roc_auc_score
from sklearn.preprocessing import label_binarize   #Binarize labels in a one-vs-all
metrics = list()
cm = dict()


for lab in coeff_labels_gamma:
   # Preciision, recall, f-score from the multi-class support function
   precision, recall, fscore, _ = score(y_test, y_pred[lab], average='weighted')

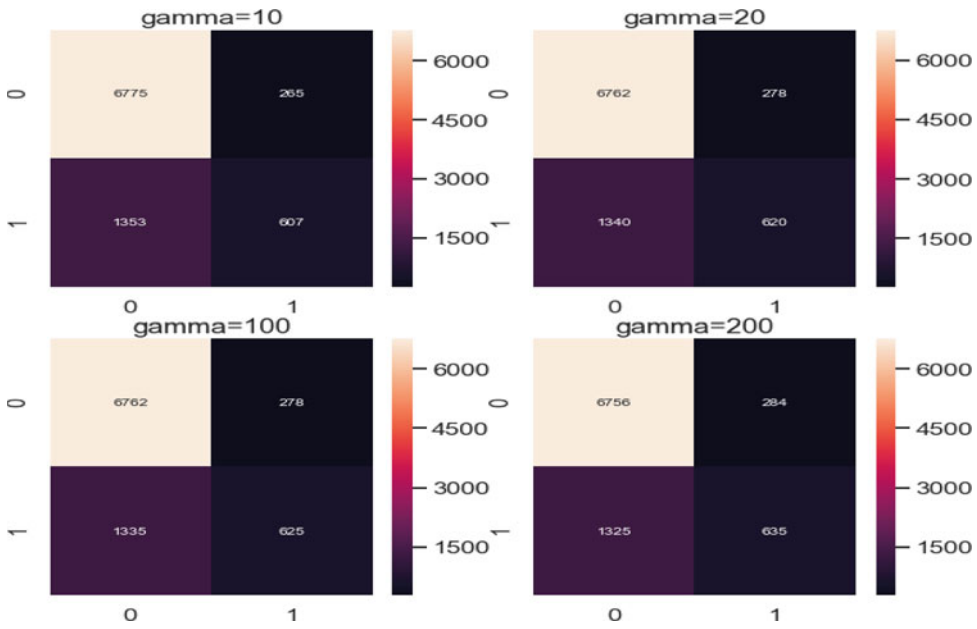

                # The usual way to calculate accuracy
                accuracy = accuracy_score(y_test, y_pred[lab])
                metrics.append(pd.Series({'precision':precision, 'recall':recall,
                                          'fscore':fscore, 'accuracy':accuracy},
                                          name=lab))
                 # Last, the confusion matrix
                cm[lab] = confusion_matrix(y_test, y_pred[lab])


            metrics = pd.concat(metrics, axis=1)
            metrics
```

	gamma=10	gamma=20	gamma=100	gamma=200
precision	0.803608	0.803208	0.803985	0.804443
	gamma=10	**gamma=20**	**gamma=100**	**gamma=200**
recall	0.820222	0.820222	0.820778	0.821222
fscore	0.792136	0.793125	0.793951	0.795055
accuracy	0.820222	0.820222	0.820778	0.821222

```
fig, axList = plt.subplots(nrows=2, ncols=2)
axList = axList.flatten()
fig.set_size_inches(10, 10)

axList[-1].axis('on')
# axList[:] will list all the 4 confusion tables; axList[:-1] list the first three confusion tables
for ax,lab in zip(axList[:], coeff_labels_gamma):
    sns.heatmap(cm[lab], ax=ax, annot=True, fmt='d');
    ax.set(title=lab);
```



```
Cs = [.1, 1, 10, 100]
coeff_labels = ['C=0.1', 'C=1.0', 'C=10','C=100']

y_pred = list()
for C,lab in zip(Cs,coeff_labels):
   clf = svm.SVC(kernel='rbf', C=C)
   lr=clf.fit(X_train,y_train)
   y_pred.append(pd.Series(lr.predict(X_test), name=lab))
y_pred = pd.concat(y_pred, axis=1)

from sklearn.metrics import precision_recall_fscore_support as score
from sklearn.metrics import confusion_matrix, accuracy_score, roc_auc_score
from sklearn.preprocessing import label_binarize    #Binarize labels in a one-vs-all

metrics = list()
cm = dict()

   for lab in coeff_labels:
      # Preciision, recall, f-score from the multi-class support function
      precision, recall, fscore, _ = score(y_test, y_pred[lab], average='weighted')
      # The usual way to calculate accuracy
      accuracy = accuracy_score(y_test, y_pred[lab])

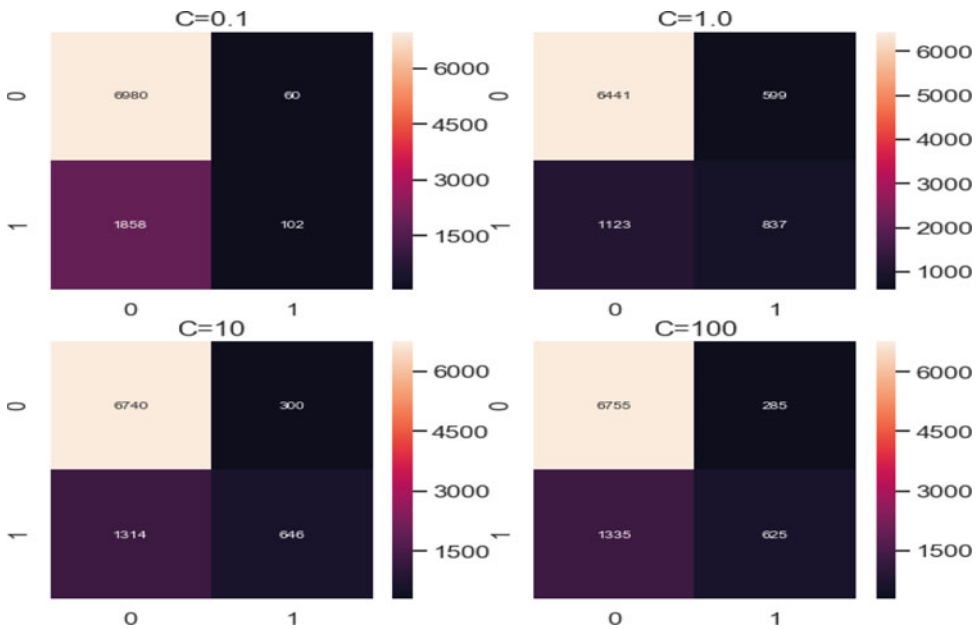
      metrics.append(pd.Series({'precision':precision, 'recall':recall,
                          'fscore':fscore, 'accuracy':accuracy}, name=lab))

      # Last, the confusion matrix
      cm[lab] = confusion_matrix(y_test, y_pred[lab])
   metrics = pd.concat(metrics, axis=1)
   metrics
```

	C=0.1	C=1.0	C=10	C=100
Precision	0.754896	0.793024	0.803319	0.802714
Recall	0.786889	0.808667	0.820667	0.820000
Fscore	0.708669	0.797338	0.795403	0.793319
Accuracy	0.786889	0.808667	0.820667	0.820000

```
fig, axList = plt.subplots(nrows=2, ncols=2)
axList = axList.flatten()
fig.set_size_inches(10, 10)
axList[-1].axis('on')

# axList[:] will list all the 4 confusion tables; axList[:-1] list the first three confusion tables
for ax,lab in zip(axList[:], coeff_labels):
    sns.heatmap(cm[lab], ax=ax, annot=True, fmt='d');
    ax.set(title=lab);
```



References

Aizerman, M. A., E. M. Braverman, and L. I. Rozonoer (1964). The probability problem of pattern recognition learning and the method of potential functions. Automation and Remote Control 25, 1175–1190.

Boser, B. E., I. M. Guyon, and V. N. Vapnik (1992). A training algorithm for optimal margin classifiers. In D. Haussler (Ed.), Proceedings Fifth Annual Workshop on Computational Learning Theory (COLT), pp. 144–152. ACM.

Csat´o, L. and Opper, M. (2002) Sparse online Gaussian processes. *Neural Computation*, 14(3): 641–669, 2002.

Nadaraya, E. A. (1964). On estimating regression. ´ Theory of Probability and its Applications 9(1), 141–142.

Powell, M. J. D. (1987). Radial basis functions for multivariable interpolation: a review. In J. Qui˜nonero-Candela, J., and C.E. Rasmussen (2005) A Unifying View of Sparse Approximate Gaussian Process Regression, *Journal of Machine Learning Research* 6 1939–1959.

Rasmussen, C.E. and Quin˜onero-Candela, J. (2005) Healing the Relevance Vector Machine through Augmentation, Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany.

Seeger, M., C. K. I. Williams, and N. Lawrence (2003) Fast forward selection to speed up sparse Gaussian process regression. In Christopher M. Bishop and Brendan J. Frey, editors, Ninth International Workshop on Artificial Intelligence and Statistics. Society for Artificial Intelligence and Statistics.

Snelson, E., and Ghahramani, Z. (2006) Sparse Gaussian processes using pseudo-inputs. In Y. Weiss, B. Sch¨olkopf, and J. Platt, editors, Advances in Neural Information Processing Systems 18, Cambridge, Massachussetts. The MIT Press.

Tipping, M.E. (2001) Sparse Bayesian learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1:211–244.

Watson, G. S. (1964). Smooth regression analysis. Sankhya: The Indian Journal of Statistics. Series A 26, 359–372.

Check for updates

13 Neural Networks and Deep Learning Algorithm

13.1 Introduction

In Chap. 11, we considered a model $f(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w}$, where the initial input vector $\boldsymbol{x}$ is replaced by feature vector $\boldsymbol{\phi}(\boldsymbol{x}) = [\phi_0(\boldsymbol{x}), \ldots, \phi_M(\boldsymbol{x})]'$. As ideal basis functions $\boldsymbol{\phi}(\boldsymbol{x})$ should be localized or adaptive w.r.t. $\boldsymbol{x}$, we cluster the input dataset $\{\boldsymbol{x}_i | 1 \leq i \leq N\} \subset R^D$ into M clusters, and let $\{\mu_j, 0 \leq j \leq M\text{-}1\}$ will be the centers of the clusters. Or, without cluster the input dataset $\{\boldsymbol{x}_i | 1 \leq i \leq N\}$, choose as many basis functions as the number of training dataset, i.e., for some radial basis function h and $1 \leq i \leq N$, we have.

$$\phi_i(\boldsymbol{x}) = \boldsymbol{h}(||\boldsymbol{x} - \boldsymbol{x}_i||)$$

Nonlinear models with radial basis functions are very flexible models; however, they are very restricted because the feature vector $\boldsymbol{\phi}$ needs to be determined first in an ad hoc way. In practice, we have no clue of the form of the feature vector $\boldsymbol{\phi}$. Neural network models provide a way to learn the feature vector $\boldsymbol{\phi}$ in a flexible problem-dependent manner.

The term 'neural network' was originated to find mathematical representations of information processing in biological systems (McCulloch and Pitts 1943; Rosenblatt 1962; Rumelhart et al. 1986). A neural network is based on a collection of connected nodes that loosely model the neurons in a biological brain. Each connection or edge, like the synapses in a biological brain, can transmit a signal to other neurons. Once a neuron receives a signal, it will process it and pass the signal to neurons connected to it. The "signal" at a connection is a real number, and the output of each neuron is computed by some nonlinear function of the sum of its inputs. For each edge, there is a weight associated with it, which adjusts the strength of the signal as learning proceeds. Neurons may have a threshold such that a signal is sent only if the aggregate signal crosses that threshold. Typically, neurons are aggregated into layers. Different layers may perform different transformations on their inputs. Signals travel from the first layer (the input layer) to the last layer (the output layer), possibly after traversing the layers multiple times.

In recent years, deep learning based on neural network architecture including feedforward neural network, recurrent neural networks (Dupond 2019; Tealab 2018; Graves et al. 2009), and convolutional neural network (Valueva et al. 2020; Zhang 1990; Coenraad et al. 2020; Collobert et al. 2008) have been applied to fields including computer vision, natural language processing, audio recognition, social network filtering, medical image analysis, and board game programs. These applications have produced outcomes comparable to and in some cases surpassing human expert performance. In the following, neural network is introduced first, and then, two types of deep learning, namely, deep feedforward network and deep convolutional neural network will be introduced.

This chapter is broken down into the following sections. Section 13.2 looks at the feedforward network functions. Section 13.3 discusses network training, Sect. 13.4 discusses gradient descent optimization, and Sect. 13.5 looks at the regularization in neural networks and early stopping. Section 13.6 compares deep feedforward network and deep convolutional neural networks. Section 13.7 discusses Python programming.

13.2 Feedforward Network Functions

The earliest type of neural networks is the feedforward neural network, in which the information moves in only one direction—forward—from the input nodes, through the hidden nodes (if any) and to the output nodes with no cycles or loops in the network.

Considered a model $y = f(\boldsymbol{x})$, where the initial input vector $\boldsymbol{x}$ is related to the target y, where the target y is either continuous or 0–1 in a classification problem with two classes. Suppose the model function is.

$$f(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{a}) \tag{13.1}$$

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_13

the quantity $a = \boldsymbol{w}'\boldsymbol{\phi}(\boldsymbol{x})$ is called the activation, $\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_0(\boldsymbol{x}),\ldots, \phi_{M-1}(\boldsymbol{x}))'$ are the M-dim basis functions, and h is a differentiable, nonlinear activation function. Examples of activation functions include logistic sigmoid function and "tanh" function. (13.1) is a single neuron model.

Figure 13.1 exhibits a single neuron model. A basic neural network of two-layers extends the single neuron model with a hidden layer consisting of H_1 hidden units as follows. Suppose the initial input vector $\boldsymbol{x}$ is related to the K targets $\boldsymbol{y} = (y_1,\ldots, y_K)'$, where y_ks are either continuous variables or in the form of 1-of-K coding in a classification problem with K classes. The inputs and outputs of the first hidden layer are $\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_0(\boldsymbol{x}),\ldots, \phi_{M-1}(\boldsymbol{x}))'$ and

$$z_k^{(1)} = h\left(\sum_{j=0}^{M-1} w_{k,j}^{(1)}\phi_j(\boldsymbol{x})\right) \tag{13.2}$$

respectively, where $1 \le k \le H_1$. If there is only one hidden layer, $z_1^{(1)}, \ldots, z_{H_1}^{(1)}$ are the inputs of the output layer, and the outputs are

$$z_k^{(2)} = h\left(\sum_{j=0}^{H_1} w_{k,j}^{(2)} z_j^{(1)}\right) \tag{13.3}$$

More general neural network can be constructed by extending the one-hidden-layer neural network with more hidden layers. Figure 13.2 exhibits a feedforward neural network with two hidden layers. For a general feedforward neural network with $L-1$ hidden layers, the outputs of the lth hidden layer are

$$z_k^{(l)} = h\left(\sum_{j=0}^{H_{l-1}} w_j^{(l)} z_j^{(l-1)}\right) \tag{13.4}$$

where $1 \le k \le H_l$, with H_l the number of hidden nodes of the lth hidden layer, $1 \le l \le L-1$.

There is a direct mapping between a mathematical function f and the corresponding neural network (13.4) in a feedforward architecture having no closed directed cycles. More complex neural network can be developed, but the architecture must be restricted to feedforward to ensure the following characteristics:

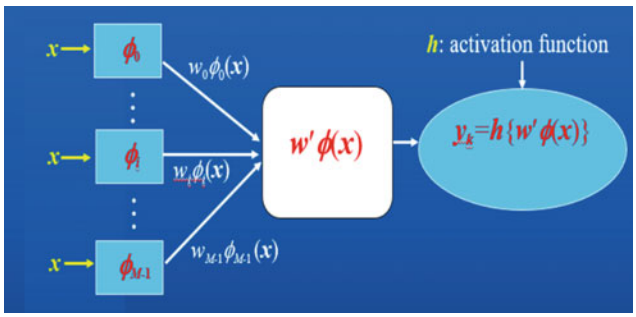


Fig. 13.1 A single neuron model

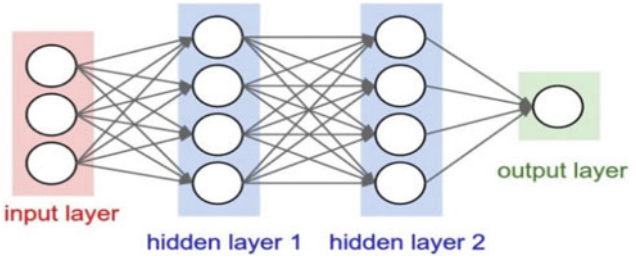


Fig. 13.2 A feedforward neural network with two hidden layers. https://mc.ai/a-to-z-about-artificial-neural-networks-ann-theory-n-hands-on/

- Sign-flip symmetry
  - If we change the sign of all of the weights and the bias feeding into a particular hidden unit, then, for a given input pattern, the sign of the activation of the hidden unit will be reversed.
  - Compensated by changing the sign of all of the weights of leading out of that hidden units.
  - For M hidden nodes, by $\tanh(-a) = -\tanh(a)$, there will be M sign-flip symmetries.
  - Any given weight vector will be one of a set 2^M equivalent weight vectors.
- Interchange symmetry.
  - We can interchange the values of all of the weights (and the bias) leading both into and out of a particular hidden unit with the corresponding values of the weights (and bias) associated with a different hidden unit.
  - This clearly leaves the network input–output mapping function unchanged.
  - For M hidden units, any given weight vector will belong to a set of $M!$ equivalent weight vectors.

13.3 Network Training: Error Backpropagation

Error backpropagation is used to train a multilayer neural network by applying gradient descent to minimize the sum-of-squares error function. It is an iterative procedure with adjustments to the weights in a sequence of steps, in which local information is sent forwards and backwards alternately through the network. At each such step, two distinct stages are involved: (1) The derivatives of the error function with respect to the weights are evaluated as the errors are propagated backwards through the network at this stage; (2) the derivatives are then used to compute the adjustments to be made to the weights.

Suppose the neural network (13.4) has L layers and is mapped to the model function $f(\boldsymbol{x}, \boldsymbol{w})$, where $\boldsymbol{w}$ contains all the unknown weight parameters. Given a training dataset of N examples with the inputs $\{\boldsymbol{x}_n | n = 1, \ldots, N\} \subset R^D$ and the corresponding targets $\{\boldsymbol{y}_n | n = 1, \ldots, N\} \subset R^K$, we want to minimize the sum-of-squares error function

$$Error(\boldsymbol{w}) = \sum_{n=1}^{N} \|\boldsymbol{y}_n - \boldsymbol{f}(\boldsymbol{x}_n, \boldsymbol{w})\|^2 \tag{13.5}$$

In the following, we start with a regression problem with continuous outputs. For the nth example in the training dataset, $1 \le n \le N$, let $\boldsymbol{x}_n$ and $\boldsymbol{y}_n = (y_{n,1},\ldots, y_{n,K})'$ be the input vector and the K outputs. Suppose the activations of all of the hidden and output units in the network by successive application of (13.4) are calculated using a forward flow of information or forward propagation through the network. In more specific, at the lth layer, $1 \le l \le L$, the input and output of the kth node of the nth example, $1 \le n \le N$, is

$$a_{n,k}^{(l)} = \sum_{j=0}^{H_{l-1}} w_{k,j}^{(l)} z_{n,j}^{(l-1)} \tag{13.6a}$$

$$z_{n,k}^{(l)} = h\left(a_{n,k}^{(l)}\right) \tag{13.6b}$$

Note the activation function of the Lth layer is the identity function, thus

$$z_{n,k}^{(L)} = \sum_{j=0}^{H_{L-1}} w_{k,j}^{(L)} z_{n,j}^{(L-1)}$$

Consider the sum of squared errors for the K outputs $\boldsymbol{y}_n = (y_{n,1},\ldots, y_{n,K})'$ of the nth example:

$$Error_n(\boldsymbol{w}) = \frac{1}{2}\sum\nolimits_{k=1}^{K}\left(\delta_{n,k}^{(L)}\right)^2 \text{for} 1 \le n \le N$$

where $\delta_{n,k}^{(L)} = \left(y_{n,k} - z_{n,k}^{(L)}\right)$, $1 \le k \le K$.

Of interest is the derivative of $Error_n(\boldsymbol{w})$ *w.r.t.* $w_{k,j}^{(l)}$, $1 \le k \le H_l$, $1 \le j \le H_{l-1}$, and $1 \le l \le L$. In order to evaluate these derivatives, we need to calculate the value of δ for each hidden and output node in the network, where δ for the kth hidden node in the lth layer, $1 \le l \le L-1$, is defined as

$$\delta_{n,k}^{(l)} = \frac{\partial Error_n(\boldsymbol{w})}{\partial a_{n,k}^{(l)}} = \sum_{j=0}^{H_{l+1}} \left(\frac{\partial Error_n(\boldsymbol{w})}{\partial a_{n,j}^{(l+1)}}\right)\left(\frac{\partial a_{n,j}^{(l+1)}}{\partial a_{n,k}^{(l)}}\right) \tag{13.7}$$

In (13.7), $a_{n,j}^{(l+1)}$ is the input to the jth hidden node in the $(l + 1)$th layer given by

$$\begin{aligned} a_{n,j}^{(l+1)} &= \sum_{k=0}^{H_l} w_{j,k}^{(l+1)} z_{n,k}^{(l)} \\ &= \sum_{k=0}^{H_l} w_{j,k}^{(l+1)} h\left(a_{n,k}^{(l)}\right) \text{for} 1 \le j \le Hl+1 \end{aligned}$$

Since

$$\frac{\partial a_{n,j}^{(l+1)}}{\partial a_{n,k}^{(l)}} = w_{j,k}^{(l+1)} h'\left(a_{n,k}^{(l)}\right) \text{for} 1 \le k \le Hl$$

By definition in (13.7),

$$\frac{\partial Error_n(\boldsymbol{w})}{\partial a_{n,j}^{(l+1)}} = \delta_{n,j}^{(l+1)} \text{for} 1 \le j \le Hl+1$$

Equation (13.7) becomes

$$\begin{aligned} \delta_{n,k}^{(l)} &= \sum_{j=0}^{H_{l+1}} \delta_{n,j}^{(l+1)} w_{j,k}^{(l+1)} h'\left(a_{n,k}^{(l)}\right) \\ &= h'\left(a_{n,k}^{(l)}\right) \sum_{j=0}^{H_{l+1}} \delta_{n,j}^{(l+1)} w_{j,k}^{(l+1)} \end{aligned} \tag{13.8}$$

Equation (13.8) indicates that the value of δ for a particular hidden node can be obtained by propagating the δ's backwards from the nodes in the next layer in the network. The backpropagation procedure can therefore be implemented as follows:

1. The inputs and activations of all of the hidden and output nodes in the network by (13.6a) and (13.6b) are calculated,
2. At the output layer, i.e., the Lth layer, evaluate the derivative

$$\frac{\partial Error_n(\boldsymbol{w})}{\partial w_{k,j}^{(L)}} = \delta_{n,k}^{(L)} z_{n,j}^{(L-1)} \text{for} 1 \le k \le HL, 0 \le j \le HL-1 \tag{13.9}$$

3. For the lth hidden layer with H_l hidden units, $1 \le l$ $L-1$, the derivative of $Error_n(\boldsymbol{w})$ *W.R.T.* $w_{k,j}^{(l)}$, $1 \le k$ H_l, $1 \le j \le H_{l-1}$, is

$$\frac{\partial Error_n(\boldsymbol{w})}{\partial w_{k,j}^{(l)}} = \left(\frac{\partial Error_n(\boldsymbol{w})}{\partial a_{n,k}^{(l)}}\right)\left(\frac{\partial a_{n,k}^{(l)}}{\partial w_{k,j}^{(l)}}\right) = \delta_{n,k}^{(l)} z_{n,j}^{(l-1)}.$$

13.4 Gradient Descent Optimization

The sum-of-squares error function (13.5), i.e., the objective function, needs to be minimized in order to train the neural network. As the gradient can be computed analytically, which is used to estimate the impact of small variations of the parameter values on the objective function, efficient gradient-based learning algorithms to minimize the objective function can be devised.

One should note that an objective function F: $R^d \rightarrow R$ can be reduced as the update is in the direction of $-\nabla_w F$ since

$$\lim_{h \to 0} \frac{f(\boldsymbol{w} + h\boldsymbol{u}) - f(\boldsymbol{w})}{h} = \nabla_w F \bullet \boldsymbol{u}$$

is the directional derivative in the direction $\boldsymbol{u}$, where $\boldsymbol{u}$ is a normed-one vector and

$$\nabla \boldsymbol{w} F = \left[\frac{\partial F}{\partial w_1}, \ldots, \frac{\partial F}{\partial w_d} \right]'$$

is the gradient. The basis of the gradient-descent learning algorithm is iteratively reduce the value of the objective function by the update

$$\boldsymbol{w}_{new} = \boldsymbol{w}_{old} - \eta \nabla_w F \tag{13.10}$$

where $\boldsymbol{w}$ is the real-valued parameter vector, and η is the learning rate.

It is very often the objective function F has the form of a sum of N functions.

$$F(\boldsymbol{w}) = \sum_{j=1}^{N} f(\boldsymbol{w}|\boldsymbol{x}_i)$$

based on N *i.i.d.* training data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$. In such cases, evaluating the gradient of the objective function F requires evaluating all the summand functions' gradients. When the training set is enormous and no simple formulas exist, evaluating the sums of gradients becomes very expensive. To economize on the computational cost at every iteration, stochastic gradient descent algorithm is devised, in which a subset of summand functions is sampled at every step.

Sum-minimization problems often arise in least squares and maximum likelihood estimation. As the training set is enormous, the stochastic gradient descent algorithm is very effective. When the stochastic gradient descent algorithm is applied to the minimization of the sum-of-squares error function (13.5), one has.

- Choose initial values of the parameter vector $\boldsymbol{w}$ and learning rate η, where $\boldsymbol{w}$ contains all the unknown weight parameters in the neural network (13.4).
- Repeat until an approximate minimum is obtained:
  - Randomly shuffle examples in the training set.
  - For $i = 1, \ldots, N$, do

$$\boldsymbol{w}_{new} = \boldsymbol{w}_{old} - \eta \nabla_w f(w|x_1)$$

The convergence of the stochastic gradient descent algorithm is due to the Lemma by Robbins and Siegmund (1971) as following:

Robbins–Siegmund Lemma When the learning rate η decreases with an appropriate rate, and subject to relatively mild assumptions, stochastic gradient descent converges almost surely to a global minimum when the objective function f is convex.

13.5 Regularization in Neural Networks and Early Stopping

As the numbers of input and output nodes in a neural network are generally determined by the dimensionality of the data set, the numbers of hidden layers and their nodes are free parameters that can be adjusted to give different predictive performance. As the larger the numbers of hidden layers and/or their nodes, the more unknown weights and biases parameters in the network, so we might expect that there is a trade-off between under-fitting and overfitting to the optimum balance performance in a maximum likelihood setting.

To control the complexity of a neural network model in order to avoid over-fitting problem, one solution is to choose relatively large numbers of hidden layers and/or hidden nodes, and then to control the complexity by the addition of a regularization term to the error function. The simplest regularizer is the quadratic, also known as weight decay giving a regularized error of the form

$$\tilde{Error}(\boldsymbol{w}) = Error(\boldsymbol{w}) + \frac{\lambda}{2} \boldsymbol{w}' \boldsymbol{w} \tag{13.11}$$

where λ is the regularization coefficient that control the model complexity as the quadratic regularizer $\frac{\lambda}{2} \boldsymbol{w}' \boldsymbol{w}$ can be considered as the negative logarithm of a zero-mean Gaussian prior distribution over the weight vector $\boldsymbol{w}$.

Another way to control the complexity of a neural network is early stopping. As the training of a feedforward neural network corresponds to an iterative reduction of the error function. For many of the optimization algorithms, such as gradient descent, the error is a nonincreasing function with respect to the training dataset. The effective number of parameters in the network therefore grows during the course of training. However, when the error of the trained neural network model is measured with respect to an

independent dataset, generally called a validation set, often shows a decrease at first, followed by an increase as the network starts to overfit. Training can therefore be stopped at the point of smallest error with respect to the validation data set to obtain a network model with good generalization performance. Early stopping is similar to weight decay by the quadratic regularizer in (13.11).

13.6 Deep Feedforward Network Versus Deep Convolutional Neural Networks

A neural network with very large number of hidden layers and/or nodes with no feedback connections is called a deep feedforward network. Due to its high degree of freedoms in the numbers of hidden layers and nodes, the deep feedforward neural network can be trained to learn high-dimensional and nonlinear mappings, which makes them candidates for complex tasks. However, there are still problems with the deep feedforward neural network for complex tasks such as image recognition, as images are large, often with several hundred variables (pixels). A deep feedforward network with, say one hundred hidden units in the first layer, would already contain several tens of thousands of weights. Such a large number of parameters increases the capacity of the system and therefore requires a larger training dataset. In addition, images have a strong $2D$ local structure: variables (or pixels) that are spatially or temporally nearby are highly correlated. Local correlations are the reasons for the well-known advantages of extracting and combining local features before recognizing spatial or temporal objects, because configurations of neighboring variables can be classified into a small number of categories (e.g., edges, corners…). Another deficiency of a feedforward network is the lack of built-in invariance with respect to translations, or local distortions of the inputs.

Convolutional neural networks (CNN) were developed with the idea of local connectivity and shared weights so the shift invariance is automatically obtained by forcing the replication of weight configurations across space. In each layer of the convolutional neural network, the input is convolved with the weight matrix (also called the filter) to create a feature map. In other words, the weight matrix slides over the input and computes the dot product between the input and the weight matrix. Note that as opposed to regular neural networks, all the values in the output feature map share the same weights. This means that all the nodes in the output detect exactly the same pattern. The local connectivity and shared weights aspect of CNNs reduces the total number of learnable parameters resulting in more efficient training. The intuition behind a convolutional neural network is thus to learn in each layer a weight matrix that will be able to extract the necessary, translation-invariant features from the input.

Consider the inputs $x_0, \ldots, x_{N-1}$. In the first layer, the input is convolved with a set of H_1 filters (weights)$\{w_h^{(1)}, 1 \leq h \leq H_1\}$ and the output is

$$z_h^{(1)}(i) = h\left(\sum_{j=-\infty}^{\infty} w_h^{(1)}(j) x_{i-j}\right) \tag{13.12}$$

where $w_h^{(1)}$ is k-dim, here k is the filter size that controls the receptive field of each output node, and $1 \leq i \leq N-1$. In a convolutional neural network, the receptive field of node a is defined as the set of nodes from previous layer with the outputs acting as the inputs of node a.

Now the output feature map $z^{(1)}$ is $(N-k+1) \times H_1$, which is convolved with a set of H_2 filters (weights) $\{w_h^{(2)}, 1 \leq h \leq H_2\}$ and becomes the inputs of the 2nd layer. Similar to the first layer, a nonlinear transformation is applied to the inputs to produce the output feature map. Repeat the same procedure, the output feature map of the lth layer, $2 \leq l \leq L$, is

$$z_h^{(l)}(i) = h\left(\sum_{j=-\infty}^{\infty} \sum_{m=1}^{H_{l-1}} w_h^{(l)}(j,m) \times z_m^{(l-1)}(i-j)\right) \tag{13.13}$$

where $w_h^{(l)}$ is $k \times H_l$, and the output feature map $z^{(l)}$ is N_l H_l, $N_l = N_{l-1}$-k + 1. The local connectivity is achieved by replacing the weighted sums from the neural network with convolutions to a local region of each node in CNN. The local connected region of a node is referred to as the receptive field of the node.

For time series inputs $x_0, \ldots, x_{N-1}$, to learn the long-term dependencies within the time series, stacked layers of dilated convolutions are used:

$$z_h^{(l)}(i) = h\left(\sum_{j=-\infty}^{\infty} \sum_{m=1}^{H_{l-1}} w_h^{(l)}(j,m) \times z_m^{(l-1)}(i-d \times j)\right). \tag{13.14}$$

In this way, the filter is applied to every dth element in the input vector, allowing the model to learn connections between far-apart data elements. In addition to dilated convolutions, for time series inputs $x_0, \ldots, x_{N-1}$, it is convenient to pad the input with zeros around the border. The size of this zero-padding depends on the size of the receptive field.

13.7 Python Programing

Consider the dataset of credit card holders' payment data in October 2005, from a bank (a cash and credit card issuer) in Taiwan. Among the total 25,000 observations, 5529 observations (22.12%) are the cardholders with default payment. Thus the target variable y is the default payment (Yes = 1, No = 0), and the explanatory variables are the following 23 variables:

- X_1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
- X_2: Gender (1 = male; 2 = female).
- X_3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
- X_4: Marital status (1 = married; 2 = single; 3 = others).
- X_5: Age (year).
- X_6–X_{11}: History of past payment from September to April 2005.
  (The measurement scale for the repayment status is: 1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; ... ; 8 = payment delay for eight months; 9 = payment delay for nine months and above).
- X_{12}–X_{17}: Amount of bill statement from September to April 2005.
- X_{18}–X_{23}: Amount of previous payment (NT dollar) from September to April 2005.

References

Coenraad, M; Myburgh, Johannes C.; Davel, Marelie H. (2020). Gerber, Aurona (ed.). "Stride and Translation Invariance in CNNs". Artificial Intelligence Research. Communications in Computer and Information Science. Cham: Springer International Publishing. **1342**: 267–281.

Collobert, Ronan, Weston, Jason (2008–01–01). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. Proceedings of the 25th International Conference on Machine Learning. ICML'08. New York, NY, USA: ACM. pp. 160–167.

Dupond, Samuel (2019). "A thorough review on the current advance of neural network structures". *Annual Reviews in Control*. **14**: 200–230.

Graves, Alex; Liwicki, Marcus; Fernandez, Santiago; Bertolami, Roman; Bunke, Horst; Schmidhuber, Jürgen (2009). "A Novel Connectionist System for Improved Unconstrained Handwriting Recognition" (PDF). *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **31** (5): 855-868.

McCulloch, W. S. and W. Pitts (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5, 115–133.

Rosenblatt, F. (1962). Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan.

Rumelhart, D. E., J. L. McClelland, and the PDP Research Group (Eds.) (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations. MIT Press.

Tealab, Ahmed (2018–12–01). "Time series forecasting using artificial neural networks methodologies: A systematic review". *Future Computing and Informatics Journal*. **3** (2): 334–340.

Valueva, M.V.; Nagornov, N.N.; Lyakhov, P.A.; Valuev, G.V.; Chervyakov, N.I. (2020). "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation". Mathematics and Computers in Simulation. Elsevier BV. **177**: 232–243.

Zhang, Wei (1990). "Parallel distributed processing model with local space-invariant interconnections and its optical architecture". Applied Optics. **29** (32): 4790–7.

Alternative Machine Learning Methods for Credit Card Default Forecasting*

14

By Huei-Wen Teng, National Yang Ming Chiao Tung University, Taiwan

This chapter is a revised and extended version of the paper: Huei-Wen Teng and Michael Lee. Estimation procedures of using five alternative machine learning methods for predicting credit card default. Review of Pacific Basin Financial Markets and Policies, 22(03):1950021, 2019. doi: https://doi.org/10.1142/S0219091519500218

14.1 Introduction

Following de Mello and Ponti (2018), Bzdok et al. (2018), and others, we can define machine learning as a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns, and make decisions with minimal human intervention. Machine learning is one of the most important tools for financial technology. Machine learning is particularly useful when the usual linearity assumption does not hold for the data. Under equilibrium conditions and when the standard assumptions of normality and linearity hold, machine learning and parametric methods, such as OLS, tend to generate similar results. Since machine learning methods are essentially search algorithms, there is the usual problem of finding global minima that minimizes some function.

Machine learning can generally be classified as (i) supervised learning, (ii) unsupervised learning, and (iii) others (reinforcement learning, semi-supervised, and active learning). Supervised learning includes (i) regression (lasso, ridge, logistic, loess, KNN, and spline) and (ii) classification (SVM, random forest, and deep learning). Unsupervised learning includes (i) clustering (K-means, hierarchical tree clustering) and (ii) factor analysis (principle component analysis, etc.). K nearest neighbors (KNN) is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the data.

Based upon the concept and methodology of machine learning and deep learning, which has been discussed in Chaps. 12 and 13, this chapter shows how five alternative machine learning methods can be used to forecast credit card default. This chapter is organized as follows. Section 14.1 is the introduction, and Sect. 14.2 reviews literature. Section 14.3 introduces the credit card data set. Section 14.4 reviews five supervised learning methods. Section 14.5 gives the study plan to find the optimal parameters and compares the learning curves among five methods. A summary and concluding remarks are provided in Sect. 14.6. Python codes are given in Appendix 14.1.

14.2 Literature Review

Machine learning is a subset of artificial intelligence that often uses general and intuitive methodology to give computers (machines) the ability to learn with data so that the performance on a specific task is improved, without explicitly programmed (Samuel 1959). Because of its flexibility and generality, machine learning has been successfully applied in the fields, including email filtering, detection of network intruders or malicious intruders working towards a data breach, optical character recognition, learning to rank, informatics, and computer vision (Mitchell 1997; Mohri et al. 2012; De Mello and Ponti 2018). In recent years, machine learning has fruitful applications in financial technology, such as fraud prevention, risk management, portfolio management, investment predictions, customer service, digital assistants, marketing, sentiment analysis, and network security.

Machine learning is closely related to statistics (Bzdok et al. 2018). Indeed, statistics is a sub-field of mathematics, whereas machine learning is a sub-field of computer science. To explore the data, statistics starts with a probability model, fits the model to the data, and verifies if this model is adequate using residuals analysis. If the model is not adequate,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_14

residuals analysis can be used to refine the model. Once the model is shown to be adequate, statistical inference about the parameters in the model can be furthermore used to determine if a factor of interests is significant. The ability to explain if a factor really matters makes statistics widely used in almost all disciplines.

In contrast, machine learning focuses more on prediction accuracy but not model interpretability. In fact, machine learning uses general purposes algorithms and aims at finding patterns with minimal assumption about the data-generating system. Classic statistics method together with machine learning techniques leads to a combined field, called statistical learning (James et al. 2013).

Applications domain of machine learning can be roughly divided into unsupervised learning and supervised learning (Hastie et al. 2008). Unsupervised learning refers to the situations one has just predictors, and attempts to extract features that represent the most distinct and striking features in the data. Supervised learning refers to the situations that one has predictors (also known as input, explanatory, or independent variables) and responses (also known as output, or dependent variables), and attempts to extract important features in the predictors that best predict responses. Using input–output pairs, supervised learning learns a function from the data set to map an input to an output using sample (Russell and Norvig 2010).

In financial technology (FinTech), machine learning has received extensive attention in recent years. For example, Heaton et al. (2017) apply deep learning for portfolio optimization. With the rapid development of high-frequency trading, intra-day algorithmic trading becomes a popular trading device and machine learning is a fundamental paralytics for predicting returns of underlying asset: Putra and Kosala (2011) use neural network and validate the validity of the associated trading strategies in the Indonesian stock market; Borovykh et al. (2018) propose a convolutional neural network to predict time series of the S&P 500 index. Lee (2020) and Lee and Lee (2020) have discussed the relationship between machine learning and financial econometrics, mathematics, and statistics.

In addition to the above applications, machine learning is also applied to other canonical problems in finance. For example, Solea et al. (2018) identify the next emerging countries using statistical learning techniques. To measure asset risk premia in empirical asset pricing, Gu et al. (2018) perform a comparative analysis of methods using machine learning, including generalized linear models, dimension reduction, boosted regression trees, random forests, and neural networks.

To predict the delinquency of a credit card holder, a credit scoring model provides a model-based estimate of the default probability of a credit card customer. The predictive models for the default probability have been developed using machine learning classification algorithms for binary outcomes (Hand and Henley 1997). There have been extensive studies examining the accuracy of alternative machine learning algorithms or classifiers. Recently, Lessmann et al. (2015) provide comprehensive classifier comparisons to date and divide machine learning algorithms into three divisions: individual classifiers, homogeneous ensembles, and heterogeneous ensembles.

Individual classifiers are those using a single machine learning algorithm, for example, the *k*-nearest neighbors, decision trees, support vector machine, and neural network. Butaru et al. (2016) test decision tree, regularized logistic regression, and random forest models with a unique large data set from six larger banks. It is found that no single model applies to all banks, and suggests the need for a more customized approach to the supervision and regulation of financial institutions, in which parameters such as capital ratios and loss reserves should be specified to each bank according to its credit risk model exposures and forecasts. Sun and Vasarhelyi (2018) demonstrate the effectiveness of a deep neural network based on clients' personal characteristics and spending behaviors over logistic regression, naïve Bayes, traditional neural networks, and decision trees in terms of better prediction performance with a data set of size 711,397 collected in Brazil.

Novel machine learning method to incorporate complex features of the data are proposed as well. For example, Fernandes and Artes (2016) incorporate spatial dependence as inputs into the logistic regression, and Maldonado et al. (2017) propose support vector machines for simultaneous classification and feature selection that explicitly incorporate attribute acquisition costs. Addo et al. (2018) provide binary classifiers based on machine and deep learning models on real data in predicting loan default probability. It is observed that tree-based models are more stable than neural network-based methods.

On the other hand, the ensemble method contains two steps: model developments and forecast combinations. It can be divided into homogeneous ensemble classifiers and heterogeneous ensemble classifiers. The former uses the same classification algorithm, whereas the latter uses different classification algorithms. Finlay (2011) and Paleologo et al. (2010) have shown that homogeneous ensemble classifiers increase predictive accuracy. Two types of homogeneous ensemble classifiers are bagging and boosting. Bagging derives independent base models from bootstrap samples of the original data (Breiman 1996), and boosting iteratively adds base models to avoid the errors of current ensembles (Freund and Schapire 1996).

Heterogeneous ensemble methods create these models using different classification algorithms, which have different views on the same data and may complement each other. In addition to base models' developments and forecast combinations, heterogeneous ensembles need a third step to

search the space of available base models. Static approaches search the base model once, and dynamic approaches repeat the selection step for every case (Ko et al. 2008; Woloszynski and Kurzynski 2011). For static approaches, the direct method maximizes predictive accuracy (Caruana et al. 2006) and the indirect method optimizes the diversity among base models (Partalas et al. 2010).

14.3 Description of the Data

We apply the machine learning techniques in the default of credit card clients' data set. There are 29,999 instances in the credit card data set. The default of credit card client's data set can be found at http://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients and was initially analyzed by Yeh and Lien (2009). This data set is the payment data of credit card holders in October 2005, from a major cash and credit card issuer in Taiwan. This data set contains 23 different attributes to determine whether or not a person would default on their next credit card payment. It contains amount of given credit, gender, education, marital status, age, and history of past payments, including how long it took someone to pay the bill, the amount of the bill, and how much they actually paid for the previous six months.

The response variable is

- Y: Default payment next month (1 = default; 0 = not default). We use the following 23 variables as explanatory variables:
- X_1: Amount of the given credit (NT dollar),
- X_2: Gender (1 = male, 2 = female),
- X_3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others),
- X_4: Marital status (1 = married; 2 = single; 3 = others),
- X_5: Age (year),
- X_6–X_{11}: History of past monthly payment traced back from September 2005 to April 2005 (−1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; ...; 8 = payment delay for eight months; 9 = payment delay for nine months and above),
- X_{12}–X_{17}: Amount of past monthly bill statement (NT dollar) traced backfrom September 2005 to April 2005.
- X_{18}–X_{23}: Amount of past payment (NT dollar) traced back from September 2005 to April 2005.

This data set is interesting because it contains two "sorts" of attributes. The first sort is about categorical attributes like education, marital status, and age. These attributes have a very small range of possible values, and if there was a high correlation between these categorical attributes then the classification algorithms would be able to easily identify them and produce high accuracies. The second sort of attribute is the past payment information. These attributes are just integers without clear differentiation of categories and have much larger possible ranges of how much money was paid. Especially, if there was not strong correlation between education, marital status, age, etc., and defaulting on payments, it could be more difficult to algorithmically predict the outcome from past payment details, except for the extremes where someone never pays their bills or always pays their bills. Figure 14.1 plots the heatmap to show pairwise correlations between attributes. It is shown that most correlations are about zeros, but high correlations exist in features of past monthly payments $(X_6, \ldots, X_{11})$ and past monthly bill statements $(X_{12}, \ldots, X_{17})$.

14.4 Alternative Machine Learning Methods

Let $X = (X_1, \ldots, X_p)$ denote the p-dimensional input vector, and let $Y = (Y_1, \ldots, Y_d)$ denote the d-dimensional output vector. In its simplest form, a learning machine is an input–output mapping, $Y = F(X)$. In statistics, $F(\cdot)$ is usually a simple function, such as a linear or polynomial function. In contrast, the form of the $F(\cdot)$ in machine learning may not be represented by simple functions.

In the following, we introduce the spirit of five machine learning methods: k-nearest neighbors, decision tree, boosting, support vector machine, and neural network, with illustrative examples. Rigorous formulations for each machine learning method will not be covered here because they are out of the scope of this chapter.

14.4.1 *k*-Nearest Neighbors

The k-Nearest Neighbors (KNN) method is intuitive and easy to implement. First, a distance metric (such as the Euclidean distance) needs to be chosen to identify the KNNs for a sample of unknown category. Second, a weighting scheme (uniform weighting or distance weighting) to summarize the score of each category needs to be decided. The uniform weighting scheme gives equal weight for all neighbors regardless of its distance to the sample of unknown category, whereas the distance weighting scheme weights distant neighbors less. Third, the score for each category is summed over these KNNs. Finally, the predicted category of this sample is the category yielding the highest score.

An example is illustrated in Fig. 14.2. Suppose there are two classes (category A and category B) for the output and two features (x_1 and x_2). A sample of unknown category is plotted as a solid circle. KNN predicts the category of this sample as follows. To start, we choose Euclidean distance and uniform distance weight. If $K = 3$, in the three nearest neighbors to the unknown sample, there are one sample of

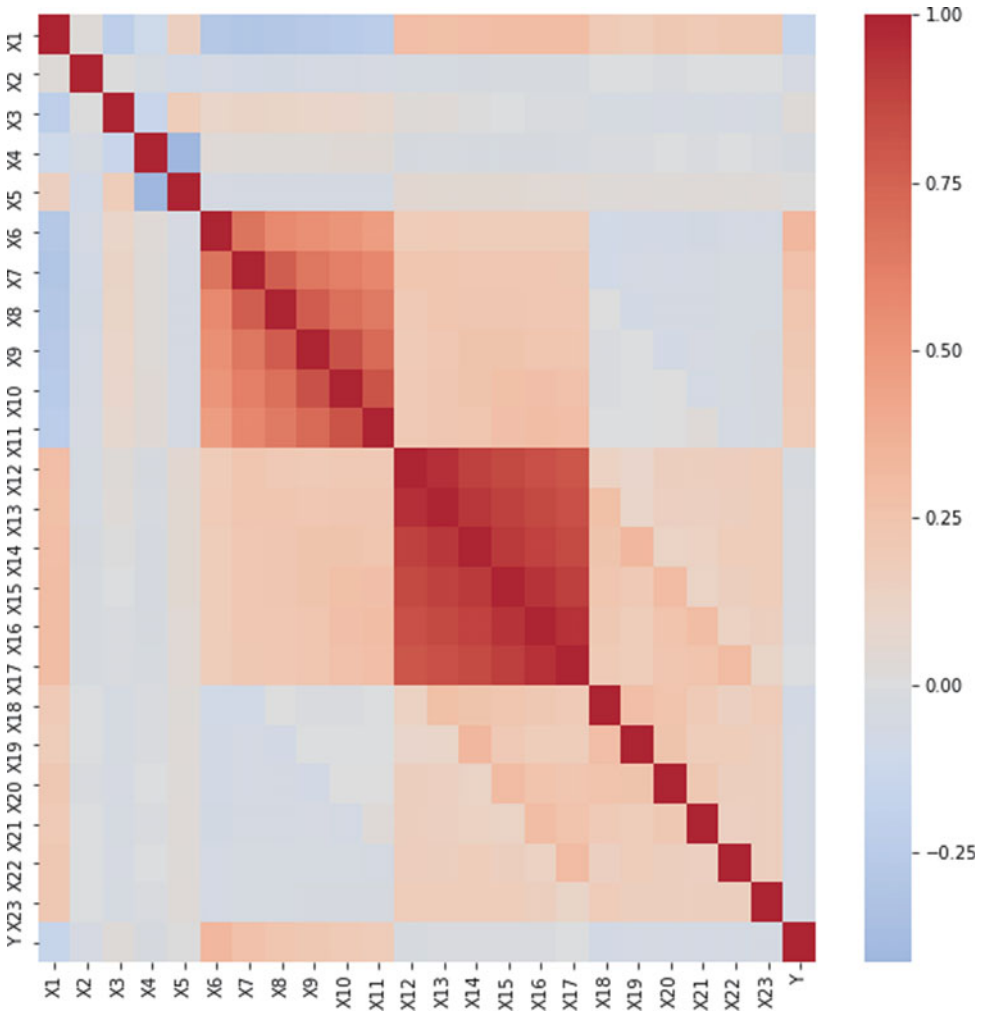


Fig. 14.1 The heatmap of correlations between the response variable and all predictors in the credit card dataset

category A and two samples of category B. Because there are more samples of category B, KNN predicts the unknown sample to be of category B. If $K = 6$, in the six nearest neighbors to the sample of unknown category, there are four samples of class A and two samples of class B. Because class A occurs more frequently than class B, KNN predicts the sample to be of category A.

In addition to the distance metric and weighting scheme, the number of neighbors K is needed to be decided. Indeed, the performance of the KNN is highly sensitive to the size of K. There is no strict rule in selecting l. In practice, the selection of K can be done by observing the predicted accuracies for various K and select the one that reach the highest training scores and cross-validation scores. Detailed descriptions about how to calculate these scores are given in Sect. 14.4.

14.4.2 Decision Trees

A decision tree is also called a classification tree when the target output variable is categorical. For a decision tree, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision tree is usually constructed top-down, by choosing a mapping of feature variables at each step that best splits the set of items. Different algorithms choose different metrics for measuring the homogeneity of the target variables within the subsets. These metrics are applied to each candidate subset and the resulting values are combined to provide a quality of the split. Common metrics include the Gini Index or Information Gain based on the concept of entropy.

Figure 14.3 depicts the structure of a decision tree: the decision tree starts with a root node and consists of internal decision nodes and leaf nodes. The decision nodes and leaf nodes are stemmed from the root node and are connected by branches. Each decision node represents a test function with discrete outcomes labeling the branches. The decision tree grows along with these branches into different depths of internal decision nodes. At each step, the data is classified by a different test function of attributes leading the data either to a deeper depth of internal decision node or it finally ends up at a leaf node.

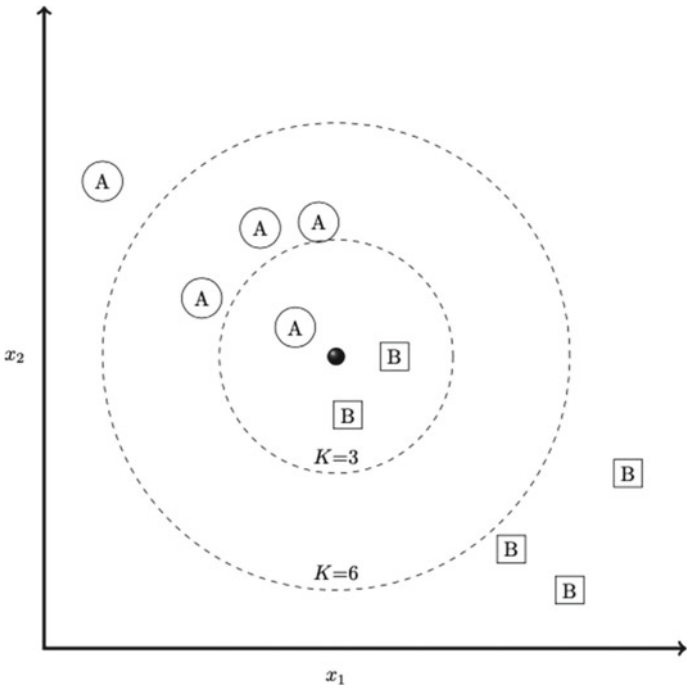


Fig. 14.2 Illustration of the k-nearest neighbors

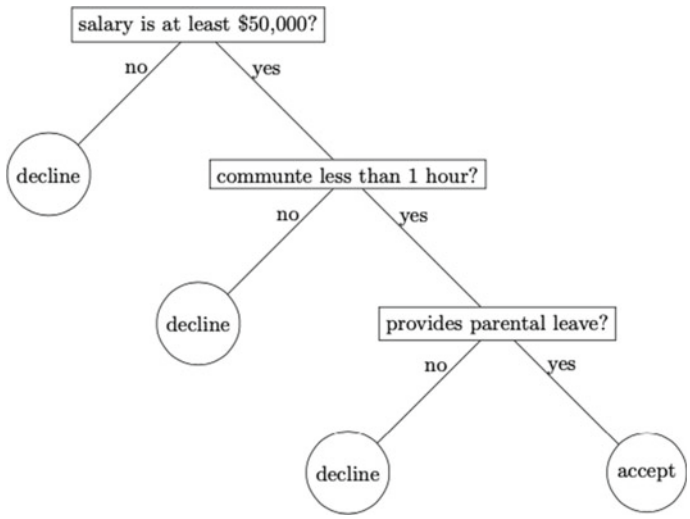


Fig. 14.3 Illustration of the decision tree

Figure 14.3 illustrates a simple example. Suppose an interviewer is classified as "decline offer" or "accept offer". The tree starts with a root node. The root node is a test function to check if the salary is at least $50,000. If data with answer "no" declines offer, the branch ends up with decline offer and hence is represented as a leaf node indicating "decline". If the answer is yes, the data remain contains samples of declining and accepting the offer. Therefore, this branch results in a second decision node to check if the interviewer needs commuting time more than 1 hour and the output could be "yes" and "no". If data with answer "yes" declines the offer, then this branch ends up with a leaf node indicating "decline". Data with answers no contains both declining and accepting the offer, so the branch ends up with another decision node to check if parental leave is provided. Again, the outcome is yes or no. For data with output no, all data decline the offer, so this branch ends up with a leaf node indicating "decline". Data with answer "yes" accepts the offer, so this branch ends up with a leaf node indicating "accept".

To apply the decision tree algorithm, we use the training data set to build a decision tree. For a sample with unknown category, we simply employ the decision tree to figure out which leaf node the sample of unknown category will end up.

Different algorithms choose different metrics for measuring the homogeneity of the target variables within the subsets. These metrics are applied to each candidate subset and the resulting values are combined to provide a quality of the split. Common metrics include the Gini Index and

Information Gain. The major difference between the Information Gain and the Gini Index is that the former produces multiple nodes, whereas the latter only produces two nodes (TRUE and FALSE, or binary classification).

The representative decision tree using Gini Index (also known as the Gini Split and Gini Impurity) to generate the next lower node is the classification and regression tree (CART), which indeed allows both classification and regression. Because CART is not limited to the types of response and independent variables, it is of wide popularity.

Suppose we would like to build up a next lower node, and the possible classification label is i, for $i = 1, ..., c$. Let p_i represent the proportion of the number of samples in the lower node classified as i. The Gini Index is defined as

$$\text{GiniIndex} = 1 - \sum_{i=1}^{c} (p_i)^2 \tag{14.1}$$

The attribute used to build the next node is the one that maximize the Gini Index.

The Information Gain is precisely the measure used by the decision tree ID3 and C4.5 to select the best attribute or feature when building the next lower node (Mitchell 1997). Let f denote a candidate feature, and D denote the data at current node and D_i denote the data classified as label i at the lower node, for $i = 1, \ldots, c$. $N = /D/$ is the number of the sample at current node, and $N_i = |D_i|$ is the number of sample classified as label i at the lower node. Then, the Information Gain is defined as

$$IG(D,f) = I(D) - \sum_{i=1}^{c} \frac{N_i}{N} I(D), \tag{14.2}$$

where I is an impurity measure, either the Gini Index as defined in Eq. (14.1) or the entropy. The entropy is defined as

$$I_e = -\sum_{i=1} p_i \log_2 p_i \tag{14.3}$$

Equation (14.2) can be regarded as the original information at current node minus the expected value of the impurity after the data D is partitioned using attribute f. Therefore, f is selected to maximize the IG. Entropy and Gini Impurity perform similarly in general, so we can focus on the adjustment of other parameters.

14.4.3 Boosting

In the filed of computer science, weak learner is a classification rule of lower accuracy, whereas strong learner is that of higher accuracy. The term "boosting" refers to a family of algorithms which convert weak learners to strong learners. There are many boosting algorithms, such as AdaBoost (Adpative Boosting), Gradient Tree Boosting, and XGBoost. Here, we focus on AdaBoost.

In an iterative process, boosting yields a sequence of weak learners which are generated by assuming different distributions for the sample. To choose the distribution, boosting proceeds as follows:

- Step 1: The base learner (or the first learning algorithm) assigns equal weight to each observation.
- Step 2: The weights of observations which are incorrectly predicted are increased to modify the distribution of the observation, so that a second learner is obtained.
- Step 3: Iterate Step 2 until the limit of base learning algorithm is reached, or higher accuracy is reached.

With the above procedures, a sequence of weak learner is obtained. The prediction of a new sample is based on the average (or weighted average) of each weak learners or that having the higher vote from all these weak learners.

14.4.4 Support Vector Machines

A support vector machine (SVM) is a recently developed technique originally used for pattern classification. The idea of SVM is to find a maximal margin hyperplane to separate data points of different categories. Figure 14.4 shows how the SVM separates the data into two categories with hyperplanes.

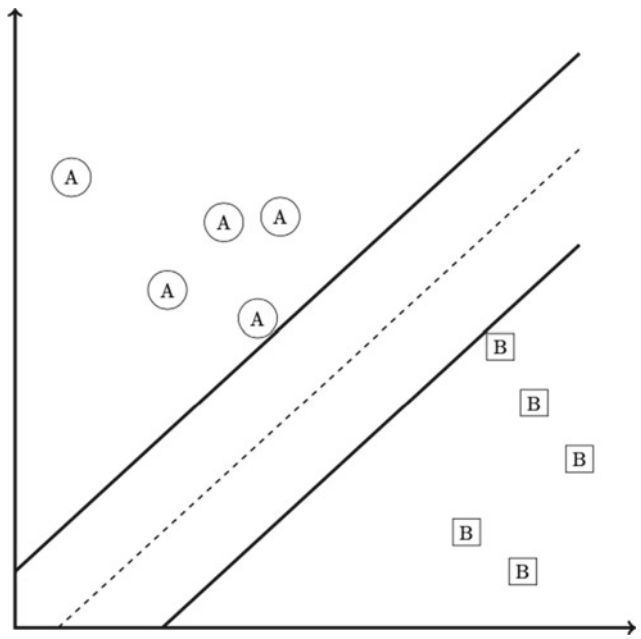


Fig. 14.4 Illustration of the support vector machine

If the classification problem cannot be separated by a linear hyperplane, the input features have to be mapped into a higher dimensional feature space by a mapping function, which is calculated through a prior chosen a kernel function. Kernel functions include linear, polynomial, sigmoid, and the radial basis function (RBF). Yang (2007) and Kim and Sohn (2010) apply SVM in credit scoring problem and show that SVM outperforms other techniques in terms of higher accuracy.

14.4.5 Neural Networks

A neural network (NN), or an artificial neural network, has the advantage of strong learning ability without any assumptions about the relationships between input and output variables. Recent studies using an NN or its variants in credit risk analysis can be found in Desai et al. (1996), Malhotra and Malhotra (2002), and Abdou et al. (2008).

NN links the input–output paired variables with simple functions called activation functions. A simple standard structure for an NN includes an input layer, a hidden layer, and an output layer. If an NN contains more than one hidden layer, it is also called as deep neural network (or deep learning neural network).

Suppose that there are unknown L layers in an NN. The original input layer and the output layer are also called the zeroth layer and $(L + 1)$th layer, respectively. The name of hidden layers implies that they are originally invisible in the data and are built artificially. The number of layers L is called the *depth* of the architecture. See Fig. 14.5 for an illustration of a structure of a neural network.

Each layer is composed of nodes (also called neurons) representing a nonlinear transformation of information from previous layer. The nodes in the input layer receive input features $X = (X_1, ..., X_p)$ of each training sample and transmit the weighted outputs to the hidden layer. The d nodes in the output layer represent the output features $Y = (Y_1, \ldots, Y_d)$.

Let $l \in \{1, 2, \ldots, L\}$ denote the index of the layers from 1 to L. NN trains a model on data to make predictions by passing learned features of data through different layers via L nonlinear transformation applied to input features. We explicitly describe a deep learning architecture as follows. For a hidden layer, various activation functions, such as logistic, sigmoid, and radial basis function (RBF), can be applied. We summarize some activation functions and their definitions in Table 14.1.

Let $f^{(0)}, f^{(1)}, \ldots, f^{(L)}$ be given univariate activation functions for these layers. For notational simplicity, let f be a given activation. Suppose $U = (U_1, \ldots, U_k)^{\mathrm{X}}$ is a k-dimensional input. We abbreviate $f(U)$ by

$$f(U) = (f(U_1), \ldots, f(U_k))^{\mathrm{X}}$$

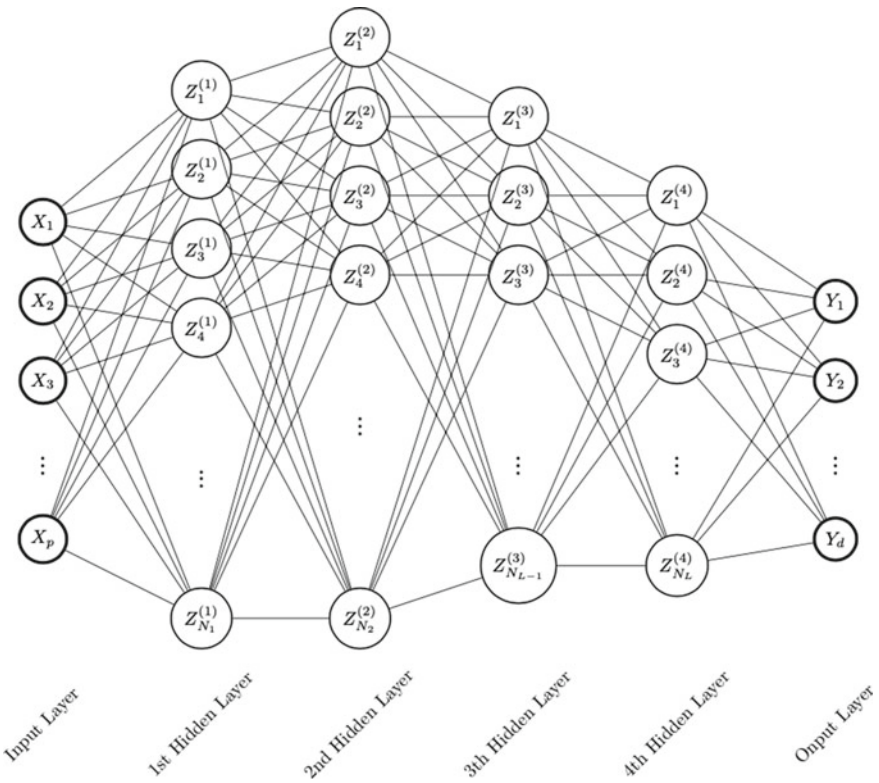


Fig. 14.5 Illustration of a neural network with four layers

Table 14.1 List of activation functions

Activation function	Definition
The identity function	$f(x) = x$
The logistic function	$f(x) = 1/(1 + \exp(-x))$
The hyperbolic tan function	$f(x) = \tanh(x)$
The rectified linear units (ReLU) function	$f(x) = \max\{x, 0\}$

Let N_l denote the number of nodes at the lth layers, for $l = 1, \ldots, L$. For notational consistency, let $N_0 = p$, and $N_{(L+1)} = d$. To build the lth layer, let $W^{(l-1)} \in R^{N_l \times N_{l-1}}$ be the weight matrix, and $b^{(l-1)} \in R^{N_l}$ be the thresholds or activation levels, for $l = 1, \ldots, L+1$. Then, these N_l nodes at the lth layers $Z^{(l)} \in R^{N_l}$ are formed by

$$Z^{(I)} = f^{(l-1)}\left(W^{(l-1)}Z^{(l-1)} + b^{(l-1)}\right),$$

for $l = 1, \ldots, L+1$. Specifically, the deep learning neural network is constructed by the following iterations:

$$\begin{aligned}
Z^{(1)} &= f^{(0)}\left(W^{(0)}X + b^{(0)}\right)\\
Z^{(2)} &= f^{(1)}\left(W^{(1)}Z^{(1)} + b^{(1)}\right)\\
Z^{(3)} &= f^{(2)}\left(W^{(2)}Z^{(2)} + b^{(2)}\right)\\
&\vdots\\
Z^{(I)} &= f^{(I-1)}\left(W^{(I-1)}Z^{(I-1)} + b^{(I-1)}\right)\\
&\vdots\\
Z^{(L)} &= f^{(L-1)}\left(W^{(L-1)}Z^{(L-1)} + b^{(L-1)}\right)\\
\hat{Y} &= f^{(L)}\left(W^{(L)}Z^{(L)} + b^{(L)}\right)
\end{aligned}$$

Finally, the deep learning neural network predicts using the Y by $\hat{Y}$ input W and the learning parameters $W = \{W^{(0)}, W^{(1)}, \ldots, W^{(L)}\}$ and $b = \{b^{(0)}, b^{(1)}, \ldots, b^{(L)}\}$. As a result, a deep learning neural network predicts Y by

$$F^{W,b}(X) := f^{(L)}\left(W^{(L)}Z^{(L)} + b^{(L)}\right).$$

Once the architecture of the deep neural network (i.e., L, and N_l for $i = 1, \ldots, L$) and activation functions $f^{(l)}$ for $l = 1, \ldots, L$ are decided, we need to solve the training problem to find the learning parameters $W = \{W^{(0)}, W^{(1)}, \ldots, W^{(L)}\}$ and $b = \{b^{(0)}, \ldots, b^{(L)}\}$, so that the solutions $\hat{W}$ and $\hat{b}$ satisfy

$$\hat{W}, \hat{b} = \arg\min_{W,b} \frac{1}{n}\sum_{i=0}^{n} L\left(Y^{(i)}, F^{W,b}\left(X^{(i)}\right)\right)$$

Here, L is the loss function.

Some drawbacks of building an NN are summarized below. First, the relationship between the input and output variables is mysterious because the structure of an NN could be very complicated. Second, how to design and optimize the NN structure is determined via a complicated experiment process. For instance, different combinations of number of hidden layers, number of nodes in each hidden layer, and activation functions in each layer, yield different classification accuracies. As a consequence, learning an NN is usually time consuming.

14.5 Study Plan

In Sect. 14.4.1, we describe how to preprocess the data, and describe the Python programming. We defer Python scripts in the appendix. Section 14.4.2 provides detailed descriptions on the tuning process to decide the optimal tuning parameter, because there is no quick access in selecting the optimal tuning parameters in each method. The performance of these five machine learning methods is compared using the learning curves.

14.5.1 Data Preprocessing and Python Programming

To start with, we preprocess the data as follows. Because the data set is quite complete, there is no missing data issue. We take log-transformation for continuous variables, such as X_{12} to X_{17} and X_{18} to X_{23}, because they are highly skewed.

Python is created by Guido van Rossum first released in 1991 and is a high-level programming language for general purpose programming. Python has been successfully applied to machine learning techniques with a wide range of applications. See Raschka (2015) for using Python for machine learning. For simplicity, we provide Python codes in the appendix to preprocess the data and apply machine learning methods to the data set.

14.5.2 Tuning Optimal Parameters

The optimal combination of parameters is decided based on criteria such as testing scores and cross-validation scores. To calculate the testing score, we split the data set randomly into 70% training set and 30% testing set. When fitting the algorithm, we only use the training set. Then, we use the remaining 30% testing set to calculate the percentage of correct classification of the method, which is also the prediction accuracy or testing score.

Furthermore, to investigate if the algorithm is stable and if the over-fitting problem exists, we calculate the cross-validation score. We further split the 70% training set into ten subsets, and fit the algorithm using nine of these subsets being the training data, and one set being the testing data. Rotating which set is the testing set, the average of these ten prediction accuracies is the cross-validation score.

Our selection rule for optimal tuning parameters goes as follows. We first plot the testing and cross-validation scores for various combinations of tuning parameters. The optimal tuning parameters are the simplest to achieve the highest testing scores, whereas the cross-validation scores are later used to check if the over-fitting problem exists.

The above procedures give a simple rule to select the optimal tuning parameters. We remark that there are other alternatives to select the optimal tuning parameters. For instance, the optimal combination of tuning parameters is selected to maximize the performance measure (such as the F1-score or AUC).

Figure 14.6 compares testing and cross-validation scores against various combinations of tuning parameters: k ranging from 1, 21, 41, ..., 81, and two weighting schemes (uniform weight and distance weight). Testing scores with uniform and distance weighting are about the same, which are also close to the two cross-validation scores. Therefore, we choose uniform weighting because it is simpler, and choose k to be 50 because all four scores appear to be stable for k larger than 50.

Figure 14.7 compares the testing and cross-validation scores for decision trees. We test both the Gini Index and entropy for the Information Gain splitting criteria. And we vary the number of samples in a node required to split it because this effectively varies the amount of pruning done to the decision tree. A low requirement lets the decision tree split the data into small groups, increases the complexity of the tree, and corresponds to low pruning. A high requirement prevents as many nodes being created, decreases the complexity of the tree, and corresponds to higher pruning.

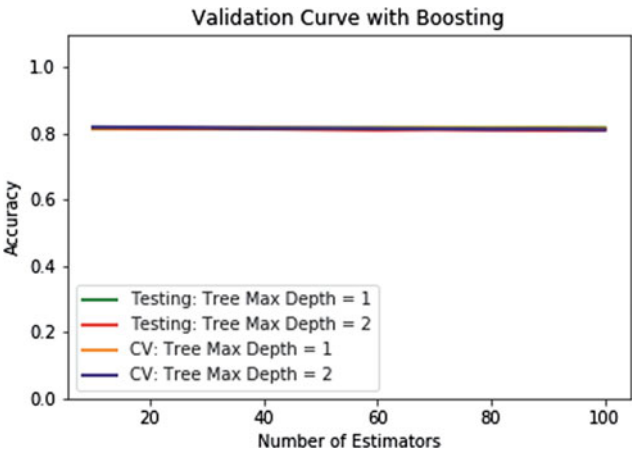


Fig. 14.7 Validation curves of decision tree against minimum samples splits with Gini Index and Information Gain using training data and cross-validation of the credit card dataset

Because testing scores of using Gini Index and entropy are close, we choose Gini Index because it is the default criteria for splitting. On the other hand, both training and cross-validation scores are not affected by the amount of pruning. Hence, we choose 80% of samples for minimum split requirement for a decision tree with a smaller maximum depth.

Figure 14.8 shows that the algorithm converges pretty quickly. This suggests, like the decision tree, that the data is fairly clustered. In terms of boosting, it would mean that there are not many hard instances, where the instance is an anomaly and the algorithm fails to compare it to other similar instances. We decide to use a maximum tree depth of one since it is more general and does not perform worse than a maximum depth of 2, and 10 estimators because it gives

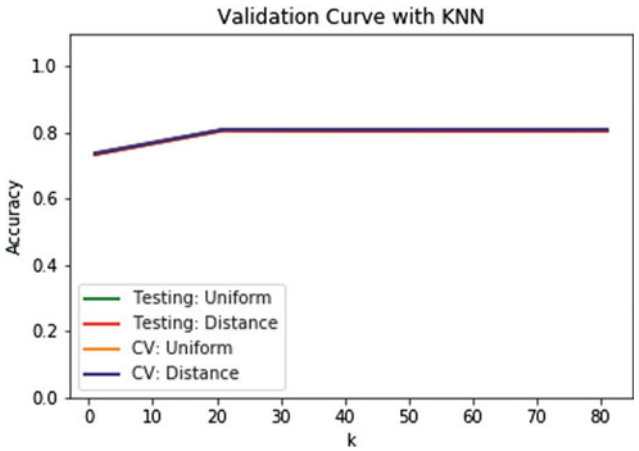


Fig. 14.6 Validation curves of the k-nearest neighbors against k with uniform weight and distance weight using training data and cross-validation of the credit card dataset

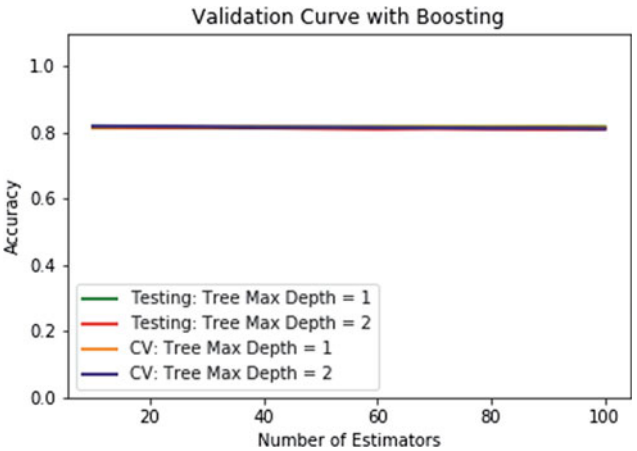


Fig. 14.8 Validation curves of boosting against a number of estimators with tree maximum depths of one and two using training data and cross-validation of the credit card dataset

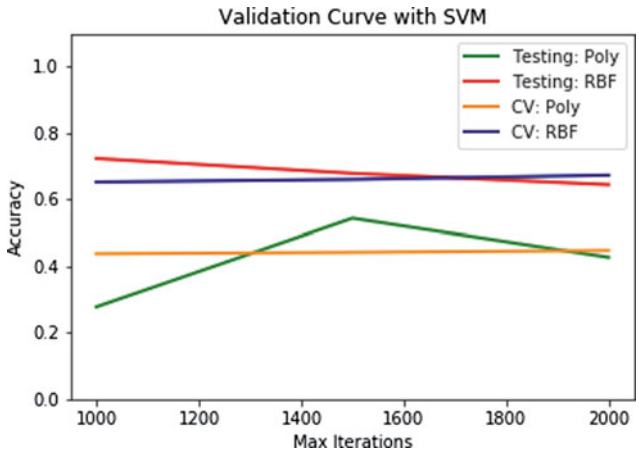


Fig. 14.9 Validation curves of the support vector machine against maximum iterations with polynomial and RBF functions using training data and cross-validation of the credit card data set

better performance and this data set does not benefit from having more estimators.

Figure 14.9 compares the testing and cross-validation scores with the SVM using both the polynomial and RBF kernels with maximum iterations ranging from 1100, 1600, 2100, and 2600. Our experiments suggest using the RBF kernel because it performs much better than the polynomial kernel, and it also runs faster than the polynomial kernel. In addition, we use a maximum iterations value of 2100, as no more improvements on the testing scores can be found with larger maximum iterations value.

We use the ReLU function as the activation function. For neural network, we decide to test the number of hidden layers and number of neurons in each hidden layer. Figure 14.10 compares the testing and cross-validation scores of neural networks. The upper panel varies the number of hidden layers, and suggests us to select the number of hidden layers to be three. With three hidden layers, the lower panel varies the number of hidden neurons in each layer, which suggests us to have 15 neurons as a suitable size in each layer.

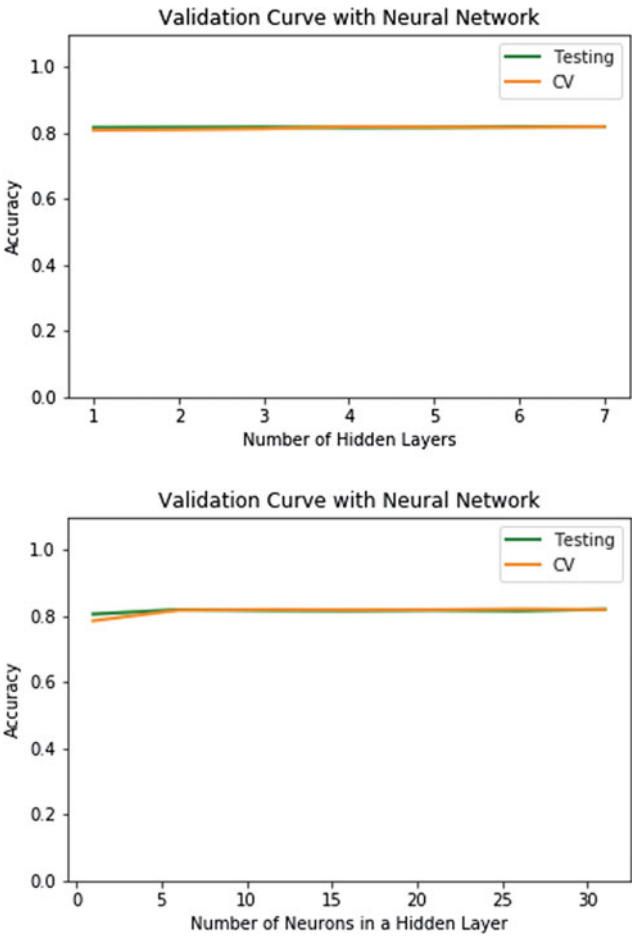


Fig. 14.10 Validation curves of neural network against number of hidden layers and number of neurons in each hidden layer, in the upper and lower panels, respectively, using training data and cross-validation of the credit card data set

14.5.3 Learning Curves

Figure 14.11 compares the accuracy with these five machine learning methods against the number of examples (the size of training set) with the optimal tuning parameters obtained in Sect. 14.4.2 to see if the accuracy appears to be stable as the number of examples increases. It is shown that KNN, decision tree, and boosting perform consistently as the number of examples increases. But, SVM's performs worse as the number of examples increases. As a conclusion, for the credit card data set, the decision tree algorithm performs the best. Not only does it yield the highest accuracy, but it runs the quickest.

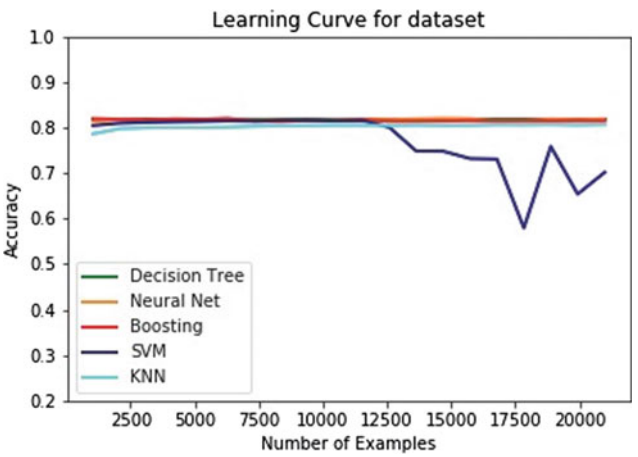


Fig. 14.11 Learning curves against number of examples with decision tree, neural network, boosting, support vector machine, and *k*-nearest neighbors, for the credit card dataset

14.6 Summary and Concluding Remarks

In this chapter, we introduce five machine learning methods: *k*-nearest neighbors, decision tree, boosting, support vector machine, and neural network, to predict the default of credit card holders. For illustration, we conduct data analysis using a data set of 29,999 instances with 23 features and provide Python scripts for implementation. It is shown in our study that the decision tree performs best in predicting the default of credit card holders in terms of learning curves.

As the risk management for personal debt is of considerable importance, it is worthy of studying the following directions for future research. One limitation in this paper is that we only use one data set. According to Butaru et al. (2016), multiple data sets should be used to illustrate the robustness of a machine learning algorithm, and pairwise-comparisons should be conducted to verify which machine learning algorithm outperforms the others (Demšar 2006; García and Herrera 2008).

This chapter only uses accuracy as a measure to compare different machine learning methods. Indeed, in addition to the standard measures, such as precision, recall, F1-score, and AUC, it is interesting to consider cost-sensitive framework or profit measures to compare different machine learning algorithms as in Verbraken et al. (2014), Bahnsen et al. (2015), and Garrido et al. (2018).

Along with the availability of voluminous data in recent days, Moeyersoms and Martens (2015) solve high-cardinality attributes in churn prediction in the energy sector. In addition, it is also interesting to predict for longer-horizon or the default time (using survival analysis). Last but not least, it is of considerable importance to develop a method for extremely rare event. All of the above-mentioned issues are worthy of future studies. In the next chapter, we will discuss how deep neural networks can be used to predict credit card delinquency.

Appendix 14.1: Python Codes

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

dataset = pd.read_csv('C:\\default␣of␣credit␣card␣clients.csv')
dataset.shape
dataset.head()
dataset = dataset.iloc[0:29999,:]
colForLog = ["X1",
             "X12","X13","X14","X15","X16","X17",
             "X18", "X19", "X20", "X21", "X22", "X23"]
for col in colForLog:
    dataset.loc[:, col] = ( dataset.loc[:, col] - dataset.loc[:,col].min()
    + dataset.loc[:,col].median()).apply(np.log)

dataset = dataset.drop(['Unnamed:␣0'], axis = 1)

X = dataset.drop(['Y'],axis=1)
y = dataset['Y']

# Get 70-30 split to get the training-testing data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.30, random_state = 42)


from sklearn import metrics # to get the accuracy
def calc_testing_score(X_train, y_train, X_test, y_test, clf):
    clf = clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)
    accuracy = metrics.accuracy_score(y_test, y_pred)
    return accuracy

from sklearn.neighbors import KNeighborsClassifier
```

```
param_range = range(1, 100, 20)
# (a) get testing score for distance = uniform
for i in range(1, 100, 20):
    clf = KNeighborsClassifier(n_neighbors = i)
    clf2 = KNeighborsClassifier(n_neighbors = i, weights = "distance")
    accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
    accuracyTest2 = calc_testing_score(X_train, y_train, X_test, y_test,
    clf2)

'''
II. Decision Tree 2/5
'''
from sklearn.tree import DecisionTreeClassifier
param_range = np.arange(.1, .9, .1)
for i in np.arange(.1, .9, .1):
    clf = DecisionTreeClassifier(min_samples_split = i)
    clf2 = DecisionTreeClassifier(criterion = "entropy",
    min_samples_split = i)
    accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
    accuracyTest2 = calc_testing_score(X_train, y_train, X_test, y_test,
    clf2)

'''
III. Boosting 3/5
'''
from sklearn.ensemble import AdaBoostClassifier
param_range = range(10, 109, 10)
for i in range(10, 109, 10):
   clf = AdaBoostClassifier(
   base_estimator = DecisionTreeClassifier(max_depth = 1),
    n_estimators = i)
   clf2 = AdaBoostClassifier(
   base_estimator = DecisionTreeClassifier(max_depth = 2),
    n_estimators = i)
   accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
   accuracyTest2 = calc_testing_score(X_train, y_train, X_test, y_test, clf2)
```

```
71
72  '''
73  IV. Support Vector Machine 4/5
74  '''
75  from sklearn.svm import SVC
76  param_range = range(1000, 5003, 500)
77  for i in range(1000, 10003, 500):
78      clf = SVC(kernel = "poly", max_iter = i)
79      clf2 = SVC(kernel = "rbf", max_iter = i)
80      accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
81      accuracyTest2 = calc_testing_score(X_train, y_train, X_test, y_test,
82      clf2)
83
84  '''
85  5/5 Neural networks
86  '''
87  from sklearn.neural_network import MLPClassifier
88  # (a) numHiddenLayers
89  param_range = range(1, 8, 1)
90  tup = (100,)
91  for i in range(1, 8, 1):
92      tup2 = (i, 10)
93      clf = MLPClassifier(activation = 'relu', hidden_layer_sizes = tup2)
94      accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
95
96  # (b) size of number of neurons in the
97  param_range = range(1, 36, 5)
98  for i in range (1, 36, 5):
99      clf = MLPClassifier(hidden_layer_sizes = (i, i, i))
100     accuracyTest = calc_testing_score(X_train, y_train, X_test, y_test, clf)
```

References

Abdou, H., Pointon, J. and Masry, A.E. (2008). Neural Nets Versus Conventional Techniques in Credit Scoring in Egyptian Banking. *Expert Systems with Applications* **35**(2), 1275–1292.

Addo, P.M., Guegan, D. and Hassani, B. (2018). Credit Risk Analysis Using Machine and Deep Learning Models. *Risks* **6**(2), 38.

Bahnsen, A.C., Aouada, D. and Ottersten, B. (2015). A Novel Cost-sensitive Framework for Customer Churn Predictive Modeling. *Decision Analytics* **2**(5), 1–15.

Borovykh, A., Bothe, S. and Oosterlee, C. (2018). Conditional Time Series Forecasting with Convolutional Neural Networks. https://arxiv.org/abs/1703.04691v4 (retrieved June 15, 2018).

Breiman, L. (1996). Bagging predictors. Machine Learning, 24, 123–140.

Butaru, F., Chen, Q., Clark, B., Das, S., Lo, A.W. and Siddique, A. (2016). Risk and Risk Management in the Credit Card Industry. *Journal of Banking and Finance* **72**, 218–239.

Bzdok, D., Altman, N. and Krzywinski, M. (2018). Statistics Versus Machine Learning.*Nature Methods* **15**(4), 233–234.

Caruana, R., Munson, A., & Niculescu-Mizil, A. (2006). Getting the most out of ensemble selection. Proceedings of the 6th international conference on data mining (pp. 828–833). Hong Kong, China: IEEE Computer Society.

De Mello, R.F. and Ponti, M.A. (2018). *Machine Learning: A Practical Approach on the Statistical Learning Theory.* Springer.

Demšar, J. (2006). Statistical Comparisons of Classifiers Over Multiple Data Sets. *Journal of Machine Learning Research* **7**, 1–30.

Desai, V.S., Crook, J.N. and Overstreet, G.A. (1996). A Comparison of Neural Networks and Linear Scoring Models in the Credit Union Environment. *European Journal of Operational Research* **95**(1), 24–47.

Fernandes, G.B. and Artes, R. (2016). Spatial Dependence in Credit Risk and its Improvement in Credit Scoring. *European Journal of Operational Research* **249**, 517–524.

Finlay, S. (2011). Multiple classifier architectures and their application to credit risk assessment. European Journal of Operational Research, 210, 368–378.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In L. Saitta (Ed.), Proceedings of the 13th international conference on machine learning (pp. 148–156). Bari, Italy: Morgan Kaufmann.

García, S. and Herrera, F. (2008). An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. *Journal of Machine Learning Research* **9**, 2677–2694.

Garrido, F., Verbeke, W. and Bravo, C. (2018). A Robust Profit Measure for Binary Classification Model Evaluation. *Expert Systems with Applications* **92**, 154–160.

Gu, S., Kelly, B. and Xiu, D. (2018). Empirical Asset Pricing via Machine Learning. Technical Report No. 18–04, Chicago Booth Research Paper.

Hand, D. J., & Henley, W. E. (1997). Statistical classification models in consumer credit scoring: A review. Journal of the Royal Statistical Society: Series A (General), 160, 523–541.

Hastie, T., Ribshirani, R. and Friedman, J. (2008). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York.

Heaton, J.B., Polson, N.G. and White, J.H. (2017). Deep Learning for Finance: Deep Portfolios. *Applied Stochastic Models in Business and Industry* **33**(3), 3–12.

James, G., Witten, D. Hastie, T. and Tibshirani, R. (2013). *An Introduction to Statistical Learning: With Applications in R*. Springer.

Kim, H.S. and Sohn, S.Y. (2010). Support Vector Machines for Default Prediction of SMEs Based on Technology Credit. *European Journal of Operational Research* **201**(3), 838–846.

Ko, A. H. R., Sabourin, R., & Britto, J. A. S. (2008). From dynamic classifier selection to dynamic ensemble selection. Pattern Recognition, 41, 1735–1748.

Kumar, P. R., & Ravi, V. (2007). Bankruptcy prediction in banks and firms via statistical and intelligent techniques—A review. European Journal of Operational Research, 180, 1–28.

Lee, C.F. (2020). Financial Econometrics, Mathematics, Statistics, and Financial Technology: An Overall View. *Review of Quantitative Finance and Accounting*. Forthcoming.

Lee, C.F. and Lee, J. (2020). *Handbook of Financial Econometrics, Mathematics, Statistics, and Machine Learning*. World Scientific, Singapore. Forthcoming.

Lessmann, S., Baesens, B., Seow, H.-V. and Thomas, L.C. (2015). Benchmarking State-of-the-Art Classification Algorithms for Credit Scoring: An Update of Research. *European Journal of Operational Research* **247**, 124–136.

Maldonado, S., Pérez, J. and Bravo, C. (2017). Cost-Based Feature Selection for Support Vector Machines: An Application in Credit Scoring. *European Journal of Operational Research* **261**, 656–665.

Malhotra, R. and Malhotra, D.K. (2002). Differentiating Between Good Credits and Bad Credits Using Neuro-Fuzzy Systems. *European Journal of Operational Research* **136**(1), 190–211.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill.

Moeyersoms, J. and Martens, D. (2015). Including High-cardinality Attributes in Predictive Models: A Case Study in Churn Prediction in the Energy Sector. *Decision Support Systems* **72**, 72–81.

Mohri, M., Rostamizadeh, A. and Talwalkar, A. (2012). *Foundations of Machine Learning*. MIT Press.

Paleologo, G., Elisseeff, A., & Antonini, G. (2010). Subagging for credit scoring models. European Journal of Operational Research, 201, 490–499.

Partalas, I., Tsoumakas, G., & Vlahavas, I. (2010). An ensemble uncertainty aware mea- sure for directed hill climbing ensemble pruning. Machine Learning, 81, 257–282.

Putra, E.F. and Kosala, R. (2011). Application of Artificial Neural Networks to Predict Intraday Trading Signals. In *Proceedings of 10th WSEAS international conference on e-activity*, Jakatar, Island of Java, pp. 174–179.

Raschka, S. (2015). *Python Machine Learning*. Packt, Birmingham, UK.

Russell, S. and Norvig, P. (2010). *Artificial Intelligence: a Modern Approach*, 3rd Edition. Prentice-Hall.

Samuel, A.L. (1959). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research of Development* **3**(3), 210–229.

Solea, E., Li, B. and Slavković, A. (2018). Statistical Learning on Emerging Economies. *Journal of Applied Statistics* **45**(3), 487–507.

Sun, T. and Vasarhelyi, M. A. (2018). Predicting Credit Card Delinquencies: An Application of Deep Neural Network. *Intelligent Systems in Accounting, Finance and Management* **25**, 174–189.

Woloszynski, T., & Kurzynski, M. (2011). A probabilistic model of classifier competence for dynamic ensemble selection. Pattern Recognition, 44, 2656–2668.

Yang, Y.X. (2007). Adaptive Credit Scoring with Kernel Learning Methods. *European Journal of Operational Research* **183**(3), 1521–1536.

Verbraken, T., Bravo, C., Weber, R. and Baesens, B. (2014). Development and Application of Consumer Credit Scoring Models Using Profit-based Classification Measures. *European Journal of Operational Research* **238**(2), 505–513.

Yeh, I.-C. and Lien, C.-H. (2009). The Comparisons of Data Mining Techniques for the Predictive Accuracy of Probability of Default of Credit Card Clients. *Expert Systems with Applications* **36**, 2473–2480.

Deep Learning and Its Application to Credit Card Delinquency Forecasting

15

By Ting Sun, The College of New Jersey

15.1 Introduction

This chapter aims to introduce the theory of deep learning (also called deep neural networks (DNNs)) and provides an example of its application to credit card delinquencies prediction. It explains the inner working of a DNN, differentiates it with traditional machine learning algorithms, describes the structure and hyper-parameters optimization, and discusses techniques that are frequently used in deep learning and other machine learning algorithms (e.g., regularization, cross-validation, and under/over sampling). It demonstrates how the algorithm can be used to solve a real-life problem. It partially adopts the data analysis part from Sun and Vasarhelyi (2018)'s research to illustrate how the theory of deep learning algorithm can be put into practice.

There is an increasing high risk of credit card delinquency globally. In the US., according to NerdWallet's statistics, "credit card balances carried from one month to the next hit $438.8 billion in March 2020," and "credit card debt has increased more than 6% in the past year and more than 31% in the past five years" (Issa 2019).

A number of machine learning techniques have been proposed to evaluate credit card related risks and performed well, such as discriminant analysis, logistic regression, decision trees, and support vector machine (Marqués et al. 2012), and traditional artificial neural networks (Koh and Chan 2002; Thomas 2000). As an emerging artificial intelligence (AI) technique, deep learning has been applied and achieved "state-of-the-art" performance in healthcare, computer games, and other areas where data is complex and large (Hamet and Tremblay 2017). This technology exhibits great potential to be used in many other fields where human decision-making is inadequate (Ohlsson 2017).

Sun and Vasarhelyi (2018) authored a paper entitled "*predicting credit card delinquencies: an application of deep neural networks.*" The data used in their paper is from a major bank in Brazil, and it contains demographic characteristics (e.g., the occupation, the age, and the region of residence) and historical transactional information (e.g., the total amount of cash withdrawals) of credit card holders. The objective is to evaluate the risk of credit card delinquencies with a deep learning approach. This research evidences the effectiveness of DNN in assisting financial institutions to quantify and manage credit risk for the decision-making of credit card issuance and loan approval. The proposed deep learning model is compared to other machine learning algorithms, and found to be superior than other ones in terms of better F_1 and AUC, which are metrics of overall predictive accuracy. The result suggests that, for a real-life data set with large volume, severe imbalance issue, and complex structure, deep learning would be an effective tool to help detect outliers.

The remainder of this chapter is organized as follows. Section two reviews prior literature other than Sun and Vasarhelyi (2018) using deep learning to predict default risks. Section three overviews deep learning method and introduces the structure of deep learning and its hyper-parameters. Section four describes the dataset and attributes. The modeling process and results are presented and reported in Section five and Section six, respectively. Section seven concludes the chapter.

15.2 Literature Review

Evaluating the risk of credit card delinquencies is a challenging problem in credit risk management. Prior research considers it a complex and non-liner problem requiring sophisticated approaches (Albanesi and Domonkos 2019). The research stream of using deep learning technology to predict credit card delinquencies contains a limited number of papers. Using a dataset from UCI machine learning repository,[1] Hamori et al. (2018) develop a list of machine learning models to predict credit card default payments. The dataset has a total number of 30,000 observations, where 6636 observations are default payments. There are 23

[1] UCI Machine Learning Repository can be accessed via https://archive.ics.uci.edu/ml/index.php.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_15

predictors, including the information about the credit card holder's basic demographic data, historical payment record, the amount of bill statements, as well as the amount of previous payments. They compare the performance of deep learning models with various activation functions to ensemble-learning techniques, bagging, random forest, and boosting. The results show that boosting has the strongest predictive power and the performance of deep learning models relies on the choice of activation function (i.e., Tanh and ReLu), the number of hidden layers, and the regularization method (i.e., Dropout).

As a simple application of deep learning, Zhang et al. (2017) also analyze a dataset from UCI machine learning repository and develop a prediction model for credit card default. Their data represents Taiwan's credit card defaults in 2005 and consists of 22 predictors, including age, education, marriage, and financial account characteristics. The result of the developed deep learning model is compared to those of linear regression and support vector machine. It finds that deep learning outperforms other models in terms of processing ability, which is suitable for large, complex financial data.

Using a dataset of 29,999 observations with 23 predictors from a major bank in Taiwan obtained from UCI machine learning repository, Teng and Lee (2019) examine the predictive capabilities of five techniques, the nearest neighbors, decision trees, boosting, support vector machine, and neural networks, for credit card default. Their work shows an inconsistent result from prior ones: the decision tree performs best among others in terms of validation curves.

Albanesi and Domonkos (2019) claim that deep learning approach is "specifically designed for prediction in environments with high dimensional data and complicated nonlinear patterns of interaction among factors affecting the outcome of interest, for which standard regression approaches perform poorly." A deep learning-based prediction model is proposed for consumer default using an anonymized credit file data from the Experian credit bureau. The data comprises more than 200 variables for 1 million households, describing information on credit cards, bank cards, other revolving credit, auto loans, installment loans, business loans, etc. For the proposed model, they apply dropout to each layer and ReLu at all neurons. Their results show that the proposed model consistently outperforms conventional credit scoring models.

15.3 The Methodology

15.3.1 Deep Learning in a Nutshell

Deep learning is also called deep neural networks (DNN). Due to technical limitations, although the concept of the artificial neural network (ANN) is decades old, ANNs have not achieved solid progress until early 2000s when deep learning was firstly introduced by Hinton et al. (2006) in a paper named "A Fast Learning Algorithm for Deep Belief Nets." In their paper, Hinton and his colleagues develop a deep neural network capable of classifying handwritten digits with high accuracy. Since then, scholars have explored this technique and demonstrated that deep learning is capable of achieving state-of-art achievements in various areas, such as self-driving car, game of Go, and Natural Language Processing (NLP).

A DNN consists of a number of layers of artificial neurons which are fully connected to one another. The central idea of DNN is that layers of those neurons automatically learn from massive amounts of observational data, recognize the underlying pattern, and classify the data into different categories. As shown in Fig. 15.1, a simple DNN consists of interconnected layers of neurons (as represented by circles in Fig. 15.1). It contains one input layer, two hidden layers,[2] and one output layer. The input layer receives the raw data, identifies the most basic element of the data, and passes it to the hidden layers. The hidden layer further analyzes, extracts data representations, and sends the output to the next layer. After receiving the data representations from its predecessor layer, the output layer categorizes the data into predefined classes (e.g., students' grade A, B, and C). Within each layer, complex nonlinear computations are executed by the neuron, and the output will be assigned with a weight. The weighted outputs are then combined through a transformation and transferred to the next layer. As the data is processed and transmitted from one layer to another, a DNN extracts higher level data representations defined in terms of other, lower-level representations (Bengio 2012a, b; Goodfellow et al. 2016; Sun and Vasarhelyi 2017).

15.3.2 Deep Learning Versus Conventional Machine Learning Approaches[3]

A DNN is a special case of a traditional artificial neural network with deeper hierarchical layers of neurons. Today's large quantity of available data and tremendous increase in computing power make it possible to train neural networks with deep hierarchical layers. With the great depth of layers and the massive number of neurons, a DNN has much greater representational capability than a traditional one with only one or two hidden layers. In a DNN, with each iteration of model training, the final classification result provided by the output layer will be compared to the actual observation

[2] A DNN typically has more than two hidden layers. For simplicity, I use two hidden layers.

[3] This subsection is partially adopted from Sun and Vasarhelyi (2018).

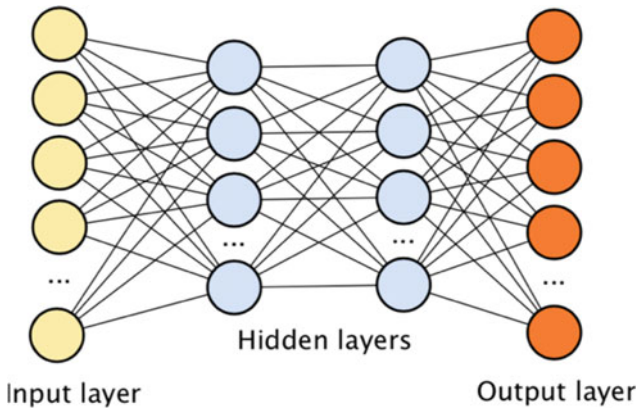


Fig. 15.1 Architecture of a simplified deep neural network Adopted from Marcus (2018)

to compute the error, and the DNN gradually "learns" from the data by updating the weight and other parameters in the next rounds of training. After numerous rounds of model training, the algorithm iterates through the data until the error cannot be reduced any further (Sun and Vasarhelyi 2017). Then the validation data is used to examine the data overfitting, and the selected model is used to predict the holdout data, which is the out-of-sample test. The paper will discuss the concepts of weights, iterations, overfitting, and out-of-sample test in the next section.

A key feature of deep learning is that it performs well in terms of feature engineering. While traditional machine learning usually relies on human experts' knowledge to identify critical data features to reduce the complexity of the data and eliminate the noise created by irrelevant attributes, deep learning automatically learns highly abstract features from the data itself without human intervention (Sun and Vasarhelyi 2017). For example, a convolutional neural network (CNN) trained for face recognition can identify basic elements such as pixels and edges in the first and second layers, then parts of faces in successive layers, and finally a high-level representation of a face as the output. This characteristic of DNNs is seen as "*a major step ahead of traditional Machine Learning*" (Shaikh 2017). Another important difference between deep learning and other machine learning techniques is its performance as the scale of data increases. Deep learning algorithms learn from past examples. As a result, they need a sufficiently large amount of data to understand the complex pattern underlying. A DNN may not perform better than traditional machine learning algorithms like decision trees when the dataset is small or simple. But their performance will significantly improve as the data scales increases (Shaikh 2017).

Therefore, deep learning performs excellently for unstructured data analysis and has produced remarkable breakthroughs. It can now automatically detect objects in images (Szegedy 2014), translate speeches (Levy 2016), understand text (Abdulkader et al. 2016), and play board game Go (Silver et al. 2016) on real-time basis at better than human-level performance (Heaton et al. 2016). Professionals in leading accounting firms delve into this technology. KPMG's Clara can review the full population of data to detect irregularities; Halo from PwC is capable of performing risk assessment; Deloitte's Argus is able to review textual documents like invoices and emails; EY develops a speech recognition system, Goldie.

15.3.3 The Structure of a DNN and the Hyper-Parameters

(1) Layers and neurons

As mentioned earlier, a DNN is composed of layers containing neurons. To construct a DNN, it firstly needs to determine the number of layers and neurons. There are many types of DNN. For example, multi-layer perceptron (MLP), convolutional neural network (CNN), recursive neural network (RNN), and recurrent neural network (RNN). The architectural of a DNN is as below:

a. The input layer

There is only one input layer as the goal of which is to receive the data. The number of neurons comprising the layer is typically equal to the number of variables in the data (sometimes, one additional neuron is included as a bias neuron).

b. The output layer

Similar to the input layer, a DNN has exactly one output layer. The number of neurons in the output layer is determined by the objective of the model. If the model is a regressor, the output layer has a single neuron, while the number of the neuron for a classifier is determined by the number of class labels for the dependent variable.

c. The hidden layers

There are no "rules of thumb" for choosing the number of hidden layers and neurons on each layer. It depends on the complexity of the problem and the nature of the data. For many problems, it starts with one single hidden layer and examines the prediction accuracy. It keeps adding more layers until the test error does not improve anymore (Bengio 2012a, b). Likewise, the choice of the number of neurons is based on "trial and error." This paper starts with minimum neurons and increases the size until the model achieves its optimal performance. In other words, it stops adding neurons when it starts to overfit the training set.

(2) Other hyper-parameters

a. Weight and bias

From the prior discussion, we learned that, in a neural network, inputs are received by the neurons in the input layer and then are transmitted between layers of neurons which are fully connected to each other. The input in a predecessor layer must be strong enough to be passed to the successor layer. To make the input data transmittable between layers, a weight along with a bias term is applied to the input data to control the strength of the connection between layers. That is, the weight affects the amount of influence the input will have on the output. Initially, a neural network will be assigned with random weights and biases before training begins. As training continues, the weights and biases are adjusted on the basis of "trial and error" until the model achieves its best predictive performance, that is the difference between desired value and model output (as represented by the cost function which will be discussed later) is minimized.[4]

Bias is a constant term added to the product of inputs and weights, with the objective of shifting the output toward the positive or negative side to reduce its variance. Assuming you want a DNN to return 2 when all the inputs are 0s. If the result of the activation function, which is the product of inputs and weights, is 0, you may add a bias value of 1 to ensure the output is 1. What will happen if you do not include the bias? The DNN is simply performing a matrix multiplication on the inputs and weights. This could easily introduce an overfitting issue (Malik 2019).

b. Cost function

A cost function is a measure of the performance of a neural network with respect to its given training sample and the expected output. An example of a cost function is Mean Squared Error (MSE), which is simply a squared difference between every output and true value and takes the average. Other more complex examples include cross-entropy cost, exponential cost, Hellinger distance, Kullback–Leibler divergence, and so on.

c. Activation function

The activation function is a mathematical function applied between the input that is received in the current neuron and the output that is transmitting to the neuron in the next layer.[5] Specifically, the activation function is used to introduce nonlinearity to the DNN. It is a nonlinear transformation performed over the input data, and the transformed output will then be passed to the next layer as the input data (Radhakrishnan 2017). Activation functions help the neural network learn complex data and provide accurate predictions. Without the activation function, the weights of the neural network would simply execute a linear transformation and even a deep stack of layers is equivalent to a single layer, which is too simple to learn complex data (Gupta 2017). In contrast, "*a large enough DNN with nonlinear activations can theoretically approximate any continuous function*" (Géron 2019). Some frequently used nonlinear activation functions include Sigmoid (also called Logistic), TanH (Hyperbolic Tangent), ReLU (Rectified Linear Unit), Leaky ReLU, Parametric ReLU, Softmax, Swish, and more. Each of them has its own advantages and disadvantages and the choice of the activation function relies on trial and error. A classification MLP often uses ReLu in its hidden layers and Softmax or Sigmoid in the output layer (Géron 2019). As shown in Fig. 15.2, a diagram describing the inner working of a neural network. In a neural network, a neuron is a basic processing unit, performing two functions: collecting inputs and producing the output. Once received by a neuron, each input is multiplied by a weight, and the products are summed and added with biases, then an activation function is applied to produce an output as shown in Fig. 15.2 (Mohamed 2019).

d. Learning rate, batch, iteration, and epoch

Since machine learning projects typically use limited size of data, to optimize the learning, this study employs an iterative process of continuously adjusting the values of model weight or bias. This strategy is called Gradient Descent (Rumelhart et al. 1986; Brownlee 2016b). Explicitly, updating the parameters once is not enough as it will lead to underfitting (Sharma 2017). Hence the entire training data needs to be passed through (forward and backward) and learned

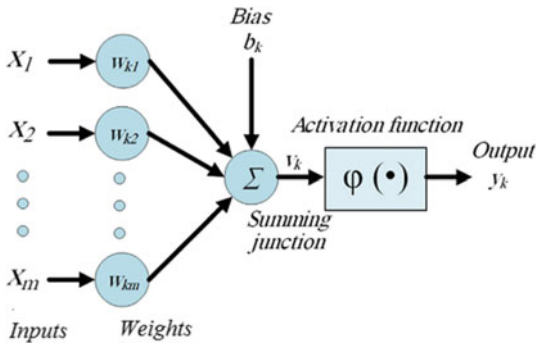


Fig. 15.2 The inner working of a neural network Adopted from Mohamed (2019)

[4] For more information about weights and biases, read https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network and https://docs.paperspace.com/machine-learning/wiki/weights-and-biases.

[5] For more information about activation functions, read https://missinglink.ai/guides/neural-network-concepts/7-types-neural-network-activation-functions-right/.

by the algorithm multiple times until it reaches the global minimum of the cost function. Each time the entire data is passed through the algorithm is called one epoch. As the number of epochs increases, a greater number of times the parameters are updated in the neural network, the training accuracy as well as the validation accuracy will increase.[6] Because it is impossible to pass the entire dataset into the algorithm at once, the dataset is divided into a number of parts called batches. the number of batches needed to complete one epoch is called the number of iterations. The learning rate is the extent to which the parameters are updated during the learning process. A lower learning rate requires more epochs, as the smaller adjustment is made to the parameters of each update, and vice versa (Ding et al. 2020).

e. Overfitting and regularization

A very complex model may cause an overfitting issue, which means that the model performs excellently on the training set, but has a low predictive accuracy on the testing set. This is because a complex model such as DNN can detect idiosyncratic patterns in training set. If the data contains lots of noises (or if it is too small), the model actually detects patterns in the noise itself, instead of generalizing to the testing set (Geron 2019). To avoid overfitting, one can employ a regularization constraint to make the model simpler to reduce the generalization error. One will tune regularization parameters to control the strength of regularization applied during the learning process.

There are several regularization techniques such as L1 and L2 regularization, dropout, and early stopping. L1 or L2 regularization works by applying a penalty term to the cost function to limit the capacity of models. The strength of regularization is controlled by the value of its parameters (e.g., lambda), By adding the regularized term, the values of weight matrices decrease, which in turn reduces the complexity of the model (Kumar 2019). Dropout is one of the most frequently used regularization techniques in DNN. At every iteration of learning, it randomly removes some neurons and all of their incoming and outgoing connections. Dropout can be applied to both the input layer and hidden layers. This approach can be considered an ensemble technique as it allows each iteration to have a different set of neurons resulting in a different set of outputs. A parameter, the probability, is used to control the number of neurons that will be deleted (Jain 2018). Early stop technique is a cross-validation strategy where we partition one part of the training set as the validation set. We learn the data patterns with the training set to construct a model and assess the performance of the model on the validation set. Specifically, the study monitors the model's predictive errors on the validation set. If the performance of the model on the validation set is not improving while the training error is decreasing, it immediately stops training the model further. Two parameters need to be configured. One is the quantity that needs to be monitored (e.g., validation error); the other is the number of epochs with no further improvement after which the training will be stopped (Jain 2018).

[6] However, when the number of epochs reaches a certain point, the validation accuracy starts decreasing while the training accuracy is still increasing. This means the model is overfitting. Thus, the optimal number of epochs is the point where the validation accuracy reaches its highest value.

15.4 Data

The credit card data in the data analysis part is from a large bank in Brazil. The final dataset consists of three subsets, including (1) a dataset describing the personal characteristics of the credit card holder (e.g., gender, age, annual income, residential location, occupation, account age, and credit score); (2) a dataset providing the accumulated transactional information at account level recorded by the bank in September 2013 (e.g., the frequency that the account has been billed, the count of payments, and the number of cash withdrawals in domestic); and (3) a dataset containing account-level transactions in June 2013 (e.g., credit card revolving payment made, the amount of authorized transaction exceeded the evolve limit of credit card payment, and the number of days past due).

The original transaction set contains 6,516,045 records at the account level based on transactions made in June 2013, among which 45,017 are made with delinquent credit card, and 6,471,028 are legitimate. For each credit card holder, the original transaction set is matched with the personal characteristics set and the accumulated transactional set. The objective of this work is to investigate the credit card holder's characteristics and the spending behaviors and use them to develop an intelligent prediction model for credit card delinquency. Some transactional data is aggregated at the level of credit card holder. For example, all the transactions made by the client are aggregated on all credit cards owned and generate a new variable, TRANS_ALL. Another derived variable, TRANS_OVERLMT, is the average amount of authorized transactions that exceed the credit limit made by the client on all credit cards owned.

Table 15.1 The data structure

Panel A: delinquent versus legitimate observations			
Dataset	Delinquent Obs. (percentage)	Legitimate Obs. (percentage)	Total (Percentage)
Credit card data	6,537 (0.92%)	704,860 (99.08%)	711,397 (100%)
Panel B: data content			
Data categories[7]	No. of data fields	Time period	
Client characteristics	15	As of September 2013	
Accumulative transactional information	6	As of September 2013	
Transactional information	23	June 2013	
Total	44		

After summarization, standardization, eliminating observations with missing variables, and discarding variables with zero variations, there are 44 input data fields (among which, 15 fields are related to credit card holders' characteristics, 6 variables provide accumulative information for all past transactions made by the credit card holder based on the bank's record as of September 2013, and 23 attributes summarize the account-level records in June 2013), which are linked to 711,397 credit card holders. In other words, for each credit card holder, there are 15 variables describing his or her personal characteristics, 6 variables summarizing his or her past spending behavior, and 23 variables reporting the transactions the client made with all credit cards owned in June 2013. The final data is imbalanced because only 6,537 clients are delinquent. In this study, a credit card client is defined as delinquent when any of his or her credit card account was permanently blocked by the bank in September 2013 due to the credit card delinquency. Table 15.1 summarized the input data. The input data fields are listed and explained in Appendix 15.1.

15.5 Experimental Analysis

The data analysis process is performed with an Intel (R) Xeon (R) CPU (64 GB RAM, 64-bit OS). The software used in this analysis is H_2O, an open source machine learning and predictive analytics platform. H_2O provides deep learning algorithms to help users train DNNs based on different problems (Candel et al. 2020). This research uses H_2O Flow, which is a notebook-style user interface for H_2O. It is a browser-based interactive environment allowing uses to import files, split data, develop models, iteratively improve them, and make predictions. H_2O Flow blends command-line computing with a graphical user interface, providing a point-and-click interface for every operation (e.g., selecting hyper-parameters).[8] This feature enables users with limited programming skills such as auditors to build their own machine learning models much easier than they do with other tools.

15.5.1 Splitting the Data

The objective of data splitting in machine learning is to evaluate how well a model will generalize to new data before putting the model into production. The entire data is divided into two sets: the training set and the test set. A data analyst typically trains the model using the training set and tests it using the test set. By evaluating the error rate on the test set, the data analyst can evaluate the error rate on new data in the future. But how to choose the best model? More specifically, how to determine what is the best set of hyper-parameters that make a model outperform others? A solution to this is to tune those hyper-parameters by holding out part of the training set as a validation set and monitoring the performance of all candidate models on the validation set. With this approach, multiple models with various hyper-parameters are trained on the reduced training set, which is the full training set minus the validation set, and the model that performs best on the validation set will be chosen. The current analysis uses cross-validation technique. Cross-validation[9] is a popular method, especially when the data size is limited. It makes fully use of all data instances in the training set and generally results in a less biased estimate than other methods (Brownlee 2018).

[7] A description of the attributes in each data category is provided in Appendix 15.1.

[8] https://www.h2o.ai/h2o-old/h2o-flow/.

[9] For more information about cross-validation, read https://towardsdatascience.com/5-reasons-why-you-should-use-cross-validation-in-your-data-science-project-8163311a1e79.

First, 20% of the data is held as a test set,[10] which will be used to give a confident estimate of the performance of the final tuned model. The stratified sampling method is applied to ensure that the test set has the same distribution of both classes (delinquent vs. legitimate class) as the overall dataset. For the remaining 80% of the data (hereafter called "remaining set"), fivefold cross-validation is applied. In H_2O, the fivefold cross-validation works as follows. Totally six models are built. The first five models are called cross-validation models. The last model is called main model. In order to develop the five cross-validation models, the remaining set is divided into five groups using stratified sampling to ensure each group has the same class distribution. To construct the first cross-validation model, group 2, 3, 4, and 5 are used as training data, and the constructed model is used to make predictions on group 1; to construct the second cross-validation model, group 1, 3, 4, and 5 are used as training data, and the constructed model is used to make predictions on group 2, and so on. So now it has five holdout predictions. Next, the entire remaining set is trained to build the main model, with training metrics and cross-validation metrics that will be reported later. The cross-validation metrics are computed as follows. The five holdout predictions are combined into one prediction for the full training dataset. This "holdout prediction" is then scored against the true labels, and the overall cross-validation metrics are computed. This approach scores the holdout predictions freshly rather than taking the average of the five metrics of the cross-validation models (H_2O.ai 2018).

15.5.2 Tuning the Hyper-Parameters

Hyper-parameters need to be configured before fitting the model (Tartakovsky et al. 2017). The choice of hyper-parameters is critical as it determines the structure and the variables controlling how the network is trained (e.g., the learning rate and weight) (Radhakrishnan 2017), which will in turn makes the difference between poor and superior predictive performance (Tartakovsky et al. 2017). To select the best value for hyper-parameters, two prevalent hyper-parameter optimization techniques are frequently used: Grid Search and Randomized Search.

The basic idea of Grid Search is that the user selects several grid points for every hyper-parameter (e.g., 2, 3, and 4 for the number of hidden layers) and trains the model using every combination of those values of hyper-parameters. The combination that performs the finest will be selected. Unlike Grid Search, Randomized Search evaluates a given number of random combinations. At each iteration, it uses one single random value for each hyper-parameter. Assuming there are 500 iterations as controlled by the user, Randomized Search uses 500 random values for each hyper-parameter.

In contrast, Grid Search tries all combinations of only several values as selected by the user for each hyper-parameter. This approach works well when we are exploring relatively few combinations, but when the hyper-parameter search space is large, Randomized Search is more preferable as you have more control over the computing cost for hyper-parameter search by controlling the number of iterations.

In this analysis, Grid Search is employed to select some key hyper-parameters and other settings in the DNN, such as the number of hidden layers and neurons as well as the activation function. The simplest form of DNN, MLP, is employed as the basic structure of the neural network. No regularization is applied because the model itself is very simple. With Grid Search, one selects the combination of hyper-parameters that produces the lowest validation error. This leads to the choice of three hidden layers. In other words, the DNN consists of five fully connected layers (one input layer, three hidden layers, and one output layer). The input layer contains 322 neurons.[11] The first hidden layer contains 175 neurons, the second hidden layer contains 350 neurons, and the third hidden layer contains 150 neurons. Finally, the output layer has 2 output neurons,[12] which is the classification result of this research (whether or not the credit card holder is delinquent). The number of hidden layers and the number of neurons determine the complexity of the structure of the neural network. It is critical to build a neural network with an appropriate structure that fits the complexity of the data. While a small number of layers or neurons may cause underfitting, an extremely complex DNN would lead to overfitting (Radhakrishnan 2017).

It uses *Uniform Distribution Initialization* method to initialize the network weights to a small random number between 0 and 0.05 generated from a uniform distribution, then forward propagate the weight throughout the network. At each neuron, the weights and the input data are multiplied, aggregated, and transmitted through the activation function.

The model uses the ReLu activation function on the three hidden layers to solve the problem of exploding/vanishing

[10] An 80:20 ratio of data splitting is used as it is a common rule of thumb (Guller 2015; Giacomelli 2013; Nisbet et al. 2009; Kloo 2015).

[11] The original inputs have 41 attributes. After creating dummies for all classes of categorical attributes, it finally has 322 attributes.

[12] For a binary classification problem, it just needs a single output neuron using the logistic activation function: the output will be a number between 0 and 1, which can be interpreted as the estimated probability of the positive class. The estimated probability of the negative class is equal to one minus that number (Géron 2019). Here, a number 2 is used to indicate there are two classes.

Table 15.2 The structure of the DNN

Layer	Number of neurons	Type	Initial weight distribution/activation function
1	322	Input	Uniform
2	175	Hidden layer 1	ReLu
3	350	Hidden layer 2	ReLu
4	150	Hidden layer 3	ReLu
5	2	Output	Sigmoid

Table 15.3 The distributions of classes

	Training (over-balanced)	5 cross-validation sets	Test
Delinquency observations	563,744	5,260	1,277
Legitimate observations	563,766	563,786	141,074
Overall	1127,530	569,046	142,351

gradient which is introduced by Bengio, Simard, and Frasconi (1994) (Jin et al. 2016; Baydin et al. 2016). The *Sigmoid* activation function is applied to the output layer as it is a binary prediction. Table 15.2 depicts the neural network's structure.

The number of epochs in the DNN model is 10. The learning rate defines how quickly a network updates its parameters. Instead of using a constant learning rate to update the parameters (e.g., network weights) for each training epoch, it employs an adaptive learning rate, which allows the specification of different learning rates per layer (Brownlee 2016a; Lau 2017). Two parameters, Rho and Epsilon, need to be specified to implement the adaptive learning rate algorithm. Rho is similar to momentum and relates to the memory of prior weight updates. Typical values are between 0.9 and 0.999. This study uses the value 0.99. Epsilon is similar to learning rate annealing during initial training and momentum at later stages where it allows forward progress. It prevents the learning process from being trapped in local optima. Typical values are between 1e–10 and 1e–4. The value of epsilon is 1e–8 in this study. Batch size is the total number of training observations present in a single batch. The batch size used here is 32.

15.5.3 Techniques of Handling Data Imbalance

The entire dataset has imbalanced classes. The vast majority of the credit card holders do not have delinquency. A total of 6,537 instances are labeled with class "delinquent," while the remaining 704,860 are labeled with class "legitimate." To avoid the data imbalance, over-sampling and under-sampling are two popular resampling techniques. While over-sampling adds copies of instances from the under-represented class (which is the delinquency class in our case), under-sampling deletes instances from the over-represented class (which is the legitimate class in our case). It applies *Grid Search* again to try both approaches and find over-sampling works better for our data. Table 15.3 summaries the distributions of classes in training, 5 cross-validation, and test set.[13]

To compare the predictive performance of DNN to that of traditional neural network, logistic regression, Naïve Bayes, and decision tree, the same dataset, and data splitting and preprocessing method are used to develop prediction models. The results of cross-validation are reported in the next section.

15.6 Results

15.6.1 The Predictor Importance

This analysis evaluates the independent contribution of each predictor in explaining the variance of the target variable. Figure 15.3 lists the top 10 important indicators and their importance scores measured by the relative importance as compared to that of the most important variable.

The most powerful predictor is TRANS_ALL, the total amount of all authorized transactions on all credit cards held by the client in June, which indicates that the more the client spent, the riskier that the client will have severe delinquency issue later in September. The second important predictor is LOCATION, suggesting that clients living in some regions

[13] When splitting frames, H_2O does not give an exact split. It's designed to be efficient on big data using a probabilistic splitting method rather than an exact split. For example, when specifying a 0.75/0.25 split, H_2O will produce a test/train split with an expected value of 0.75/0.25 rather than exactly 0.75/0.25. On small datasets, the sizes of the resulting splits will deviate from the expected value more than on big data, where they will be very close to exact. http://h2o-release.s3.amazonaws.com/h2o/master/3552/docs-website/h2o-docs/datamunge/splitdatasets.html.

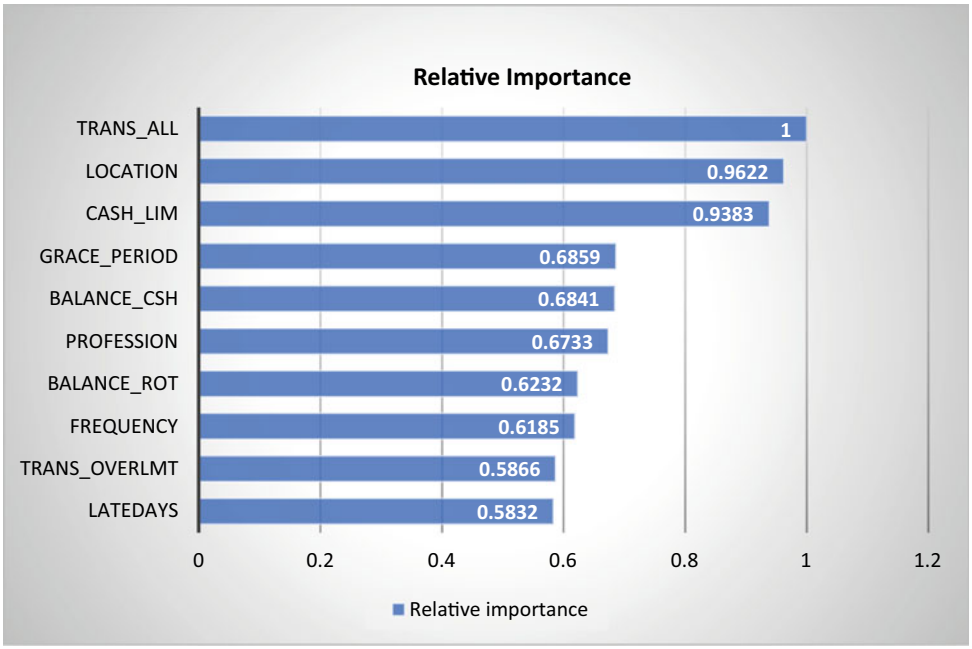


Fig. 15.3 The importance of top ten predictors

are more likely to default on credit card debt. Compared to TRANS_ALL, whose relative importance is 1 as it is the most important indicator, LOCATION's relative importance is 0.9622. It is followed by the limit of cash withdrawal (CASH_LIM) and the number of days given to the client to pay off the new balance without paying finance charges (GRACE_PERIOD). This result suggests that the flexibility the bank provides to the client facilitates the occurrence of delinquencies. Other important data fields include BALANCE_CSH (the current balance of cash withdrawal), PROFESSION (the occupation of the client), BALANCE_ROT (the current balance of credit card revolving payment), FREQUENCY (the number of times the client has been billed until September 2013), and TRANS_OVERLMT (the average amount of the authorized transactions exceeded the limit on all credit card accounts owned by the client). The last predictor is the average number of days the client's payments (on all credit cards) in June 2013 have passed the due dates.

15.6.2 The Predictive Result for Cross-Validation Sets

A list of metrics is applied to evaluate the predictive performance of the constructed DNN for cross-validation. The current analysis also uses a traditional neural network algorithm with a single hidden layer and a comparative number of neurons to build a similar prediction model. Logistic regression, Naïve Bayes, and decision tree techniques are also employed to conduct the same task. Next, it uses those metrics to compare the prediction result of the DNN and other models.

As shown in Table 15.4, the DNN has an overall accuracy of 99.54%, slightly lower than the traditional neural network and decision tree, but higher than the other two approaches. Since there is a large class imbalance in the validation data, the classification accuracy alone cannot provide useful information for model selection as it is possible that a model can predict the value of the majority class for all predictions and achieve a high classification accuracy. Therefore, I consider a set of additional metrics.

Specificity (also called True Negative Rate (TNR)) measures the proportion of negatives that are correctly identified as such. In this case it is the percentage of legitimate holders who are correctly identified as non-delinquent. The TNR of DNN is 0.9990, which is the second highest score of all algorithms. This result shows that the DNN classifier performs excellently in correctly identifying legitimate clients. Decision tree has a slightly higher specificity, which is 0.9999. Traditional neural network and logistic regression also have a high score of specificity. However, Naïve Bayes has a low TNR, which is 0.5913. This means that many legitimate observations are mistakenly identified by the Naïve Bayes model as delinquent ones. False negative rate (FNR) is the Type II error rate. It is the proportion of positives that are incorrectly identified as negatives. A FNR of 0.3958 of DNN indicates 39.58% of delinquent clients are undetected by the classifier. This is the second lowest score. The lowest one is 0.1226 generated by Naïve Bayes. So far, it seems like that the Naïve Bayes model tends to consider all observations as default ones because of the low level of

Table 15.4 Predictive performance[14]

Metrics	DNN	Traditional NN	Decision tree (J48)	Naïve Bayes	Logistic regression
Overall accuracy	0.9954	0.9955	0.9956	0.5940	0.9938
recall	0.6042	0.5975	0.5268	0.8774	0.4773
precision	0.8502	0.8739	0.9922	0.0196	0.7633
Specificity	0.9990	0.9980	0.9999	0.5913	0.9986
F_1	0.7064	0.6585	0.6882	0.0383	0.5874
F_2	0.6413	0.6204	0.5813	0.0898	0.5166
$F_{0.5}$	0.7862	0.7016	0.8432	0.0243	0.6816
FNR	*0.3958*	*0.4027*	*0.4732*	*0.1226*	*0.5227*
FPR	*0.0010*	*0.0020*	*0.0001*	*0.4087*	*0.0014*
AUC	*0.9547*	*0.9485*	*0.881*	*0.7394*	*0.8889*
Model building time	8 h 3 min 13 s	13 min 56 s	0.88 s	9 s	34 s

TNR and FNR. False positive rate (FPR) is called Type I error rate. It is the proportion of negatives that are incorrectly classified as positives. The table shows that the Type I error rate of decision tree is 0.01%, higher than that of DNN, which is 0.1%. This result suggests that it is unlikely that a normal client will be identified by Decision Tree and DNN as a problematic one.

Precision and recall are two important measures for the ability of the classifier for delinquency detection, where precision[15] measures the percentage of actual delinquencies in all perceived ones. The precision score, 0.8502, of DNN is lower than that of decision tree and traditional neural network, which is 0.9922 and 0.8739, respectively, but higher than that of the other two algorithms. Specifically, Naïve Bayes model receives an extremely low score, 0.0196. This number shows that approximately all perceived delinquencies are actually legitimate observations. Recall,[16] on the other hand, indicates that, for all actual delinquencies, how many of them are successfully identified by the classifier. It is also called Sensitivity or the True Positive Rate (TPR), which can be thought of as a measure of a classifier's completeness. The Recall score of DNN is 0.6042, the highest score of all models except Naïve Bayes. This number also means 39.58% of delinquent observations are not identified by our model, which is consistent with the result of FNR.

While the decision tree and traditional neural network models perform better than the DNN in terms of precision, the DNN outperforms them in terms of recall. Thus, it is necessary to evaluate the performance of models by considering both precision and recall. Three F scores, F_1, F_2, and $F_{0.5}$, , are frequently used by existing data mining research to conduct this job (Powers 2011). The F_1 score[17] is the harmonic mean of precision and recall, treating precision and recall equally. While F_2[18] treats recall with more importance than precision by weighting recall higher than precision, $F_{0.5}$[19] weighs recall lower than precision. The F_1, F_2, and $F_{0.5}$ score of the DNN is 0.7064, 0.6413, and 0.7862, respectively. The result shows that, with the exception of $F_{0.5}$, DNN exhibit the highest overall performance than other models.

The overall capability of the classifier can also be measured by the Area Under the Receiver Operating Characteristic (ROC) curve, AUC. The ROC curve (see Fig. 15.4) plots the recall versus the false positive rate as the discriminative threshold is varied between 0 and 1. Again, the DNN provides the highest AUC of 0.9547 compared to other models, showing its strong ability to discern between the two classes. Finally, the model building time shows that it is a time-consuming procedure (more than 8 h) to develop a DNN due to the complexity of computing.

15.6.3 Prediction on Test Set

The results of cross-validation show the performance of the model with optimal hyper-parameters. The actual predictive capability of the model is measured by the out-of-sample test on the test set. Table 15.5 is the confusion matrix for the test set. 85 legitimate credit card holders are classified as

[14] We choose the threshold that gives us the highest F1 score, and the reported value of the metric is based on the selected threshold.
[15] Precision = true positive/(true positive + false positive).
[16] Recall = true positive/(true positive + false negative).
[17] $F_1 = 2 \times (\text{precision} \times \text{recall})/(\text{precision} + \text{recall})$.
[18] $F_2 = 5 \times (\text{precision} \times \text{recall})/(4 \times \text{precision} + \text{recall})$.
[19] $F_{0.5} = \frac{5}{4} \times (\text{precision} \times \text{recall})/(\frac{1}{4} \times \text{precision} + \text{recall})$.

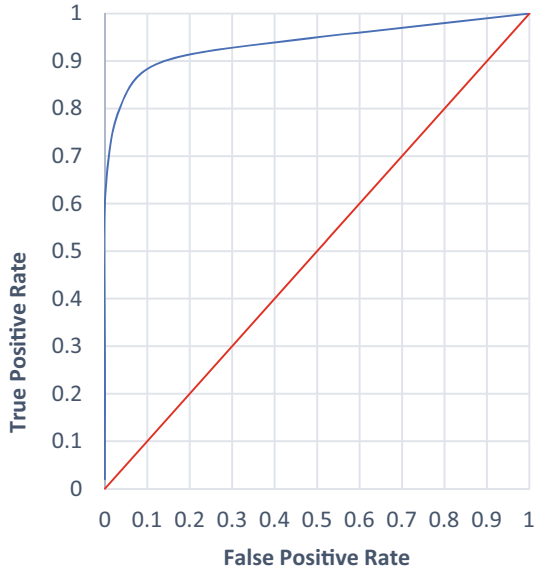


Fig. 15.4 The ROC curve—cross-validation metrics

Table 15.5 The confusion matrix of DNN (test set)

Actual/predicted	Legitimate obs	Delinquent obs	Total
Legitimate obs	140,989	85	141,074
Delinquent obs	504	773	1277
Total	141,493	858	142,351

delinquent ones by the DNN. In addition, 773 out of 1277 delinquent clients are successfully detected.

The result of out-of-sample test in Table 15.6 and the ROC curve in Fig. 15.5 both show that the DNN model generally performs effectively in detecting delinquencies, as reflected by the highest AUC value, 0.9246. The recall is 0.6053, which is the second highest value. The highest value of recall is 0.8677 for the Naïve Bayes model. The precision of the DNN is also the second highest, which is 0.9009. Considering both precision and recall, the DNN outperforms other models with the highest F_1 score, 0.7241. This result is consistent with the result for all models on the cross-validation sets. Specifically, the F_1 score for test set is higher than that for the cross-validation set. The remaining metrics support that, compared to others, the DNN performs more effectively in identifying credit card delinquency.

15.7 Conclusion

This chapter introduces deep learning and its application to credit card delinquency forecasting. It describes the process of DNN training and validation, hyper-parameters tuning, and how to handle data overfitting and imbalance issues, etc. Using real-life data from a large bank in Brazil, a DNN is built to predict severe credit card delinquencies based on the

Table 15.6 The result of out-of-sample test

Metrics	DNN	Traditional NN	Naïve Bayes	Logistic	Decision tree (J48)
Overall accuracy	0.9959	0.9941	0.6428	0.9949	0.9944
Recall	0.6053	0.5521	0.8677	0.5770	0.4527
Precision	0.9009	0.7291	0.0217	0.8047	0.9080
Specificity	0.9994	0.9981	0.6407	0.9987	0.9996
F_1	0.7241	0.6283	0.0424	0.6721	0.6042
F_2	0.6478	0.5802	0.0987	0.6116	0.5032
$F_{0.5}$	0.8208	0.6851	0.0270	0.7459	0.7559
False negative Rate	0.3947	0.4479	0.1323	0.4230	0.5473
False positive Rate	0.0006	0.0019	0.3593	0.0013	0.0004
AUC	0.9246	0.9202	0.7581	0.8850	0.8630

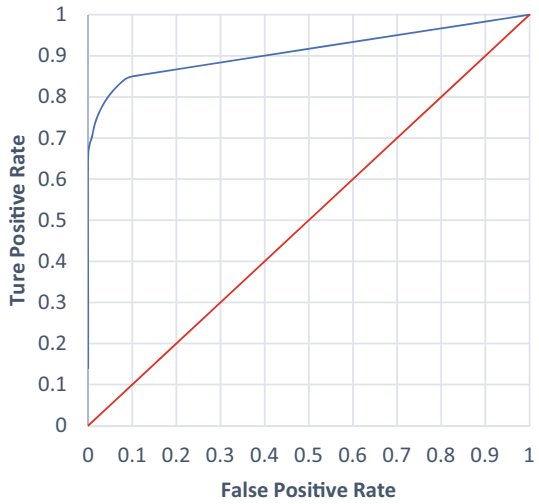


Fig. 15.5 The ROC curve-testing metrics

clients' basic demographic information and records of historical transactions. Compared to a traditional neural network, logistic regression, Naïve Bayes, and decision tree models, deep learning is superior in terms of predictive accuracy as shown by the results of the out-of-sample test.

Appendix 15.1: Variable Definition

Target variable	Description[20]
INDICATOR	It indicates if any of the client's credit card is permanently blocked in September 2013 due to credit card delinquency
Input variables	Description
1. Personal characteristics	
SEX	The gender of the credit card holder
Individual	The code indicating if the holder is an individual or a corporation
AGE	The age of the credit card holder
INCOME_CL	The annual income claimed by the holder
INCOME_CF	The annual income of the holder confirmed by the bank
ADD_ASSET	The number of additional assets owned by the holder
LOCATION	The code indicating the holder's region of residence
PROFESSION	The code indicating the occupation of the holder
ACCOUNT_AGE	The oldest age of the credit card accounts owned by the client (in months)
CREDIT_SCORE	The credit score of the holder
SHOPPING_CRD	The number of products in shopping cards
VIP	The VIP code of the holder
CALL	It equals 1 if the client requested an increase of the credit limit; 0 otherwise
PRODUCT	The number of products purchased
CARDS	The number of credit cards held by the client (issued by the same bank)
2. Information about accumulative transactional activities (as of September 2013)	
FREQUENCY	The frequency that the client has been billed
PAYMENT_ACC	The frequency of the payments made by the client
WITHDRAWAL	The accumulated amount of cash withdrawals (domestic)
BEHAVIOR	The behavior code of the client is determined by the bank
BEHAVIOR_SIMPLE	The simplified behavior score provided by the bank
CREDIT_LMT_PRVS	The maximum credit limit in the last period
3. Transactions in June 2013	
CREDIT_LMT_CRT	The maximum credit limit
LATEDAYS	The average number of days that the client's credit card payments have passed the due date
UNPAID_DAYS	the average number of days that previous transactions have remained unpaid
BALANCE_ROT	The current balance of credit card revolving payment
BALANCE_CSH	The current balance of cash withdrawal
GRACE_PERIOD	The remaining number of days that the bank gives the credit card holder to pay off the new balance without paying finance charges. The time window starts from the end of June 2013 to the next payment due date

[20] The unit of the amount is Brazilian Real.

Input variables	Description
3. Transactions in June 2013	
INSTALL_LIM_ACT	The available installment limits. It equals the installment limit plus the installment paid[21]
CASH_LIM	The limit of cash withdrawal
INSTALL_LIM	The limit of installment
ROT_LIM	The revolve limit of credit card payment
DAILY_TRANS	The maximum number of authorized daily transactions
TRANS_ALL	The amount of all authorized transactions (including all credit card revolving payment, installment, and cash withdrawal) on all credit card accounts owned by the client
TRANS_OVERLMT	The average amount of the authorized transactions exceeded the limit on all credit card accounts owned by the client
BALANCE_ALL	The average balance for authorized unpaid transactions (including all revolving credit card payment, installment, and cash withdrawal) on all credit card accounts owned by the client
BALANCE_PROCESSING	The average balance of all credit card transactions under the authorization process
ROT_PAID	The total amount of credit card revolving payment that has been made
CASH_OVERLMT_PCT	The average percentage of cash withdrawal exceeded the limit on all credit card accounts owned by the client
PAYMENT_PROCESSING	The average payment under processing
INSTALLMENT_PAID	The total installment amount that has been paid
INSTALLMENT	The total number of installments, including the paid installments and the unpaid ones
ROT_OVERLMT	The average amount of credit card revolving payment exceeded the revolve limit
INSTALLMENT_OVERLMT_PCT	The average percentage of the installment exceeded the limit

(continued)

[21] The actual amount of installment limit could exceed the installment limit provided by the bank for the customer. This happens when the customer made some payments, so those funds become available for borrowing again.

References

Abdulkader, A., Lakshmiratan, A., & Zhang, J. (2016). Introducing DeepText: Facebook's text understanding engine. https://backchannel.com/an-exclusive-look-at-how-ai-and-machine-learning-work-at-apple-8dbfb131932b

Albanesi, S., & Vamossy, D. F. (2019). Predicting consumer default: A deep learning approach (No. w26165). National Bureau of Economic Research

Baydin, A.G., Pearlmutter, B.A. & Siskind, J.M. (2016). Tricks from Deep Learning. *arXiv preprint* arXiv:1611.03777

Bengio, Y., Simard, P. & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5,157-166.

Bengio, Y. (2012a). Deep learning of representations for unsupervised and transfer learning. *In Proceedings of ICML Workshop on Unsupervised and Transfer Learning,* June, 17–36

Bengio, Y. (2012b). Practical recommendations for gradient-based training of deep architectures. arXiv:1206.5533v2

Brownlee, J. (2016a). Using Learning Rate Schedules for Deep Learning Models in Python with Keras. Machine Learning Mastery. https://machinelearningmastery.com/using-learning-rate-schedules-deep-learning-models-python-keras/

Brownlee, J. (2016b). Gradient Descent for Machine Learning. Machine Learning Mastery. https://machinelearningmastery.com/gradient-descent-for-machine-learning/

Brownlee, J. (2018). A gentle introduction to K-fold cross-validation. https://machinelearningmastery.com/k-fold-cross-validation/

Candel, A., Parmar, V., LeDell, E., & Arora, A. (2020). Deep Learning with H_2O. *Working paper*. http://h2o-release.s3.amazonaws.com/h2o/master/5288/docs-website/h2o-docs/booklets/DeepLearningBooklet.pdf

Ding, K., Lev, B., Peng, X., Sun, T., & Vasarhelyi, M. A. (2020). Machine learning improves accounting estimates: evidence from insurance payments. Review of Accounting Studies, 1–37

Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media

Giacomelli, P., (2013). Apache mahout cookbook. *Packt Publishing Ltd*

Goodfellow.I., Bengio. Y., & Courville, A. (2016). Deep Learning. *MIT Press*. http://www.deeplearningbook.org

Guller, M. (2015). Big Data Analytics with Spark: A Practitioner's Guide to Using Spark for Large Scale Data Analysis. Apress, 155

Gupta, D. (2017). Fundamentals of Deep Learning – Activation Functions and When to Use Them? Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/10/fundamentals-deep-learning-activation-functions-when-to-use-them/

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. Neural computation, 18(7), 1527–1554.

H2O.ai. (2018). Cross-Validation. H_2O Documents. http://docs.h2o.ai/h2o/latest-stable/h2o-docs/cross-validation.html

Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism*, 1–5

Hamori, S., Kawai, M., Kume, T., Murakami, Y., & Watanabe, C. (2018). Ensemble learning or deep learning? Application to default risk analysis. Journal of Risk and Financial Management, 11(1), 12.

Heaton, J.B., Polson, N.G. & Witte, J.H. (2016). Deep learning in finance. arXiv preprint arXiv:1602.06561

Issa, E. (2019). Nerdwallet's 2019 American Household Credit Card Debt Study. https://www.nerdwallet.com/blog/average-credit-card-debt-household/

Jain, S. (2018). An Overview of Regularization Techniques in Deep Learning (with Python code). https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/

Jin, X., Xu, C., Feng, J., Wei, Y., Xiong, J. & Yan, S. (2016). Deep Learning with S-Shaped Rectified Linear Activation Units. *In AAAI*, 2, 1737-1743.

Kloo, I. (2015). Textmining: Clustering, Topic Modeling, and Classification. http://data-analytics.net/cep/Schedule_files/Textmining%20%20Clustering,%20Topic%20Modeling,%20and%20Classification.htm

Koh, H. C., & Chan, K. L. G. (2002). Data mining and customer relationship marketing in the banking industry. *Singapore Management Review*, 24, 1–27.

Kumar, N. (2019). Deep Learning Best Practices: Regularization Techniques for Better Neural Network Performance. https://heartbeat.fritz.ai/deep-learning-best-practices-regularization-techniques-for-better-performance-of-neural-network-94f978a4e518

Lau, S. (2017). Learning Rate Schedules and Adaptive Learning Rate Methods for Deep Learning. Towards Data Science. https://towardsdatascience.com/learning-rate-schedules-and-adaptive-learning-rate-methods-for-deep-learning-2c8f433990d1

Levy, S. (Aug 24, 2016). An exclusive inside look at how artificial intelligence and machine learning work at Apple. Backchannel. https://backchannel.com/an-exclusive-look-at-how-ai-and-machine-learning-work-at-apple-8dbfb131932b

Malik, F. (2019). Neural networks bias and weights. https://medium.com/fintechexplained/neural-networks-bias-and-weights-10b53e6285da

Marcus, G. (2018). Deep learning: a critical appraisal. https://arxiv.org/abs/1801.00631

Marqués, A.I., García, V. & Sánchez, J.S. (2012). Exploring the behavior of base classifiers in credit scoring ensembles. *Expert Systems with Applications*, *39*, 10244-10250.

Mohamed, Z. (2019). Using the artificial neural networks for prediction and validating solar radiation. Journal of the Egyptian Mathematical Society. 27(47). https://doi.org/10.1186/s42787-019-0043-8

Nisbet, R., Elder, J. & Miner, G. (2009). Handbook of statistical analysis and data mining applications. *Academic Press.*

Ohlsson, C. (2017). Exploring the potential of machine learning: How machine learning can support financial risk management. Master's Thesis. Uppsala University

Powers, D.M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation

Radhakrishnan, P. (2017). What are Hyperparameters and How to tune the Hyperparameters in a Deep Neural Network? *Towards Data Science*. https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536

Silver, D., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529, 484-489

Sun, T. & Vasarheyi, M.A. (2017). Deep learning and the future of auditing: how an evolving technology could transform analysis and improve judgment. *The CPA Journal*. 6, 24-29

Sun, T., & Vasarhelyi, M. A. (2018). Predicting credit card delinquencies: An application of deep neural networks. Intelligent Systems in Accounting, Finance and Management, 25(4), 174-189

Shaikh, F. (2017). Deep learning vs. machine learning-the essential differences you need to know. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/04/comparison-between-deep-learning-machine-learning/

Sharma, S. (2017). Epoch vs Batch Size vs Iterations. *Towards Data Science*. https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9

Szegedy, C. (2014). Building a deeper understanding of images. Google Research Blog (September 5, 2014). https://research.googleblog.com/2014/09/building-deeper-understanding-of-images.html

Tartakovsky, S., Clark, S., & McCourt, M (2017) Deep Learning Hyperparameter Optimization with Competing Objectives. NVIDIA Developer Blog. https://devblogs.nvidia.com/parallelforall/sigopt-deep-learning-hyperparameter-optimization/

Teng, H. W., & Lee, M. (2019). Estimation procedures of using five alternative machine learning methods for predicting credit card default. Review of Pacific Basin Financial Markets and Policies, 22 (03), 1950021

Thomas, L. C. (2000). A survey of credit and behavioral scoring: Forecasting financial risk of lending to consumers. *International Journal of Forecasting*, 16, 149–172

Zhang, B. Y., Li, S. W., & Yin, C. T. (2017). A Classification Approach of Neural Networks for Credit Card Default Detection. *DEStech Transactions on Computer Science and Engineering*, (AMEIT 2017). DOI https://doi.org/10.12783/dtcse/ameit2017/12303

Binomial/Trinomial Tree Option Pricing Using Python

16

16.1 Introduction

The Binomial Tree Option Pricing model is one the most famous models used to price options. The binomial tree pricing process produces more accurate results when the option period is broken up into many binomial periods. One problem with learning the Binomial Tree Option pricing model is that it is computationally intensive as the number of periods of a Binomial Tree is large. A ten period Binomial Tree would require 2047 calculations for both call and put options. As a result, most books do not present Binomial Trees with more than three periods.

To solve the computationally intensive problem of a binomial option pricing model, we will use Python programming. This chapter will do its best to present the Binomial Tree Option model in a less mathematical matter. In Sect. 16.2, Binomial Tree model to price European call and put options are given. Some basic finance concepts will also be included. In Sect. 16.3, Binomial Tree model to price American options is given. In addition to Binomial Tree Option model, trinomial tree option pricing model is also given in Sect. 16.4. Section 16.5 concludes.

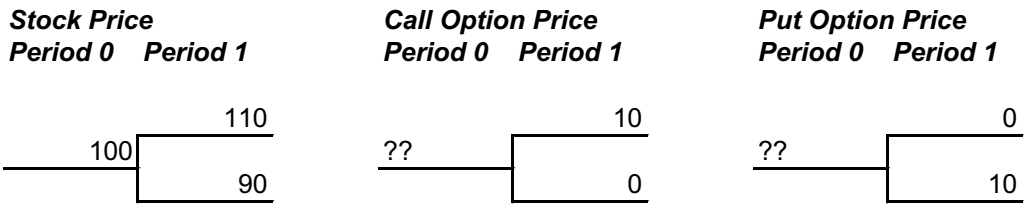
16.2 European Option Pricing Using Binomial Tree Model

A European option is a contract that limits execution to its expiration date. In other words, if the underlying security such as a stock has moved in price, an investor would not be able to exercise the option early and take delivery of or sell the shares. Instead, the call or put action will only take place on the date of option maturity. In a competitive market, to avoid arbitrage opportunities, assets with identical payoff structures must have the same price. Valuation of options has been a challenging task and pricing variations lead to arbitrage opportunities. Black–Scholes remains one of the most popular models used for pricing options but has limitations. The binomial tree option pricing model is another popular method used for pricing options.

In the following, we consider the value of a European option for one period using the binomial tree option pricing model. A stock price can either go up or go down. Let's look at a case where we know for certain that a stock with a price of $100 will either go up 10% or go down 10% in the next period and the exercise after one period is $100. Below shows the decision tree for the stock price, the call option price, and the put option price.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_16



Let's first consider the issue of pricing a call option. Using a one period Binomial Tree, we can illustrate the price of a stock if it goes up and the price of a stock if it goes down. Since we know the possible endings values of a stock, we can derive the possible ending values of a call option. If the stock price increases to \$110, the price of the call option will then be \$10 (\$110 − \$100). If the stock price decreases to \$90, the value of the call option will be worth \$0 because it would be below the exercise price of \$100. We have just discussed the possible ending value of a call option in period 1. But, what we are really interested is the value now of the call option knowing the two resulting value of a call option.

To help determine the value of a one period call option, it's useful to know that it is possible to replicate the resulting two states of the value of the call option by buying a combination of stocks and bonds. Below is the formula to replicate the situation where the price increases to \$110. We will assume that the interest rate for the bond is 7%.

$$110S + 1.07B = 10$$
$$90S + 1.07B = 0$$

We can use simple algebra to solve for both S and B. The first thing that we need to do is to rearrange the second equation as follows:

$$1.07B = -90S$$

With the above equation, we can rewrite the first equation as

$$110S + (-90S) = 10$$
$$20S = 10$$
$$S = .5$$

We can solve for B by substituting the value .5 for S in the first equation.

$$110(.5) + 1.07B = 10$$
$$55 + 1.07B = 10$$
$$1.07B = -45$$
$$B = -42.05607$$

Therefore, from the above simple algebraic exercise, we should at period 0 buy .5 shares of IBM stock and borrow 42.05607 at 7% to replicate the payoff of the call option. This means the value of a call option should be .5 × 100 − 42.05607 = 7.94393. If this were not the case, there would then be arbitrage profits. For example, if the call option were sold for \$8 there would be a profit of .056607. This would result in an increase in the selling of the call option. The increase in the supply of call options would push the price down for the call options. If the call options were sold for \$7, there would be a saving of .94393. This saving would result in an increased demand for the call option. The equilibrium point would be 7.94393.

Using the above mentioned concept and procedure, Benninga (2000) has derived a one period call option model as

$$C = q_u \times \text{Max}[S(1+u) - X, 0] + q_d \times \text{Max}[S(1+d) - X, 0] \tag{16.1}$$

where

$$q_u = \frac{i-d}{(1+i)(u-d)}$$

$$q_d = \frac{u-i}{(1+i)(u-d)}$$

u = increase factor
d = down factor
i = interest rate

Let $i = r$, and $p = (r - d)/(u - d)$, $1 - p = (u - r)/(u - d)$, $R = 1/(1 + r)$. Then

$$C_u = \text{Max}[S(1+u) - X, 0]$$

$$C_d = \text{Max}[S(1+d) - X, 0]$$

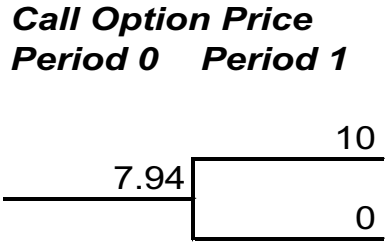
where C_u = call option price after up and C_d = call option price after down. Then, the value of the call option is

$$C = [pC_u + (1-p)C_d]/R \tag{16.2}$$

Below calculates the value of the above one period call option where the strike price, X, is \$100 and the risk-free interest rate is 7%. We will assume that the price of a stock for any given period will either increase or decrease by 10%.

$$\begin{aligned} X &= \$100 \\ S &= \$100 \\ u &= 1.10 \\ d &= .9 \\ R &= 1 + r = 1 + .07 \\ p &= (1.07 - .90)/(1.10 - .90) \\ C &= [.85(10) + .15(0)]/1.07 = \$7.94 \end{aligned}$$

Therefore, from the above calculations, the value of the call option is \$7.94. From the above calculations, the call option pricing binomial tree should look like the following:



For a put option, as the stock price decreases to \$90, one has

$$\begin{aligned} 110S + 1.07B &= 0 \\ 90S + 1.07B &= 10 \end{aligned}$$

S and B will be solved as

$$\begin{aligned} S &= -.5 \\ B &= 51.04 \end{aligned}$$

This tells us that we should in period 0 lend \$51.04 at 7% and sell .5 shares of stock to replicate the put option payoff for period 1. And, the value of the put option should be 100* (−.5) + 51.40 = −50 + 51.40 = 1.40. Using the same arbitrage argument that we used in the discussion of the call option, 1.40 has to be the equilibrium price of the put option. As with the call option, Benninga (2000) has derived a one period put option model as

$$P = q_u \text{Max}[X - S(1+u),\ 0] + q_d \text{Max}[X - S(1+d),\ 0] \tag{16.3}$$

where

$$q_u = \frac{i-d}{(1+i)(u-d)}$$

$$q_d = \frac{u-i}{(1+i)(u-d)}$$

u = increase factor

d = down factor

i = interest rate

Let $i = r$, $p = (r - d)/(u - d)$, $1 - p = (u - r)/(u - d)$, $R = 1/(1 + r)$. Then the put option price after increase and decrease are, respectively

$$P_u = \text{Max}[X - S(1+u),\ 0]$$

$$P_d = \text{Max}[X - S(1+d),\ 0]$$

then we have

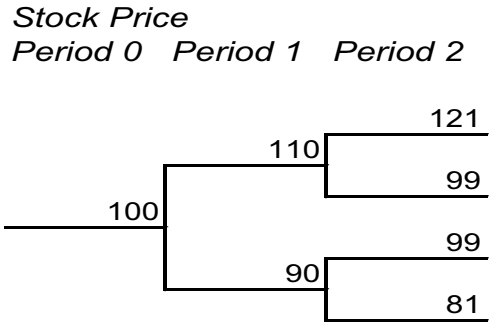
$$P = [pP_u + (1-p)P_d]/R \tag{16.4}$$

As an example, suppose the strike price, X, is \$100 and the risk-free interest rate is 7%. Then

$$P = [.85(0) + .15(10)]/1.07 = \$1.40$$

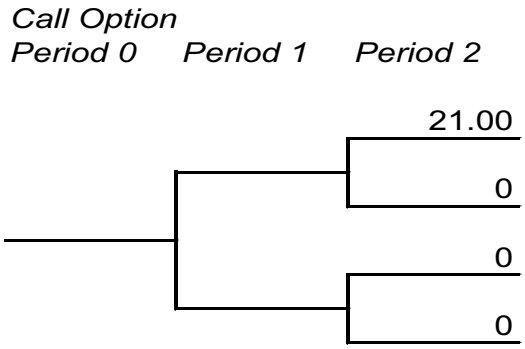
16.2.1 European Option Pricing—Two Period

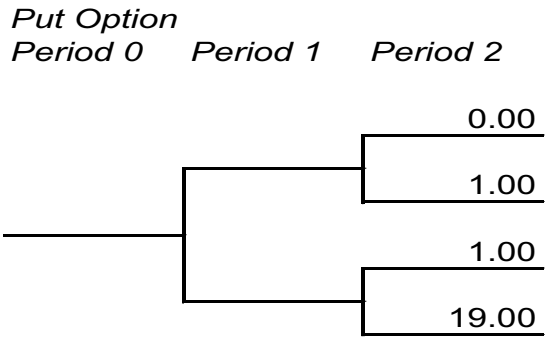
We will now look at pricing options for two periods. Below shows the stock price Binomial tree based on the parameters indicated in the last section.



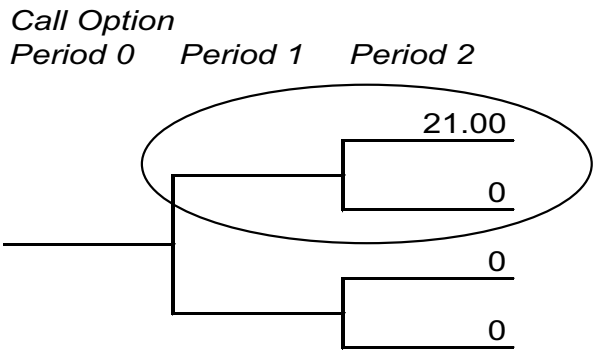
We can assume a stock price will either increase by 10% or decrease by 10%. The highest possible value for our stock based on our assumption is \$121. The lowest possible value for our stock based on our assumptions is \$81. In period two, the value of a call option when a stock price is \$121 is the stock price minus the exercise price, \$121 − 100, or \$21 dollars. In period two, the value of a put option when a stock price \$121 is the exercise price minus the stock price, \$100 − \$121, or −\$21. A negative value has no value to an investor so the value of the put option would be \$0. In period two, the value of a call option when a stock price is \$81, is

the stock price minus the exercise price, \$81 − \$100, or − \$19. A negative value has no value to an investor so the value of a call option would be \$0. In period two, the value of a put option when a stock price is \$81 is the exercise price minus the stock price, \$100 − \$81, or \$19. We can derive the call and put option value for the other possible value of the stock in period 2 in the same fashion. The following shows the possible call and put option values for period 2.

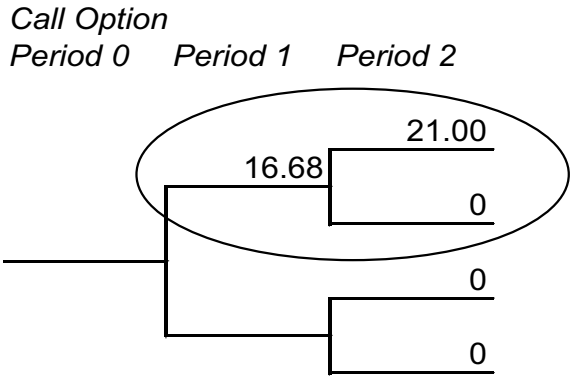




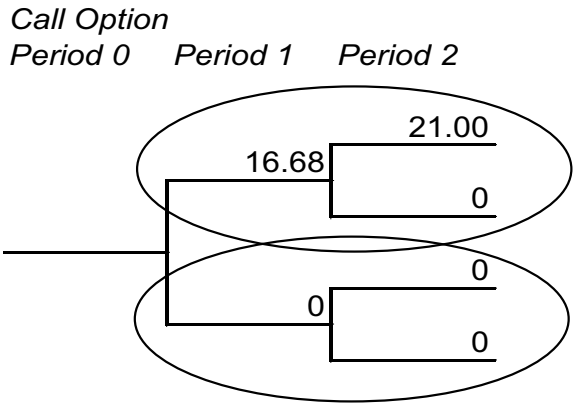
We cannot calculate the value of the call and put option in period 1 the same way as we did in period 2, because it's not the ending value of the stock. In period 1, there are two possible call values. One value is when the stock price increased and one value is when the stock price decreased. The call option Decision Tree shown above shows two possible values for a call option in period 1. If we just focus on the value of a call option when the stock price increases from period one, we will notice that it is like the Decision Tree for a call option for one period. This is shown below.



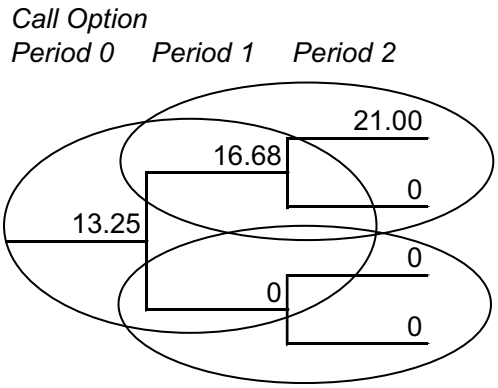
As the pricing of a call option for one period, the price of a call option when the stock price increases from period 0 will be \$16.68. The resulting Binomial Tree is shown below.



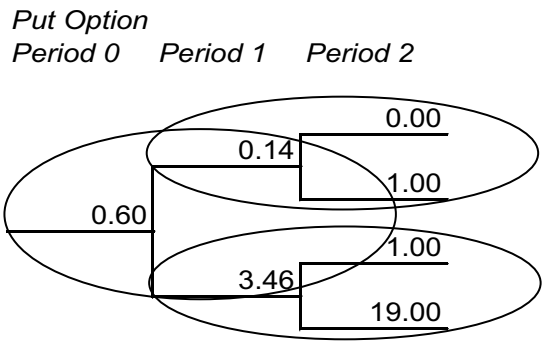
In the same fashion, we can price the value of a call option when a stock price decreases. The price of a call option when a stock price decreases from period 0 is \$0. The resulting Decision Tree is shown below.



In the same fashion, we can price the value of a call option in period 0. The resulting Binomial Tree is shown below.



We can calculate the value of a put option in the same manner as we did in calculating the value of a call option. The Binomial Tree for a put option is shown below.



16.2.2 European Option Pricing—*N* Periods

Benninga (2000, p 260) has derived the price of a call and a put option, respectively, by a Binomial Option Pricing model with n periods as

$$C = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[S(1+u)^i(1+d)^{n-i} - X, 0] \tag{16.5}$$

$$P = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[X - S(1+u)^i(1+d)^{n-i}, 0] \tag{16.6}$$

Chapter 5 has shown how Excel VBA can be used to estimate the binomial option pricing model. Appendix 16.1 has shown how the Python program can be used to estimate the binomial option pricing model. By using the python program in Appendix 16.1, Figs. 16.1, 16.2 and 16.3 illustrate the simulation results of binomial tree option pricing using initial stock price $S_0 = 100$, strike price $X = 100$, $n = 4$ periods, interest rate r = 0.07, the up factor $u = 1.175$, and down factor $d = 0.85$. Figure 16.1 illustrates the simulated stock prices, and Figs. 16.2 and 16.3 illustrate the corresponding European call and put prices, respectively. As

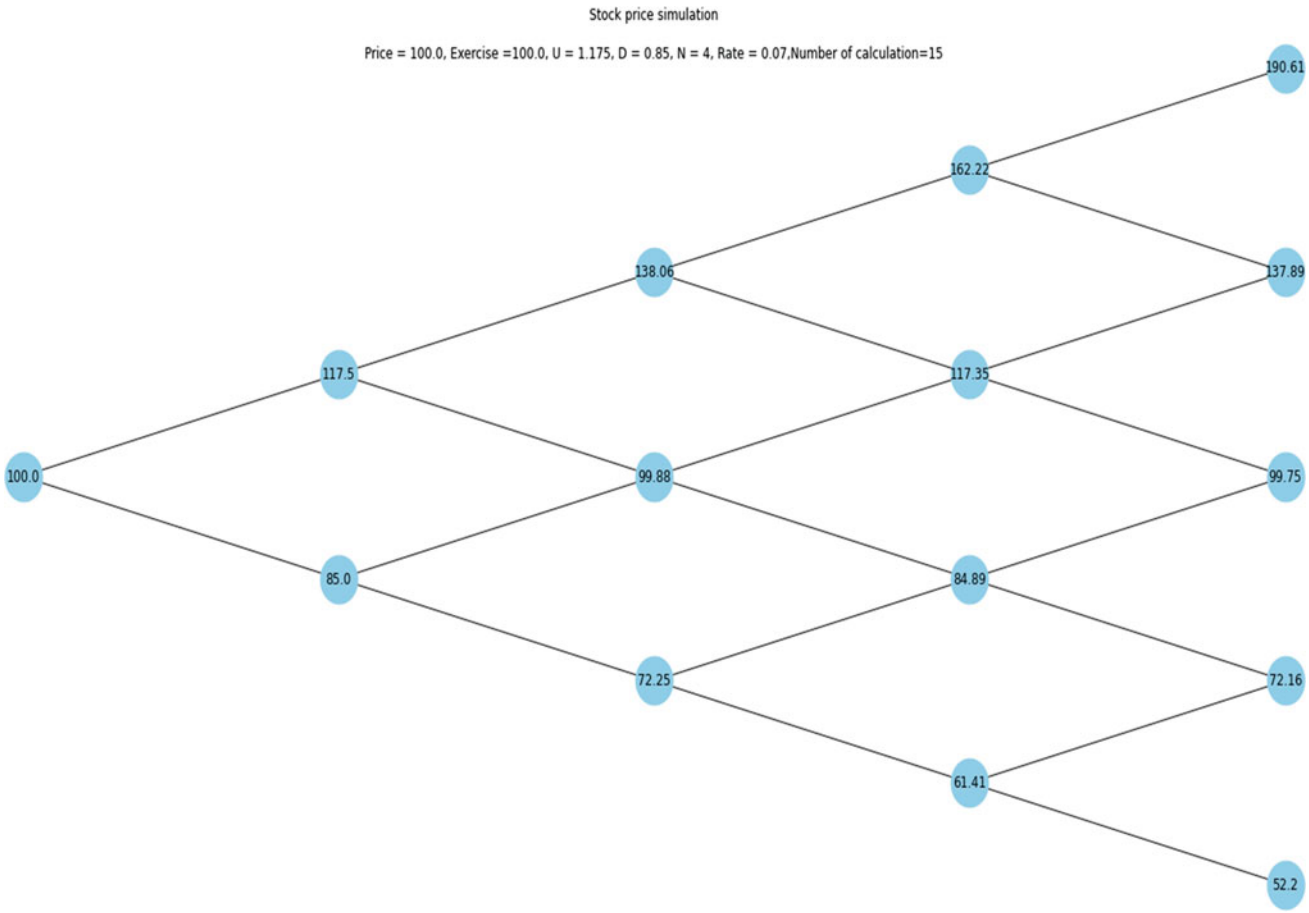


Fig. 16.1 Stock price simulation

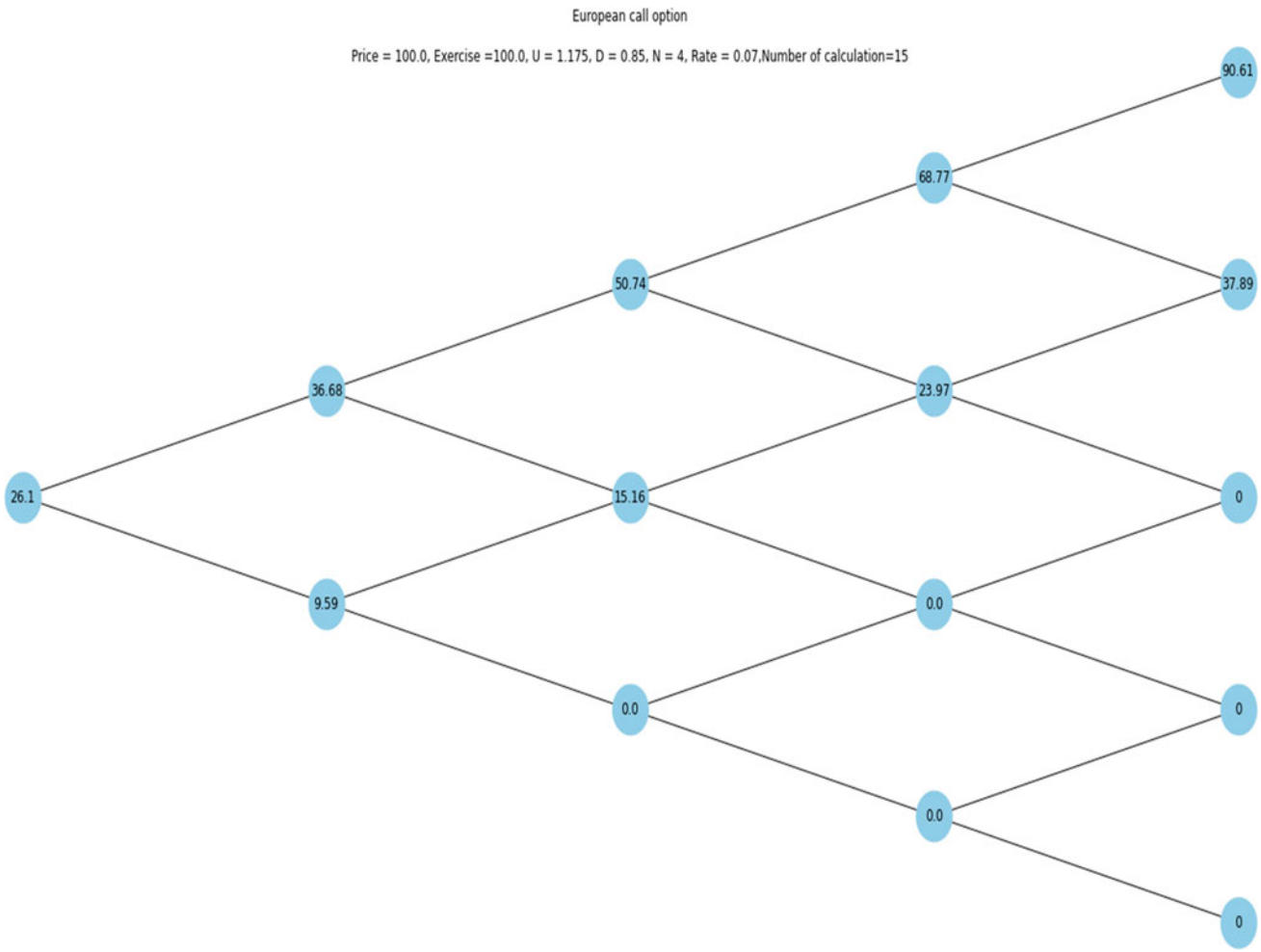


Fig. 16.2 European call option prices by binomial tree

can be seen, for example, as the stock price at the 4th period $S = 190.61$, the European call and put prices are 90.61 and 0, respectively. As the stock price at the 4th period $S = 52.2$, the European call and put prices are 0 and 47.8, respectively.

16.3 American Option Pricing Using Binomial Tree Model

An American option is an option the holder may exercise at any time between the start date and the maturity date. Therefore, the holder of an American option faces the dilemma of deciding when to exercise. Binomial tree valuation can be adapted to include the possibility of exercise at intermediate dates and not just the maturity date. This feature needs to be incorporated into the pricing of American options. The binomial option pricing model presents two advantages for option sellers over the Black–Scholes model. The first is its simplicity, which allows for fewer errors in commercial application. The second is its iterative operation, which adjusts prices in a timely manner so as to reduce the opportunity for buyers to execute arbitrage strategies. For example, since it provides a stream of valuations for a derivative for each node in a span of time, it is useful for valuing derivatives such as American options—which can be executed anytime between the purchase date and expiration date. It is also much simpler than other pricing models such as the Black–Scholes model.

The first step of pricing an American option is the same as a European option. For an American option, the second step relates to the difference between the strike price of the option and the price of the stock. A simplified example is

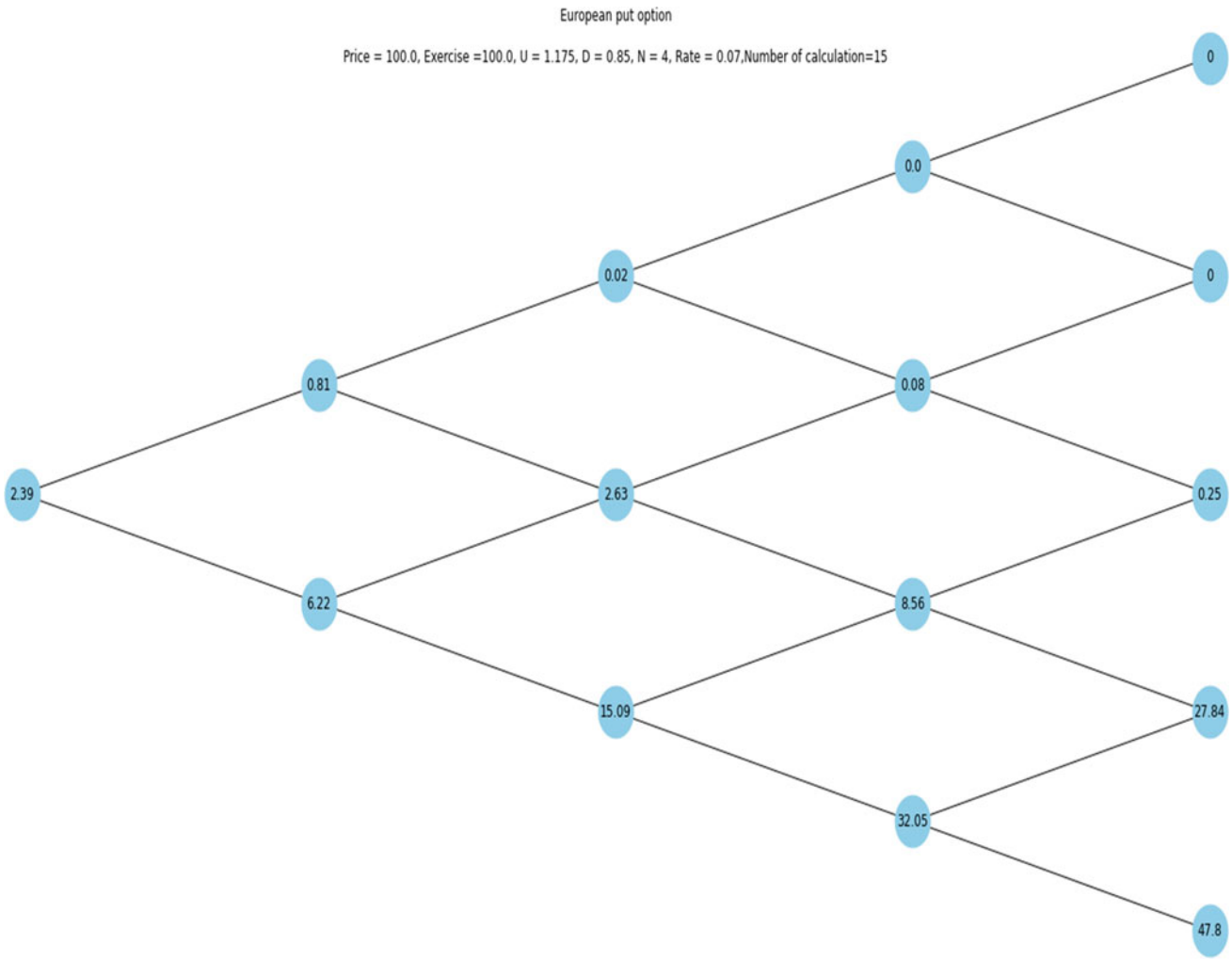


Fig. 16.3 European put option prices by binomial tree

given as follows. Assume there is a stock that is priced at $S = \$100$ per share. In one month, the price of this stock will go up by \$10 or go down by \$10, creating this situation

$$S = \$100$$
$$\text{Stock price in one month (up state)} = \$110$$
$$\text{Stock price in one month (down state)} = \$90$$

Suppose there is a call option available on this stock that expires in one month and has a strike price of \$100. In the up state, this call option is worth \$10, and in the down state, it is worth \$0. Assume an investor purchases one-half share of stock and writes or sells one call option. The total investment today is the price of half a share less the price of the option, and the possible payoffs at the end of the month are

$$\text{Cost today} = \$50 - \text{option price}$$
$$\text{Portfolio value (up state)} = \$55 - max(\$110 - \$100,\ 0) = \$45$$
$$\text{Portfolio value (down state)} = \$45 - max(\$90 - \$100,\ 0) = \$45$$

The portfolio payoff is equal no matter how the stock price moves. Given this outcome, assuming no arbitrage opportunities, an investor should earn the risk-free rate over the course of the month. The cost today must be equal to the

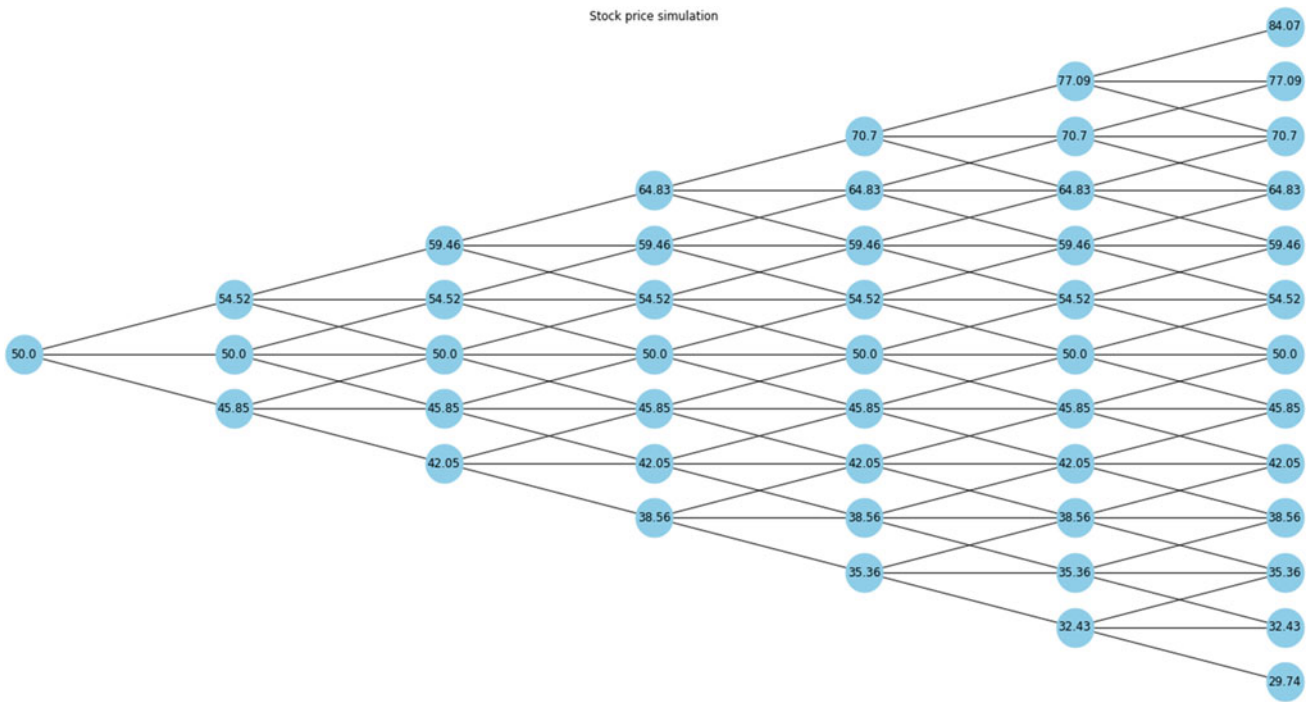


Fig. 16.4 Stock price simulation by trinomial tree

payoff discounted at the risk-free rate for one month. The equation to solve is thus

$$\text{Option price} = \$50 - \$45 \times e^{-rT},$$

where e is the mathematical constant 2.7183

Assuming the risk-free rate is 3% per year, and T equals 0.0833 (one divided by 12), then the price of the call option today is $5.11.

16.4 Alternative Tree Models

In this section, we will introduce three binomial tree methods and one trinomial tree method to price option values. Three binomial tree methods include Cox et al. (1979), Jarrow and Rudd (1983), and Leisen and Reimer (1996). These methods will generate different kinds of underlying asset trees to represent different trends of asset movement. Kamrad and Ritchken (1991) extend the binomial tree method to multinomial approximation models. The trinomial tree method is one of the multinomial models.

16.4.1 Cox, Ross, and Rubinstein Model

Cox et al. (1979) (hereafter CRR) propose an alternative choice of parameters that also creates a risk-neutral valuation environment. The price multipliers, u and d, depend only on volatility σ and on δt, not on drift

$$u = e^{\sigma\sqrt{\delta t}}$$

$$d = \frac{1}{u}$$

To offset the absence of a drift component in u and d, the probability of an up move in the CRR tree is usually greater than 0.5 to ensure the expected value of the price increases by a factor of $\exp[(r - q)\delta t]$ on each step. The formula for p is

$$p = \frac{e^{(r-q)\delta t} - d}{u - d}$$

Let $f_{i,j}$ denotes the option value in node (i, j), where i denotes the ith node in period j (j = 0,1,2,…, n). Note in a

binomial tree model, $i = 0, \ldots, j$. Thus, the underlying asset price in a node (i, j) is $Su^j d^{i-j}$. At the expiration we have

$$f_{i,N} = \max\left[Su^i d^{n-i} - X, 0\right] \quad i = 0, 1, \ldots, n$$

Going backward in time (decreasing j), we get

$$f_{i,j} = e^{-r\delta t}[pf_{i+1,j+1} + (1-p)f_{i,j+1}]$$

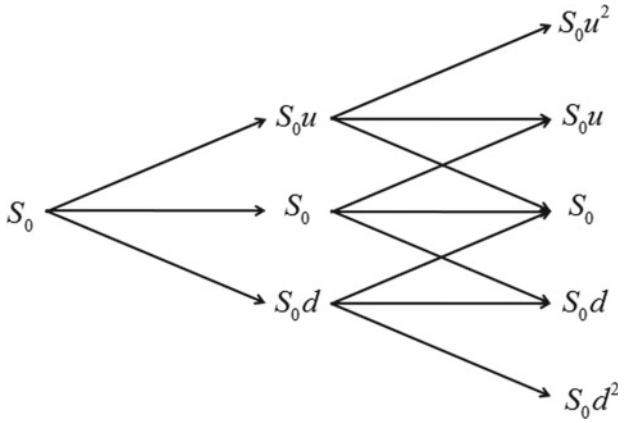
Lee et al. (2000, p 237) has derived the pricing of a call and a put option, respectively, a Binomial Option Pricing model with N period as

$$C = \frac{1}{R^n}\sum_{k=0}^{n}\frac{n!}{k!(n-k!)}p^k(1-p)^{n-k}\max[0,(1+u)^k(1+d)^{n-k}S - X] \tag{16.7}$$

$$P = \frac{1}{R^n}\sum_{k=0}^{n}\frac{n!}{k!(n-k!)}p^k(1-p)^{n-k}\max[0, X - (1+u)^k(1+d)^{n-k}S] \tag{16.8}$$

16.4.2 Trinomial Tree

Because binomial tree methods are computationally expensive, Kamrad and Ritchken (1991) propose multinomial models. New multinomial models include as special cases existing models. The more general models are shown to be computationally more efficient.



Expressed algebraically, the trinomial tree parameters are

$$u = e^{\lambda\sigma\sqrt{\delta t}}$$

$$d = \frac{1}{u}$$

The formula for probability p

$$p_u = \frac{1}{2\lambda^2} + \frac{(r - \sigma^2/2)\sqrt{\delta t}}{2\lambda\sigma}$$

$$p_m = 1 - \frac{1}{\lambda^2}$$

$$p_d = 1 - p_u - p_m$$

If parameter λ is equal to 1, then the trinomial tree model reduces to a binomial tree model. Below is the underlying asset price pattern base on the trinomial tree model.

Appendix 16.2 has shown how the Python program can be used to estimate the trinomial option pricing model. Figures 16.4, 16.5 and 16.6 illustrate the simulation results of trinomial tree option pricing using initial stock price $S_0 = 50$, strike price $X = 50$, $n = 6$ periods, interest rate $r = 0.04$, and $\lambda = 1.5$. Figure 16.4 illustrates the simulated stock prices, and Figs. 16.5 and 16.6 illustrate the corresponding European call and put prices, respectively. As can be seen, for example, as the stock price at the 6th period $S = 84.07$, the European call and put prices are 34.07 and 0, respectively. As the stock price at the 6th period $S = 29.74$, the European call and put prices are 0 and 20.25, respectively.

16.5 Summary

Although using computer programs can make these intensive calculations easy, the prediction of future prices remains a major limitation of binomial models for option pricing. The finer the time intervals, the more difficult it gets to predict the payoffs at the end of each period with high-level precision. However, the flexibility to incorporate the changes expected at different periods is a plus, which makes it suitable for pricing American options, including early-exercise valuations. The values computed using the binomial model

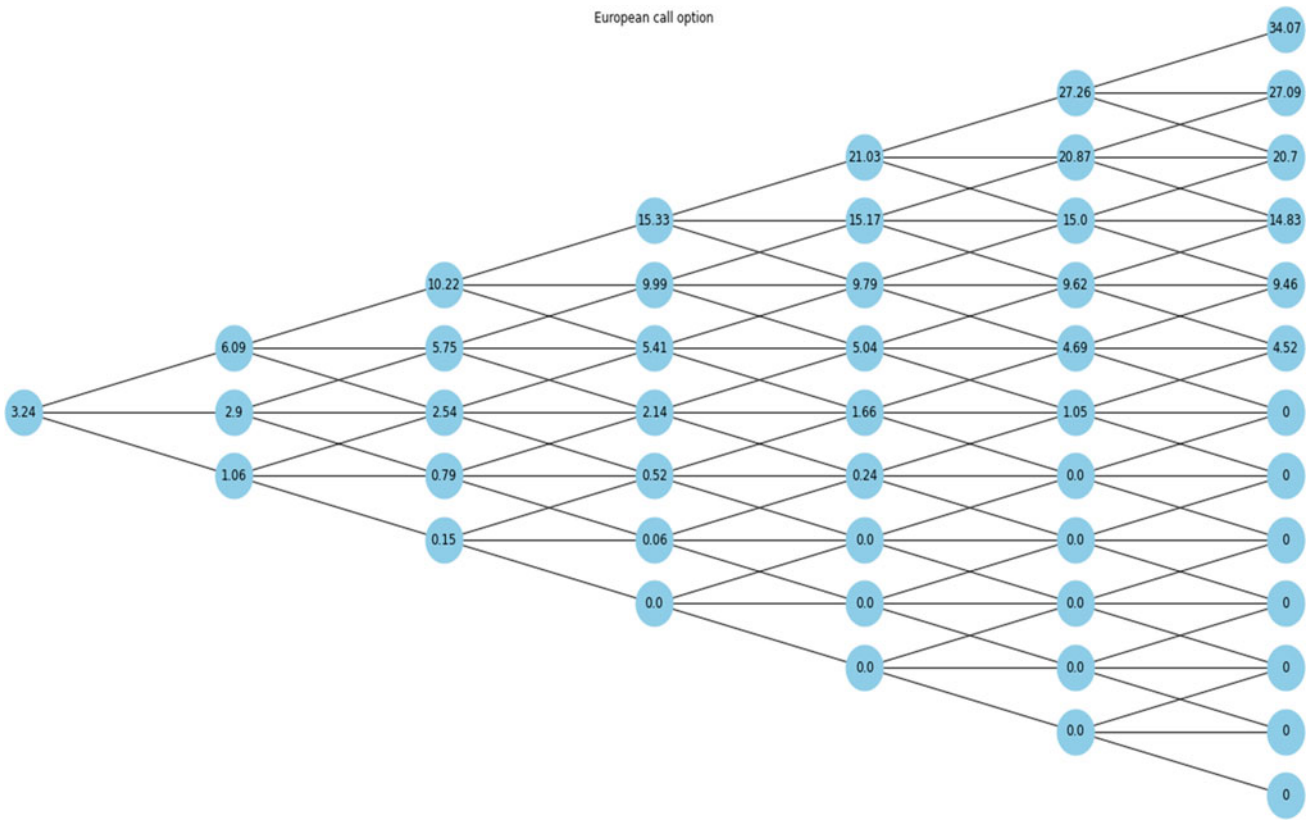


Fig. 16.5 European call prices by trinomial tree

closely match those computed from other commonly used models like Black–Scholes, which indicates the utility and accuracy of binomial models for option pricing. Binomial pricing models can be developed according to a trader's preferences and can work as an alternative to Black–Scholes.

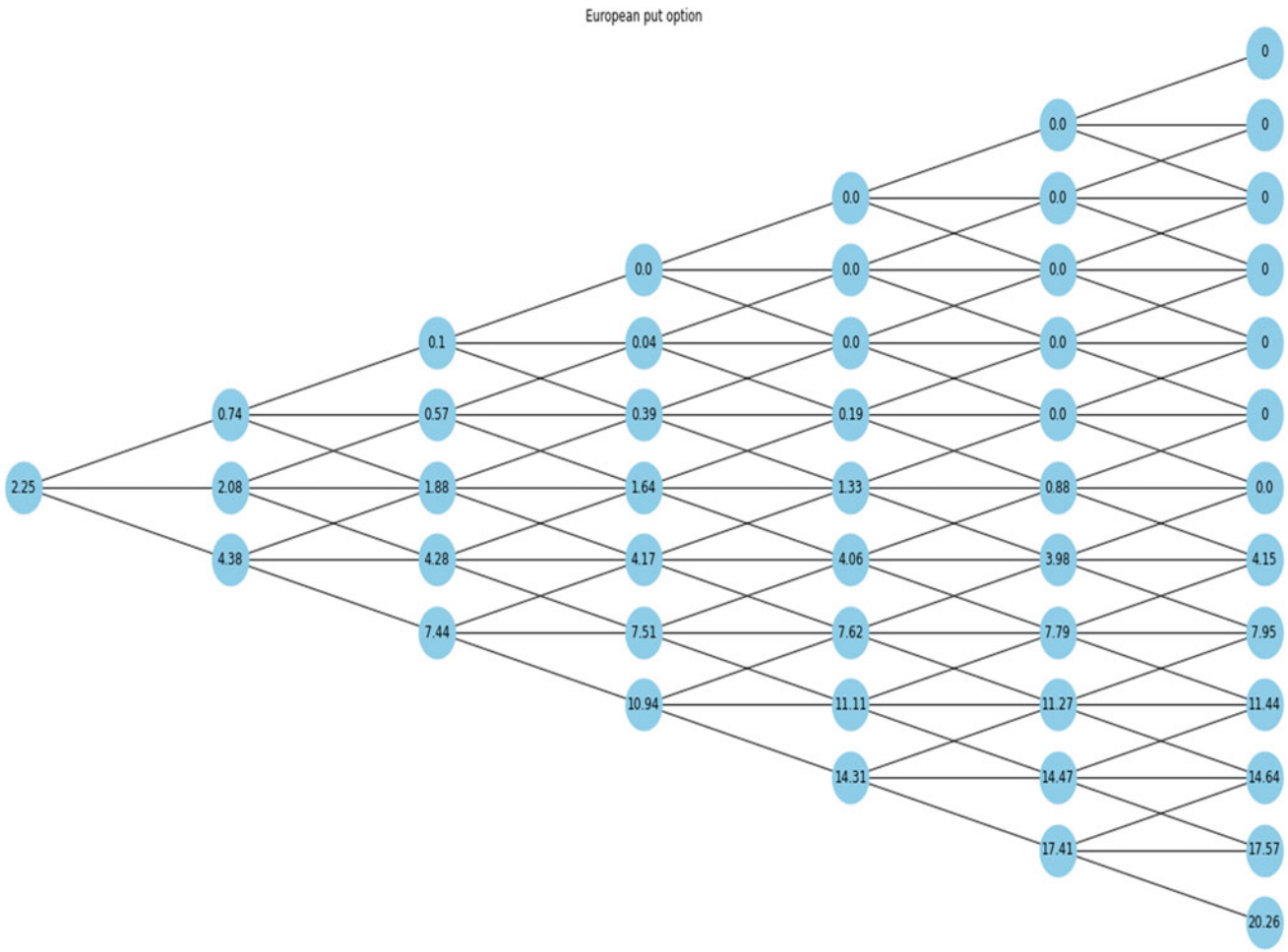


Fig. 16.6 European put prices by trinomial tree

Appendix 16.1: Python Programming Code for Binomial Tree Option Pricing

Input the parameters required for a Binomial Tree:

```
        ' S... stock price
        ' K... strike price
        ' N... time steps of the binomial tree
        . r... Interest Rate
        . sigma... Volatility
        . deltaT ... time duration of a step

import networkx as nx
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
define a balanced binary tree

class Binode(object):

  def __init__(self,element=None,down=None,up=None):
    self.element = element
    self.up = up
    self.down = down

  def dict_form(self):
    dict_data = {'up':self.up,'down':self.down,'element':self.element}
    return dict_data

class Tree(object):

  def __init__(self,root=None):
    self.root = root

#add node from bottom up
  def add_node(self,element):
    new_node = Binode(element)
    if self.root == None:
      self.root = new_node
    else:
      node_queue = list()
      node_queue.append(self.root)
      while len(node_queue):
        cur_node = node_queue.pop(0)
        if cur_node.down == None:
          cur_node.down = new_node
        elif cur_node.up == None:
          cur_node.up = new_node
        else:
          node_queue.append(curnode.down)
          node_queue.append(curnode.up)
Find position for each node(prepare for doubling node)
def hierarchy_pos(G, root=None, width=1., vert_gap = 0.2, vert_loc = 0, leaf_vs_root_factor =
0.5):

    if not nx.is_tree(G):
        raise TypeError('Need to define a tree')

    if root is None:
        if isinstance(G, nx.DiGraph):
            root = next(iter(nx.topological_sort(G)))
        else:
            root = random.choice(list(G.nodes))
```

```
def _hierarchy_pos(G, root, leftmost, width, leafdx = 0.2, vert_gap = 0.2, vert_loc = 0,
            xcenter = 0.5, rootpos = None,
            leafpos = None, parent = None):

   if rootpos is None:
      rootpos = {root:(xcenter,vert_loc)}
   else:
      rootpos[root] = (xcenter, vert_loc)
   if leafpos is None:
      leafpos = {}
   children = list(G.neighbors(root))
   leaf_count = 0
   if not isinstance(G, nx.DiGraph) and parent is not None:
      children.remove(parent)
   if len(children)!=0:
      rootdx = width/len(children)
      nextx = xcenter - width/2 - rootdx/2
      for child in children:
         nextx += rootdx
         rootpos, leafpos, newleaves = _hierarchy_pos(G,child, leftmost+leaf_count*leafdx,
                     width=rootdx, leafdx=leafdx,
                     vert_gap = vert_gap, vert_loc = vert_loc-vert_gap,
                     xcenter=nextx, rootpos=rootpos, leafpos=leafpos, parent = root)
         leaf_count += newleaves
         leftmostchild = min((x for x,y in [leafpos[child] for child in children]))
         rightmostchild = max((x for x,y in [leafpos[child] for child in children]))
         leafpos[root] = ((leftmostchild+rightmostchild)/2, vert_loc)
      else:
         leaf_count = 1
         leafpos[root]  = (leftmost, vert_loc)
#       pos[root] = (leftmost + (leaf_count-1)*dx/2., vert_loc)
#       print(leaf_count)
      return rootpos, leafpos, leaf_count

   xcenter = width/2.
   if isinstance(G, nx.DiGraph):
      leafcount = len([node for node in nx.descendants(G, root) if G.out_degree(node)==0])
   elif isinstance(G, nx.Graph):
      leafcount = len([node for node in nx.node_connected_component(G, root) if
G.degree(node)==1 and node != root])
   rootpos, leafpos, leaf_count = _hierarchy_pos(G, root, 0, width,
                                    leafdx=width*1./leafcount,
                                    vert_gap=vert_gap,
                                    vert_loc = vert_loc,
                                    xcenter = xcenter)
   pos = {}
   for node in rootpos:
      pos[node] = (leaf_vs_root_factor*leafpos[node][0] + (1-
leaf_vs_root_factor)*rootpos[node][0], leafpos[node][1])
#   pos = {node:(leaf_vs_root_factor*x1+(1-leaf_vs_root_factor)*x2, y1) for ((x1,y1), (x2,y2)) in
(leafpos[node], rootpos[node]) for node in rootpos}
   xmax = max(x for x,y in pos.values())
   for node in pos:
      pos[node]= (pos[node][0]*width/xmax, pos[node][1])
   return pos
Final stage

###construct labels for the graph
def construct_labels(initial_price,N,u,d):
```

```
#define a dict contains first layer [layer0:initial price]
 list_node = {'layer0':[initial_price]}
#set a for loop to from 1 to N-1
 for layer in range(1,N+1):
#construct a layer in each loop
   cur_layer = list()
   prev_layer = list_node['layer'+str(layer-1)]
   for ele in range(len(prev_layer)):
     cur_layer.append(round(d*prev_layer[ele],10))
     cur_layer.append(round(u*prev_layer[ele],10))
     #cur_layer = np.unique(cur_layer)
     dict_data = {'layer'+str(layer):cur_layer}
     list_node.update(dict_data)

   return list_node
  #store cur-1 layer
  #for each ele in cur-1 layer, update value in cur layer

 def construct_Ecallput_node(list_node,K,N,u,d,r,call_put):
  p_tel = (1+r-d)/(u-d)
  q_tel = 1-p_tel
  #store the last layer of the list node to a new dict
  last_layer = list_node['layer'+str(N)]
  #use max(x-k,0) to recalculate the value of that layer
  if call_put=='call':
    last_layer = np.subtract(last_layer,K)
  else:
    last_layer = np.subtract(K,last_layer)
  last_layer = [max(ele,0) for ele in last_layer]
  #construct a new dict to store next layer's value
  call_node = {'layer'+str(N):last_layer}
  #construct for loop from layer end-1 to 0
  for layer in reversed(range(N)):
    cur_layer = list()
    propagate_layer = call_node['layer'+str(layer+1)]
    #instide the for loop.construct another for loop from the first element to end-1
    for ele in range(len(propagate_layer)-1):
      #calculate the value for the next layer and add to it
      val = (propagate_layer[ele]*q_tel+propagate_layer[ele+1]*p_tel)/(1+r)
      cur_layer.append(round(val,10))
    dict_data = {'layer'+str(layer):cur_layer}
    call_node.update(dict_data)

  return call_node

  #need to reconstruct plot, can't use netwrokx

 def construct_Acallput_node(list_node,K,N,u,d,r,call_put):
  p_tel = (1+r-d)/(u-d)
  q_tel = 1-p_tel
  #store the last layer of the list node to a new dict
  last_layer = list_node['layer'+str(N)]
  #use max(x-k,0) to recalculate the value of that layer
  if call_put=='call':
    last_layer = np.subtract(last_layer,K)
  else:
    last_layer = np.subtract(K,last_layer)
  last_layer = [max(ele,0) for ele in last_layer]
  #construct a new dict to store next layer's value
```

```
  call_node = {'layer'+str(N):last_layer}
  #construct for loop from layer end-1 to 0
  for layer in reversed(range(N)):
    cur_layer = list()
    propagate_layer = call_node['layer'+str(layer+1)]
    #instide the for loop.construct another for loop from the first element to end-1
    for ele in range(len(propagate_layer)-1):
      #calculate the value for the next layer and add to it
      val = (propagate_layer[ele]*q_tel+propagate_layer[ele+1]*p_tel)/(1+r)

      ## the main difference between european and american option is the following##
      ##need to calculate all the pre-exericise value
      if call_put=='call':
        pre_exercise = max(list_node['layer'+str(layer)][ele]-K,0)# the difference between call and
put
      else:
        pre_exercise = max(K-list_node['layer'+str(layer)][ele],0)

      val = max(val,pre_exercise)#compare new val with pre_exercised one
      cur_layer.append(round(val,10))
    dict_data = {'layer'+str(layer):cur_layer}
    call_node.update(dict_data)

  return call_node

  #need to reconstruct plot, can't use netwrokx

#input price variation and Put option for American
def color_map(list_node_o,list_node_a,N,K):
  #construct a dictionary to store labels
  color_map = []
#define a for loop from 0 to N
  for layer in range(N+1):
#define a for loop from 0 to len(list_node['layer])
    for ele in range(len(list_node_o['layer'+str(layer)])):
      pre_exercise = max(K-list_node_o['layer'+str(layer)][ele],0)
      val = list_node_a['layer'+str(layer)][ele]

      if val<pre_exercise:
        color_map.append('red')
      else:
        color_map.append('skyblue')
#dict.append(counter:list_node['layer][])
#counter++
  return color_map

def construct_nodelabel(list_node,N):
#construct a dictionary to store labels
  nodelabel = {}
#define a for loop from 0 to N
  for layer in range(N+1):
#define a for loop from 0 to len(list_node['layer])
    for ele in range(len(list_node['layer'+str(layer)])):
      dict_data = {str(layer)+str(ele):round(list_node['layer'+str(layer)][ele],2)}
      nodelabel.update(dict_data)
#dict.append(counter:list_node['layer][])
#counter++
  return nodelabel
```

```
def construct_node(node_list,N):
#set a for loop from 0 to n-1
 G = nx.Graph()
 for layer in range(N):
#store layer current and layer next
   cur_layer = node_list['layer'+str(layer)]
#for each ele in current layer, add_edge to ele on next layer and next ele on next layer
   for ele in range(len(cur_layer)):
     G.add_edge(str(layer)+str(ele),str(layer+1)+str(ele))
     G.add_edge(str(layer)+str(ele),str(layer+1)+str(ele+1))
 return G

def construct_nodepos(node_list):
 position = {}
 for layer in range(len(node_list)):
  cur_layer = node_list['layer'+str(layer)]

  for element in range(len(cur_layer)):
    ele_tuple = (layer, -1*layer+2*element) #ele*2 for the gap between up and down is 2
    dict_data = {str(layer)+str(element):ele_tuple}
    position.update(dict_data)

 return position
```

Input the parameters required for a Binomial Tree:

```
        ' S... stock price
        ' K... strike price
        ' N... time steps of the binomial tree
        . r... Interest Rate
        . sigma... Volatility
        . deltaT ... time duration of a step


def usr_input():

 initial_price = input('Stock Price - S (Defualt : 100) --> ') or 100
 K = input('Strike price - K (Default 100) --> ') or 100
 u = input('Increasae Factor - u (Default 1.175) --> ') or 1.175
 d = input('Decrease Factor - d (Default 0.85) --> ') or .85
 N = input('Periods (less than 9) (Default 4)  --> ') or 4
 r = input('Interest Rate - r (Default 0.07) --> ') or .07
 A_E = input('American or European (Default European) --> ') or 'European'
 return int(N),float(initial_price),float(u),float(d),float(r),float(K), A_E
N,initial_price,u,d,r,K,A_E = usr_input()
number_of_calculation = 0
for i in range(N+2):
 number_of_calculation = number_of_calculation+i
Stock Price - S (Defualt : 100) -->
Strike price - K (Default 100) -->
Increasae Factor - u (Default 1.175) -->
Decrease Factor - d (Default 0.85) -->
Periods (less than 9) (Default 4)  -->
Interest Rate - r (Default 0.07) -->
American or European (Default European) -->
```

The price fluctuation tree plot

```
##customize node size and fontsize here
size_of_nodes = 1500
size_of_font = 12
```

```
plt.figure(figsize=(20,10))
vals = construct_labels(initial_price,N,u,d)
labels = construct_nodelabel(vals,N)
nodepos = construct_nodepos(vals)
G = construct_node(vals,N)
nx.set_node_attributes(G, labels, 'label')
nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',alpha=1
,font_weight="bold",font_color='darkblue',font_size=size_of_font)
plt.title('Stock price simulation')
plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()

if A_E =='European':

  plt.figure(figsize=(20,10))
  call_vals = construct_Ecallput_node(vals,K,N,u,d,r,'call')
  labels = construct_nodelabel(call_vals,N)
  nodepos = construct_nodepos(call_vals)
  G = construct_node(call_vals,N)
  nx.set_node_attributes(G, labels, 'label')

nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',alpha=1
,font_weight="bold",font_color='darkblue',font_size=size_of_font)
  plt.title('European call option')
  plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
  nx.draw_networkx_labels(G, nodepos, labels)
  plt.show()

  plt.figure(figsize=(20,10))
  put_vals = construct_Ecallput_node(vals,K,N,u,d,r,'put')
  labels = construct_nodelabel(put_vals,N)
  nodepos = construct_nodepos(put_vals)
  G = construct_node(put_vals,N)
  nx.set_node_attributes(G, labels, 'label')

nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',alpha=1
,font_weight="bold",font_color='darkblue',font_size=size_of_font)
  plt.title('European put option')
  plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
  nx.draw_networkx_labels(G, nodepos, labels)
  plt.show()

else:

  plt.figure(figsize=(20,10))
  call_vals_A= construct_Acallput_node(vals,K,N,u,d,r,'call')
  labels = construct_nodelabel(call_vals_A,N)
  nodepos = construct_nodepos(call_vals_A)
  G = construct_node(call_vals_A,N)
  nx.set_node_attributes(G, labels, 'label')
  nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',alpha=1
  ,font_weight="bold",font_color='darkblue',font_size=size_of_font)
    plt.title('American call option')
    plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
  calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
```

```
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()

plt.figure(figsize=(20,10))
put_vals = construct_Ecallput_node(vals,K,N,u,d,r,'put')
put_vals_A = construct_Acallput_node(vals,K,N,u,d,r,'put')
Color_map = color_map(vals,put_vals,N,K)#should use put_vals instead of put_vals_A
labels = construct_nodelabel(put_vals_A,N)
nodepos = construct_nodepos(put_vals_A)
G = construct_node(put_vals_A,N)
nx.set_node_attributes(G, labels, 'label')

nx.draw(G,pos=nodepos,node_color=Color_map,node_size=size_of_nodes,node_shape='o',alpha
=1,font_weight="bold",font_color='darkblue',font_size=size_of_font)
plt.title('American put option')
plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()
```

Appendix 16.2: Python Programming Code for Trinomial Tree Option Pricing

Input the parameters required for a Trinomial Tree:

```
  ' S... stock price
  ' K... strike price
  ' T ... Time to mature
  ' N... time steps of the binomial tree
  . r... Interest Rate
  . sigma... Volatility
  . lambdA ...  tunable parameter

import networkx as nx
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

###construct labels for the graph
def construct_labels(initial_price,N,T,sigma,lambdA):
  u = np.exp(lambdA*sigma*np.sqrt(T/N))
  d = 1/u
#define a dict contains first layer [layer0:initial price]
  list_node = {'layer0':[initial_price]}
#set a for loop to from 1 to N+1
  for layer in range(1,N+1):
#construct a layer in each loop
    cur_layer = list()
    #add the last node to the layer
    cur_layer.append(initial_price*d**layer)
    #every up node is u times the down node
    for i in range(layer*2):
      cur_layer.append(cur_layer[i]*u)
    dict_data = {'layer'+str(layer):cur_layer}
    list_node.update(dict_data)

  return list_node
#store cur-1 layer
#for each ele in cur-1 layer, update value in cur layer


def construct_Ecallput_node(list_node,K,N,r,T,lambdA,sigma,call_put):
    dt = T/N
    erdt = np.exp(r*dt)
    pu = 1/(2*lambdA**2)+(r-sigma**2/2)*np.sqrt(dt)/(2*lambdA*sigma)
    pm = 1-1/lambdA**2
    pd = 1-pu-pm
    #store the last layer of the list node to a new dict
    last_layer = list_node['layer'+str(N)]
    #use max(x-k,0) to recalculate the value of that layer
    if call_put=='call':
      last_layer = np.subtract(last_layer,K)
    else:
      last_layer = np.subtract(K,last_layer)
    last_layer = [max(ele,0) for ele in last_layer]
    #construct a new dict to store next layer's value
    call_node = {'layer'+str(N):last_layer}
    #construct for loop from layer end-1 to 0
    for layer in reversed(range(N)):
      cur_layer = list()
      propagate_layer = call_node['layer'+str(layer+1)]
      #instide the for loop.construct another for loop from the first element to end-2
      for ele in range(len(propagate_layer)-2):
```

```
      #calculate the value for the next layer and add to it
      val =
(propagate_layer[ele]*pd+propagate_layer[ele+1]*pm+propagate_layer[ele+2]*pu)/erdt
      cur_layer.append(np.round(val,10))
    dict_data = {'layer'+str(layer):cur_layer}
    call_node.update(dict_data)

  return call_node
  #need to reconstruct plot, can't use netwrokx

def construct_nodelabel(list_node,N):
#construct a dictionary to store labels
  nodelabel = {}
#define a for loop from 0 to N
  for layer in range(N+1):
#define a for loop from 0 to len(list_node['layer])
    for ele in range(len(list_node['layer'+str(layer)])):
      dict_data = {str(layer)+str(ele):round(list_node['layer'+str(layer)][ele],2)}
      nodelabel.update(dict_data)
#dict.append(counter:list_node['layer][])
#counter++
  return nodelabel


def construct_node(node_list,N):
#set a for loop from 0 to n-1
  G = nx.Graph()
  for layer in range(N):
#store layer current and layer next
    cur_layer = node_list['layer'+str(layer)]
#for each ele in current layer, add_edge to ele on next layer and next ele on next layer
    for ele in range(len(cur_layer)):
      G.add_edge(str(layer)+str(ele),str(layer+1)+str(ele))
      G.add_edge(str(layer)+str(ele),str(layer+1)+str(ele+1))
      G.add_edge(str(layer)+str(ele),str(layer+1)+str(ele+2))
  return G

def construct_nodepos(node_list):
  position = {}
  for layer in range(len(node_list)):
    cur_layer = node_list['layer'+str(layer)]

    for element in range(len(cur_layer)):
      ele_tuple = (layer, -1*layer+element) #ele*2 for the gap between up and down is 2
      dict_data = {str(layer)+str(element):ele_tuple}
      position.update(dict_data)

  return position

def usr_input():

  initial_price = float(input('Stock Price - S (Defualt : 50) --> ') or 50)
  K = float(input('Strike price - K (Default 50) --> ') or 50)
  sigma = float(input('Volatility - sigma (Default 0.2) --> ') or 0.2)
  T = float(input('Time to mature - T (Default 0.5) --> ') or .5)
  N = int(input('Periods  (Default 6)  --> ') or 6)
  r = float(input('Interest Rate - r (Default 0.04) --> ') or .04)
  lambdA = float(input('Lambda (Default 1.5)-->') or 1.5)
  return initial_price,K,sigma,T,N,r,lambdA
```

```
initial_price,K,sigma,T,N,r,lambdA = usr_input()
number_of_calculation = 0

for i in range(N+2):
  number_of_calculation = number_of_calculation+i
Stock Price - S (Defualt : 50) -->
Strike price - K (Default 50) -->
Volatility - sigma (Default 0.2) -->
Time to mature - T (Default 0.5) -->
Periods  (Default 6)  -->
Interest Rate - r (Default 0.04) -->
Lambda (Default 1.5)-->


size_of_nodes = 1500
size_of_font = 12

plt.figure(figsize=(20,10))
vals = construct_labels(initial_price,N,T,sigma,lambdA)
labels = construct_nodelabel(vals,N)
nodepos = construct_nodepos(vals)
G = construct_node(vals,N)
nx.set_node_attributes(G, labels, 'label')
nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',
alpha=1,font_weight="bold",font_color='darkblue',font_size=size_of_font)
plt.title('Stock price simulation')
#plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()

plt.figure(figsize=(20,10))
call_vals = construct_Ecallput_node(vals,K,N,r,T,lambdA,sigma,'call')
labels = construct_nodelabel(call_vals,N)
nodepos = construct_nodepos(call_vals)
G = construct_node(call_vals,N)
nx.set_node_attributes(G, labels, 'label')
nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',
alpha=1,font_weight="bold",font_color='darkblue',font_size=size_of_font)
plt.title('European call option')
#plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()


plt.figure(figsize=(20,10))
call_vals = construct_Ecallput_node(vals,K,N,r,T,lambdA,sigma,'put')
labels = construct_nodelabel(call_vals,N)
nodepos = construct_nodepos(call_vals)
G = construct_node(call_vals,N)
nx.set_node_attributes(G, labels, 'label')
nx.draw(G,pos=nodepos,node_color='skyblue',node_size=size_of_nodes,node_shape='o',
alpha=1,font_weight="bold",font_color='darkblue',font_size=size_of_font)
plt.title('European put option')
#plt.suptitle('Price = {}, Exercise ={}, U = {}, D = {}, N = {}, Rate = {},Number of
calculation={}'.format(initial_price,K,u,d,N,r,number_of_calculation))
nx.draw_networkx_labels(G, nodepos, labels)
plt.show()
```

References

Benninga, S. *Financial Modeling*. Cambridge, MA: MIT Press, 2000.

Cox, J., S. A. Ross and M. Rubinstein. "Option Pricing: A Simplified Approach." *Journal of Financial Economics*, v. 7 (1979), pp. 229–263.

Jarrow, Robert, and Andrew Rudd. "A comparison of the APT and CAPM a note." Journal of Banking & Finance 7.2 (1983): 295–303.

Kamrad, Bardia, and Peter Ritchken. "Multinomial approximating models for options with k state variables." Management science 37.12 (1991): 1640–1652.

Lee, C. F., J. C. Lee and A. C. Lee (2000). *Statistics for Business and Financial Economics*. 3rd edition. Springer, New York, 2000.

Leisen, Dietmar PJ, and Matthias Reimer. "Binomial models for option valuation-examining and improving convergence." Applied Mathematical Finance 3.4 (1996): 319–346.

Part IV
Financial Management

Check for updates

Financial Ratio Analysis and Its Applications

17

17.1 Introduction

In this chapter, we will briefly review four financial statements from Johnson & Johnson. By using this data, we try to demonstrate how financial ratios are calculated. In addition, sustainable growth rate, DOL, DFL, and DCL will also be discussed in detail. Applications of Excel program to calculate the above-mentioned information will also be demonstrated.

In Sect. 17.2, a brief review of financial statements is given. In Sect. 17.3, an analysis of static ratio is provided. In Sect. 17.4, two possible methods to estimate sustainable growth rate are discussed. In Sect. 17.5, DFL, DOL, and DCL are discussed. A chapter summary is provided in Sect. 17.6. Appendix 17.1 calculates financial ratios with Excel, Appendix 17.2 shows how to use Excel to calculate sustainable growth rate, and finally Appendix 17.3 shows how to compute DOL, DFL, and DCL with Excel.

17.2 Financial Statements: A Brief Review

Corporate annual and quarterly reports generally contain four basic financial statements: balance sheet, statement of earnings, statement of retained earnings, and statement of changes in financial position. Using Johnson & Johnson (JNJ) annual consolidated financial statements as examples, we discuss the usefulness and problems associated with each of these statements in financial analysis and planning. Finally, the use of annual versus quarterly financial data is addressed.

17.2.1 Balance Sheet

The balance sheet describes a firm's financial position at one specific point in time. It is a static representation, such as a snapshot, of the firm's financial composition of assets and liabilities at one point in time. The balance sheet of JNJ, shown in Table 17.1, is broken down into two basic areas of classification—total assets (debit) and total liabilities and shareholders' equity (credit).

On the debit side, accounts are divided into six groups: current assets, marketable securities—non-current, property, plant, and equipment (PP&E), intangible assets, deferred taxes on income, and other assets. Current assets represent short-term accounts, such as cash and cash equivalents, marketable securities and accounts receivable, inventories, deferred tax on income, and prepaid expenses. It should be noted that deferred tax on income in this group is a current deferred tax and will be converted into income tax within one year.

Property encompasses all fixed or capital assets such as real estate, plant and equipment, special tools, and the allowance for depreciation and amortization. Intangible assets refer to the assets of research and development (R&D).

The credit side of the balance sheet in Table 17.1 is divided into current liabilities, long-term liabilities, and shareowner's equity. Under current liabilities, the following accounts are included: accounts, loans, and notes payable; accrued liabilities; accrued salaries and taxes on income. Long-term liabilities include various forms of long-term debt, deferred tax liability, employee-related obligations, and other liabilities. The stockholder's equity section of the balance sheet represents the net worth of the firm to its investors. For example, as of December 31, 2012, JNJ had $0 million preferred stock outstanding, $3,120 million in common stock outstanding, and $85,992 million in retained earnings. Sometimes there are preferred stock and hybrid securities (e.g., convertible bond and convertible preferred stock) on the credit side of the balance sheet.

The balance sheet is useful because it depicts the firm's financing and investment policies. The use of comparative balance sheets, those that present several years' data, can be used to detect trends and possible future problems. JNJ has presented on its balance sheet information from eight periods: December 31, 2012, December 31, 2013, December 31,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_17

Table 17.1 Consolidated balanced sheets of JNJ corporation and subsidiaries

Consolidated balance sheets—USD ($) in millions	2012	2013	2014	2015	2016	2017	2018	2019
Assets								
Current assets								
Cash and cash equivalents	14,911	20,927	14,523	13,732	18,972	17,842	18,107	17,305
Marketable securities	6,178	8,279	18,566	24,644	22,935	472	1,580	1,982
Accounts receivable trade, less allowances for doubtful accounts	11,309	11,713	10,985	10,734	11,699	13,490	14,098	14,481
Inventories	7,495	7,878	8,184	8,053	8,144	8,765	8,599	9,020
Deferred taxes on income	3,139	3,607	–	–	–	–	–	–
Prepaid expenses and other receivables	3,084	4,003	3,486	3,047	3,282	2,537	2,699	2,392
Total current assets	**46,116**	**56,407**	**55,744**	**60,210**	**65,032**	**43,088**	**46,033**	**45,274**
Property, plant and equipment, net	16,097	16,710	16,126	15,905	15,912	17,005	17,053	17,658
Intangible assets, net	28,752	27,947	27,222	25,764	26,876	53,228	47,611	47,643
Goodwill	22,424	22,798	21,832	21,629	22,805	31,906	30,453	33,639
Deferred taxes on income	4,541	3,872	6,202	5,490	6,148	7,105	7,640	7,819
Other assets	3,417	4,949	3,232	4,413	4,435	4,971	4,182	5,695
Total assets	**121,347**	**132,683**	**130,358**	**133,411**	**141,208**	**157,303**	**152,954**	**157,728**
Liabilities and shareholders' equity								
Current liabilities								
Loans and notes payable	4,676	4,852	3,638	7,004	4,684	3,906	2,769	1,202
Accounts payable	5,831	6,266	7,633	6,668	6,918	7,310	7,537	8,544
Accrued liabilities	7,299	7,685	6,553	5,411	5,635	7,304	7,610	9,715
Accrued rebates, returns, and promotions	2,969	3,308	4,010	5,440	5,403	7,201	9,380	10,883
Accrued compensation and employee related obligations	2,423	2,794	2,751	2,474	2,676	2,953	3,098	3,354
Accrued taxes on income	1,064	770	446	750	971	1854	818	2,266
Total current liabilities	**24,262**	**25,675**	**25,031**	**27,747**	**26,287**	**30,537**	**31,230**	**35,964**
Long-term debt	11,489	13,328	15,122	12,857	22,442	30,675	27,684	26,494
Deferred taxes on income	3,136	3,989	2,447	2,562	2,910	8,368	7,506	5,958
Employee related obligations	9,082	7,784	9,972	8,854	9,615	10,074	9,951	10,663
Other liabilities	8,552	7,854	8,034	10,241	9,536	9,017	8,589	11,734
Total liabilities	**56,521**	**58,630**	**60,606**	**62,261**	**70,790**	**97,143**	**93,202**	**98,257**
Shareholders' equity								
Preferred stock—without par value	–	–	–	–	–	–	–	–
Common stock—par value $1.00 per share	3,120	3,120	3,120	3,120	3,120	3,120	3,120	3,120
Accumulated other comprehensive income	(5,810)	(2,860)	(10,722)	(13,165)	(14,901)	(13,199)	(15,222)	(15,891)
Retained earnings	85,992	89,493	97,245	103,879	110,551	101,793	106,216	110,659
Stockholders' equity before treasury stock	83,302	89,753	89,643	93,834	98,770	91,714	94,144	97,888
Less: common stock held in treasury, at cost	18,476	15,700	19,891	22,684	28,352	31,554	34,632	38,417
Total shareholders' equity	**64,826**	**74,053**	**69,752**	**71,150**	**70,418**	**60,160**	**59,752**	**59,471**
Total liabilities and shareholders' equity	**121,347**	**132,683**	**130,358**	**133,411**	**141,208**	**157,303**	**152,954**	**157,728**

2014, December 31, 2015, December 31, 2016, December 31, 2017, December 31, 2018, and December 31, 2019. The balance sheet, however, is static and therefore should be analyzed with caution in financial analysis and planning.

17.2.2 Statement of Earnings

JNJ's statement of earnings is presented in Table 17.2 and describes the results of operations for a 12-month period ending December 31. The usual income-statement periods are annual, quarterly, and monthly. Johnson has chosen the annual approach. Both the annual and quarterly reports are used for external as well as internal reporting. The monthly statement is used primarily for internal purposes, such as the estimation of sales and profit targets, judgment of controls on expenses, and monitoring progress toward longer-term targets. The statement of earnings is more dynamic than the balance sheet, because it reflects changes for the period. It provides an analyst with an overview of a firm's operations and profitability on a gross, operating, and net income basis. JNJ's income includes sales, interest income, and other income/expenses. Costs and expenses for JNJ include the cost of goods sold, selling, marketing, and administrative expenses, depreciation, depletion, and amortization. The difference between income and cost and expenses results in the company's Net Earnings. A comparative statement of earnings is very useful in financial analysis and planning

Table 17.2 Consolidated statements of earnings of JNJ corporation and subsidiaries

(Dollars in millions except per share figures)	2012	2013	2014	2015	2016	2017	2018	2019
Sales to customers ($)	67,224	71,312	74,331	70,074	71,890	76,450	81,581	82,059
Cost of products sold	21,658	22,342	22,746	21,536	21,685	25,354	27,091	27,556
Gross profit	45,566	48,970	51,585	48,538	50,101	51,011	54,490	54,503
Selling, marketing, and administrative expenses	20,869	21,830	21,954	21,203	20,067	21,520	22,540	22,178
Research expense	7,665	8,183	8,494	9,046	9,143	10,594	10,775	11,355
Purchased in-process research and development	1,163	580	178	224	29	408	1,126	890
Interest income	(64)	(74)	(67)	(128)	(368)	(385)	(611)	(357)
Interest expense, net of portion capitalized	532	482	533	552	726	934	1,005	318
Other (income) expense, net	1,626	2,498	(70)	(2,064)	210	(42)	1,405	2,525
Restructuring	–	–	–	509	491	509	251	266
Earnings before provision for taxes on income	13,775	15,471	20,563	19,196	19,803	17,673	17,999	17,328
Provision for taxes on income	3,261	1,640	4,240	3,787	3,263	16,373	2,702	2,209
Net earnings	10,514	13,831	16,323	15,409	16,540	1,300	15,297	15,119
Basic net earnings per share ($)	3.50	3.76	3.67	4.62	6.04	0.48	5.70	5.72
Diluted net earnings per share ($)	3.46	3.73	3.63	4.57	5.93	0.47	5.61	5.63

because it allows insight into the firm's operations, profitability, and financing decisions over time. For this reason, JNJ presents the statement of earnings for six consecutive years: 2012, 2013, 2014, 2015, 2016, 2017, 2018, and 2019. Armed with this information, evaluating the firm's future is easier.

17.2.3 Statement of Equity

JNJ's statements of equity are shown in Table 17.3. These are the earnings that a firm retains for reinvestment rather than paying them out to shareholders in the form of dividends. The statement of equity is easily understood if it is viewed as a bridge between the balance sheet and the statement of earnings. The statement of equity presents a summary of those categories that have an impact on the level of retained earnings: the net earnings and the dividends declared for preferred and common stock. It also represents a summary of the firm's dividend policy and shows how net income is allocated to dividends and reinvestment. JNJ's equity is one source of funds for investment, and this internal source of funds is very important to the firm. The balance sheet, the statement of earnings, and the statement of equity allow us to analyze important firm decisions on the capital structure, cost of capital, capital budgeting, and dividend policy of that firm.

17.2.4 Statement of Cash Flows

Another extremely important part of the annual and quarterly report is the statement of cash flows. This statement is very helpful in evaluating a firm's use of its funds and in determining how these funds were raised. Statements of cash flow for JNJ are shown in Table 17.4. These statements of cash flow are composed of three sections: cash flows from operating activities, cash flows from investing activities, and

Table 17.3 Consolidated statements of equity of JNJ corporation and subsidiaries (2012–2019) (dollars in millions)

Consolidated statements of equity—USD ($) in millions	Total	Retained earnings	Accumulated other comprehensive income	Common stock issued amount	Treasury stock amount
Balance at Dec. 30, 2012	**$ 64,826**	**85,992**	**(5,810)**	**3,120**	**(18,476)**
Net earnings	13,831	13,831	–	–	–
Cash dividends paid	(7,286)	(7,286)	–	–	–
Employee compensation and stock option plans	3,285	(82)	–	–	3,367
Repurchase of common stock	(3,538)	(2,947)	–	–	(591)
Payments for repurchase of common stock	3,538	–	–	–	–
Other	(15)	(15)	–	–	–
Other comprehensive income (loss), net of tax	2,950	–	2,950	–	–
Balance at Dec. 29, 2013	**$ 74,053**	**89,493**	**(2,860)**	**3,120**	**(15,700)**
Net earnings	16,323	16,323	–	–	–
Cash dividends paid	(7,768)	(7,768)	–	–	–
Employee compensation and stock option plans	2,164	(769)	–	–	2,933
Repurchase of common stock	(7,124)	–	–	–	(7,124)
Other	(34)	(34)	–	–	–
Other comprehensive income (loss), net of tax	(7,862)	–	(7,862)	–	–
Balance at Dec. 28, 2014	**$ 69,752**	**97,245**	**(10,722)**	**3,120**	**(19,891)**
Net earnings	15,409	15,409	–	–	–
Cash dividends paid	(8,173)	(8,173)	–	–	–
Employee compensation and stock option plans	1,920	(577)	–	–	2,497
Repurchase of common stock	(5,290)	–	–	–	(5,290)

(continued)

Table 17.3 (continued)

Consolidated statements of equity—USD ($) in millions	Total	Retained earnings	Accumulated other comprehensive income	Common stock issued amount	Treasury stock amount
Other	(25)	(25)	–	–	–
Other comprehensive income (loss), net of tax	(2,443)	–	(2,443)	–	–
Balance at Jan. 03, 2016	**$ 71,150**	**103,879**	**(13,165)**	**3,120**	**(22,684)**
Net earnings	16,540	16,540	–	–	–
Cash dividends paid	(8,621)	(8,621)	–	–	–
Employee compensation and stock option plans	2,130	(1,181)	–	–	3,311
Repurchase of common stock	(8,979)	–	–	–	(8,979)
Other	(66)	(66)	–	–	–
Other comprehensive income (loss), net of tax	(1,736)	–	(1,736)	–	–
Balance at Jan. 01, 2017	**$ 70,418**	**110,551**	**(14,901)**	**3,120**	**(28,352)**
Net earnings	1,300	1,300	–	–	–
Cash dividends paid	(8,943)	(8,943)	–	–	–
Employee compensation and stock option plans	2,077	(1,079)	–	–	3,156
Repurchase of common stock	(6,358)	–	–	–	(6,358)
Other	(36)	(36)	–	–	–
Other comprehensive income (loss), net of tax	1,702	–	1,702	–	–
Balance at Dec. 31, 2017	**$ 60,160**	**101,793**	**(13,199)**	**3,120**	**(31,554)**
Net earnings	15,297	15,297	–	–	–
Cash dividends paid	(9,494)	(9,494)	–	–	–
Employee compensation and stock option plans	1,949	(1,111)	–	–	3,606
Repurchase of common stock	(5,868)	–	–	–	(5,868)
Other	(15)	(15)	–	–	–
Other comprehensive income (loss), net of tax	(1,791)	–	(1,791)	–	–
Balance at Dec. 30, 2018	**$ 59,752**	**106,216**	**(15,222)**	**3,120**	**(34,362)**
Net earnings	15,119	15,119	–	–	–
Cash dividends paid	(9,917)	(9,917)	–	–	–
Employee compensation and stock option plans	1,933	(758)	–	–	2,691
Repurchase of common stock	(6,746)	–	–	–	(6,746)
Other	(1)	(1)	–	–	–
Other comprehensive income (loss), net of tax	(669)	–	(669)	–	–
Balance at Dec. 29, 2019	**$ 59,471**	**110,659**	**(15,891)**	**3,120**	**(38,417)**

Table 17.4 Comparative cash flow statement (**2012–2019**)

(Dollars in millions)	2012	2013	2014	2015	2016	2017	2018	2019
Cash flows from operating activities								
Net earnings	10,514	13,831	16,323	15,409	16,540	1,300	15,297	15,119
Adjustments to reconcile net earnings to cash flows								
Depreciation and amortization of property and intangibles	3,666	4,104	3,895	3,746	3,754	5,642	6,929	7,009
Stock-based compensation	662	728	792	874	878	962	978	977
Non-controlling interest	339	–	87	122	–	–	–	–
Venezuela adjustments	–	108	–	–	–	–	–	–
Asset write-downs	2,131	739	410	624	283	795	1,258	1,096
Net gain on sale of assets/businesses or equity investment		−417	−2,383	−2,583	−563	−1,307	−1,217	−2,154
Deferred tax provision	−39	−607	441	−270	−341	2,406	−1,016	−2,476
Accounts receivable allowances	92	−131	−28	18	−11	17	−31	−20
Changes in assets and liabilities, net of effects from acquisitions								
Increase in accounts receivable	−9	−632	−247	−433	−1,065	−633	−1,185	−289
(Increase)/decrease in inventories	−1	−622	−1,120	−449	−249	581	−644	−277
(Decrease)/increase in accounts payable and accrued liabilities	2,768	1,821	1,194	287	656	1,725	3,951	4,060
Decrease/(increase) in other current and non-current assets	−2,172	−1,806	442	65	−529	−411	−275	−1,054
Increase in other current and non-current liabilities	−2,555	298	−1,096	2,159	−586	8,979	−1,844	1,425
Net cash flows from operating activities	**15,396**	**17,414**	**18,710**	**19,569**	**18,767**	**21,056**	**22,201**	**23,416**
Cash flows from investing activities								
Additions to property, plant, and equipment	−2,934	−3,595	−3,714	−3,463	−3,226	−3,279	−3,670	−3,498
Proceeds from the disposal of assets	1,509	458	4,631	3,464	1,267	1,832	3,302	3,265
Acquisitions, net of cash acquired	−4,486	−835	−2,129	−954	−4,509	−35,151	−899	−5,810
Purchases of investments	−13,434	−18,923	−34,913	−40,828	−33,950	−6,153	−5,626	−3,920
Sales of investments	14,797	18,058	24,119	34,149	35,780	28,117	4,289	3,387
Other (primarily intangibles)	38	−266	−299	−103	−123	−234	−464	44
Net cash used by investing activities	**−4,510**	**−5,103**	**−12,305**	**−7,735**	**−4,761**	**−14,868**	**−3,176**	**−6,194**
Cash flows from financing activities								
Dividends to shareholders	−6,614	−7,286	−7,768	−8,173	−8,621	−8,943	−9,494	−9,917
Repurchase of common stock	−12,919	−3,538	−7,124	−5,290	−8,979	−6,358	−5,868	−6,746
Proceeds from short-term debt	3,268	1,411	1,863	2,416	111	869	80	39
Retirement of short-term debt	−6,175	−1,397	−1,267	−1,044	−2,017	−1,330	−2,479	−100
Proceeds from long-term debt	45	3,607	2,098	75	12,004	8,992	5	3
Retirement of long-term debt	−804	−1,593	−1,844	−68	−2,223	−1,777	−1,555	−2,823
Proceeds from the exercise of stock options	2,720	2,649	1,543	1,005	1,189	1,062	949	954
Other	−83	56	–	−57	−15	−188	−148	575
Net cash used by financing activities	**−20,562**	**−6,091**	**−12,499**	**−11,136**	**−8,551**	**−7,673**	**−18,510**	**−18,015**
Effect of exchange rate changes on cash and cash equivalents	45	−204	310	1,489	−215	337	−241	−9

(continued)

Table 17.4 (continued)

(Dollars in millions)	2012	2013	2014	2015	2016	2017	2018	2019
Increase/ (Decrease) in cash and cash equivalents	−9,631	6,016	6,404	791	5240	−1,148	283	−802
Cash and cash equivalents, beginning of year	24,542	14,911	20,927	14,523	13,732	18,972	17,824	18,107
Cash and cash equivalents, end of year	**14,911**	**20,927**	**14,523**	**13,732**	**18,972**	**17,824**	**18,107**	**17,305**
Supplemental cash flow data								
Cash paid during the year for								
Interest	616	596	603	617	730	960	1,049	576
Interest, net of amount capitalized	501	491	488	515	628	866	963	492
Income taxes	2,507	3,155	3,536	2,865	2,843	3,312	4,570	2,970
Supplemental schedule of noncash investing and financing activities								
Treasury stock issued for employee compensation and stock option plans, net of cash proceeds	615	743	1,409	1,486	2,043	2,062	2,095	995
Conversion of debt	–	22	17	16	35	16	6	1
Acquisitions								
Fair value of assets acquired	19,025	1,028	2,167	1,174	4,586	36,937	1,047	7,228
Fair value of liabilities assumed	−1,204	−193	−38	−220	−77	−1,786	−148	−1,418
Net cash paid for acquisitions	4,486	835	2,129	954	4,509	35,151	899	5,810

cash flows from financing activities. The statement of cash flows can be compiled by either the direct or indirect method. Most companies, such as Johnson & Johnson, compile their cash flow statements using the indirect method. For JNJ, the sources of cash are essentially provided by operations. Application of these funds includes dividends paid to stockholders and expenditures for property, plant, equipment, etc. Therefore, this statement reveals some important aspects of the firm's investment, financing, and dividend policies; making it an important tool for financial planning and analysis.

The cash flow statement shows how the net increase or decrease in cash has been reflected in the changing composition of current assets and current liabilities. It highlights changes in short-term financial policies. It should be noted that the balance of cash flow statement should be equal to the first item of the balance sheet (i.e., cash and cash equivalents). Furthermore, it is well known that investment policy, financial, dividend, and production policies are four important policies in the financial management and decision-making process. Most of the information of these four policies can be obtained from the cash flow statement. For example, cash flow associated with operation activity gives information about operation and production policy. Cash flow associated with investment activity gives information about investments policy. Finally, cash flow associated with financial activity gives information about dividend and financing policy.

The statement of cash flows can be used to help resolve differences between finance and accounting theories. There is value for the analyst in viewing the statement of cash flow over time, especially in detecting trends that could lead to technical or legal bankruptcy in the future. Collectively, the balance sheet, the statement of retained earnings, the statement of equity, and the statement of cash flow present a fairly clear picture of the firm's historical and current position.

17.2.5 Interrelationship Among Four Financial Statements

It should be noted that the balance sheet, statement of earnings, statement of equity, and statement of cash flow are interrelated. These relationships are briefly described as follows:

(1) Retained earnings calculated from the statement of equity for the current period should be used to replace the retained earnings item in the balance sheet of the previous period. Therefore, the statement of equity is regarded as a bridge between the balance sheet and the statement of earnings.
(2) We need the information from the balance sheet, the statement of earnings, and the statement of equity to compile the statement of cash flow.

(3) Cash and cash equivalents item can be found in the statement of cash flow. In other words, the statement of cash flow describes how the cash and cash equivalent changed during the period. It is known that the first item of the balance sheet is cash and cash equivalent.

17.2.6 Annual Versus Quarterly Financial Data

Both annual and quarterly financial data are important to financial analysts; which one is the most important depends on the time horizon of the analysis. Depending upon pattern changes in the historical data, either annual or quarterly data could prove to be more useful. It is well-known that understanding the implications of using quarterly data versus annual data is important for proper financial analysis and planning.

Quarterly data has three components: trend-cycle, seasonal, and irregular or random components. It contains important information about seasonal fluctuations that "reflects an intra-year pattern of variation which is repeated constantly or in evolving fashion form year to year." Quarterly data has the disadvantage of having a large irregular, or random, component that introduces noise into the analysis.

Annual data has both the trend-cycle component and the irregular component, but it does not have the seasonal component. The irregular component is much smaller in annual data than in quarterly data. While it may seem that annual data would be more useful for long-term financial planning and analysis, seasonal data reveals important permanent patterns that underlie the short-term series in financial analysis and planning. In other words, quarterly data can be used for intermediate-term financial planning to improve financial management.

Use of either quarterly or annual data has a consistent impact on the mean-square error of regression forecasting, which is composed of variance and bias. Changing from quarterly to annual data will generally reduce variance while increasing bias. Any difference in regression results, due to the use of different data, must be analyzed in light of the historical patterns of fluctuation in the original time-series data.

17.3 Static Ratio Analysis

In order to make use of financial statements, an analyst needs some form of measurement for analysis. Frequently, ratios are used to relate one piece of financial data to another. The ratio puts the two pieces of data on an equivalent base, which increases the usefulness of the data. For example, net income as an absolute number is meaningless to compare across firms of different sizes. However, if one creates a net profitability ratio (NI/Sales), comparisons are easier to make. Analysis of a series of ratios will give us a clear picture of a firm's financial condition and performance.

Analysis of ratios can take one of two forms. First, the analyst can compare the ratios of one firm with those of similar firms or with industry averages at a specific point in time. This is a type of cross-sectional analysis technique that may indicate the relative financial condition and performance of a firm. One must be careful, however, to analyze the ratios while keeping in mind the inherent differences between a firm's production functions and its operations. Also, the analyst should avoid using "rules of thumb" across industries because the composition of industries and individual firms varies considerably. Furthermore, inconsistency in a firm's accounting procedures can cause accounting data to show substantial differences between firms, which can hinder ratio comparability. This variation in accounting procedures can also lead to problems in determining the "target ratio" (to be discussed later).

The second method of ratio comparison involves the comparison of a firm's present ratio with its past and expected ratios. This form of time-series analysis will indicate whether the firm's financial condition has improved or deteriorated. Both types of ratio analysis can take one of the two following forms: static determination and its analysis, or dynamic adjustment and its analysis. In this section, we only discussed static determination of financial ratios. The dynamic adjustment and its analysis can be found in Lee and Lee (2017).

17.3.1 Static Determination of Financial Ratios

The static determination of financial ratios involves the calculation and analysis of ratios over a number of periods for one company, or the analysis of differences in ratios among individual firms in one industry. An analyst must be careful of extreme values in either direction, because of the interrelationships between ratios. For instance, a very high liquidity ratio is costly to maintain, causing profitability ratios to be lower than they need to be. Furthermore, ratios must be interpreted in relation to the raw data from which they are calculated, particularly for ratios that sum accounts in order to arrive at the necessary data for the calculation. Even though this analysis must be performed with extreme caution, it can yield important conclusions in the analysis for a particular company. Table 17.5 presents six alternative types of ratios for Johnson & Johnson. These six ratios are short-term solvency, long-term solvency, asset management, profitability ratios, market value ratios, and policy ratios. We now discuss these six ratios in detail.

Table 17.5 Alternative financial ratios for Johnson & Johnson (2016–2019)

Ratio classification	Formula	JNJ			
		2019	2018	2017	2016
I. Short-term solvency, or liquidity ratios (times)					
(1) Current ratio	(Current asset)/(current liabilities)	1.26	1.47	1.41	2.47
(2) Quick ratio	(Cash + MS + receivables)/(current liabilities)	0.94	1.08	1.04	2.04
(3) Cash ratio	(Cash + MS)/(current liabilities)	0.54	0.63	0.60	1.59
(4) Net working capital to total asset	(Net working capital)/(total asset)	0.06	0.10	0.08	0.27
II. Long-term solvency, or financial leverage ratios (times)					
(5) Debt to asset	(Total debt)/(total asset)	0.62	0.61	0.62	0.50
(6) Debt to equity	(Total debt)/(total equity)	1.65	1.56	1.61	1.01
(7) Equity multiplier	(Total asset)/(total equity)	2.65	2.56	2.61	2.01
(8) Times interest paid	(EBIT)/(interest expenses)	54.49	17.91	18.92	28.28
(9) Long-term debt ratio	(Long-term debt)/(long-term debt + total equity)	0.31	0.32	0.34	0.24
(10) Cash coverage ratio	(EBIT + depreciation)/(interest expenses)	76.53	24.80	24.96	33.45
III. Asset management, or turnover (activity) ratios (times)					
(11) Day's sales in receivables (average collection period)	(Account receivable) /(sales/365)	64.41	63.08	64.41	59.40
(12) Receivable Turnover	(Sales)/(account receivable)	5.67	5.79	5.67	6.14
(13) Day's sales in inventory	(Inventory)/(cost of goods cold/365)	119.48	115.86	126.18	137.08
(14) Inventory turnover	(Cost of goods sold) (inventory)	3.05	3.15	2.89	2.66
(15) Fixed asset turnover	(Sales)/(fixed assets)	4.65	4.78	4.50	4.52
(16) Total asset turnover	(Sales)/(total assets)	0.52	0.53	0.49	0.51
(17) Net working capital turnover	(Sales)/(net working capital)	8.81	5.51	6.09	1.86
IV. Profitability ratios (percentage)					
(18) Profit margin	(Net income)/(sales)	18.42	18.75	1.70	23.01
(19) Return on assets (ROA)	(Net income)/total assets)	9.59	10.00	0.83	11.71
(20) Return on equity (ROE)	(Net income)/(total equity)	25.42	25.60	2.16	23.49
V. Market value ratios (times)					
(21) Price-earnings ratio	(Mkt price per share)/(earnings per share)	30.08	25.96	289.33	18.70
(22) Market-to-book ratio	(Mkt price per share)/(book value per share)	2.88	2.60	2.39	2.19
(23) Earnings yield	(Earnings per share)/(mkt price per share)	0.03	0.04	0.00	0.05
(24) Dividend yield	(Dividend per share) /(mkt price per share)	0.02	0.02	0.02	0.03
(25) PEG ratio	(Price-earnings ratio)/(earnings growth rate)	343.85	267.28	–2277.37	166.27
(26) Enterprise value-EBITDA ratio	(Enterprise value)/(EBITDA)	18.97	17.68	18.81	14.46
(27) Dividend payout ratio	(Dividend payout)/(net income)	0.66	0.62	6.88	0.52
VI. Policy ratios (percentage)					
(5) Debt to asset	(Total debt)/(total asset)	62.30	60.93	61.76	50.13
(27) Dividend payout ratio	(Dividend payout)/(net income)	65.59	62.06	687.92	52.12
(28) Sustainable growth rate	[(1 − payout ratio) * ROE]/[1 − (1 − payout ratio) * ROE]	9.59	10.76	–11.27	12.67

Short-Term Solvency, or Liquidity Ratios

Liquidity ratios are calculated from information on the balance sheet; they measure the relative strength of a firm's financial position. Crudely interpreted, these are coverage ratios that indicate the firm's ability to meet short-term obligations. The current ratio (ratio 1 in Table 17.5) is the most popular of the liquidity ratios because it is easy to calculate, and it has intuitive appeal. It is also the most broadly defined liquidity ratio, as it does not take into account the differences in relative liquidity among the individual components of current assets. A more specifically defined liquidity ratio is the quick or acid-test ratio (ratio 2), which excludes the least liquid portion of current assets and inventories. In other words, the numerator of this ratio includes cash, marketable securities (MS), and receivables. Cash ratio (ratio 3) is the ratio of the company's total cash and cash equivalents (marketable securities, MS) to its current liabilities. It is most often used as a measure of company liquidity. A strong cash ratio is useful to creditors when deciding how much debt they are willing to extend to the asking party (Investopedia.com).

The net working capital to total asset ratio (ratio 4) is the NWC divided by the total assets of the company. A relatively low value might indicate relatively low levels of liquidity.

Long-Term Solvency, or Financial Leverage Ratios

If an analyst wishes to measure the extent of a firm's debt financing, a leverage ratio is the appropriate tool to use. This group of ratios reflects the financial risk posture of the firm. The two sources of data from which these ratios can be calculated are the balance sheet and the statement of earnings.

The balance sheet leverage ratio measures the proportion of debt incorporated into the capital structure. The debt–equity ratio measures the proportion of debt that is matched by equity; thus this ratio reflects the composition of the capital structure. The debt–asset ratio (ratio 5), on the other hand, measures the proportion of debt-financed assets currently being used by the firm. Other commonly used leverage ratios include the equity multiplier ratio (7) and the time interest paid ratio (8).

Debt-to-equity (6) is a variation in the total debt ratio. Its total debt is divided by total equity.

Long-term debt ratio (9) is long-term debt divided by the sum of long-term debt and total equity.

Cash coverage ratio (10) is defined as the sum of EBIT and depreciation divided by interest. The numerator is often abbreviated as EBITDA.

The income-statement leverage ratio measures the firm's ability to meet fixed obligations of one form or another. The time interest paid, which is earnings before interest and taxes over interest expense, measures the firm's ability to service the interest expense on its outstanding debt. A more broadly defined ratio of this type is the fixed-charge coverage ratio, which includes not only the interest expense but also all other expenses that the firm is obligated by contract to pay (This ratio is not included in Table 17.5 because there is not enough information on fixed charges for these firms to calculate this ratio).

Asset Management, or Turnover (Activity) Ratios

This group of ratios measures how efficiently the firm is utilizing its assets. With activity ratios, one must be particularly careful about the interpretation of extreme results in either direction; very high values may indicate possible problems in the long term, and very low values may indicate a current problem of low sales or not taking a loss for obsolete assets. The reason that high activity may not be good in the long term is that the firm may not be able to adjust to an even higher level of activity and therefore may miss out on a market opportunity. Better analysis and planning can help a firm get around this problem.

The days-in-accounts-receivable or average collection period ratio (11) indicates the firm's effectiveness in collecting its credit sales. The other activity ratios measure the firm's efficiency in generating sales with its current level of assets, appropriately termed turnover ratios. While there are many turnover ratios that can be calculated, there are three basic ones: inventory turnover (14), fixed assets turnover (15), and total assets turnover (16). Each of these ratios measures a different aspect of the firm's efficiency in managing its assets.

Receivables turnover (12) is computed as credit sales divided by accounts receivable. In general, a higher accounts receivable turnover suggests more frequent payment of receivables by customers.

In general, analysts look for higher receivables turnover and shorter collection periods, *but this combination may imply that the firm's credit policy is too strict,* allowing only the lowest risk customers to buy on credit. Although this strategy could minimize credit losses, it may hurt overall sales, profits, and shareholder wealth.

Day's sales in inventory ratio (13) estimate how many days, on average, a product sits in the inventory before it is sold.

Net working capital turnover (17) measures how much per dollar of net working capital can generate dollar of sales. For example, if this ratio is 3, this means the per dollar of net working capital can generate $3 of sales.

Profitability Ratios

This group of ratios indicates the profitability of the firm's operations. It is important to note here that these measures are based on past performance. Profitability ratios are generally the most volatile, because many of the variables affecting them are beyond the firm's control. There are three groups of profitability ratios; those measuring margins, those measuring returns, and those measuring the relationship of market values to book or accounting values.

Profit-margin ratios show the percentage of sales dollars that the firm was able to convert into profit. There are many such ratios that can be calculated to yield insightful results, namely, profit margin (18), return on asset (19), and return on equity (20).

Return ratios are generally calculated as a return on assets or equity. The return on assets ratio (19) measures the profitability of the firm's asset utilization. The return on equity ratio (20) indicates the rate of return earned on the book value of owner's equity. Market-value analyses include (i) market-value/book-value ratio and (ii) price per share/earnings per share (P/E) ratio, and other ratios as indicated in Table 17.5.

Overall, all four different types of ratios (as indicated in Table 17.5) have different characteristics stemming from the firm itself and the industry as a whole. For example, the collection period ratio (which is Accounts Receivable times 365 over Net Sales) is clearly the function of the billings, payment, and collection policies of the pharmaceutical industry. In addition, the fixed-asset turnover ratios for those firms are different, which might imply that different firms have different capacity utilization.

Market Value Ratios

A firm's profitability, risk, quality of management, and many other factors are reflected in its stock and security prices. Hence, market value ratios indicate the market's assessment of the value of the firm's securities.

The price-earnings (PE) ratio (21) is simply the market price of the firm's common stock divided by its annual earnings per share. Sometimes called the *earnings multiple*, the PE ratio shows how much the investors are willing to pay for each dollar of the firm's earnings per share. Earnings per share comes from the income statement. Therefore, earnings per share is sensitive to the many factors that affect the construction of an income statement, such as the choice of GAAP to management decisions regarding the use of debt to finance assets. Although earnings per share cannot reflect the value of patents or assets, the quality of the firm's management, or its risk, and stock prices can reflect all of these factors. Comparing a firm's PE ratio to that of the stock market as a whole, or with the firm's competitors, indicates the market's perception of the true value of the company.

Market-to-book ratio (22) measures the market's valuation relative to balance sheet equity. The book value of equity is simply the difference between the book values of assets and liabilities appearing on the balance sheet. The price-to-book-value ratio is the market price per share divided by the book value of equity per share. A higher ratio suggests that investors are more optimistic about the market value of a firm's assets, its intangible assets, and the ability of its managers.

Earnings yield (23) is defined as earnings per share divided by market price per share and is used to measure return on investment. Dividend yield (24) is defined as dividend per share divided by the market price per share, which is used to determine whether this company's stock is an income stock or a gross stock. A gross stock dividend yield is very small or even zero. For example, the stock from a utility industry dividend yield is very high.

PEG ratio (25) is defined as price-earnings ratio divided by earnings growth rate. The price/earnings to growth (PEG) ratio is used to determine a stock's value while taking the company's earnings growth into account and is considered to provide a more complete picture than the PE ratio. While a high PE ratio may make a stock look like a good buy, factoring in the company's growth rate to get the stock's PEG ratio can tell a different story. The lower the PEG ratio, the more the stock may be undervalued given its earnings performance. The PEG ratio that indicates an over or underpriced stock varies by industry and by company type, though a broad rule of thumb is that a PEG ratio below one is desirable. Also, the accuracy of the PEG ratio depends on the inputs used. Sustainable growth rate is usually used to estimate earnings growth rate. In Appendix 17.2, we introduce two possible methods to calculate it. However, using historical growth rates, for example, may provide an inaccurate PEG ratio if future growth rates are expected to deviate from historical growth rates. To distinguish between calculation methods using future growth and historical growth, the terms "forward PEG" and "trailing PEG" are sometimes used.

Enterprise value is an estimate of the market value of the company's operating assets, which means all the assets of the firm except cash. Since market values are usually unavailable, we use the right-hand side of the balance sheet and calculate the enterprise value as

$$\begin{aligned}\text{Enterprise value} &= \text{Total Market Value of Equity} \\ &\quad + \text{Book Value of Total Liabilities} - \text{Cash}\end{aligned}$$

Notice that the sum of the value of the market values of the stock and all liabilities equals the value of the firm's

assets from the balance sheet identity. Total market value of equity = market price per share times basic number of shares outstanding.

Enterprise value is often used to calculate the Enterprise value-EBITDA ratio (26):

$$\text{EBITDA ratio} = \text{Enterprise value}/\text{EBITDA}$$

where EBITDA is defined as earnings before interest, taxes, depreciation, and amortization.

This ratio is similar to the PE ratio, but it relates the value of all the operating assets to a measure of the operating cash flow generated by those assets.

Policy Ratios

Policy ratios include debt-to-asset ratio, dividend payout ratio, and sustainable growth rate. Debt-to-asset ratio has been discussed in Group 2 of Table 17.5. Dividend payout ratio is defined as (dividend payout)/(net income). The dividend payout ratio is the ratio of the total amount of dividends paid out to shareholders relative to the net income of the company. It is the percentage of earnings paid to shareholders in dividends. The amount that is not paid to shareholders is retained by the company to pay off debt or to reinvest in core operations. It is sometimes simply referred to as the "payout ratio."

Sustainable growth rate is defined as [(1 − payout ratio) *ROE]/[1 − (1 − payout ratio)*ROE]. Appendix 2B will discuss sustainable growth rate in further detail.

Table 17.5 summarizes all 28 ratios for Johnson & Johnson during 2016, 2017, 2018, and 2019. Appendix 2A shows how to use Excel to calculate the first 26 ratios with the data of 2018 and 2019 from JNJ Financial Statement.

Estimation of the Target of a Ratio

An issue that must be addressed at this point is the determination of an appropriate proxy for the target of a ratio. For an analyst, this can be an insurmountable problem if the firm is extremely diversified, and if it does not have one or two major product lines in industries where industry averages are available. One possible solution is to determine the relative industry share of each division or major product line, then apply these percentages to the related industry averages. Lastly, derive one target ratio for the firm as a whole with which its ratio can be compared. One must be very careful in any such analysis, because the proxy may be extremely over- or underestimated. The analyst can also use Standard Industrial Classification (SIC) codes to properly define the industry of diversified firms. The analyst can then use 3- or 4-digit codes and compute their own weighted industry average.

Often an industry average is used as a proxy for the target ratio. This can lead to another problem, the inappropriate calculation of an industry average, even though the industry and companies are fairly well defined. The issue here is the appropriate weighting scheme for combining the individual company ratios in order to arrive at one industry average. Individual ratios can be weighted according to equal weights, asset weights, or sales weights. The analyst must determine the extent to which firm size, as measured by asset base or market share, affects the relative level of a firm's ratios and the tendency for other firms in the industry to adjust toward the target level of this ratio. One way this can be done is to calculate the coefficients of variation for a number of ratios under each of the weighting the schemes and to compare them to see which scheme consistently has the lowest coefficient variation. This would appear to be the most appropriate weighting scheme. Of course, one could also use a different weighting scheme for each ratio, but this would be very tedious if many ratios were to be analyzed. Note, that the median rather than the average or mean can be used to avoid needless complications with respect to extreme values that might distort the computation of averages. Dynamic financial ratio analysis is to compare individual company ratios with industry averages over time. In general, this kind of analysis needs to rely upon regression analysis. Lee and Lee (2017, Chap. 2) have discussed this kind of analysis in detail.

17.4 Two Possible Methods to Estimate the Sustainable Growth Rate

Sustainable growth rate (SGR) can be either estimated by (i) using both external and internal source of fund or (ii) using only internal source of fund.

We present these two methods in detail as follows:

Method 1: The sustainable growth rate with both external and internal source of fund can be defined as (Lee 2017)

$$SGR = \frac{\text{Retention Rate*ROE}}{1-(\text{Retention Rate*ROE})} = \frac{(1-\text{Dividend Payout Ratio})\text{*ROE}}{1-[(1-\text{Dividend Payout Ratio})*ROE]} \quad (17.1)$$

$$\text{Dividend Payout Ratio} = \text{Dividends}/\text{Net Income}$$

Method 2: The sustainable growth rate: considering internal source of fund

$$\begin{aligned} \text{ROE} &= \text{Net Income/Total Equity} \\ \text{ROE} &= (\text{Net Income/Assets}) \times (\text{Assets/Equity}) \\ \text{ROE} &= (\text{Net Income/Sales}) \times (\text{Sales/Assets}) \\ &\quad \times (\text{Assets/Equity}) \\ \text{SGR} &= ROE \times (1 - Dividend\ Payout\ Ratio) \end{aligned} \tag{17.2}$$

Example

With the data from JNJ financial statement of 2019 fiscal year, we estimate obtain

$$\begin{aligned} \text{ROE} &= \text{Net Income/Total Equity} = 15,119/59,471 \\ &= 0.2542 \end{aligned}$$

$$\begin{aligned} \text{Dividend Payout Ratio} &= \text{Dividends/Net Income} \\ &= 9,917/15,119 = 0.6559 \end{aligned}$$

$$\begin{aligned} &\text{According to the method 1, SGR} \\ &= (1-0.6559) * 0.2542/1-[(1-0.6559) * 0.2542] \\ &= 0.0959 \end{aligned}$$

$$\begin{aligned} &\text{According to the method 2, SGR} = 0.2542 * (1-0.6559) \\ &= 0.0875. \end{aligned}$$

The difference between method 1 and method 2

Technically, as $ROE \times (1 - D)$ is the numerator of $\frac{ROE \times (1-D)}{1-ROE \times (1-D)}$ and $1 > [1 - ROE \times (1 - D)] \geq 0$, it is easy to prove $\frac{ROE \times (1-D)}{1-ROE \times (1-D)} \geq ROE \times (1 - D)$.

In addition, we can transform $\frac{ROE \times (1-D)}{1-ROE \times (1-D)}$ into $\frac{\text{Retained Earnings}}{\text{Equity–Retained Earnings}}$ and transform $ROE \times (1 - D)$ into $\frac{\text{Retained Earnings}}{\text{Equity}}$. It is obvious to see $\frac{\text{Retained Earnings}}{\text{Equity–Retained Earnings}} \geq \frac{\text{Retained Earnings}}{\text{Equity}}$ since Equity − Retained Earnings $\leq$ Equity. If we use equity value at the end of this year, then (Equity − Retained Earnings) can be interpreted as the equity value at the beginning of this year under the condition of no external finance.

Consequently, the SGR from method 1 is usually greater than that from method 2. The numerical result $0.0959 > 0.0875$ confirms this. In Appendix 17.2, we use Excel to show how to calculate SGR with two methods.

17.5 DFL, DOL, and DCL

It is well known that financial leverage can lead to higher expected earnings for a corporation's stockholders. The use of borrowed funds to generate higher earnings is known as *financial leverage*. But this is not the only form of leverage available to increase corporate earnings. Another form is *operating leverage*, which pertains to the proportion of the firm's fixed operating costs. In this section, we discuss degree of financial leverage (DFL), degree of operating leverage (DOL), and degree of combined leverage (DCL).

17.5.1 Degree of Financial Leverage

Suppose that a levered corporation improves its performance of the previous year by increasing its operating income by 1 percent. What is the effect on earnings per share? If you answered "a 1 percent increase," you have ignored the influence of leverage. To illustrate, consider the corporation of Table 17.6. In the current year, as we saw earlier, this firm produces earnings per share of $2.49.

The firm's operating performance improves next year, to the extent that earnings before interest and taxes increase by 1 percent, from $270 million to $272.7 million. Other relevant factors are unchanged. Interest payments are $104 million, and with a corporate tax rate of 40 percent, 60 percent of earnings after interest are available for distribution to stockholders. Thus, earnings available to stockholders = 0.60 (272.7 − 104) = $101.22 million. Therefore, with 40 million shares outstanding, earnings per share next year will be

$$EPS = \frac{\$101.22}{40} = \$2.5305$$

Hence, the percentage increase in earnings per share is

$$\%change\ in\ EPS = \frac{2.5305 - 2.49}{2.49} \times 100 = 1.6265\%$$

We see that a 1 percent increase in EBIT leads to a greater percentage increase in EPS. The reason is that none of the increased earnings need be paid to debtholders. All of this increase goes to equity holders, who therefore benefit disproportionately. The argument is symmetrical. If EBIT were to fall by 1 percent, then EPS would fall by 1.6265%.

The extent to which a given percentage increase in operating income produces a greater percentage increase in earnings per share provides a measure of the effect of leverage on stockholders' earnings. This is known as the *degree of financial leverage* (DFL) and is defined as

$$DFL = \frac{\%change\ in\ EPS}{\%change\ in\ EBIT}$$

We now develop an expression for the degree of financial leverage. Suppose that a firm has earnings before interest and tax of EBIT, and debt of B, on which are interest payments at rate i. If the corporate tax rate is τ_c, then

$$earnings\ available\ to\ stockholders = (1 - \tau_c)(EBIT - iB) \tag{17.3}$$

If the firm increases operating income by 1 percent to 1.01 EBIT, with everything else unchanged, we have

$$earnings\ available\ to\ stockholders = (1 - \tau_c)(1.01 \times EBIT - iB) \tag{17.4}$$

Comparing Eqs. (17.3) and (17.4), the increase in earnings available to stockholders is

$$(1 - \tau_c)(1.01EBIT - iB) - (1 - \tau_c)(EBIT - iB) = .01(1 - \tau_c)EBIT$$

It follows that the percentage change in stockholders' earnings, and hence in earnings per share, is

$$\%change\ in\ EPS = \frac{(.01)(1 - \tau_c)EBIT}{(1 - \tau_c)(EBIT - iB)} \times 100 = \frac{(.01)EBIT}{EBIT - iB} \times 100$$

Since the increase in EBIT is 1 percent, it follows from our definition that the degree of financial leverage is

$$DFL = \frac{(.01)EBIT}{(EBIT - iB).01)} = \frac{EBIT}{EBIT - iB} = 1.6265 \tag{17.5}$$

Thus, the degree of financial leverage can be found as the ratio of net operating income to income remaining after interest payments on debt. This is illustrated in Fig. 17.1, which plots the degree of financial leverage against interest payments for a given level of net operating income. If there are no interest payments, so that the firm is unlevered, DFL is 1. That is, each 1 percent increase in earnings before interest and tax leads to a 1 percent increase in earnings per share. As interest payments increase, so does the degree of financial leverage, to the point where, if interest payments equal net operating income, DFL is infinite. This is not surprising, for in this case there would be no earnings available to stockholders. Hence, any increase in net operating income would, proportionately, yield an infinitely large improvement. The relationship between DFL and interest payments is presented in Fig. 17.1.

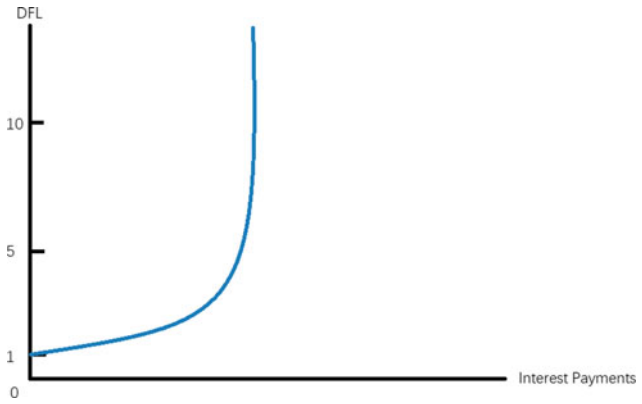


Fig. 17.1 Relation between degree of financial leverage and interest payments

17.5.2 Operating Leverage and the Combined Effect

Net earnings are the difference between total sales value and total operating costs. We now look in detail at operating costs, which we break down into two components: fixed costs and variable costs. *Fixed costs* are costs that the firm must incur, whatever its level of production. Such costs include rent and equipment depreciation. *Variable costs* are costs that increase with production, such as wages. The mix of fixed and variable costs in a firm's total operating cost structure provides operating leverage. Let us consider a firm with a single product, under the following conditions:

- The firm incurs fixed costs F, which must be paid whatever the level of output.
- Each unit of output costs an additional amount V.
- Each unit of output can be sold at price P.
- A total of Q units of output are produced and sold.

XYZ Corporation produces parts for the automobile industry. Information for this corporation can be found in Table 17.6. Its current net operating income is derived from the sale of 10 million units, priced at $150 each. Operating cost consist of $310 million of fixed costs and variable costs of $92 per unit.

Suppose this corporation increases its sales volume by 1 percent to 10.1 million units next year, with other factors unchanged. Would you guess that earnings before interest and tax also increase by 1 percent? In fact, net operating income will rise by more than 1 percent. The reason is that while the value of sales and variable operating costs increases proportionately, fixed operating costs remain uncharged. These costs, then, constitute a source of

Table 17.6 Consolidated balance sheets of J&J corporation and subsidiaries

Information for XYZ corporation		
Value of assets = $2,400 million		
Value of debt = $1,300 million		
Interest paid on debt = $104 million		
Corporate tax rate = 40%		
Shares outstanding = 40 million		
Earnings before interest and taxes = $270 million		
Value of sales		$1,500 million
Fixed operating costs	$310 million	
Variable operating costs	920 million	
Total operating costs		1,230 million
Earnings before interest and taxes		$270 million
Volume of sales: 10 million units		
Price per unit: $150		

operating leverage. The greater the share of total cost attributable to fixed costs, the greater this leverage.

The extent to which a given percentage increase in sales volume produces a greater percentage increase in earnings before interest and taxes are used to measure the degree of operating leverage.

The degree of operating leverage (DOL) is given by

$$DOL = \frac{\%change\ in\ EBIT}{\%change\ in\ sales\ volume}$$

Let us now find a measure of degree of operating leverage. If Q units are sold at price P, then

$$value\ of\ sales = QP$$

Total operating costs consist of fixed costs F and total variable costs QV, so that

$$\begin{aligned} total\ operating\ costs &= fixed\ costs + variable\ costs \\ &= F + QV \end{aligned}$$

Therefore, we can write

$$\begin{aligned} &earnings\ before\ interest\ and\ taxes \\ &= value\ of\ sales - total\ operating\ costs \end{aligned}$$

So that

$$EBIT = QP - (F + QV) = Q(P - V) - F \qquad (17.6)$$

Suppose that sales volume increases by 1 percent from Q to $1.01Q$. In this case, we have

$$EBIT = 1.01(Q(P - V) - F)$$

So that, by comparison with (11.14), the increase in EBIT is .01Q(P − V). It follows that

$$\begin{aligned} \%change\ in\ EBIT &= \frac{.01Q(P - V)}{Q(P - V) - F} \times 100 \\ &= \frac{Q(P - V)}{Q(P - V) - F} \end{aligned}$$

Since there is a 1 percent increase in sales volume, it follows from our definition of degree of operating leverage that

$$DOL = \frac{Q(P - V)}{Q(P - V) - F} \qquad (17.7)$$

Therefore,

$$DOL = \frac{value\ of\ sales - variable\ costs}{value\ of\ sales - variable\ costs - fixed\ costs}$$

Let us compute the degree of operating leverage for the firm of Table 17.6:

$$DOL = \frac{1,500 - 920}{1,500 - 920 - 310} = 2.1481$$

For this firm, each 1 percent increase in sales volume leads to an increase of 2.1481 percent in earnings before interest and taxes.

The source of operating leverage is illustrated in Fig. 17.2, which plots degree of operating leverage against the proportion of total fixed costs. If there are no fixed costs, then, as is clear from Eq. 17.7, the degree of operating leverage is 1. In other words, there is no operating leverage, and a 1 percent increase in sales volume leads to a 1 percent

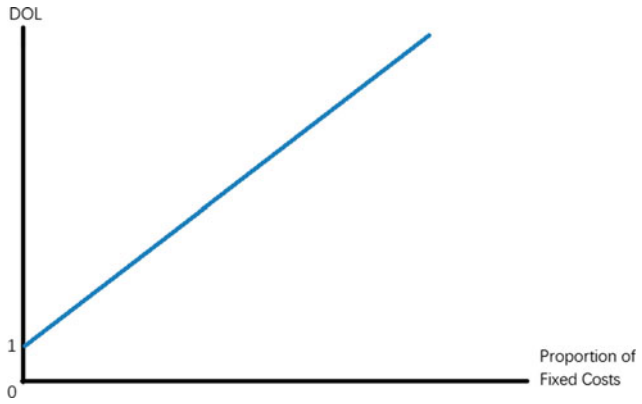


Fig. 17.2 Relation between degree of operating leverage and proportion of fixed costs

increase in earnings before interest and taxes. As the proportion of fixed costs increases, so does the degree of operating leverage.

Operating leverage and financial leverage may act in combination, so that the impact of a change in corporate performance, as measured by volume of sales, is magnified in its effect on earnings per share. We can think of this combined leverage effect as developing through two stages:

1. To the extent that there are fixed costs in a firm's total cost structure, an increase (decrease) in sales volume produces a greater percentage increase (decrease) in earnings before interest and taxes, through the effect of operating leverage.
2. To the extent that interest payments must be made to debtholders, an increase (decrease) in earnings before interest and taxes produces a greater percentage increase (decrease) in earnings per share.

The combined leverage effect measures the extent to which a given percentage increase in sales volume leads to a greater percentage increase in earnings per share.

The *combined leverage effect* (CLE) is given by

$$CLE = \frac{\%\, change\ of\ EPS}{\%\, change\ in\ sales\ volume}$$

We can express the combined leverage effect as

$$\begin{aligned} CLE &= \frac{\%\, change\ in\ EPS}{\%\, change\ in\ EBIT} \times \frac{\%\, change\ in\ EBIT}{\%\, change\ in\ sales\ volume} \\ &= DFL \times DOL \end{aligned} \tag{17.8}$$

Therefore, we see that combined leverage is the product of the degrees of financial and operating leverage. For the firm in Table 17.6, we find from our previous calculations that

$$CLE = (1.625)(2.1481) = 3.49$$

For this firm, each 1 percent increase in sales volume leads to an increase of 3.49 percent in earnings per share. Thus, the combined effects of operating and financial leverage produce for stockholders a magnification of variations in business performance, in the sense that percentage changes in sales volume are reflected in percentage changes of almost three-and-one-half their size in earnings per share.

We conclude this discussion by giving an algebraic expression that allows direct evaluation of the combined leverage effect. Writing earnings before interest and taxes as

$$EBIT = Q(P - V) - F$$

and using Eq. 17.5, the degree of financial leverage is

$$DFL = \frac{Q(P - V) - F}{Q(P - V) - F - iB}$$

Therefore, using Eq. 17.8, we can find the combined leverage effect:

$$\begin{aligned} CLE &= DFL \times DOL \\ &= \frac{Q(P - V) - F}{Q(P - V) - F - iB} \times \frac{Q(P - V)}{Q(P - V) - F} \\ &= \frac{Q(P - V)}{Q(P - V) - F - iB} \end{aligned}$$

Thus, combined leverage can be found as follows:

$$CLE = \frac{Q(P - V)}{Q(P - V) - F - iB} \tag{17.9}$$

It is the final two terms in the denominator of Eq. 17.9, acting in combination, that produce leverage. If there were no fixed operating costs and no interest payments on the debt, then there would be no leverage. Each dollar increases in either term, all else equal, produces the same leverage as a dollar increase the other. Moreover, we see that if an increase in interest payments is matched by a decrease of the same amount in fixed operating costs, then leverage will be unchanged.

We now verify Eq. 17.9 for the firm in Table 17.6. For this firm, value of sales = \$1,500; variable operating costs = \$920; fixed operating costs = \$310; and interest payments on debt = \$104 (all figures are in millions of dollars). Therefore,

$$CLE = \frac{1,500 - 920}{1,500 - 920 - 310 - 104} = \frac{580}{166} = 3.49$$

confirming our earlier finding.

In Appendix 17.3, we used Johnson & Johnson data to calculate DOL, DFL, and CLE, which are defined in this section.

The Trade-off between Business Risk and Financial Risk

Leverage is a two-edged sword. If stockholders know that a corporation could improve its operating performance, they would prefer a high degree of leverage. As we have just seen, a relatively small sales-growth rate can, through the combined effects of operating leverage and financial leverage, lead to a much larger proportionate increase in earnings per share. However, the economic climate in which corporations operate is too complex and unpredictable to allow such certainty in judgments. Sales could fall short of expectations, and quite possibly fall from earlier levels. In this case, leverage works against stockholders, and a small decrease in sales leads to a proportionately greater drop in earnings per share for the levered corporation. Therefore, in conjunction with leverage, it is also necessary to consider uncertainty or risk.

Just as there are two types of leverage, we must also examine two types of risk. As discussed earlier in this book, business risk describes uncertainty about future earnings before interest and taxes. Such uncertainty can arise for a number of reasons. First, it is impossible to forecast sales demand with complete precision, so that there will be some uncertainty about future sales volume. A related issue involves the prices at which a corporation is able to sell its products. In markets where there is intense competition among firms, competitors may react to slack demand by price-cutting, offering temporary discounts, providing generous loan terms, and other inducements to potential customers. To compete successfully, our firm will probably have to match its competitors' moves, which eats into profits. A further source of uncertainty arises because production costs cannot be predicted with certainty. Prices of raw materials used by a manufacturer can fluctuate dramatically over time.

These sources of uncertainty about business conditions must be considered in the context of operating leverage. We have seen that, if the business climate is favorable for our corporation, the higher the degree of operating leverage, the higher the expected net operating income. On the other hand, the higher the degree of operating leverage, all else equal, the greater the uncertainty about earnings before interest and taxes. The typical position is illustrated in Fig. 17.3, which shows probability distributions representing likely earnings before interest and taxes for two corporations. These firms are identical, except that one has a greater degree of operating leverage. The following points emerge from this graph:

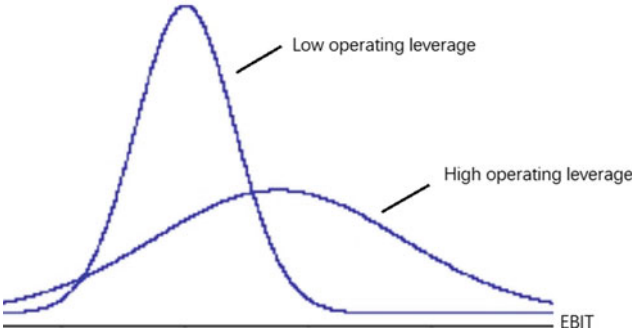


Fig. 17.3 Probability density functions for earnings before interest and taxes for firms with low and high degrees of operating leverage

1. The mean of the EBIT distribution for the firm with the higher degree of operating leverage is greater than that for the other firm. This reflects the increase in expected EBIT that can arise from operating leverage.
2. The variance of the EBIT distribution for the firm with the higher degree of operating leverage is greater than that for the firm with less leverage; that is, the former distribution is more widely dispersed about its mean than the latter distribution. This reflects the increase in business risk associated with a high degree of operating leverage.

Next, we consider financial risk. In the first section of this chapter, we saw that a high proportion of debt in a firm's capital structure can lead to higher expected earnings per share, but also to greater uncertainty about such earnings. This uncertainty is known as financial risk. Figure 17.4 shows the probability distributions of earnings per share for two corporations. The probability distributions of EBIT are the same for these two corporations, but one firm has a higher degree of financial leverage than the other. From this figure, we see the following:

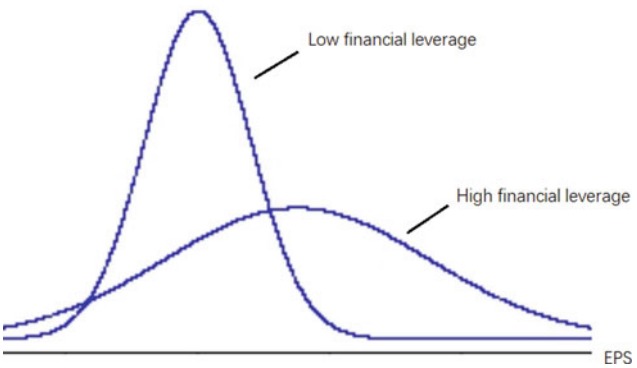


Fig. 17.4 Probability density functions for earnings per share for firms with low and high degrees of financial leverage

1. The mean of the EPS distribution for the firm with the higher degree of financial leverage exceeds the mean of the other firm. This reflects the potential for higher expected EPS resulting from financial leverage.
2. The variance of the EPS distribution is higher for the firm with the greater degree of financial leverage. This reflects the increase in financial risk resulting from financial leverage.

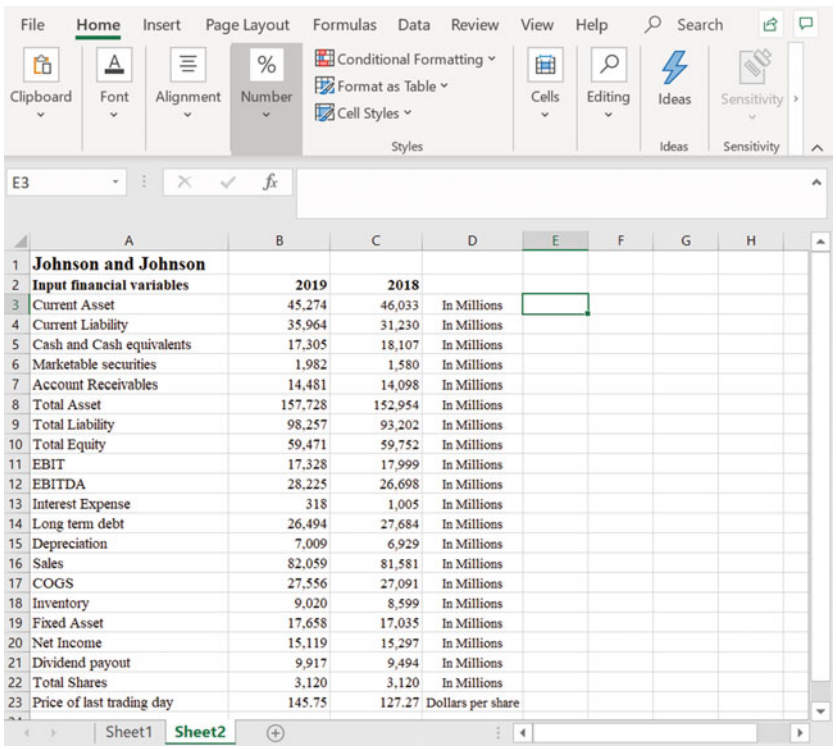
Thus, the overall risk faced by corporate stockholders is a combination of business risk and financial risk. We might think of the possibility of a trade-off between these two types of risk. Suppose that a firm operates in a risky business environment. Perhaps it trades in volatile markets and is highly capital-intensive, so that a large proportion of its costs are fixed. This riskiness will be exacerbated if the firm also has substantial debt, so that the firm has considerable financial risk. On the other hand, a low degree of financial leverage, and hence of financial risk, can mitigate the impact of high business risk on the overall riskiness of stockholders' equity. Management of a corporation subject to low business risk might feel more sanguine about taking on additional debt and thereby increasing financial risk.

17.6 Summary

This chapter reviews economic, financial, market, and accounting information to provide some environmental backgrounds to understand and apply sound financial management. Also covered are financial ratios, cost-volume-profit (CVP) analysis, break-even analysis, and degree of leverage (DOL) analysis. Financial ratios are an important tool by which managers and investors evaluate a firm's market value as well as understand the reasons for the fluctuations of the firm's market value. Factors that affect the industry in general and the firm in particular should be investigated. The best way to understand the common factors is to study economic information associated with the fluctuations or to look at the leading indicators. Accounting information, market information, and economic information are the three basic sources of data used in the financial decision-making process. In addition to analyzing the various types of information at one point in time and over time, the financial analyst is also interested in how the information changes over time. This area of study is known as *dynamic analysis and a detailed discussion can be found in Lee and* Lee (2017).

Appendix 17.1: Calculate 26 Financial Ratios with Excel

In this appendix, we use the data of 2018 and 2019 fisical year of Johnson & Johnson annual report as the example and show how to calculate the 26 basic financial ratios across five groups. The following figure lists 21 basic input variables from the Financial statements of fisical year 2019 and 2018. The colunm A is the name of the input variable. The column B shows the value of each variable in 2019 and column C shows that in 2018.



	A	B	C	D	E	F	G	H
1	**Johnson and Johnson**							
2	**Input financial variables**	**2019**	**2018**					
3	Current Asset	45,274	46,033	In Millions				
4	Current Liability	35,964	31,230	In Millions				
5	Cash and Cash equivalents	17,305	18,107	In Millions				
6	Marketable securities	1,982	1,580	In Millions				
7	Account Receivables	14,481	14,098	In Millions				
8	Total Asset	157,728	152,954	In Millions				
9	Total Liability	98,257	93,202	In Millions				
10	Total Equity	59,471	59,752	In Millions				
11	EBIT	17,328	17,999	In Millions				
12	EBITDA	28,225	26,698	In Millions				
13	Interest Expense	318	1,005	In Millions				
14	Long term debt	26,494	27,684	In Millions				
15	Depreciation	7,009	6,929	In Millions				
16	Sales	82,059	81,581	In Millions				
17	COGS	27,556	27,091	In Millions				
18	Inventory	9,020	8,599	In Millions				
19	Fixed Asset	17,658	17,035	In Millions				
20	Net Income	15,119	15,297	In Millions				
21	Dividend payout	9,917	9,494	In Millions				
22	Total Shares	3,120	3,120	In Millions				
23	Price of last trading day	145.75	127.27	Dollars per share				

Liquidity Ratio

First, we focus on the Liquidity ratio, which measures the relative strength of a firm's financial position. It usually includes current ratio, quick ratio, cash ratio, and networking capital to total asset ratio. The formula for each ratio is defined as follows:

$$\text{Current ratio (CR)} = \frac{\text{Current asset}}{\text{Current liability}}$$

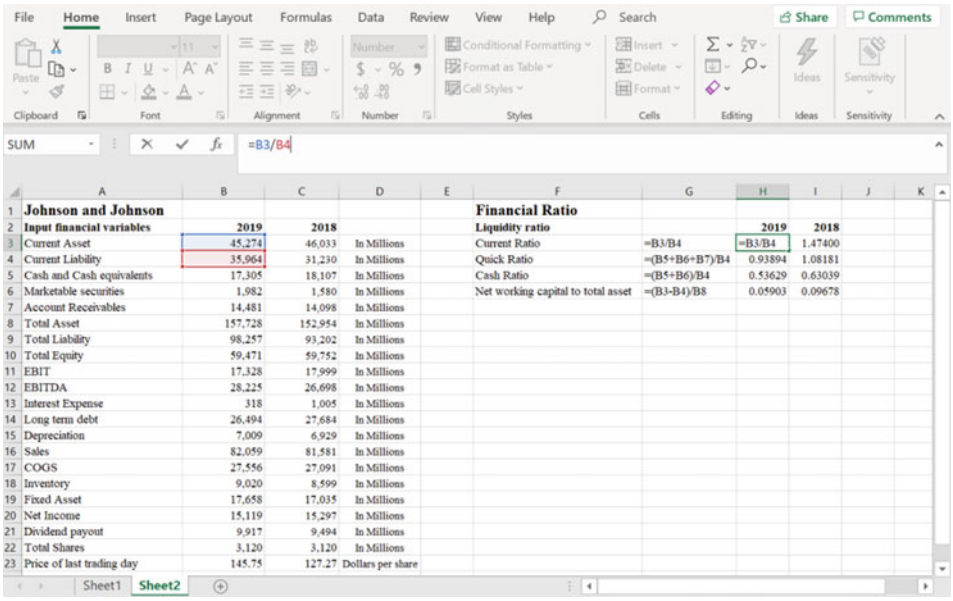
$$\text{Quick ratio} = \frac{\text{Cash} + \text{MS} + \text{receivables}}{\text{Current liability}}$$

$$\text{Cash ratio} = \frac{\text{Cash} + \text{MS}}{\text{Current liability}}$$

$$\text{Net working capital to total asset} = \frac{\text{Net working capital}}{\text{Current asset}}$$

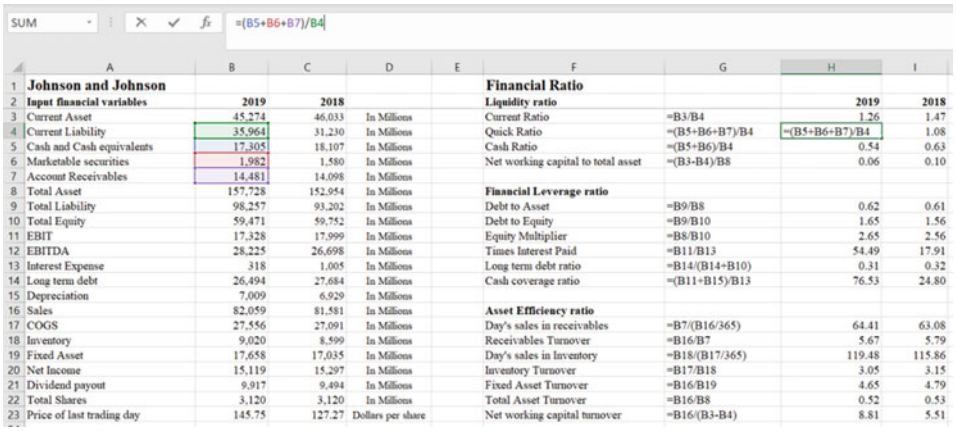
The following figure shows how to calculate these ratios based on the formulae with Excel.

To compute the Current ratio, we only need to find the cell in which the value of current asset locates (B3) and the cell in which the value of current liabilty belongs to (B4) and then find an empty cell to input "= B3/B4," which means divding current asset by current liability. The Excel will show the results "1.25887."

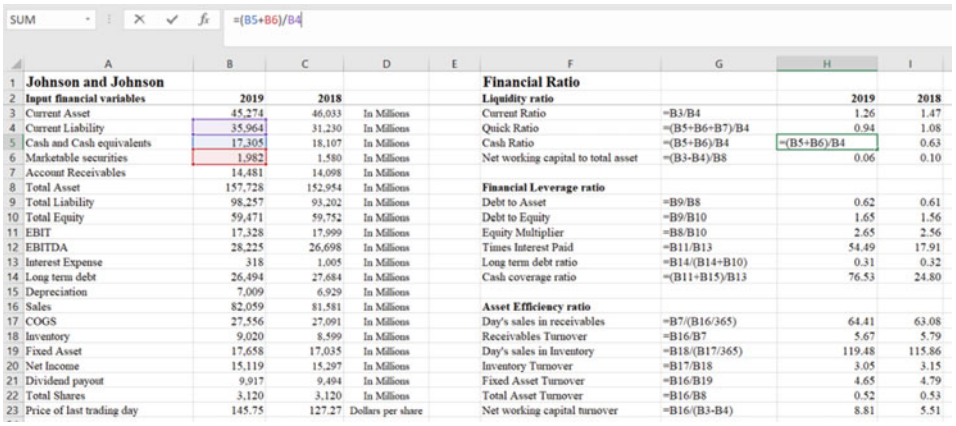


	A	B	C	D	E	F	G	H	I	J	K
1	**Johnson and Johnson**					**Financial Ratio**					
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**		
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	=B3/B4	1.47400		
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.93894	1.08181		
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.53629	0.63039		
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.05903	0.09678		
7	Account Receivables	14,481	14,098	In Millions							
8	Total Asset	157,728	152,954	In Millions							
9	Total Liability	98,257	93,202	In Millions							
10	Total Equity	59,471	59,752	In Millions							
11	EBIT	17,328	17,999	In Millions							
12	EBITDA	28,225	26,698	In Millions							
13	Interest Expense	318	1,005	In Millions							
14	Long term debt	26,494	27,684	In Millions							
15	Depreciation	7,009	6,929	In Millions							
16	Sales	82,059	81,581	In Millions							
17	COGS	27,556	27,091	In Millions							
18	Inventory	9,020	8,599	In Millions							
19	Fixed Asset	17,658	17,035	In Millions							
20	Net Income	15,119	15,297	In Millions							
21	Dividend payout	9,917	9,494	In Millions							
22	Total Shares	3,120	3,120	In Millions							
23	Price of last trading day	145.75	127.27	Dollars per share							

Similarly, we can compute the Quick ratio and Cash Ratio as the following two figures instruct. Compared with calculating the current ratio, the only difference for computing the Quick ratio or the Cash ratio is that different numerator is used. We have to use the sum of Cash and cash equivalent and Marketable securities [= (B5 + B6)] as the numerator in order to calculate the Cash ratio or use the sum of Cash and cash equivalent, Marketable securities and Accounting receivables[= (B5 + B6 + B7)] as the numerator in order to calculate the Quick ratio.

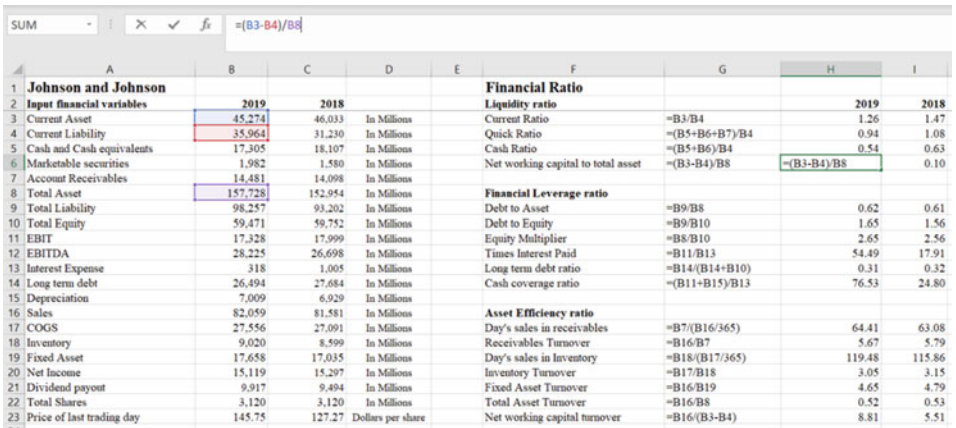


	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.26	1.47
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	=(B5+B6+B7)/B4	1.08
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.54	0.63
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.06	0.10
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.62	0.61
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.65	1.56
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.65	2.56
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.49	17.91
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.31	0.32
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.53	24.80
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.26	1.47
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.94	1.08
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	=(B5+B6)/B4	0.63
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.06	0.10
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.62	0.61
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.65	1.56
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.65	2.56
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.49	17.91
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.31	0.32
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.53	24.80
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51

For the net working capital to total asset ratio, we firstly need to calculate "Net working capital" and then divide it by current asset. As net working capital is defined as "Current asset minus current liability," we compute this ratio by inputting "= (B3 − B4)/B8," which gives us 0.06 in the figure below.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.26	1.47
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.94	1.08
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.54	0.63
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	=(B3-B4)/B8	0.10
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.62	0.61
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.65	1.56
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.65	2.56
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.49	17.91
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.31	0.32
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.53	24.80
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51

Financial Leverage Ratio

In this section, we compute the financial leverage ratios, which reflect the financial risk posture of a firm, with Excel. There are six ratios that are commonly used in financial analysis.

$$\text{Debt to Asset} = \frac{\text{Total liability}}{\text{Total asset}}$$

$$\text{Debt to Equity} = \frac{\text{Total liability}}{\text{Total equity}}$$

$$\text{Equity Multiplier} = \frac{\text{Total asset}}{\text{Total equity}}$$

$$\text{Times interest paid} = \frac{\text{EBIT}}{\text{Interest expense}}$$

$$\text{Long} - \text{term debt ratio} = \frac{\text{long} - \text{term debt}}{\text{long} - \text{term debt} + \text{Total equity}}$$

$$\text{Cash coverage ratio} = \frac{\text{EBIT} + \text{Depreciation}}{\text{Interest expense}}$$

For the first four ratios, their calculations are quite simple. We input "= B9/B8" to get 0.6230 for the Debt to Asset ratio, "= B9/B10" to get 1.6522 for the Debt to Equity ratio, "= B8/B10" to get 2.6522 for the Equity Multiplier, "= B11/B13" to get 54.4906 for the Times interested paid.

The following figure shows how to calculate the long-term debt ratio. We input "= B14/(B14 + B10)" in an empty cell, where (B14 + B10) equals the sum of long-term debt and total equity. Excel gives us 0.3082.

SUM =B14/(B14+B10)

	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	=B14/(B14+B10)	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51

Similarly, the Cash coverage ratio can be computed based on the formula by inputting "= (B11 + B15)/B13." Then we obtain 76.5314 as the value of this ratio.

SUM =(B11+B15)/B13

	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	=(B11+B15)/B13	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51

Asset Efficiency Ratios

These ratios mainly reflect how a firm is utilizing its asset. We list 7 common ratios used in financial analysis. They are Day's sales in receivables, Receivables Turnover, Day's sales in Inventory, Inventory Turnover, Fixed Asset Turnover, Total Asset Turnover, and Net working capital turnover.

$$\text{Day's sales in receivables} = \frac{\text{Account Receivable}}{\text{Sale}/365}$$

$$\text{Receivables Turnover} = \frac{\text{Sales}}{\text{Account Receivable}}$$

$$\text{Day's sales in Inventory} = \frac{\text{Inventory}}{\text{COGS}/365}$$

$$\text{Inventory Turnover} = \frac{\text{COGS}}{\text{Inventory}}$$

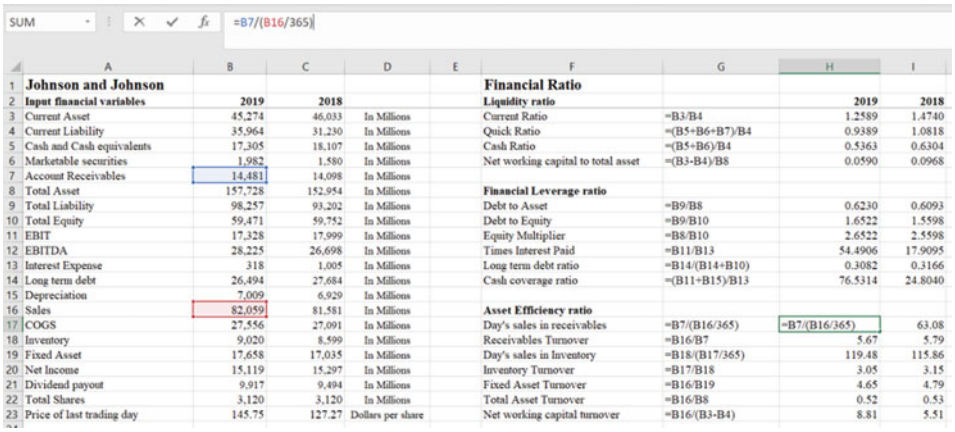
$$\text{Fixed Asset Turnover} = \frac{\text{Sales}}{\text{Fixed Assets}}$$

$$\text{Total Asset Turnover} = \frac{\text{Sales}}{\text{Total Assets}}$$

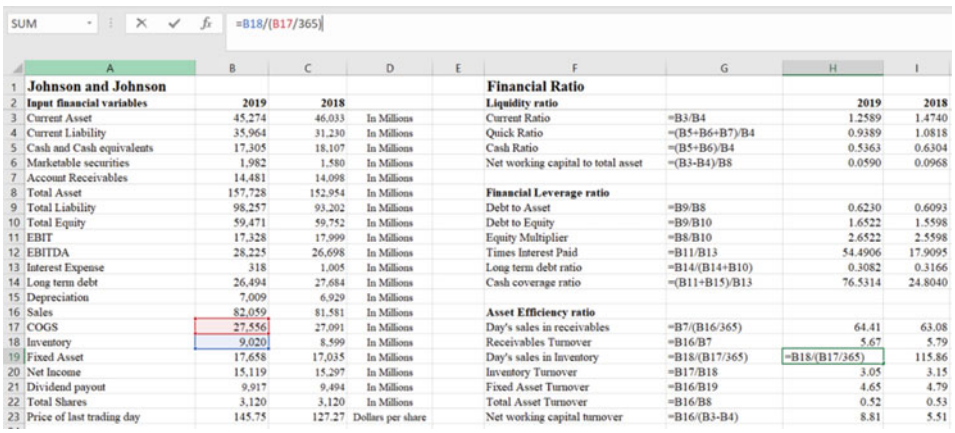
$$\text{Net Working capital Turnover} = \frac{\text{Sales}}{\text{Net Working capital}}$$

It is very simple to compute Receivable turnover by inputting "B16/B7," to calculate Inventory Turnover by inputting "= B17/B18," to obtain Fixed Asset Turnover via inputting "= B16/B19" and to get Total Asset Turnover via inputting "= B16/B8." Excel will show all these values.

The following two figures shows that we calculate the Day's sales in Receivables by inputting "= B7/(B16/365)" and that we calculate the Day's sales in Inventory by inputting "= B18/(B17/365)." The key point here is to add a bracket to the denominator when we calculate "Sales/365."

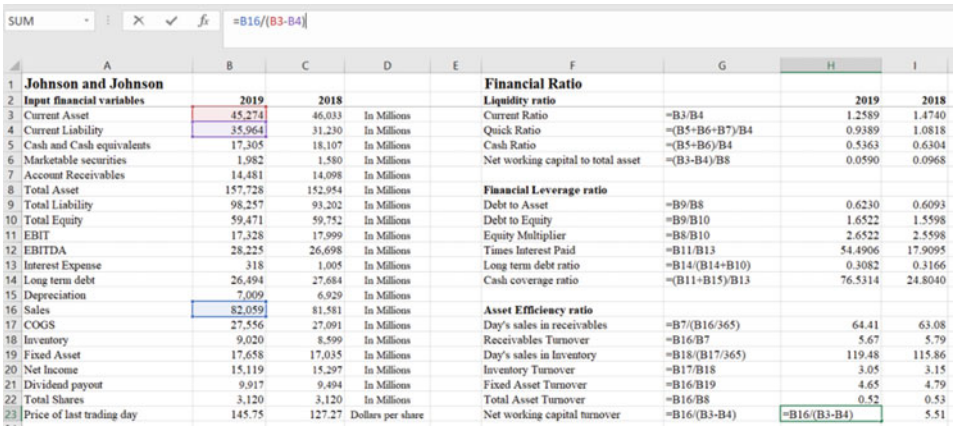


	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	=B7/(B16/365)	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	=B18/(B17/365)	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51

In order to calculate the Net Working capital Turnover, we input "= B16/(B3 − B4)" since "B3 − B4" equals to the working capital of JNJ in 2019. Excel shows the final value of 8.81.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	=B16/(B3-B4)	5.51

Profitability Ratios

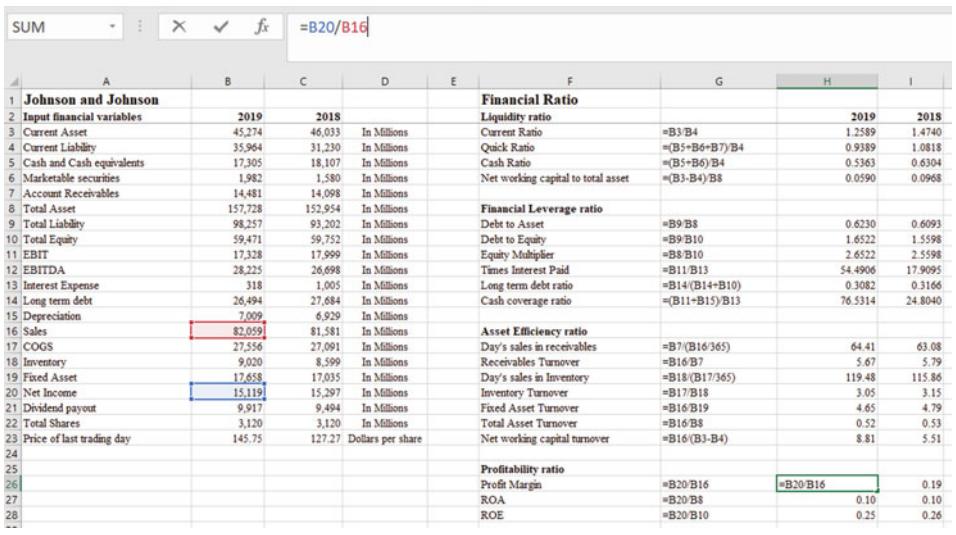
These ratios reflect the profitability of a firm's operations. Profit Margin, Return on Asset, and Return on Equity are widely used in empirical research.

$$\text{Profit Margin} = \frac{\text{Net Income}}{\text{Sales}}$$

$$\text{Return on Equity} = \frac{\text{Net Income}}{\text{Total equity}}$$

$$\text{Return on Asset} = \frac{\text{Net Income}}{\text{Total asset}}$$

Similar to the skills used before, we only need to divide one variable (X1) by another one (X2) with inputting "= X1/X2" to obtain the ratios. The figure below gives an example of how to calculate the Profit Margin (0.18). ROA and ROE can be obtained in a similar way.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	=B20/B16	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26

Market Value Ratios

The last group includes market value ratios, which indicate an assessment of value of a firm's stock. We calculate six ratios in this section.

$$\text{Price} - \text{earnings ratio(PE)} = \frac{\text{Price per share}}{\text{Earnings per share}}$$

$$\text{Market} - \text{Book ratio(MB)} = \frac{\text{Price per share}}{\text{Book value per share}}$$

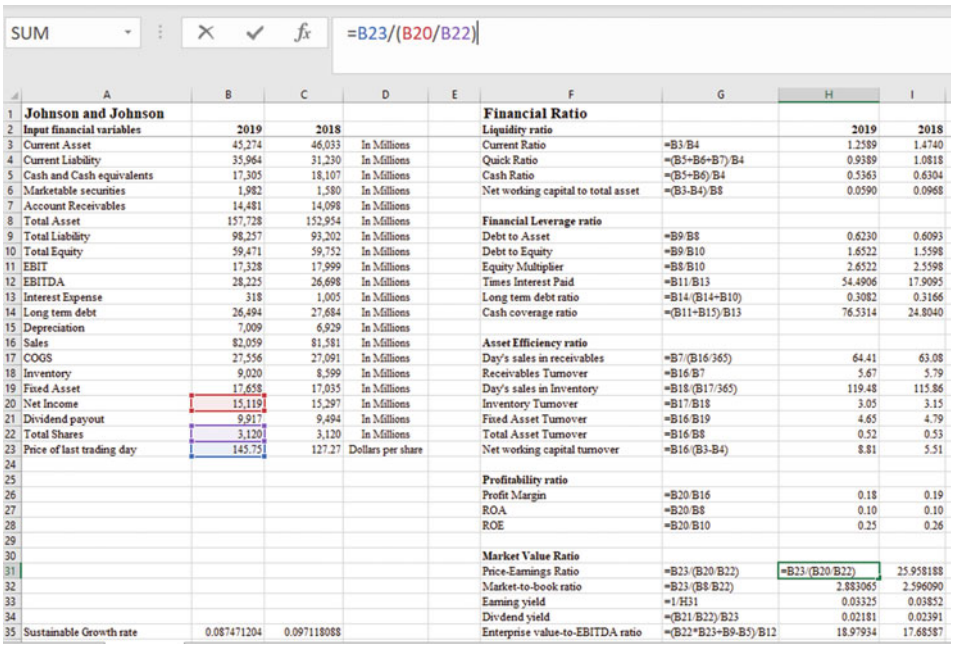
$$\text{Earnings yield} = \frac{\text{Earnings per share}}{\text{Price per share}}$$

$$\text{Dividend yield} = \frac{\text{Dividend per share}}{\text{Price per share}}$$

$$\text{PEG ratio} = \frac{\text{PE}}{\text{Earnings growth rate}}$$

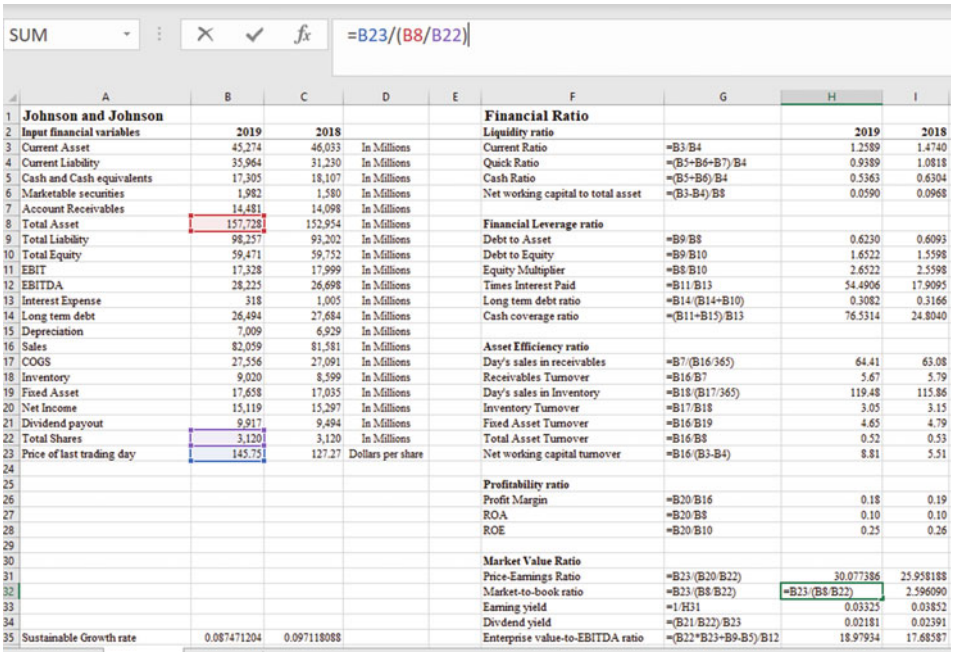
$$\text{Enterprise} - \text{EBDTA ratio} = \frac{\text{Enterprise value}}{\text{EBITDA}}$$

The following two figures show how to compute PE ratio and MB ratio. Since the price per share is input into cell B23, we only need to find EPS or book value per share. According to the definition of EPS, it is computed via dividing net income by total shares (= B20/B22). Similarly, book value per share can be obtained by inputting "= B8/B22." In order to calculate PE ratio or MB ratio in one-step, we directly input "= B23/(B20/B22)" or "= B23/(B8/B22)," respectively. The values are 30.0774 and 2.8831, respectively.



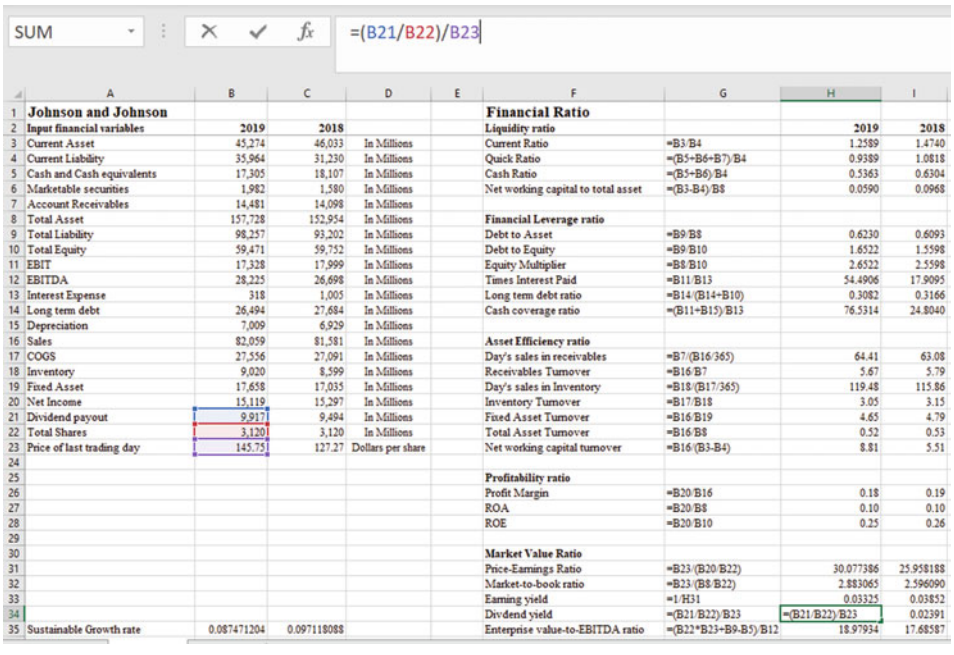
SUM | =B23/(B20/B22)

	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	=B23/(B20/B22)	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	2.883065	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	0.02181	0.02391
35	Sustainable Growth rate	0.087471204	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B12	18.97934	17.68587



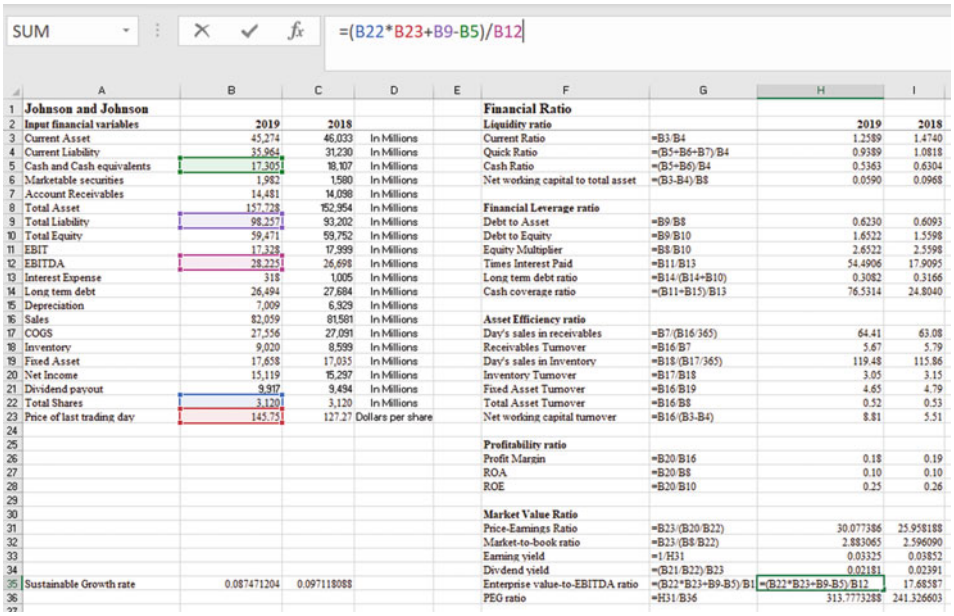
	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	30.077386	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	=B23/(B8/B22)	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	0.02181	0.02391
35	Sustainable Growth rate	0.087471204	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B12	18.97934	17.68587

Additionally, the Earnings yield is simply the reciprocal of PE so that we get it (1/30.0774 = 0.03325), and Dividend yield can be computed via inputting "= (B21/B22)/B23" and equals 0.0218. The following figure shows the result.



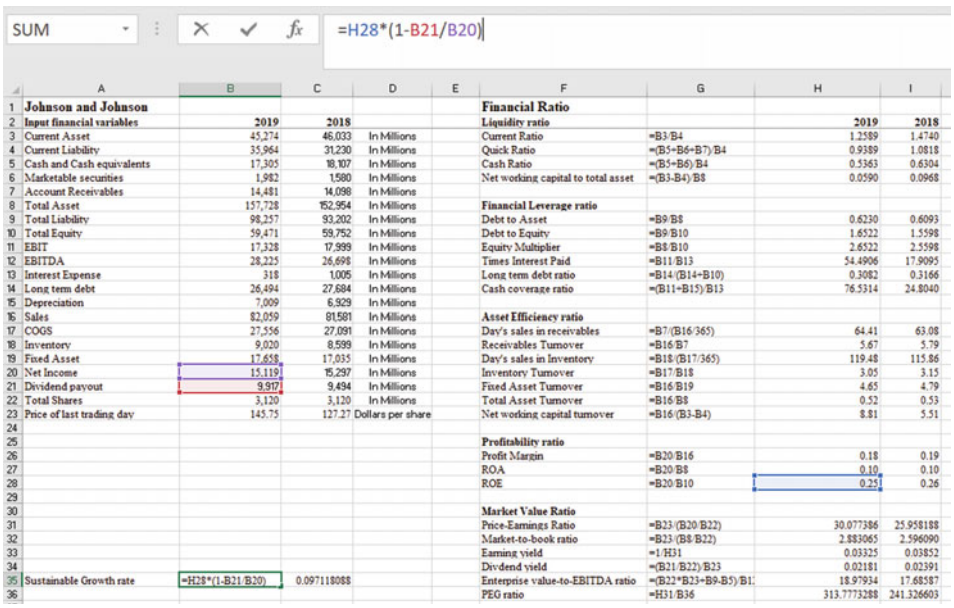
	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	30.077386	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	2.883065	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	=(B21/B22)/B23	0.02391
35	Sustainable Growth rate	0.087471204	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B12	18.97934	17.68587

For enterprise-EBITDA ratio, we firstly calculate enterprise value on the numerator according to the definition "Total market value of equity + Book value of Total Liability-Cash" and then input "= B22*B23 + B9 − B5" into an empty cell. Next, we divide enterprise value by EBITDA. So the one-step formula is "= (B22*B23 + B9 − B5)/B12." Excel gives us the value of 18.9793.



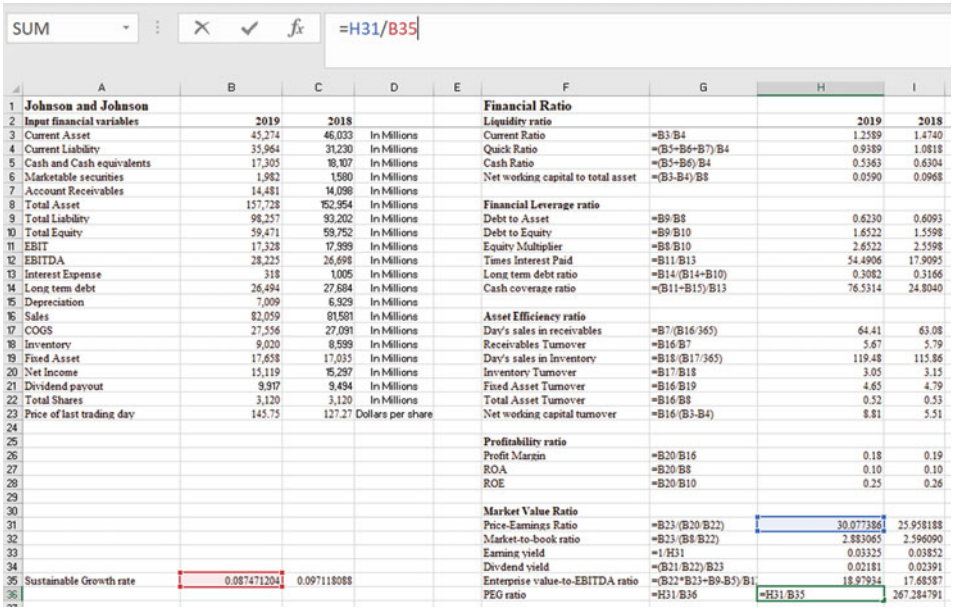
	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	30.077386	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	2.883065	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	0.02181	0.02391
35	Sustainable Growth rate	0.087471204	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B1	=(B22*B23+B9-B5)/B12	17.68587
36						PEG ratio	=H31/B36	313.7773288	241.326603

The last ratio is PEG ratio, which equals to PE ratio divided by sustainable growth rate. Since we already have PE ratio, we only need to find the value of sustainable growth rate. Based on the formula: sustainable growth rate = ROE*(1 − dividend payout ratio), we input sustainable growth rate = H28*(1 − B21/B20) in the cell B35 to get the value of sustainable growth rate (0.0875). The figure below shows the result.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	30.077386	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	2.883065	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	0.02181	0.02391
35	Sustainable Growth rate	=H28*(1-B21/B20)	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B1	18.97934	17.68587
36						PEG ratio	=H31/B36	313.7773288	241.326603

Therefore, we get PEG ratio by inputting "= H31/B35," which equals to 343.8547. The result is as follows.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26						Profit Margin	=B20/B16	0.18	0.19
27						ROA	=B20/B8	0.10	0.10
28						ROE	=B20/B10	0.25	0.26
29									
30						**Market Value Ratio**			
31						Price-Earnings Ratio	=B23/(B20/B22)	30.077386	25.958188
32						Market-to-book ratio	=B23/(B8/B22)	2.883065	2.596090
33						Earning yield	=1/H31	0.03325	0.03852
34						Divdend yield	=(B21/B22)/B23	0.02181	0.02391
35	Sustainable Growth rate	0.087471204	0.097118088			Enterprise value-to-EBITDA ratio	=(B22*B23+B9-B5)/B1	18.97934	17.68587
36						PEG ratio	=H31/B36	=H31/B35	267.284791

Appendix 17.2: Using Excel to Calculate Sustainable Growth Rate

Sustainable growth rate (SGR) can be either estimated by (i) using both external and internal source of fund or (ii) using only internal source of fund.

We present these two methods in detail as follows:

Method 1: The sustainable growth rate with both external and internal source of fund can be defined as (Lee 2017):

$$SGR = \frac{\text{Retention Rate*ROE}}{1-(\text{Retention Rate*ROE})} = \frac{(1-\text{Dividend Payout Ratio})\text{*ROE}}{1-[(1-\text{Dividend Payout Ratio})*ROE]} \quad (17A.1)$$

$$\text{Dividend Payout Ratio} = \text{Dividends/Net Income}$$

Method 2: The sustainable growth rate: considering the internal source of fund

$$\begin{aligned}\text{ROE} &= \text{Net Income/Total Equity}\\ \text{ROE} &= (\text{Net Income/Assets}) \times (\text{Assets/Equity})\\ \text{ROE} &= (\text{Net Income/Sales}) \times (\text{Sales/Assets}) \times (\text{Assets/Equity})\\ \text{SGR} &= (\text{Net Income/Sales}) \times (\text{Retention Rate}) \times (\text{Sales/Assets}) \times (\text{Assets/Equity})\\ &= ROE \times (1 - \textit{Dividend Payout Ratio}) \end{aligned} \quad (17A.2)$$

Example:

With the data from JNJ financial statement for the 2019 fiscal year, we estimate obtain

$$\begin{aligned}\text{ROE} &= \text{Net Income/Total Equity} = 15,119/59,471 = 0.2542\\ \text{Dividend Payout Ratio} &= \text{Dividends/Net Income} = 9,917/15,119 = 0.6559\end{aligned}$$

$$\begin{aligned}\text{According to the method 1, SGR} &= (1-0.6559)*0.2542/1 - [(1-0.6559)*0.2542] = 0.0959\\ \text{According to the method 2, SGR} &= 0.2542*(1-0.6559) = 0.0875\end{aligned}$$

The difference between method 1 and method 2

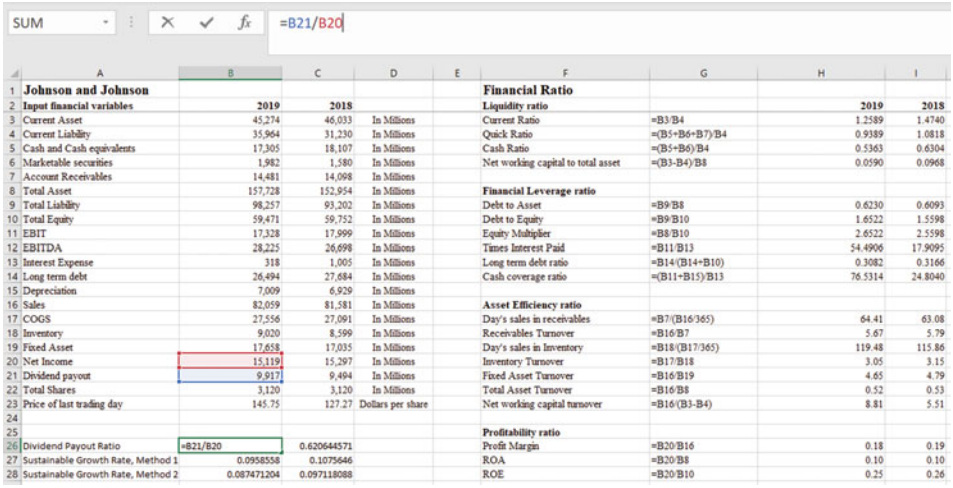
Technically, as $ROE \times (1-D)$ is the numerator of $\frac{ROE\times(1-D)}{1-ROE\times(1-D)}$ and $1 > [1-ROE\times(1-D)] \geq 0$, it is easy to prove $\frac{ROE\times(1-D)}{1-ROE\times(1-D)} \geq ROE \times (1-D)$.

In addition, we can transform $\frac{ROE\times(1-D)}{1-ROE\times(1-D)}$ into $\frac{\text{Retained Earnings}}{\text{Equity}-\text{Retained Earnings}}$ and transform $ROE \times (1-D)$ into $\frac{\text{Retained Earnings}}{\text{Equity}}$. It is obvious to see $\frac{\text{Retained Earnings}}{\text{Equity}-\text{Retained Earnings}} \geq \frac{\text{Retained Earnings}}{\text{Equity}}$ since $\text{Equity} - \text{Retained Earnings} \leq \text{Equity}$. If we use equity value at the end of this year, then $(\text{Equity} - \text{Retained Earnings})$ can be interpreted as the equity value at the beginning of this year under the condition of no external finance.

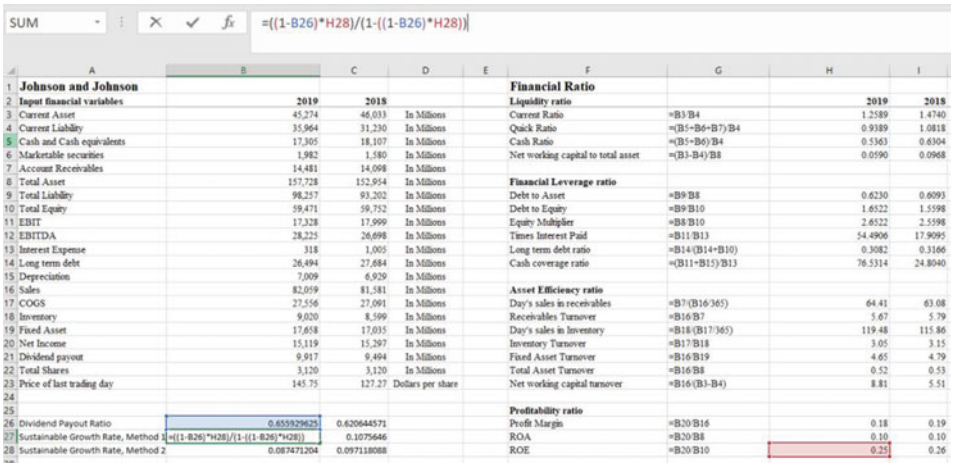
Consequently, the SGR from method 1 is usually greater than that from method 2. The numerical result $0.0959 > 0.0875$ confirms this.

How to calculate SGR with two methods with Excel

First, we calculate the dividend payout ratio by inputting "= B21/B20." We compute the SGR with method 1 by inputting = ((1 – B26)*H28)/(1 – ((1 – B26)*H28))" and then obtain 0.0958558 and with method 2 by inputting = H28* (1 – B26)" and then obtain 0.087471204. The following figures show the calculation.



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26	Dividend Payout Ratio	=B21/B20	0.620644571			Profit Margin	=B20/B16	0.18	0.19
27	Sustainable Growth Rate, Method 1	0.0958558	0.1075646			ROA	=B20/B8	0.10	0.10
28	Sustainable Growth Rate, Method 2	0.087471204	0.097118088			ROE	=B20/B10	0.25	0.26



	A	B	C	D	E	F	G	H	I
1	**Johnson and Johnson**					**Financial Ratio**			
2	**Input financial variables**	**2019**	**2018**			**Liquidity ratio**		**2019**	**2018**
3	Current Asset	45,274	46,033	In Millions		Current Ratio	=B3/B4	1.2589	1.4740
4	Current Liability	35,964	31,230	In Millions		Quick Ratio	=(B5+B6+B7)/B4	0.9389	1.0818
5	Cash and Cash equivalents	17,305	18,107	In Millions		Cash Ratio	=(B5+B6)/B4	0.5363	0.6304
6	Marketable securities	1,982	1,580	In Millions		Net working capital to total asset	=(B3-B4)/B8	0.0590	0.0968
7	Account Receivables	14,481	14,098	In Millions					
8	Total Asset	157,728	152,954	In Millions		**Financial Leverage ratio**			
9	Total Liability	98,257	93,202	In Millions		Debt to Asset	=B9/B8	0.6230	0.6093
10	Total Equity	59,471	59,752	In Millions		Debt to Equity	=B9/B10	1.6522	1.5598
11	EBIT	17,328	17,999	In Millions		Equity Multiplier	=B8/B10	2.6522	2.5598
12	EBITDA	28,225	26,698	In Millions		Times Interest Paid	=B11/B13	54.4906	17.9095
13	Interest Expense	318	1,005	In Millions		Long term debt ratio	=B14/(B14+B10)	0.3082	0.3166
14	Long term debt	26,494	27,684	In Millions		Cash coverage ratio	=(B11+B15)/B13	76.5314	24.8040
15	Depreciation	7,009	6,929	In Millions					
16	Sales	82,059	81,581	In Millions		**Asset Efficiency ratio**			
17	COGS	27,556	27,091	In Millions		Day's sales in receivables	=B7/(B16/365)	64.41	63.08
18	Inventory	9,020	8,599	In Millions		Receivables Turnover	=B16/B7	5.67	5.79
19	Fixed Asset	17,658	17,035	In Millions		Day's sales in Inventory	=B18/(B17/365)	119.48	115.86
20	Net Income	15,119	15,297	In Millions		Inventory Turnover	=B17/B18	3.05	3.15
21	Dividend payout	9,917	9,494	In Millions		Fixed Asset Turnover	=B16/B19	4.65	4.79
22	Total Shares	3,120	3,120	In Millions		Total Asset Turnover	=B16/B8	0.52	0.53
23	Price of last trading day	145.75	127.27	Dollars per share		Net working capital turnover	=B16/(B3-B4)	8.81	5.51
24									
25						**Profitability ratio**			
26	Dividend Payout Ratio	0.655929625	0.620644571			Profit Margin	=B20/B16	0.18	0.19
27	Sustainable Growth Rate, Method 1	=((1-B26)*H28)/(1-((1-B26)*H28))	0.1075646			ROA	=B20/B8	0.10	0.10
28	Sustainable Growth Rate, Method 2	0.087471204	0.097118088			ROE	=B20/B10	0.25	0.26

Appendix 17.3: How to Compute DOL, DFL, and DCL with Excel

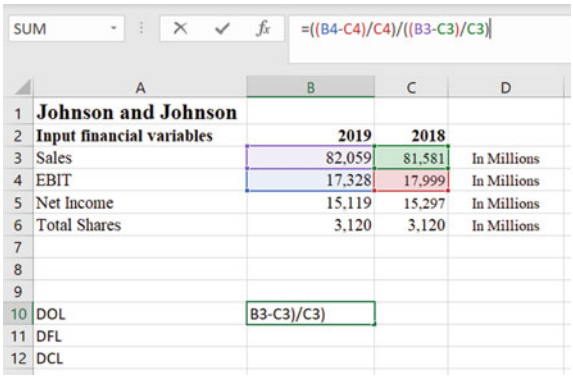
In this appendix, we first define the definitions of DOL, DFL, and DCL in terms of elasticity definition, then we show how Excel program can be used to calculate these three variables in terms of financial statement data. In Chap. 11, we will theoretically and empirically discuss these three variables in further detail.

How Excel program can be used to calculate these three variables in terms of financial statement data.

1. The definition of degree of operating leverage is:

$$DOL = \frac{\%change\ in\ EBIT}{\%change\ in\ Sale} \tag{17.12}$$

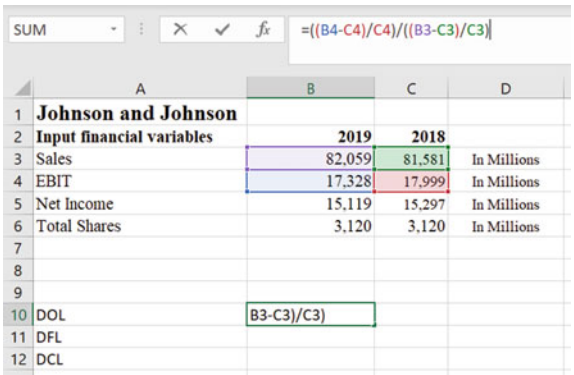
To calculate the degree of operating leverage, we firstly compute the percentage change in EBIT by inputting "(B4 – C4)/C4." Then, compute the percentage change in Sales by inputting "(B3 – C3)/C3." Put them together, we input "= ((B4 – C4)/C4)/((B3 – C3)/C3)" to get DOL = −6.3626.

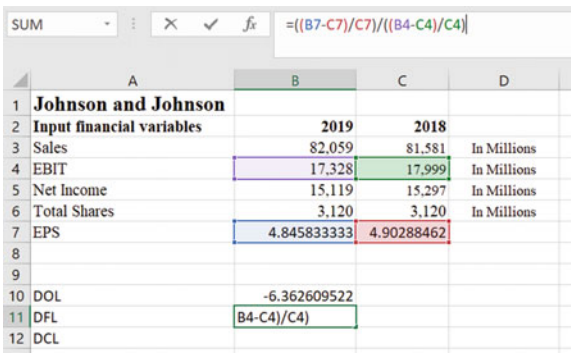


2. The definition of degree of financial leverage is:

$$DFL = \frac{\%change\ in\ EPS}{\%change\ in\ EBIT} \tag{17.13}$$

To calculate the degree of operating leverage, we firstly compute EPS (Net income/Total shares) by inputting "= B5/B6." And then we compute the percentage change in EPS by inputting "(B7 – C7)/C7," which is the percentage change in EPS. Then, compute the percentage change in EBIT in inputting "(B4 – C4)/C4." Put them together, we input "= ((B7 – C7)/C7)/((B4 – C4)/C4)," to get DFL = 0.312132932.



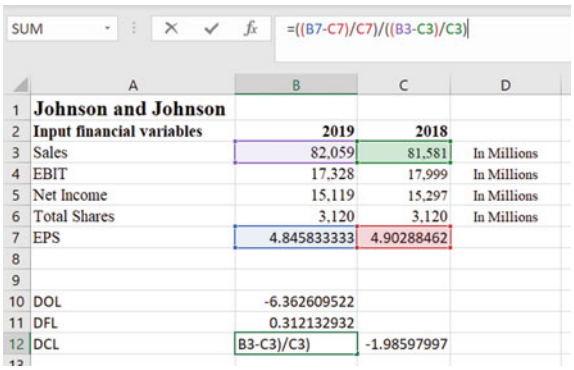


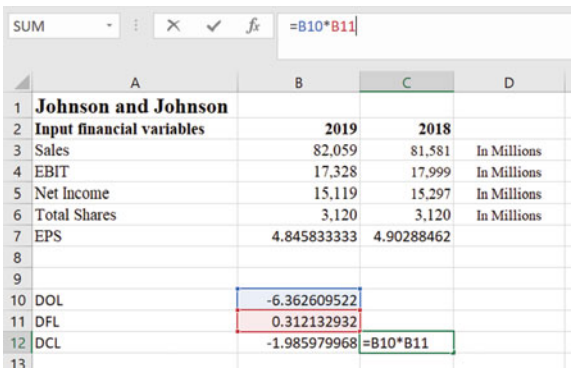
3. The definition of degree of combined leverage is

$$DCL = \frac{\%change\ in\ EPS}{\%change\ in\ EBIT} * \frac{\%change\ in\ EBIT}{\%change\ in\ Sale} = \frac{\%change\ in\ EPS}{\%change\ in\ Sale} \tag{17.14}$$

To calculate the degree of combined leverage, we firstly compute the percentage change in EPS by inputting (B7 – C7)/C7," which is the percentage change in EPS. Then, compute the percentage change in sale in inputting "(B3 – C3)/C3." Put them together, we input = ((B7 – C7)/C7)/((B3 – C3)/C3)," to get DFL = −1.986.

Alternatively, we can also input "= B10*B11" to get the same result since DCL = DFL*DOL = −1.986.





Questions and Problems

1. Define the following terms:
   a. Real versus financial market
   b. M_1 and M_2
   c. Leading economic indicators
   d. NYSE, AMEX, and OTC
   e. Primary versus the secondary stock market
   f. Bond market
   g. Options and futures markets

2. Briefly discuss the definition of liquidity, asset management, capital structure, profitability, and market value ratio. What can we learn from examining the financial ratio information of GM in 1984 and 1985 as listed in Table 17.6?
3. Discuss the major difference between the linear and nonlinear break-even analysis.
4. ABC Company's financial records are as follows:
   Quantity of goods sold = 10,000
   Price per unit sold = $20
   Variable cost per unit sold = $10
   Total amount of fixed cost = $50,000
   Corporate tax rate = 50%
   a. Calculate EAIT.
   b. What is the break-even quantity?
   c. What is the DOL?
   d. Should the ABC Company produce more for greater profits?
5. ABC Company's predictions for next year are as follows:

	Probability	Quantity	Price	Variable cost/unit	Corporate tax rate
State 1	0.3	1,000	$10	$5	.5
State 2	0.4	2,000	$20	$10	.5
State 3	0.3	3,000	$30	$15	.5

In addition, we also know that the fixed cost is $15,000. What is the next year's expected EAIT?

6. Use an example to discuss four alternative depreciation methods.
7. XYX, Inc. currently produces one product that sells for $330 per unit. The company's fixed costs are $80,000 per year; variable costs are $210 per unit. A salesman has offered to sell the company a new piece of equipment which will increase fixed costs to $100,000. The salesman claims that the company's break-even number of units sold will not be altered if the company purchases the equipment and raises its price (assuming variable costs remain the same).
   a. Find the company's current break-even level of units sold.
   b. Find the company's new price if the equipment is purchased and prove that the break-even level has not changed.
8. Consider the following financial data of a corporation:
   Sales = $500,000
   Quantity = 25,000
   Variable cost = $300,000
   Fixed cost = $50,000
   a. Calculate the DOL at the above quantity of output.
   b. Find the break-even quantity and sales levels.
9. On the basis of the following firm and industry norm ratios, identify the problem that exists for the firm:

Ratio	Firm	Industry
Total asset utilization	2.0	3.5
Average collection period	45 days	46 days
Inventory turnover	6 times	6 times
Fixed asset utilization	4.5	7.0

10. The financial ratios for Wallace, Inc., a manufacturer of consumer household products, are given below along with the industry norm:

Ratio		Firm		Industry
	1986	1987	1988	
Current ratio	1.44	1.31	1.47	1.2
Quick ratio	.66	.62	.65	.63
Average collection period	33 days	37 days	32 days	34 days
Inventory turnover	7.86	7.62	7.72	7.6
Fixed asset turnover	2.60	2.44	2.56	2.8
Total asset utilization	1.24	1.18	1.40	1.20
Debt to total equity	1.24	1.14	.84	1.00
Debt to total assets	.56	.54	.46	.50
Times interest earned	2.75	5.57	7.08	5.00
Return on total assets	.02	.06	.07	.06
Return on equity	.06	.12	.12	.13
Net profit margin	.02	.05	.05	.05

Analyze Wallace's ratios over the three-year period for each of the following categories:

   a. Liquidity
   b. Asset utilization
   c. Financial leverage
   d. Profitability

11. Below are the Balance Sheet and the Income Statement for Nelson Manufacturing:

Balance Sheet for Nelson on 12/31/88

Assets	
Cash and marketable securities	$ 125,000
Accounts receivable	239,000
Inventories	225,000
Prepaid expenses	11,000
Total Current Assets	$ 600,000
Fixed assets (net)	40,000
Total Assets	$1,000,000
Liabilities and Stockholder's Equity	
Accounts payable	$ 62,000
Accruals	188,000
Long-term debt maturing in 1 year	8,000
	$ 258,000
Long-term debt	221,000
Total Liabilities	$ 479,000
Stockholder's Equity	
Preferred stock	5,000
Common stock (at par)	175,000
Retained earnings	341,000
Total Stockholder's Equity	$ 521,000
Total Liabilities and Shareholder's Equity	$1,000,000

Income Statement for Nelson for Year Ending 12/31/88

Net sales	$800,000
Less: Cost of goods sold	381,600
Selling, general, and administrative expense	216,800
Interest Expense	20,000
Earnings before taxes	$181,200
Less: Tax expense (40 percent)	72,480
Net income	$108,720

a. Calculate the following ratios for Nelson.

	Nelson	Industry
(1) Current ratio		3.40
(2) Quick ratio		2.43
(3) Average collection period		88.65
(4) Inventory turnover		6.46
(5) Fixed asset turnover		4.41
(6) Total asset utilization		1.12
(7) Debt to total equity		.34
(8) Debt to total assets		5.25
(9) Times interest earned		12.00
(10) Return on total assets		.12
(11) Return on equity		.18
(12) Net profit margin		.12

b. Identify Nelson's strengths and weaknesses relative to the industry norm.

References

Johnson & Johnson 2016, 2017, 2018, and 2019 Annual Reports.
Lee, C. F. & John Lee *Financial Analysis and Planning: Theory and Application* (Singapore: World Scientific, 2017).

Time Value of Money Determinations and Their Applications

18

18.1 Introduction

The concepts of present value, discounting, and compounding are frequently used in most types of financial analysis. This chapter discusses the concepts of the time value of money and the mechanics of using various forms of the present value model. These ideas provide a foundation that is used throughout this book.

The first two sections of this chapter introduce the basic concept of the present value model. Section 18.2 discusses the basic concepts of present values, and Sect. 18.3 discusses the foundation of net present value rules. Section 18.4 covers the compounding and discounting processes. Section 18.5 covers the use of present and future value tables, Sect. 18.6 discusses present values are basic tools for financial management decisions, and Sect. 18.7 discusses the net present value and internal rate of return. Finally, a chapter summary is offered in Sect. 18.8. Three hypotheses about inflation and the firm's value are given in Appendix 18A, book value, replacement cost, and Tobin's q are discussed in Appendix 18B, Appendix 18C discusses continuous compounding, Appendix 18D discusses applications of Excel for calculating time value of money, and Appendix 18E presents four time value of money tables.

18.2 Basic Concepts of Present Values

Suppose that we offered to give you either $1,000 today or $1,000 one year from today; which would you prefer? Surely you would choose the former! Even if you were in the happy position of having no immediate unfulfilled desires, you could invest $1,000 today and, at no risk, obtain an amount in excess of $1,000 in one year's time. For example, you could purchase government securities with maturity in one year. Suppose that the annual interest rate on such securities is 8%, then $1,000 today would be worth $1,080 a year from today.

This simple example illustrates a significant fact motivating our analysis in this chapter. Put simply, a dollar today is worth more than a dollar at some time in the future. There are two basic reasons for this: (1) human nature being what it is, immediate gratification has a higher value than gratification sometime in the future, and (2) inflation erodes the purchasing power of an individual's dollar the longer it is held in the form of cash. Therefore, we say that money has a *time value*. The time value is reflected in the interest rate that one earns or pays to have the right to use money at various points in time. Even in the absence of inflation, money has time value as long as it has an alternative use that pays a positive interest rate. When an author signs a contract with a publisher, one important element of the contract involves payment to the author of an advance on royalties. When the book is published and the royalties become due, the amount of the advance is subtracted from the royalties. Nevertheless, because of the preference to have the money sooner rather than later, authors will negotiate, all other things being equal, for as large an advance as possible. Conversely, of course, publishers prefer to keep the advance payments to authors as low as possible.

We prefer to have $1,000 today rather than in the future because interest rates are positive. Why is interest paid on loans? There are two related rationales, even in the absence of inflation. These are the *liquidity preference* and the *time preference* theories. The liquidity preference theory asserts that rational people prefer assets with higher liquidity to assets with lower liquidity. Since cash is the most liquid asset of all, we can view interest payments as providing compensation to the lender for the sacrifice of some liquidity. The time preference theory asserts that people prefer current consumption to the same real level of consumption in the future and will sacrifice current consumption only in the expectation of being able to achieve, through interest payments, higher future consumption levels. Lenders view interest as a payment to induce consumers to give up the current use of their funds for a certain period of time.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_18

Borrowers view interest as a payment or rental fee for the privilege of being able to have the immediate use of cash that they otherwise would have to save over time.

We answered the question posed at the beginning of this section by noting that if the risk-free annual interest is 8%, then \$1,000 today will be worth \$1,080 a year from now. This is calculated as follows:

$$(1 + .08)1{,}000 = \$1{,}080$$

We can turn this statement around to determine the value today of \$1,000 received a year from now; that is, the present value of this future receipt. To do this, it is necessary to determine how much we would have to invest today, at 8% annual interest, to obtain \$1,000 in a year's time. This is done as follows:

$$\frac{1{,}000}{1.08} = \$925.93$$

Therefore, to the nearest cent, given an interest rate of 8%, the present value of \$1,000 a year from now is \$925.93.

The concept of present value is crucial in corporate finance. Investors commit resources now in the expectation of receiving future earnings flows. To properly evaluate the returns from an investment, it is necessary to consider that returns are derived in the future. These future monetary amounts must be expressed in present value terms to assess the worth of the investment when compared to its cost or current market value. Additionally, the cast receipts received at different points in time are not directly comparable without employing the present value (PV) method.

18.3 Foundation of Net Present Value Rules

We begin our study of procedures for determining present values with a simple example. Suppose that an individual, or a firm, has the opportunity to invest C_0 dollars today in a project that will yield a return of C_1 dollars in one year. Assume further that the risk-free annual interest rate, expressed as a percentage, is r. To evaluate this investment, we need to know the present value of the future return of C_1 dollars. In general, for each dollar invested today at interest rate r, we would receive in one year's time, an amount

$$\textit{future value per dollar} = (1 + r)$$

The term (1 + r) is an important enough variable in finance to warrant its own name. It is called a *wealth relative* and is part of all present value formulations. Returning to our discussion, it follows that the present value of a dollar to be received in our year is

$$\textit{present value per dollar} = \frac{1}{1 + r}$$

Therefore, the present value of C_1 dollars to be received in the future is

$$PV = \frac{C_1}{1 + r}$$

In assessing our proposed investment, the present value of the return must be compared with the amount invested. The difference between these two quantities is called the *net present value* of the investment. For the convenience of notation, we will write

$$C_0 = -cost$$

So that C_0, which is negative, represents the "cost" today. The net present value, then, is the sum of today's "cost" and the present value of the future return; that is

$$NPV = C_0 + \frac{C_1}{1 + r}$$

Provided this quantity is positive, the investment is worth making.

As another example, suppose you are offered the opportunity today to invest \$1,000, with an assured return of \$1,100 one year from now and a risk-free interest rate of 8%. In our notation, then, $C_0 = -1{,}000$; $C_1 = 1{,}100$; and $r = .08$. The present value of the \$1,000 return is

$$\frac{C_1}{1 + r} = \frac{1{,}100}{1.08} = \$1{,}018.52$$

where again we have rounded to the nearest cent. Thus, it requires \$1,018.52 invested at 8% to yield \$1,000 in one year. Therefore, the net present value of our investment opportunity is

$$\begin{aligned} NPV &= C_0 + \frac{C_1}{1 + r} \\ &= -1{,}000 + 1{,}018.52 = \$18.52 \end{aligned}$$

Offering you this investment opportunity then is equivalent to an increase of \$18.52 in your current wealth.

This example is quite restrictive in that it assumes that all of the investment's returns will be realized precisely in one year. In the next section, we see how this situation can be generalized to allow for the possibility of returns spread over time.

18.4 Compounding and Discounting Processes

In this section, we extend our analysis of present values to consider the valuation of a stream of cash flows. We consider two cases. In the first, a single payment is to be received at some specified time in the future; in the second, a sequence of annual payments is to be received. For each, we will consider both future and present values.

18.4.1 Single Payment Case—Future Values

Suppose that \$1 is invested today for a period of t years at a risk-free annual interest rate r_t, with interest to be compounded annually. How much money will be returned at the end of t years? We find the answer by proceeding in annual steps. At the end of the first year, an amount of interest r_1 is added, giving a total of $\$(1 + r_1)$. Since interest is compounded, second-year interest is paid on this whole amount, so that the interest paid at the end of the second year is $\$r_2(1 + r_1)$. Hence, the total amount at the end of the second year is

$$\begin{aligned} \textit{future value per dollar after 2 years} &= (1+r_1)+r_2(1+r_1) \\ &= 1+r_1+r_2+r_1r_2 \end{aligned}$$

In words, the future value in two years comprises four quantities: the value you started with, \$1; the interest accrued on the principal during the first year, r_1; the interest earned on the principal during the second year, r_1r_2. If the interest rate is constant—that is, $r_1 = r_2 = r_t$—then the compound term r_1r_2 can be written r_t^2. This assumes that the term structure of interest rates is flat. Continuing in this way, interest paid at the end of the third year is $\$r_3(1 + r_t)^2$, so that

$$\begin{aligned} \textit{future values per dollar after 3 years} &= (1+r_t)^2+r_3(1+r_t)^2 \\ &= 1+r_1+r_2+r_3+r_1r_2 \\ &\quad +r_1r_3+r_2r_3+r_1r_2r_3 \\ &= (1+r_t)^3 \end{aligned}$$

In words, the future value in three years comprises eight terms: the principal you started with; three terms for the interest on the principal each year, r_1, r_2, r_3; three terms for the interest on the interest, r_1r_2, r_1r_3, r_2r_3; and a term for the interest during year 3 on the compound interest from years 1 and 2, $r_1r_2r_3$. Again, if $r_1 = r_2 = r_3 = r_t$, this can all be reduced to $(1 + r_t)^3$. It is interesting to note that as t increases, the r_t terms increase linearly, whereas the compound terms increase geometrically. That is, for each year, there is only one yearly interest payment, but for the compounding terms, the number for $t = 4$ is 16 compounding terms, and for $t = 5$, it is 32 compounding terms. This compounding of interest on interest is an important concept to any investor. Large increases in value are not caused by yearly interest but by the reinvestment of the interest. The general line of reasoning should be clear. After t years, where t is any positive integer, we have

$$\textit{future value per dollar} = (1+r_t)^t \qquad (18.1)$$

To illustrate, suppose that \$1,000 is invested at an annual interest rate of 8%, with interest compounded annually, for a period of five years. At the end of this term, the total amount received, in dollars, will be

$$1{,}000(1+.08)^5 = 1{,}000(1.08)^5 = 1{,}469.33$$

The total interest of \$469.33 consists of \$400 yearly interest (\$80 per year $\times$ 5 years) and \$69.33 of compounding. If $t = 64$ years, the future value is \$137,759.11, which consists of \$5,120 yearly interest (80 per year $\times$ 64) and \$131,639.11 of compounding.

18.4.2 Continuous Compounding

There is no difficulty in adapting Eq. (4.1) to a situation where interest is compounded at an interval of less than one year. Simply replace the word *year* with the word *period* compounding (the interval) in the above discussion. For example, suppose the interest is compounded semiannually, with an annual rate of 8%. This implies that 4% is to be added to the balance at the end of each half year. Suppose, again, that \$1,000 is to be invested for a term of five years. Since this is the same as a term of ten half years, the total amount (in dollars), to be received is

$$1{,}000(1+.04)^{10} = 1{,}000(1.04)^{10} = 1{,}480.24$$

The additional \$10.91 (\$480.24–\$469.33) arises because the compounding effect is greater when it occurs ten times than when it occurs five times.

The extreme case is when interest is compounded continuously. (This is discussed in Appendix 3C in greater detail.) The total amount per dollar to be received after t years, if interest is compounded continuously at an annual rate r_t, is

$$\textit{future value per dollar} = e^{tr_t} \qquad (18.2)$$

where $e = 2.71828\ldots$ is a constant. If \$1,000 is invested for five years at an annual interest rate of 8%, with interest compounded continuously, the end-of-period return would be

$$1{,}000e^{5(.08)} = 1{,}000e^{.4} = \$1{,}491.80$$

Many investment opportunities offer daily compounding. The formula we present for continuous compounding provides a close approximation to daily compounding.

18.4.3 Single Payment Case—Present Values

Since many investments generate returns during several different years in the future, it is important to assess the present value of future payments. Suppose that a payment is to be received in t years' time and that the risk-free annual interest rate for a period of t years is r_t. In Eq. (18.1), we saw that future value at the end of t years is $(1+r_t)^t$ per dollar. Conversely, it follows that the present value of a dollar received at the end of t years is

$$present\ value\ per\ dollar = \frac{1}{(1+r_t)^t} \tag{18.3}$$

For example, suppose that \$1,000 is to be received in four years. At an annual interest rate of 8%, the present value of this future receipt is

$$1{,}000 \times \frac{1}{(1+.08)^4} = \frac{1{,}000}{(1.08)^4} = \$735.03$$

More generally, we can consider a stream of annual receipts, which may be positive or negative. Suppose that, in dollars, we are to receive C_0 now, C_1 in one year, C_2 in two years, and so on, and finally in year N we receive C_N. Again, let r_t denote the annual rate of interest for a period of t years. To find the net present value of this stream of receipts, we simply add the individual present values, obtaining

$$NPV = C_0 + \frac{C_1}{(1+r_1)^1} + \frac{C_2}{(1+r_2)^2} + \ldots + \frac{C_N}{(1+r_2)^N} \tag{18.4}$$

$$NPV = \sum_{t=0}^{N} \frac{C_t}{(1+r_t)^t}$$

Typically, the rate of interest, r_t, depends on the period t. When a constant rate, r, is assumed for each period, the net present value formula (Eq. 18.4) simplifies to

$$NPV = \sum_{t=0}^{N} \frac{C_t}{(1+r)^t} \tag{18.5}$$

Example 18.1 A corporation must choose between two projects. Each project requires an immediate investment, and further costs will be incurred in the next year. The returns from these projects will be spread over a four-year period. The following table shows the dollar amounts involved.

		Year 0	Year 1	Year 2	Year 3	Year 4
Project A	Costs	80,000	20,000	0	0	0
	Returns	0	20,000	30,000	50,000	50,000
Project B	Costs	50,000	50,000	0	0	0
	Returns	0	40,000	60,000	30,000	10,000

At first glance, this data might suggest that, for project A, total returns exceed total costs by \$50,000, while the same figure that project B is only \$40,000, indicating a preference for project A. However, this neglects the timing of the returns. Assuming an annual interest rate of 8% over the four-year period, we can calculate the present values of the net receipts for each project as follows:

		Year 0	Year 1	Year 2	Year 3	Year 4
Project A	Net returns	−80,000	0	30,000	50,000	50,000
	Present values	−80,000	0	25,720	39,692	36,751
Project B	Net returns	−50,000	−10,000	60,000	30,000	10,000
	Present values	−50,000	−9,259	51,440	23,815	7,350

It is the sums of the present values that must be compared in evaluating the projects. For project A, substituting $r = .08$ into Eq. 18.5

$$\begin{aligned} NPV &= -80{,}000 + \frac{0}{(1.08)^1} + \frac{30{,}000}{(1.08)^2} + \frac{50{,}000}{(1.08)^3} + \frac{50{,}000}{(1.08)^4} \\ &= -80{,}000 + 0 + 25{,}720 + 39{,}692 + 36{,}751 \\ &= \$22{,}163 \end{aligned}$$

Similarly, for project B

$$\begin{aligned} NPV &= -50{,}000 - \frac{10{,}000}{(1.08)^1} + \frac{60{,}000}{(1.08)^2} + \frac{30{,}000}{(1.08)^3} + \frac{10{,}000}{(1.08)^4} \\ &= -50{,}000 - 9{,}259 + 51{,}440 + 23{,}815 + 7{,}350 \\ &= \$23{,}346 \end{aligned}$$

It emerges that, if future returns are discounted at an annual rate of 8%, the net present value is higher for project B than for project A. Hence, project B is preferred, because it provides larger cash flows in the early years, which gives the firm more opportunity to reinvest the funds, thereby adding greater value to the firm.

18.4.4 Annuity Case—Present Values

An *annuity* is a special form of income stream in which regularly spaced equal payments are received over a period of time. Common examples of annuities are payments on home mortgages and installment credit loans.

Suppose that an amount C dollars is to be received at the end of each of the next n time periods (which could, for example, be months, quarters, or years). Assume further that, irrespective of the term, the interest rate period is fixed at r. Then the present value of the payment to be received at the end of the first period is $C/(1+r)$, the present value of the next payment is $C/(1+r)^2$, and so on. Hence, the present value of the N period annuity is

$$PV = \frac{C}{(1+r)^1} + \frac{C}{(1+r)^2} + \ldots + \frac{C}{(1+r)^n} = \sum_{t=1}^{N} \frac{C_t}{(1+r)^t}$$

In fact, it can be shown[1] that this expression simplifies to

$$PV = C\left[\frac{1}{r} - \frac{1}{r(1+r)^N}\right] \tag{18.6}$$

Suppose that an annuity of $1,000 per year is to be received for each of the next ten years. The total dollar amount is $10,000, but because receipts stretch far into the future, we would expect the present value to be much less. Assuming an annual interest rate of 8%, we can find the present value of this annuity by using Eq. 4.6

$$\$1{,}000\left[\frac{1}{,08} - \frac{1}{.08(1.08)^{10}}\right] = \$6{,}710$$

This annuity, then, has the same value as an immediate cash payment of $6,710.

Perpetuity

An extreme type of annuity is a *perpetuity*, in which payments are to be received forever. Certain British government bonds, known as "consol", are perpetuities. The principal need not be repaid, but a fixed interest payment to the bondholder is made every year. To find the present value of a perpetuity, we need only let the term—N, in the annuity case—grow infinitely large. Consequently, the second expression in brackets on the right-hand side of Eq. 18.6 becomes zero, so that the present value of perpetuity payments of C dollars per period, when the per period interest rate is r, is

$$PV = \frac{C}{r}$$

For example, given an 8% annual interest rate, the present value of $1,000 per annum in perpetuity is

$$\frac{\$1{,}000}{.08} = \$12{,}500$$

Notice that this sets an upper limit on the possible value of an annuity. Thus, if the interest rate is 8% per annum, annuity payments of $1,000 per year must have a present value of less than $12,500, whatever the term.

18.4.5 Annuity Case—Future Values

With an annuity of C dollars per year, we can also calculate a future value (FV) by using Eq. 18.7

$$FV = C(1+r)^N + C(1+r)^{N-1} + \ldots + C(1+r)^1 \tag{18.7}$$

This is very similar to the single value case discussed earlier; each of the terms on the right-hand side of Eq. 18.7 is identical to the values shown by Eq. 18.1.

18.4.6 Annual Percentage Rate

The annual percentage rate (APR) is the actual or effective interest rate that the borrower is paying. Quite often, the stated or nominal rate of a loan is different from the actual amount of interest or cost the lender is paying. This results from the differences created by using different compounding periods. The main benefit of calculating the APR is that it allows us to compare interest rates on loans or investments that have different compounding periods.

The Consumer Credit Protection Act (Truth-in-Lending Act), enacted in 1968, provides for disclosure of credit terms so that the borrower can make a meaningful comparison of alternative sources of credit. This act was the cornerstone for Regulation Z of the Federal Reserve. The finance charge and the annual percentage rate must be given explicitly to the borrower. The *finance charge* is the actual dollar amount that the borrower must pay if given the loan. The APR also must be explained to individual borrowers and the actual figure must be given.

[1] Let x = 1/(1 + r). Then

$$PV = C(x)\left(1 + x + \ldots + x^{N-1}\right) = C(x)\left(\frac{1-x^N}{1-x}\right)$$
$$= C * \frac{1}{1+r} * \frac{1+r}{r}\left[1 - \frac{1}{(1+r)^N}\right]$$

From which Eq. (18.6) follows.

Exhibit 18.1 shows the amount of interest paid and the APR for a $1,000 loan at 10% interest for 1 year, to be repaid in 12 equal monthly installments.

Exhibit 18.1: Interest Paid and APR

Amount borrowed = $1,000.

Nominal interest rate = 10% per year or 0.83% per month.

$$\text{Annuity or monthly payment} = \frac{\text{amount borrowed}}{\sum_{t=1}^{N} \frac{1}{\left(1+\frac{r}{12}\right)^t}} = \frac{1{,}000}{11.3745} = \$87.92$$

Month	Payment	Interest	Principal paid off	Remaining principal unpaid
0	–	–	–	$1,000
1	$87.92	$8.33	$79.58	$920.42
2	87.92	7.67	80.25	840.17
3	87.92	7.00	80.91	759.26
4	87.92	6.33	81.59	677.67
5	87.92	5.65	82.27	595.40
6	87.92	4.96	82.95	512.45
7	87.92	4.27	83.65	428.80
8	87.92	3.57	84.34	344.46
9	87.92	2.87	85.05	259.41
10	87.92	2.16	85.75	173.66
11	87.92	1.45	86.47	87.19
12	87.92	0.73	87.19	0.00
Total	$1,054.99	$54.99	$1,000.00	

$$\text{Average loan balance} = \frac{\text{beginning balance} - \text{ending balance}}{2} = \frac{1{,}000 - 0}{2} = 500$$

$$APR = \frac{\text{interest}}{\text{average loan outstanding}} = \frac{54.99}{500} = 10.9981\%$$

From Exhibit 18.1, we see that the total interest paid is $54.99 and the APR is 10.9981%. The nominal rate and the APR will be different for all annuity arrangements, because the more frequent the repayment, the greater the APR. This calculation is useful to individuals in evaluating home mortgages and to corporations borrowing with term loans to finance assets.

18.5 Present and Future Value Tables

In the previous section, we presented formulae for various present and future value calculations. However, the arithmetic involved can be rather tedious and time-consuming. Because the present and future values are frequently needed, tables have been prepared to make the computational task easier. When using present value tables, keep in mind the following: (1) they cannot be used for $r < 0$, (2) the interest or discount rate must be constant over time for use of annuity tables, and (3) the tables are constructed by assuming that all cash flows are reinvested at the discount rate or interest rate.

18.5.1 Future Value of a Dollar at the End of t Periods

Suppose that a dollar is invested now at an interest rate of r per period, with interest compounded at the end of each period. Equation 18.1 gives the future value of a dollar at the end of t periods. Values of this expression for various interest rates, r, and the number of periods, t, are tabulated in Table 1, which presents the future value of annuity. Table 18.3 of Appendix 18C presents the Excel approach to calculate this future value.

To illustrate, suppose that a dollar is invested now for 20 years at an annual rate of interest of 10% compounded annually. Table 1 shows that the future value—the amount to be received at the end of this period—is $6,728. (It follows, of course, that the future value of an investment of $1,000 is $6,728.)

Example 18.2 Suppose you deposit $1,000 at an annual interest rate of 12% for two years. How much extra interest would you receive at the end of the term if interest was compounded monthly instead of annually?

Annual compounding is straightforward. Table 18.20 shows that the future value per dollar for a term of two years at an annual interest rate of 12% is $1.254.

If the interest is compounded monthly, then the number of periods would be 4 and the monthly interest rate is 6%. According to Table 18.20, the future value factor for 4 periods with an interest rate to be 6% is 1.2625. Hence, the future value of $1,000 is $1,270.

Therefore, the extra interest we would receive (the gain in future value) from monthly compounding is

$$\$1{,}270 - \$1{,}254 = \$16$$

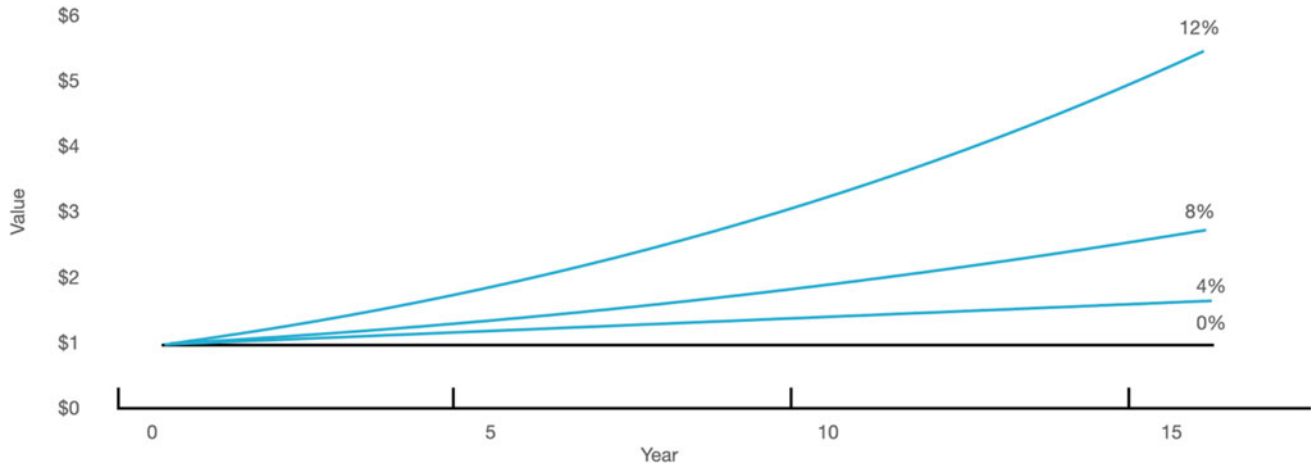


Fig. 18.1 Future value over time of $1 invested at different interest rates

Using the information in Table 18.20 of the appendix, we can construct graphs showing the effect over time of compound interest. Figure 18.1 shows the future values over time of a dollar invested at interest rates of 0, 4, 8, and 12%. At 0%, the future value is always $1. The other three curves were constructed from the future values taken from the 4, 8, and 12% interest columns in Table 18.20. Notice that these curves exhibit exponential growth; that is, as a result of compounding, annual changes in future values increase nonlinearly. Of course, the higher the rate of interest, the greater the growth rate; or the longer the time, the greater the compounding effect.

In Fig. 18.2, we compare future values of a dollar over time under simple and annually compounded interest, both at a 10% annual interest rate. By simple interest, we usually mean the interest calculated for a given period by multiplying the interest rate times the principal. The future values for compound interest are listed in Table 1 of the appendix. Under simple interest, ten cents is accumulated each year, so that the future value after t years is $\$(1 + .10t)$. Notice that, while the future values grow exponentially under compounding, they do so only linearly with simple interest, so that the two curves diverge over time.

18.5.2 Future Value of a Dollar Continuously Compounded

Table 18.21 in the appendix of this book shows the future value of a dollar invested for t periods at an interest rate of r per period, continuously compounded. The entries in this table are computed from Eq. 18.2, which states that the future value is e^{rt}. Table 18.21 shows the corresponding future values for specific values of rt.

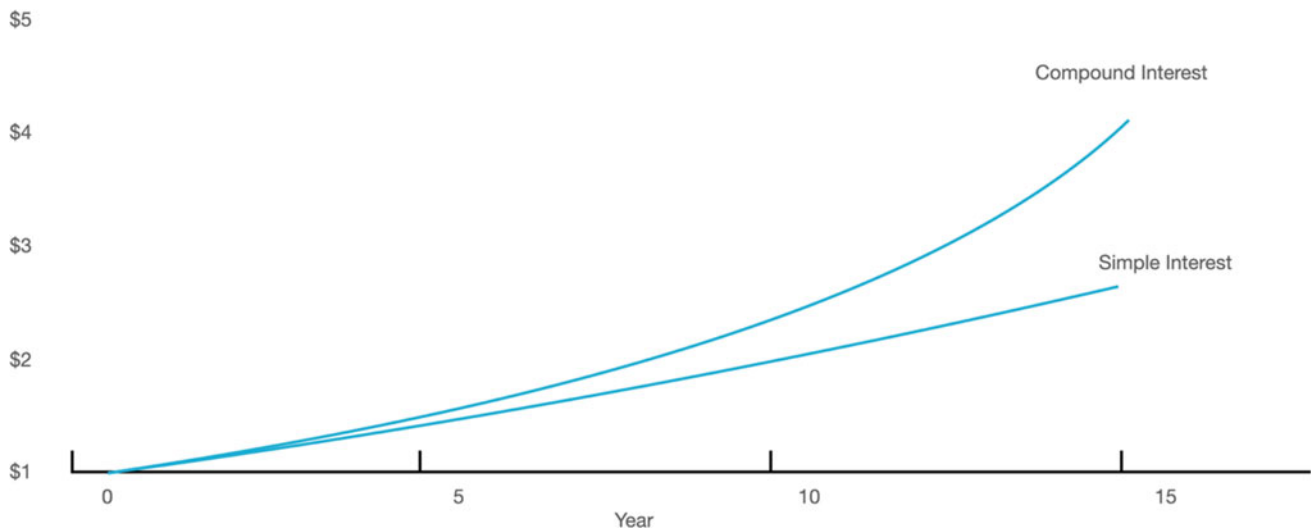


Fig. 18.2 Future value over time of $1 invested at 10% per annum simple and compound interest

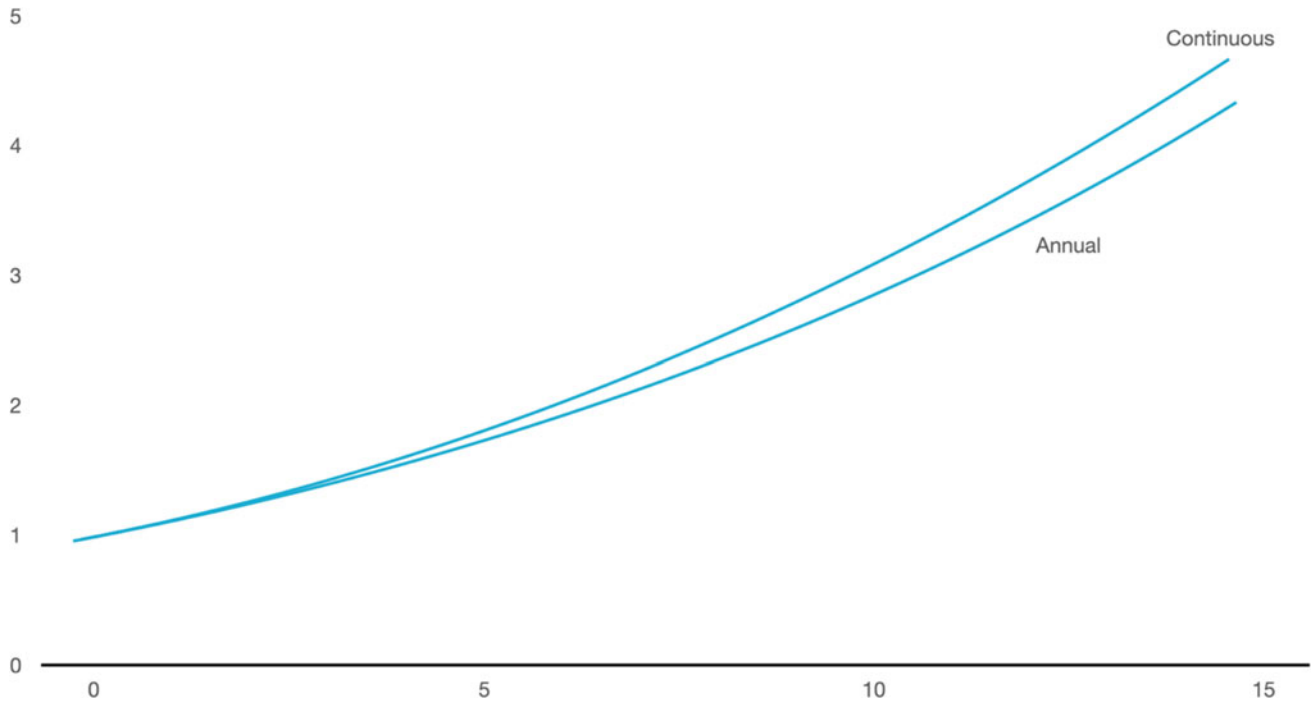


Fig. 18.3 Future value time of $1 invested at 10% per annum, compounded annually and continuously

To illustrate, suppose a dollar is invested now for 20 years at an annual interest rate of 10%, with continuous compounding. The future value at the end of the term can be read from Table 2, using $r = 0.10$, $t = 20$, $rt = 2$. From the table, we find, corresponding to an rt value of 2, the future value is $7.389.

Figure 18.3 compares, over time, the future value of a dollar invested at 10% per annum under both annual and continuous compounding. The two curves were constructed from the information in Tables 1 and 2 of the appendices. Notice that, over time, the curves diverge, reflecting the faster growth rate of future values as the interval for compounding decreases.

18.5.3 Present Value of a Dollar Received t Periods in the Future

Suppose that a dollar is to be received t periods in the future and that the rate of interest is r, with compounding at the end of each period. The present value of this future receipt can be computed from Eq. 18.3. The results of various combinations of values of r and t are tabulated in Table 3 of the appendix at the back of this volume. For example, the table shows that the present value of a dollar to be received in 20 years' time, at an annual interest rate of 10% compounded annually, is $0.149. (It follows that the present value of $1,000 under these conditions is $149.)

Using the information in Table 3, we can construct graphs showing the effect over time of the discounting process involved in present value calculations. Figure 18.4 shows the present values of a dollar received at various points in the future, discounted at interest rates of 0, 4, 8, and 12%. Notice that the present values decrease the further into the future the payment is to be received; the higher the interest rate, the sharper the decrease. A comparison of Figs. 18.1, 18.2, 18.3 and 18.4 reveals the connection between compound interest and present values. This is also clear from Eqs. 18.1 and 18.4. If the future value after t years of a dollar invested today, at annual interest rate r, is K, then, using the same interest rate, the present value of K to be received in t years' time is $1.

Example 18.3 A corporation is considering a project for which both costs and returns extend into the future, as set out in the following table (in dollars).

Year	0	1	2	3	4	5
Costs	130,000	70,000	50,000	0	0	0
Returns	0	20,000	25,000	50,000	60,000	75,000
Year	6	7	8	9	10	
Costs	0	0	0	0	0	
Returns	75,000	60,000	50,000	25,000	20,000	

Assuming that future returns are discounted at an annual rate of 8%, find the net present value of this project.

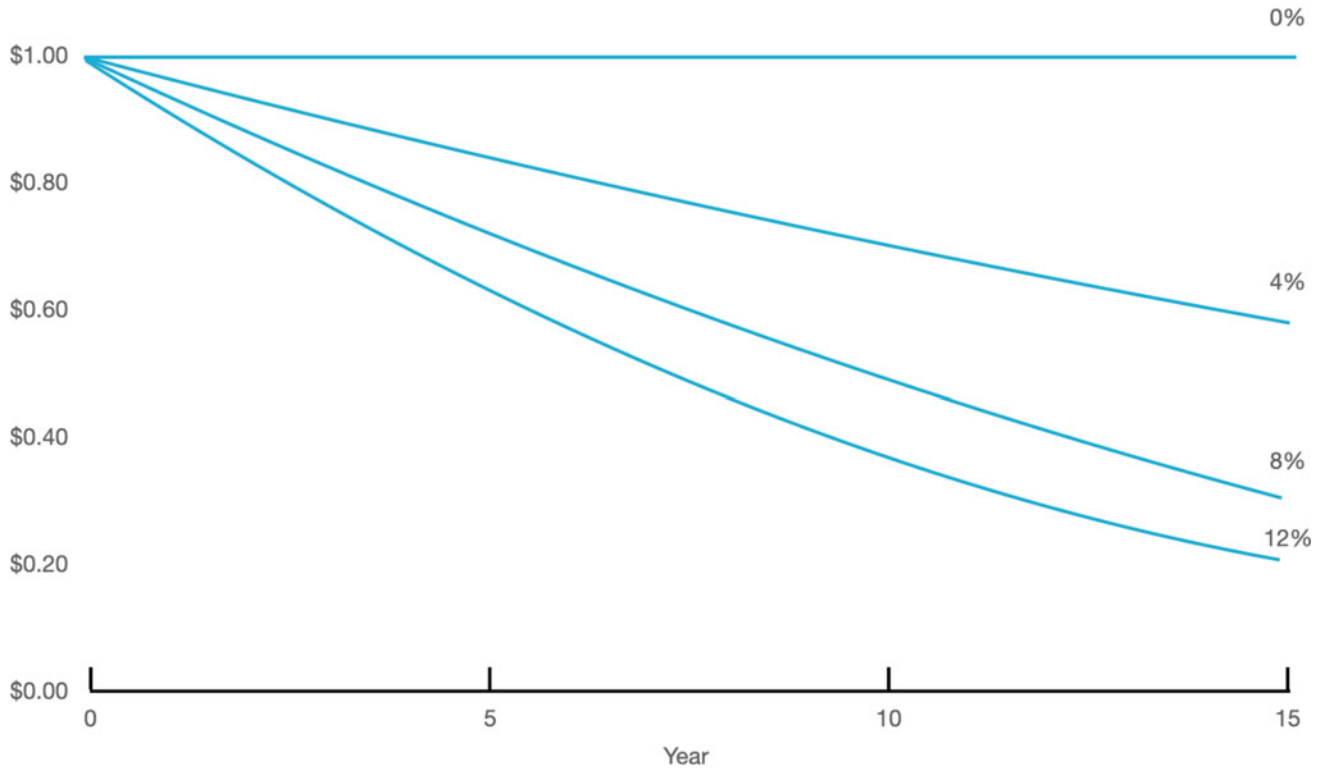


Fig. 18.4 Present value, at different discount rates, of $1 to be received in the future

As in Example 18.1, we could solve this problem by using an equation; in this case, Eq. 18.5. However, we can save time and effort by obtaining the present value per dollar figures directly from Table 18.22 of the appendix of this book. Multiplying these figures by the corresponding net returns and then summing gives us the net present value

$$\begin{aligned}\text{NPV} &= -130{,}000 - (50{,}000)(.9259) - (25{,}000)(.8573) \\ &\quad + (50{,}000)(.7938) + (60{,}000)(.7350) + (75{,}000)(.6806) \\ &\quad + (75{,}000)(.6302) + (60{,}000)(.5835) + (50{,}000)(.5403) \\ &\quad + (25{,}000)(.5002) + (2000)(0.4632) \\ &= \$68{,}163.06\end{aligned}$$

18.5.4 Present Value of an Annuity of a Dollar Per Period

Suppose that a dollar is to be received at the end of each of the next N periods. If the interest rate per period is r, the present value of this annuity is obtained by using C = 1 in Eq. 18.6. These present values are tabulated for various interest rates in Table 18.23 in the appendix of this book. For example, at an annual interest rate of 6%, the present value of $1 per year for 20 years is $11,470. (It follows that the present value of an annuity of $1,000 per year is $11,470.)

18.6 Why Present Values Are Basic Tools for Financial Management Decisions

An unrealistic feature of our discussion of present values has been the assumption that monetary amounts of future returns on an investment are known with certainty. However, in most management decision problems, while it is possible to estimate future returns, these estimates will not be precisely equal to the actual outcomes. In practice, then, it is necessary to take into account some element of risk. To do this, we discount future returns by using r_t, which is not the risk-free interest rate but rather the interest rate on some equivalent, equally risky security or investment. In principle, with this modification, a financial manager can compute the net present value of any risky project. Our aim in this section is to show that such present value calculations are important basic tools in the financial management decision-making process.

Another way to incorporate risk into the analysis is through *certainty equivalence*. Suppose that a project will yield an estimated $10,000 next year, but that there is some risk attached, so that this result is not certain. Typically, an investor will be averse to risk and so would prefer an alternative project in which $10,000 was certain to be realized. However, other investors may prefer the risky project with a sure return of somewhat less than $10,000, being prepared to accept some risk in the expectation of a higher

return. For example, the original project may be seen as equivalent to one in which a return of $9,000 is certain. We can then value the project by discounting the certainty equivalent return at the risk-free rate.

18.6.1 Managing in the Stockholders' Interest

Consider the dilemma of a corporate manager who makes investment decisions on behalf of the corporation's stockholders. Because stockholders do not constitute a homogeneous entity, the manager is faced with the problem of accommodating an array of tastes and preferences. In particular:

- Stockholders are not uniform in their time preferences for consumption. Some prefer relatively high levels of current consumption, while others prefer less current consumption in order to obtain higher consumption levels in the future.
- Stockholders have different attitudes toward the risk-return trade-off. Some are happier than others to accept an element of risk in anticipation of higher potential returns.

Even if the manager is able to elicit accurate information about the various tastes and preferences of individual stockholders, the problem of making decisions for the benefit of all seems formidable. Fortunately, Irving Fisher, in 1930, developed a simple resolution. Essentially, Fisher demonstrated that, given certain assumptions, whatever the array of stockholder tastes and preferences, the optimal management strategy is to *maximize the firm's net present value*.

To illustrate, suppose that a particular stockholder has a current cash flow of $50,000 and a future cash flow, next year, of $64,800.[2] This stockholder could plan to consume $50,000 this year and $64,800 next year. However, this is not the only consumption pattern that can be achieved with these resources.

At the heart of our analysis is the assumption that there is access to the capital markets, in which cash on hand can be lent, or that an investor can borrow against future cash receipts. This allows our stockholders to consume either more or less than $50,000 this year, which affects next year's consumption level. Moreover, the investor is not restricted to risk-free market instruments, but is free to opt for riskier securities with higher expected returns. For our conclusions to follow, we need to assume perfect competition in the capital markets; that is:

1. Access to the market is open and free, with securities readily traded.
2. No individual, or group of individuals acting in collusion, has sufficient market power for the actions of the individual or group to significantly influence market prices.
3. All relevant information about the price and risk of securities is readily available, at no cost, to all.

Certainly, these assumptions are an idealization of reality. Nevertheless, they are sufficiently close to reality for our analysis to be appropriate.

Now, in considering the consumption patterns available to our individual investor, we will assume borrowing or lending at the risk-free rate, which, for purposes of illustration, is 8%. The investor may, instead, prefer to assume some level of risk, which trading in the capital market allows for, and for such an investor this example can be carried through in terms of certainty equivalent amounts.

Let us begin by computing the present value and future value of this investor's cash flow stream. At an interest rate of 8%, the present value is

$$\begin{aligned} PV &= 50{,}000 + \frac{64{,}800}{1.08} \\ &= 50{,}000 + 60{,}000 \\ &= \$110{,}000 \end{aligned}$$

This investor could consume $110,000 this year and nothing next year by borrowing $60,000 at 8%interest. All of next year's income will then be needed to repay this loan.

The future value, next year, of the cash flow stream is

$$\begin{aligned} FV &= (50{,}000)(1.08) + 64{,}800 \\ &= 54{,}000 + 64{,}800 \\ &= \$118{,}800 \end{aligned}$$

It follows that another option available to our investor is to consume nothing this year and $118,800 next year. This can be achieved by investing all of this year's cash flow at 8% interest.

Our results are depicted in Fig. 18.5, which represents possible two-period consumption levels. These levels are found by plotting current consumption on the horizontal axis and future consumption on the vertical axis; a point on the curve represents a specific combination of current and future consumption levels. Thus, our two extreme cases are (0; 118,800) and (110,000; 0).

Between these extremes, many combinations are possible. If the investor wants to consume only $30,000 of the

[2] 2 The restriction of our analysis to two periods is convenient for graphical exposition. However, the same conclusions follow when this restriction is dropped.

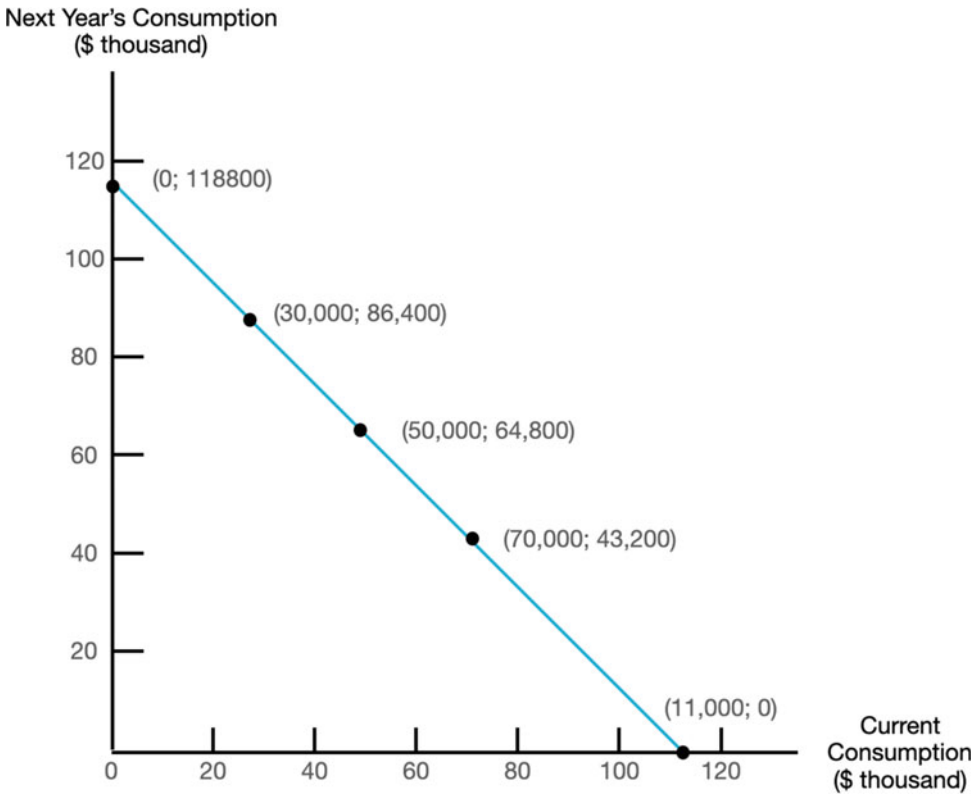


Fig. 18.5 Trade-offs in two-period consumption levels

current year's cash flow, the remaining $20,000 can be invested at 8% to yield $21,600 next year. Adding this to next year's cash flow produces a future consumption total of $86,400.

Conversely, $70,000 can be consumed this year by borrowing $20,000 at 8% interest. This requires repayment of $21,600 next year, leaving $43,200 available for consumption at that time (Table 18.1).

The consumption possibilities discussed so far are listed in Table 18.1 and plotted in Fig. 18.5. But these are not the only possibilities. Notice that the five points all lie on the same straight line. The reason is that, at 8% annual interest, each $1 of current consumption can be traded for $1.08 of consumption next year, and vice versa; therefore, any pair of consumption levels on the line in Fig. 18.5 is possible. The slope of the consumption trade-off line in Fig. 18.5 is $-(1+r)$, i.e., −1.08.

In addition to the time preference discussed in this section, positive interest rates also indicate a liquidity preference on the part of some investors. Keynes (1936) gives three reasons why individuals require cash: (1) to pay bills (transaction balances), (2) to protect against uncertain adverse future events (precautionary balances), and 93) for speculative reasons (for example, if interest rates are expected to rise in the future, it may be best to stay liquid today to take advantage of the future higher rates). Each rationale for holding cash makes individuals more partial to maintaining liquidity. An incentive must be offered in the form of a positive interest rate to induce these individuals to give up some of their liquidity.

For a corporation, the management of cash and working capital is an important treasury function that takes these factors into consideration.

18.6.2 Productive Investments

So far, we have assumed that the only opportunities for our investor are in the capital market. Suppose that there are productive investment opportunities, which may yield, in certainty equivalent terms, rates of return in excess of 8% per annum. Each dollar invested now that produces a return in excess of $1.08 in a year's time will increase the net present value for the investor.

Table 18.1 Consumption possibilities as plotted in Fig. 18.5 (in dollars)

Current year	0	30,000	50,000	70,000	110,000
Next year	118,800	86,400	64,800	43,200	0

To illustrate, suppose the investor finds $80,000 worth of such opportunities that will yield $97,200 next year. (Notice that the amount invested can exceed the current year's cash flow, because any excess can be borrowed in the capital market.) The net present value of these investment opportunities is

$$NPV = -80{,}000 + \frac{97{,}200}{1.08} = \$10{,}000$$

These productive investments would raise the present value of our investor's cash flow stream from $110,000 to $120,000. Similarly, future value is raised by (1.08) (10,000) = $10,800, from $118,800 to $129,600.

Taking advantage of such productive opportunities does not affect the investor's access to the capital market. Therefore, our investor could consume $120,000 now and nothing next year, or nothing now and $129,600 next year. It is also possible to have intermediate consumption level combinations by trading $1 of current consumption for $1.08 of future consumption.

This position is illustrated in Fig. 18.6, which shows the shift in the consumption possibilities line resulting from the productive investments. As compared with the earlier position, it is possible to consume more both now and in the future. Hence, we find that, whatever the time preference for consumption, the investor is better off as a result of a productive investment that raises net present value. Neither is it necessary to worry about the investor's attitude toward risk, as this too can be accommodated through capital market investments.

We have now established Irving Fisher's concept. Viewing this individual stockholder's cash flows as shares of those of the corporation, it follows that, to act in the stockholders' interest, management's objective should be to seek those productive investments that increase the net present value of the corporation as much as possible.

It follows from this discussion that the concept of net present value does considerably more than provide a convenient and sensible way of interpreting future receipts. As we have just seen, the net present value provides a basis on which financial managers can judge whether a proposed productive investment is in the best interest of corporate stockholders. The manager's task is to ascertain whether or not the project raises the firm's net present value by more than would competing projects, without having to pay attention to the personal tastes and preferences of stockholders.

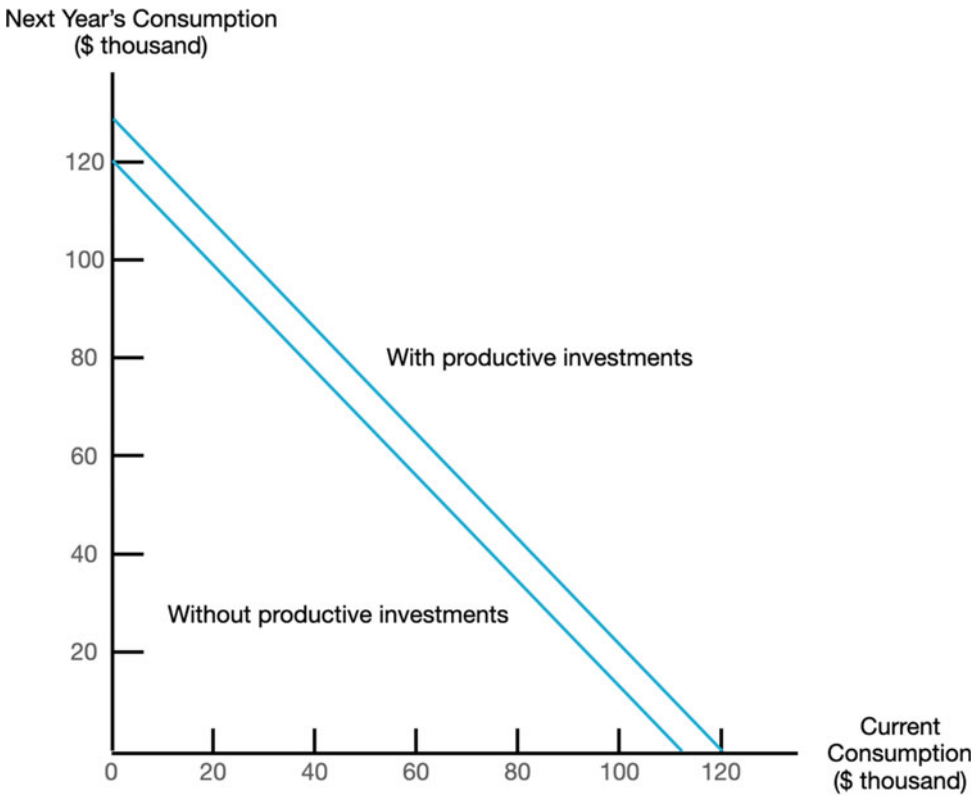


Fig. 18.6 Trade-offs in two-period consumption levels with and without productive assets

Table 18.2 Partial Inputs information for NPV method and IRR method

		Year 0	Year 1	Year 2	Year 3	Year 4
Project A	Cost	−$80,000	−$20,000	0	0	0
	Return	0	$20,000	$30,000	$50,000	$50,000
Project B	Cost	−$50,000	−$50,000	0	0	0
	Return	0	$40,000	$60,000	$30,000	$10,000

18.7 Net Present Value and Internal Rate of Return

Both Net present value (NPV) method and internal rate of return (IRR) method can be used to do the capital budgeting decision. For example, for project A and project B, the initial outlays and net cash inflow for year 0 to year 4 are presented in Table 18.2. In Table 18.2, we know that the initial outlay at year 0 for Project A and B are $80,000 and $50,000, respectively. In year 1, additional investments for projects A and B are $20,000 and $50,000, respectively. The net cash inflow of project A for the next four years are $20,000, $30,000, $50,000, and $50,000, respectively. The net cash inflow of project B for the next four years are $40,000, $60,000, $30,000, and $10,000, respectively.

The net present value of a project is computed by discounting the project's cash flows to the present by the appropriate cost of capital. The formula used to calculate NPV can be defined as follow:

$$NPV = \sum_{t=1}^{N} \frac{CF_t}{(1+k)^t} - I \tag{18.8}$$

where

k = the appropriate discount rate.

CFt = Net Cash flow (positive or negative) in period t,

I = Initial outlay,

N = Life of the project.

Using the excel function, we can calculate NPV for both projects A and B. We can also calculate the example above using the Excel NPV function. NPV is a function to calculate the net present value of an investment by using a discount rate and a series of future payments (negative values) and income (positive values). The NPV function in Cell H10 is equal to

$$= \text{NPV}(\text{C2}, \text{D10} : \text{G10}) + \text{C10}$$

Based upon the NPV function in Fig. 18.7, the NPV results are shown in Fig. 18.8.

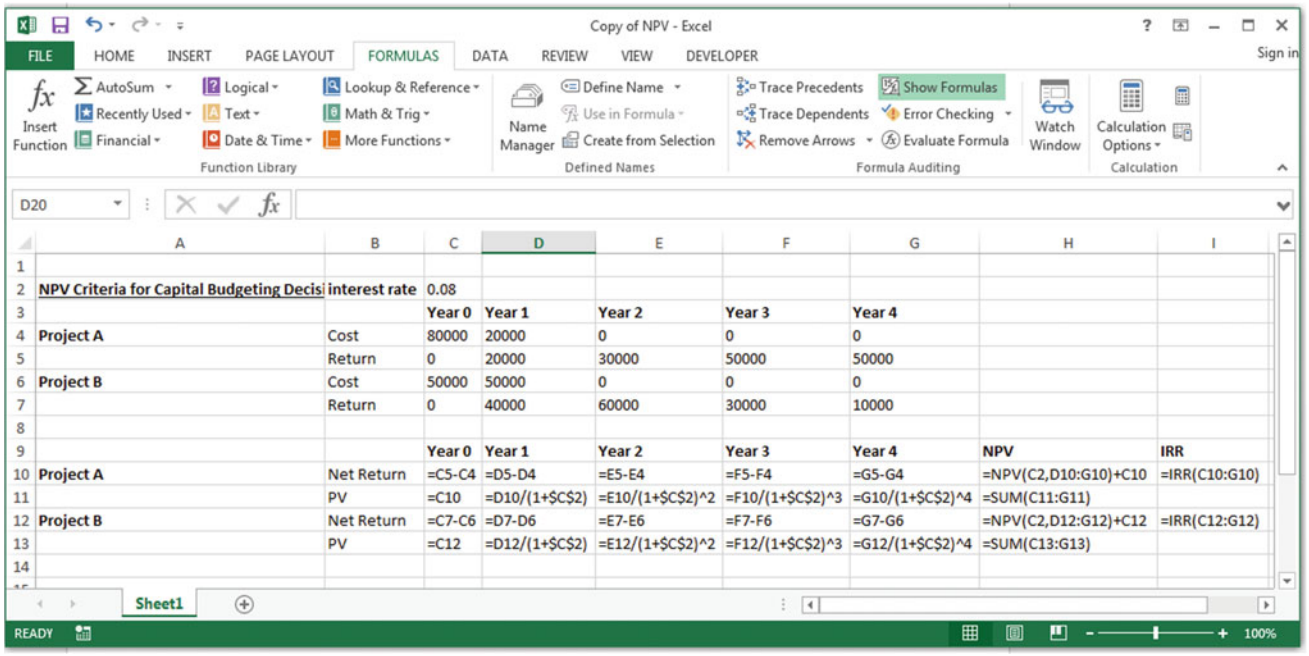
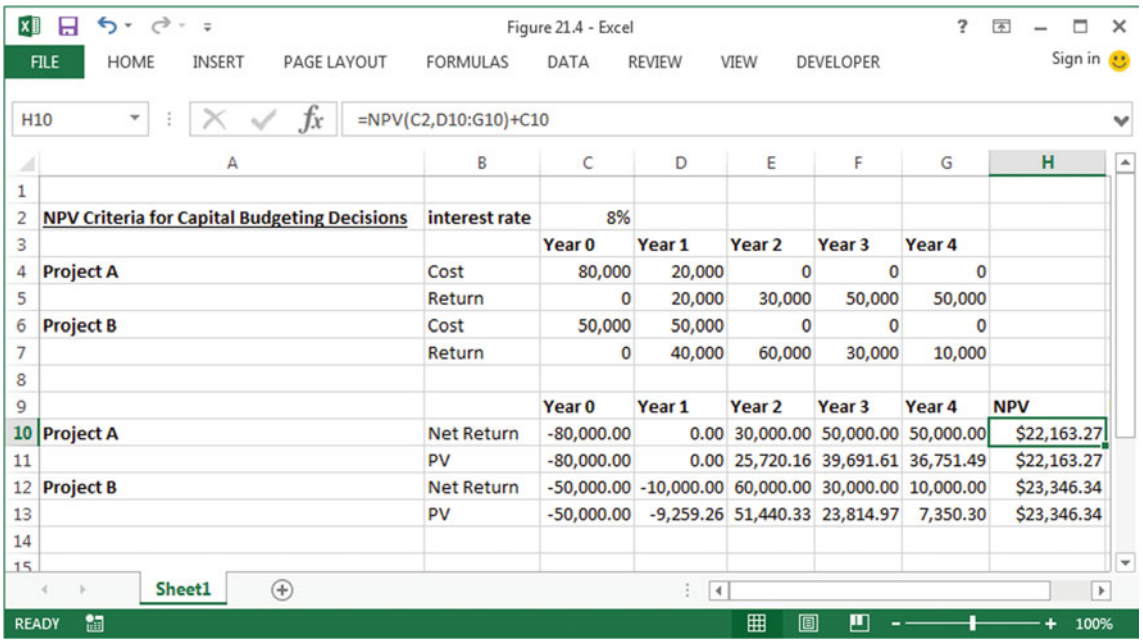


Fig. 18.7 Excel calculation functions for NPV method



	A	B	C	D	E	F	G	H
1								
2	NPV Criteria for Capital Budgeting Decisions	interest rate	8%					
3			Year 0	Year 1	Year 2	Year 3	Year 4	
4	Project A	Cost	80,000	20,000	0	0	0	
5		Return	0	20,000	30,000	50,000	50,000	
6	Project B	Cost	50,000	50,000	0	0	0	
7		Return	0	40,000	60,000	30,000	10,000	
8								
9			Year 0	Year 1	Year 2	Year 3	Year 4	NPV
10	Project A	Net Return	-80,000.00	0.00	30,000.00	50,000.00	50,000.00	$22,163.27
11		PV	-80,000.00	0.00	25,720.16	39,691.61	36,751.49	$22,163.27
12	Project B	Net Return	-50,000.00	-10,000.00	60,000.00	30,000.00	10,000.00	$23,346.34
13		PV	-50,000.00	-9,259.26	51,440.33	23,814.97	7,350.30	$23,346.34
14								
15								

Fig. 18.8 Excel calculation results for NPV method

The internal rate of return (IRR, r) is the discount rate which equates the discounted cash flows from a project to its investment. Thus, one must solve iteratively for the r in Eq. (18.9)

$$\sum_{t=1}^{N} \frac{CF_t}{(1+r)^t} = I \tag{18.9}$$

where

CFt = Net Cash flow (positive or negative) in period t,
I = Initial investment,
N = Life of the project.
r = the internal rate of return.

In addition, we can use Excel function IRR to calculate the internal rate of return. IRR is a function to calculate the internal rate of return which is the rate of return received for an investment consisting of payments (negative values) and income (positive values) that occur at regular periods. The IRR function in Cell I10 is

$$= \text{IRR}(\text{C10} : \text{G10})$$

Based upon the IRR function in Fig. 18.7, the IRR results in terms of excel calculations are shown in Fig. 18.9.

18.8 Summary

In this chapter, we have introduced the concept of the present value of a future receipt. For each dollar to be received in t years at an annual interest rate over t years of r_t, the present value is

$$PV = \frac{1}{(1+r_t)^t}$$

The rationale is that at interest rate r_t, present value is the amount that would need to be deposited now to receive one dollar in t years.

Using the concept of present values, we can evaluate an investment for which returns are to be received in the future. Denoting C_0, C_1, C_2, … C_n as the dollar returns in current and future years, and r_t as the $_t$-year annual interest rate, net present value is given as

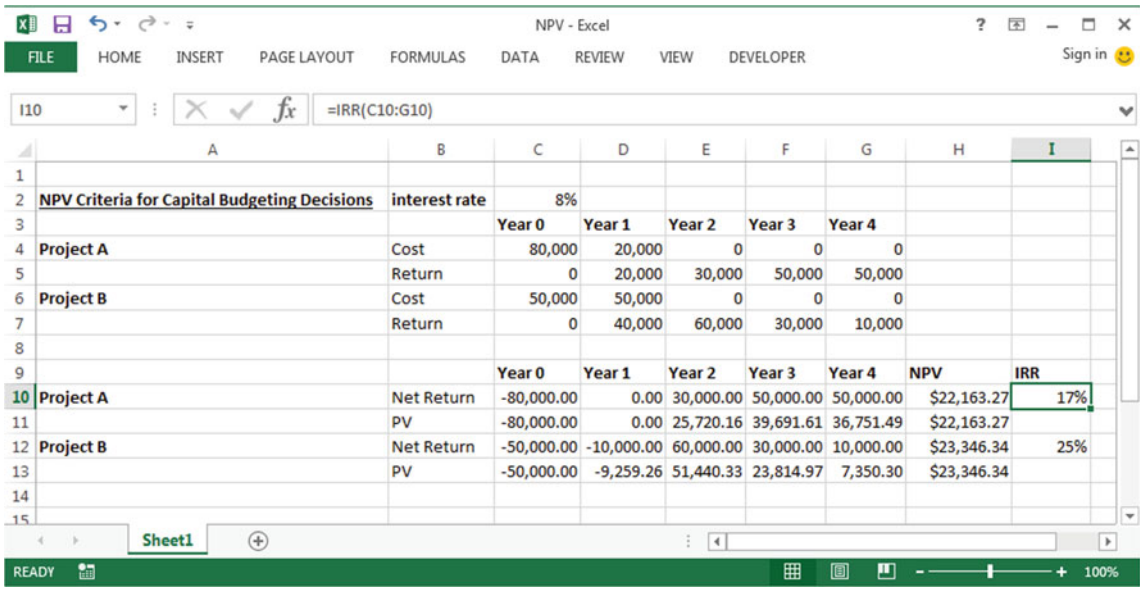
$$NPV = \sum_{t=0}^{N} \frac{C_t}{(1+r_t)^t}$$

We have seen that net present value is a basic tool for financial management decision-making. Under fairly reasonable assumptions, since stockholders have access to the capital market, it follows that to act in the interests of existing stockholders, the objective of management should be to maximize the net present value of the corporation.

Appendix 18A

Three Hypotheses about Inflation and the Firm's Value

We began this chapter by asking whether you would prefer to receive $1,000 today or $1,000 a year from now. One reason for selecting the first option is that, as a result of



	A	B	C	D	E	F	G	H	I
1									
2	NPV Criteria for Capital Budgeting Decisions	interest rate	8%						
3			Year 0	Year 1	Year 2	Year 3	Year 4		
4	Project A	Cost	80,000	20,000	0	0	0		
5		Return	0	20,000	30,000	50,000	50,000		
6	Project B	Cost	50,000	50,000	0	0	0		
7		Return	0	40,000	60,000	30,000	10,000		
8									
9			Year 0	Year 1	Year 2	Year 3	Year 4	NPV	IRR
10	Project A	Net Return	-80,000.00	0.00	30,000.00	50,000.00	50,000.00	$22,163.27	17%
11		PV	-80,000.00	0.00	25,720.16	39,691.61	36,751.49	$22,163.27	
12	Project B	Net Return	-50,000.00	-10,000.00	60,000.00	30,000.00	10,000.00	$23,346.34	25%
13		PV	-50,000.00	-9,259.26	51,440.33	23,814.97	7,350.30	$23,346.34	
14									
15									

Fig. 18.9 The excel calculation results for IRR

inflation, $1,000 will buy less in a year than it does today. In this appendix, we explore the possible effects of inflation on a firm's value. According to Van Horne and Glassmire (1972), unanticipated inflation affects the firm in three ways, characterized by the following hypotheses:

1. Debtor-creditor hypothesis.
2. Tax-effects hypothesis.
3. Operating income hypothesis.

The *debtor-creditor hypothesis* postulates that the impact of unanticipated inflation depends on a firm's net borrowing position. In periods of high inflation, fixed money amounts borrowed today will be repaid in the future in a currency with lower purchasing power. Thus, while the rate of interest on the loan reflects expected inflation rates over the term of the loan, a higher than anticipated rate of inflation should result in a transfer of wealth from creditors to debtors. Conversely, if the inflation rate turns out to be lower than expected, wealth is transferred from debtors to creditors. Hence, according to the debtor-creditor hypothesis, a higher than anticipated rate of inflation should, all other things being equal, raise the value of firms with heavy borrowings.

The *tax-effects hypothesis* concerns the influence of inflation on those firms with depreciation and inventory tax shields. Since these shields are based on historical costs, their real values decline with inflation. Hence, unanticipated inflation should lower the value of the firms with such shields. The magnitude of these tax effects could be very high indeed. For example, Feldstein and Summers (1979) estimated that the use of depreciation and inventory accounting on a historical cost basis raised corporate tax liabilities by $26 billion in 1977.

In principle, the effects of general inflation should only be felt when parties are forced to comply with nominal contracts, the terms of which fail to anticipate inflation. Hence, in theory, wealth transfers caused by general inflation should be due primarily to the debtor-creditor or tax-effects hypothesis discussed above. Apart from these considerations, if all prices move in unison, real profits should not be affected. Nevertheless, there is strong empirical evidence of a negative association between corporate profitability and the general inflation rate. One possible explanation, called the *operating income hypothesis*, is that high inflation rates lead to restrictive government fiscal and monetary policies, which, in turn, depress the level of business activity, and hence profits. Further, operating income may be adversely affected if prices of inputs, such as labor and materials, react more quickly to inflationary trends than prices of outputs. Viewed in this light, we might expect firms to react differently to inflation, depending on the reaction speed in the markets in which the firms operate.

Van Horne and Glassmire suggest that, of these three effects of unanticipated inflation on the value of the firm, the operating income effect is likely to dominate. Some support for this contention is provided by French et al. (1983), who find that debtor-creditor effects and tax effects are rather small.

Appendix 18B

Book Value, Replacement Cost, and Tobin's q

An objective of financial management should be to raise the firm's net present value. We have not, however, discussed what constitutes a firm's value.

An accounting measure of value is the total value of all a firm's assets, including plant and equipment, plus inventory. Generally, in a firm's accounts, the book values of the assets are reported. However, this is an inappropriate measure for two reasons. First, it takes no account of the growth rate of capital goods prices since the assets were acquired, and second, it does not account for the economic depreciation of those assets. Therefore, in considering a firm's value, it is preferable to consider current accounting measures that incorporate inflation and depreciation. The relevant measure of accounting value, then, is replacement cost, which is the cost today of purchasing assets of the same vintage as those currently held by the firm.

However, this accounting concept of value is not the one used in financial management, as it does not incorporate the potential for future earnings through the exploitation of productive investment opportunities. If this broader definition is considered, the value of a firm will depend not only on the accounting value of its assets, but also on the ability of management to make productive use of those assets. In finance theory, the relevant concept of values of common stock, preferred stock, and debt, all of which are determined by the financial markets.[3]

The ratio of a firm's market value to the replacement cost of its assets is known as Tobin's q, as shown in Tobin and Brainard (1977). One reason for looking at this relationship is that if the acquisition of new capital adds more to the firm's value than the cost of acquiring that capital—that is, it has a positive NPV—then shareholders immediately benefit from the acquisition. On the other hand, if the addition of new capital adds less than its cost to market value, shareholders would be better off if the money were distributed to them as dividends. Therefore, the relationship between market value and replacement cost is crucial in financial management decision-making.

Appendix 18C

Continuous Compounding and Continuous Discounting

In this appendix, we will show how continuous compounding and discounting can be theoretically derived. In addition, we also give some examples to show how these two processes to the real world.

[3] In the next chapter, we will discuss the valuation of the financial instruments.

Continuous Compounding

In the general calculation of interest, the amount of interest earned plus the principal is

$$principal + interest = principal \times \left(1 + \frac{r}{m}\right)^{T} \qquad (18.10)$$

where r = annual interest rate, m = number of compounding periods per year, and T = number of compounding periods (m) times the number of years N.

There are three variables: the initial amount of principal invested, the periodic interest rate, and the time period of the investment. If we assume that you invest \$100 for 1 year at 10% interest, you will receive the following:

$$principal + interest = \$100\left(1 + \frac{.10}{1}\right)^{1} = \$110$$

For a given interest rate, the greater frequency with which interest is compounded affects the interest and the time variables of the above equation; the interest per period decreases, but the number of compounding periods increases. The greater the frequency with which interest is compounded, the larger the amount of interest earned. For interest compounded annually, semiannually, quarterly, monthly, weekly, daily, hourly, or continuously, we can see the increase in the amount of interest earned as follows:

$$principal + interest = P_0\left(1 + \frac{r}{m}\right)^{T}$$

Annual	\$110	=	$100\left(1+\frac{.10}{1}\right)1$
Semiannual	110.25	=	$100\left(1+\frac{.10}{2}\right)2$
Quarterly	110.38	=	$100\left(1+\frac{.10}{4}\right)4$
Monthly	110.47	=	$100\left(1+\frac{.10}{12}\right)12$
Weekly	110.51	=	$100\left(1+\frac{.10}{52}\right)52$
Daily	110.52	=	$100\left(1+\frac{.10}{365}\right)365$
Hourly	110.52	=	$100\left(1+\frac{.10}{8760}\right)8760$
Continuously	110.52	=	$100\left(e^{.1(1)}\right) = 100(2.7183)^{.1}$

In the case of continuous compounding, the term $\left(1+\frac{r}{m}\right)^{T}$ goes to e^{rN} as m gets infinitely large. To see this, we start with

$$P_0 + I = P_0\left(1 + \frac{r}{m}\right)^{T}$$

where $T = m(N)$ and N = number of years.

If we multiply T by r/r, we can rearrange Eq. 18.10 as follows:

$$P_0 + I = P_0\left[1 + \frac{r}{m}\right]^{\frac{mNr}{r}} = P_0\left[\left(1 + \frac{r}{m}\right)^{\frac{m}{r}}\right]^{Nr} \tag{18.11}$$

Let $x = m/r$, and substitute this value into Eq. (18.11)

$$P_0 + I = P_0\left[(1 + \frac{1}{x})^{x}\right]^{Nr} \tag{18.12}$$

The term $(1 + 1/x)^x$ is equal to e as

$$\lim_{x\to\infty}\left(1 + \frac{1}{x}\right)^{x} = e$$

This says that as the frequency of compounding becomes instantaneous or continuous, Eq. 18.10 can be written as

$$P_N = P_0 + I = P_0 e^{rN} \tag{18.13}$$

Figure 18.10 provides graphs of the value of P = I as a function of the frequency of compounding and the number of years. We can see that for low interest rates and shorter periods, the differences between the various compounding methods are very small. However, as either r or N becomes large the difference becomes substantial. In general, as either r or N or both variables become larger, the frequency of compounding will have a greater effect on the amount of interest that is earned.

Continuous Discounting

As we have seen in this chapter, there is a relationship between calculating future values and present values. Starting from Eq. 18.10, which calculates future value, we can rearrange to find the present value

$$P_0 = \frac{P_0 + I}{\left(1 + \frac{r}{m}\right)^{T}} \tag{18.14}$$

As we mentioned earlier, as $m \to \infty$ we see that the term $(1 + r/m)^T$ goes to e^{Nr}. Rewriting Eq. 18.14

$$P_0 = \frac{P + I}{e^{Nr}} = (P_0 + I)\left(e^{-Nr}\right) \tag{18.15}$$

Equation 18.15 tells us that the present value (P_0) of a future amount (P + I) is related by the continuous discounting factor e^{-Nt}. Similarly, the present value of an annuity of future flows can be viewed as the integral of Eq. 18.15 over the relevant time period

$$P_0 = \int_0^N F_t e^{-Nr} dt \tag{18.16}$$

where F_t is the future cash flow received in period t. In fact, F_t can be viewed as a continuous cash flow. For most business organizations, it is more realistic to assume that the cast inflows and outflows occur more or less continuously throughout a given time period instead of at the end or beginning of the period as is the case with the discrete formulation of present value.

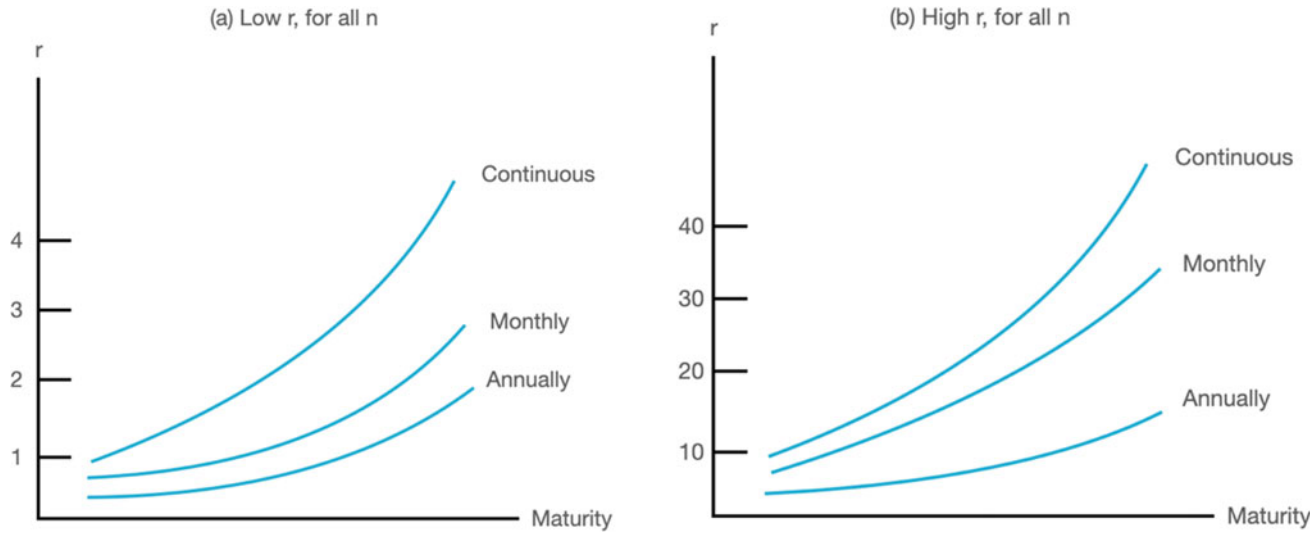


Fig. 18.10 Graphical relationships between frequency of compounding r and N

Appendix 18D: Applications of Excel for Calculating Time Value of Money

In this appendix, we will show how to use Excel to calculate: (i) the future value of a single amount, (ii) the present value of a single amount, (iii) the future value of an ordinary annuity, and (iv) the present value of an ordinary annuity.

Future Value of a Single Amount

Suppose the principal is $1000 today and the interest rate is 5% per year.

The future value of the principal can be calculated as $FV = PV(1+r)^n$, where n is the number of years.

Case 1. Suppose there is only one period, i.e. n = 1. The future value in one year will be $1000(1+5\%)^1 = 1050$.

We can use Excel to directly compute it by inputting "=B1*(1+B2)," as presented in Table 18.3

Or we can also use the function in Excel to compute the future value by inputting "=FV(B2,1, ,B1,0)"

There are five options in this function.

Rate: The interest rate per period.

Nper: The number of payment periods.

Pmt: The payment in each period; If "pmt" is omitted, we should include the "pv" argument below.

Pv: The present value. If "pv" is omitted, it is assumed to be 0. Then we should include the "pmt" argument above.

Type: The number 0 or 1 shows when payments are due. If payments are due at the end of the period, Excel sets it as 0; If payments are due at the beginning of the period, Excel sets it as 1.

The FV function gives us the same amount as what we calculate according to the formula except the sign is negative. Actually, the FV function in Excel is to compute the Future value of the principal that one party should pay back to another party. Therefore, Excel adds a negative sign to indicate the amount needed to pay back, as presented in Table 18.4.

Case 2. Now suppose there are 4 periods. The future value of $1,000 at the end of the 4th year will be $1000(1+5\%)^4 = 1215.51$.

We use two methods to compute the future value and obtain the same result.

First, we calculate it directly according to the formula, as presented in Table 18.5.

Second, we use the FV function in Excel to calculate it, as presented in Table 18.6.

Table 18.3 Future value of single period

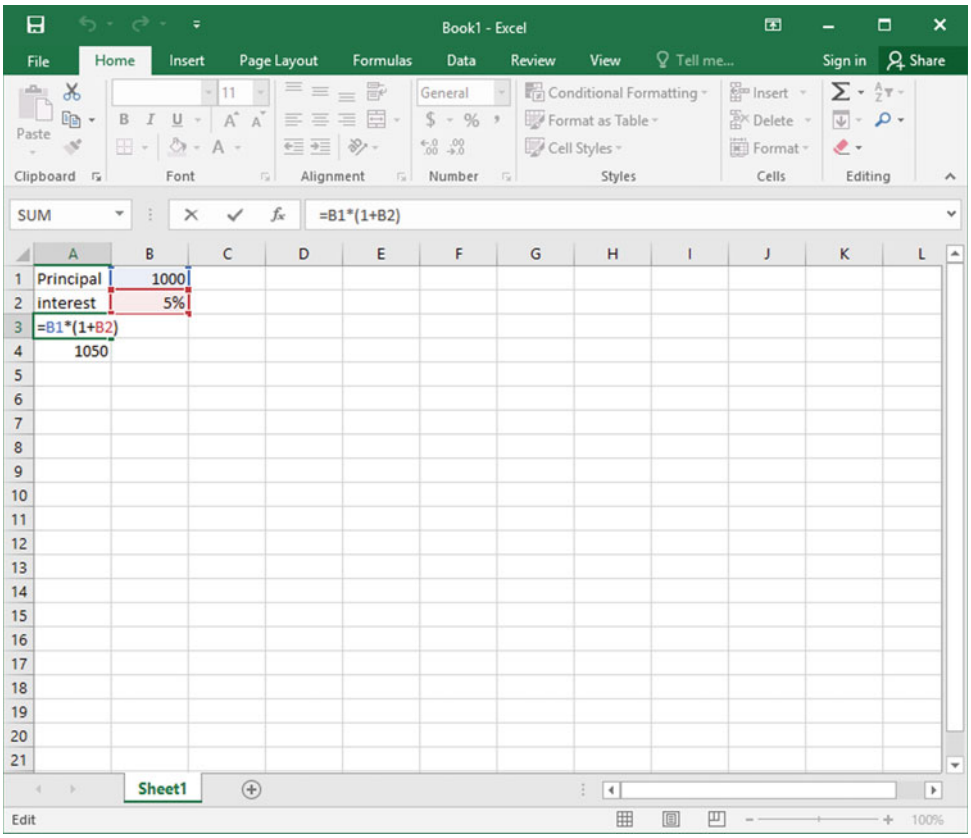
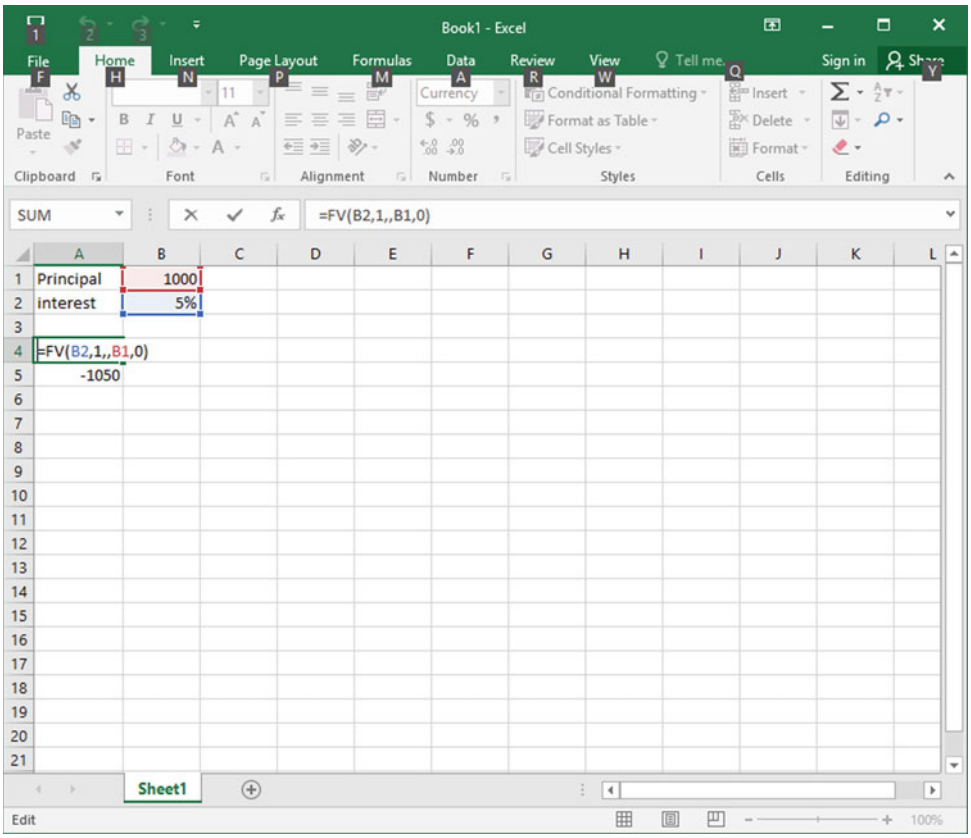


Table 18.4 Future value of single period in terms of excel formula



The FV function gives us the same amount as what we calculate according to the formula except the sign is negative. Actually, the FV function in Excel is to compute the Future value of the principal that one party should pay back to another party. Therefore, Excel adds a negative sign to indicate the amount needed to pay back.

Present Value of a Single Amount

The present value of the future sum of money can be calculated as $PV = FV/(1+r)^n$, where n is the number of years.

Case 1. Suppose a project will end in one year and it pays $1000 at the end of that year. The interest rate is 5% for one year.

The present value will be $1000/(1+5\%)^1 = 952.38$.

We can use Excel to directly compute it by inputting "=B1/(1+B2)," as presented in Table 18.7.

Or, we can use the FV function which is quite similar to the FV function we used before. The result is presented in Table 18.8.

Case 2. Suppose a project will end in four years and it would pay $1000 only at the end of the last year. The interest rate is 5% for one year.

The present value will be $1000/(1+5\%)^4 = 952.38$.

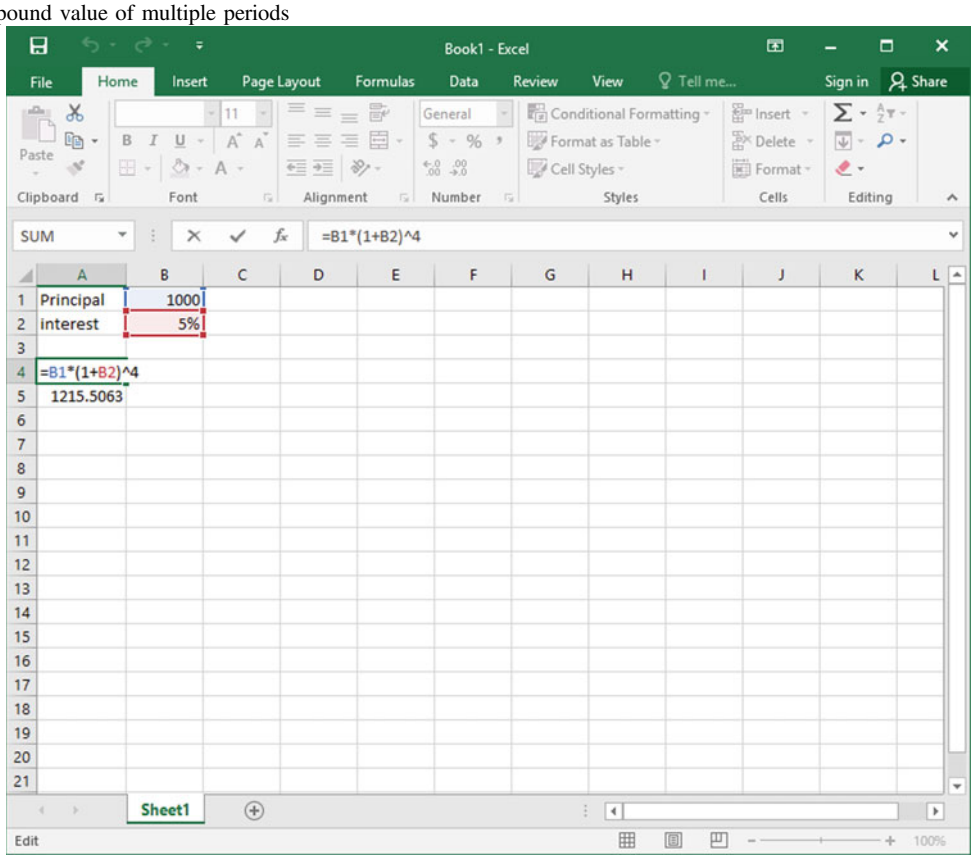
We can use Excel to directly compute it by inputting "=B1/(1+B2)^4," as presented in Table 18.9.

Or we use the PV formula in Excel by inputting "=PV (B2,4,,B1,0)," as presented in Table 18.10.

Future Value of an Ordinary Annuity

Annuity is a series of cash flow of a fixed amount for n periods of equal length. It can be divided into Ordinary Annuity (the first payment occurs at the end of period) and Annuity Due (the first payment is at the beginning of the period)

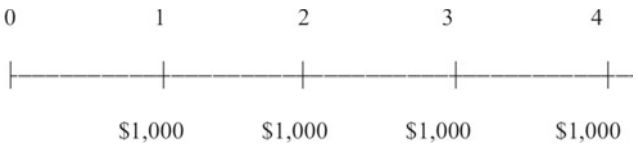
Table 18.5 Compound value of multiple periods



Case 1. Future Value of an ordinary annuity.

The formula is $FV = \sum_{k=1}^{n} PMT(1+r)^{k-1}$, *where PMT is* the payment in each period.

Suppose a project will pay you $1,000 at the end of each year for 4 years at 5% annual interest, and the following graph shows the process:



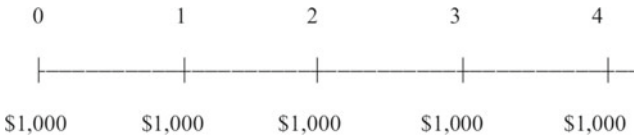
We still use two methods to calculate the future value of this ordinary annuity. First, we directly use the formula to compute it and obtain the value of 4310.125. The result is presented in Table 18.11.

Then we use the FV function in Excel to compute the future value and obtain 4310.125. Hence, the two methods give us the same result, as presented in Table 18.12.

Case 2. Future Value of an Annuity Due.

The formula is $FV = \sum_{k=1}^{n} PMT(1+r)^{k}$, *where PMT is* the payment in each period.

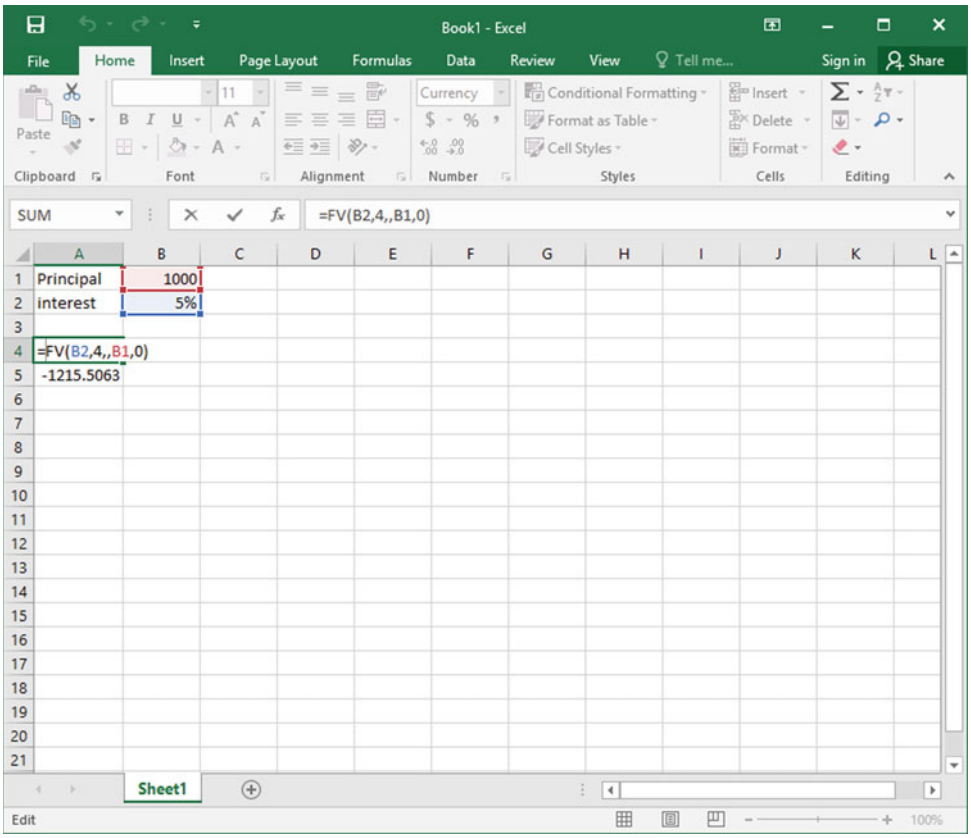
Suppose a project will pay you $1,000 at the beginning of each year for 4 years at 5% annual interest, and the following graph shows the process:



First, we directly use the formula to compute it and obtain the future value of 4525.631. The result is presented in Table 18.13.

Then we use the FV function in Excel to compute the future value and obtain 4525.63. The only difference between calculating annuity due and computing ordinary annuity is to choose "1" in "type term" of the FV function rather than to choose "0". The two methods give us the same result, as presented in Table 18.14.

Table 18.6 Compound value of multiple period in terms of excel formula



Present Value of an Ordinary Annuity

Case 1. Present Value of an ordinary annuity.

The formula is $FV = \sum_{k=1}^{n} PMT/(1+r)^k$, *where PMT is* the payment in each period.

Suppose a project will pay you $1500 at the end of each year for 4 years at 5% annual interest.

According to this formula, we directly input "=B1/(1+B5)^4+B2/(1+B5)^3+B3/(1+B5)^2+B4/(1+B5)^1" to get the present value of 5318.93, as presented in Table 18.15.

In addition, we can use the PV function in Excel directly and obtain the same amount as above, as presented in Table 18.16.

Case 2. Present Value of an annuity due.

The formula is $PV = \sum_{k=0}^{n-1} PMT/(1+r)^k$, *where PMT is* the payment in each period.

Suppose a project will pay you $1500 at the end of each year for 4 years at 5% annual interest.

According to this formula, we directly input "=B1/(1+B5)^3+B2/(1+B5)^2+B3/(1+B5)^1+B4/(1+B5)^0" to get the present value of 5584.87, as presented in Table 18.17.

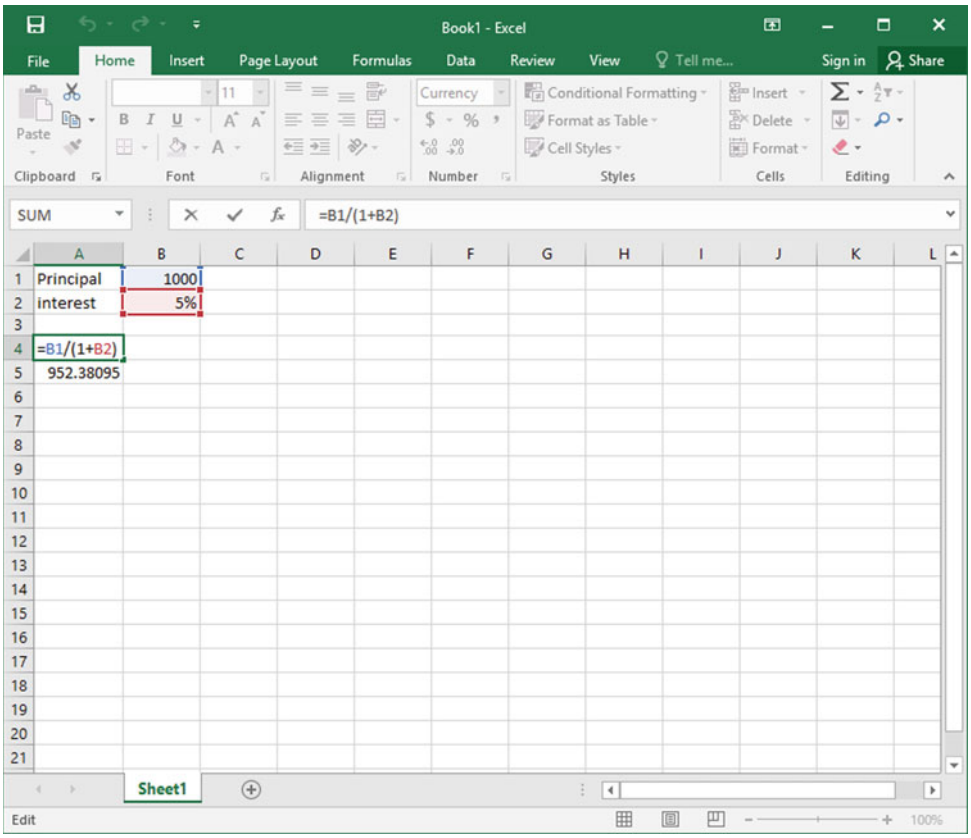
Similarly, the PV function gives us the same result as presented in Table 18.18.

Case 3. An annuity that pays forever (Perpetuity).

$$PV = \frac{PMT}{r}$$

In Excel, we directly input "=B1/B2" to get PV = 30,000, as presented in Table 18.19.

Table 18.7 Present value for single period



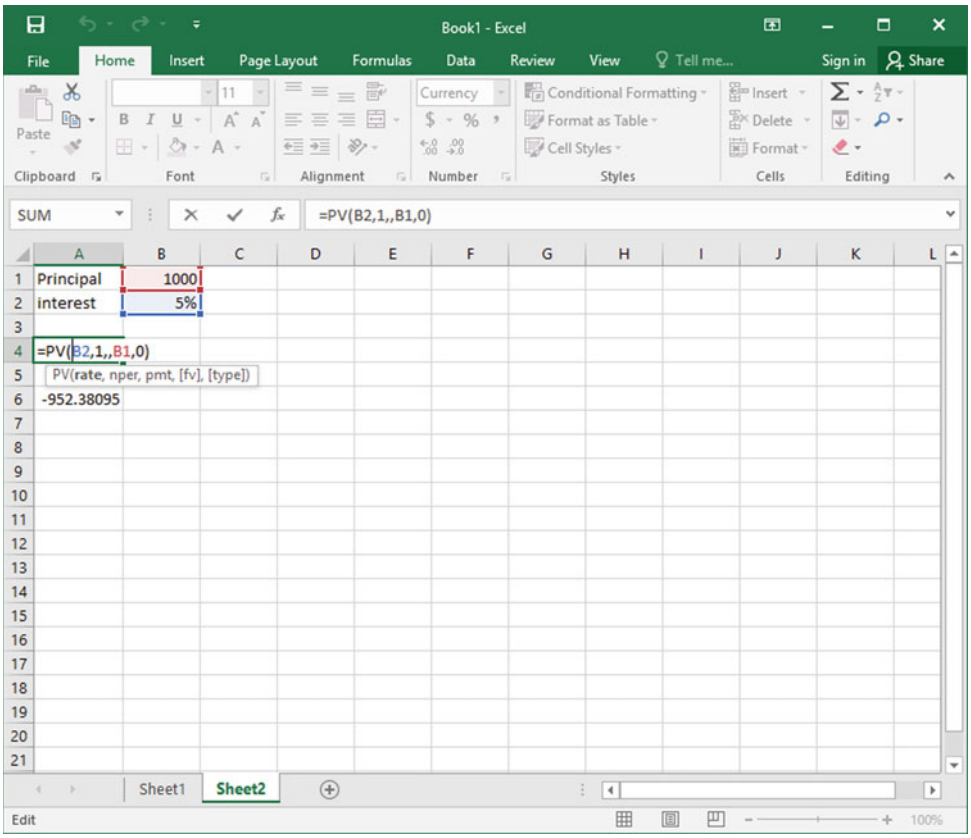
Appendix 18E: Tables of Time Value of Money

See Tables 18.20, 18.21, 18.22 and 18.23.

Questions and Problems

1. Define following terms:
   a. Present values and future value.
   b. Compounding and discounting process.
   c. Discrete versus continuous compounding.
   d. Liquidity preference.
   e. Debtor-creditor hypothesis.
   f. Operating income hypothesis.
2. Discuss how the following four tables listed at the end of the book are compiled.
   a. Present value table.
   b. Future value table.
   c. Present value of annuity table.
   d. Compound value of annuity table.
3. Suppose that $100 is invested today at an annual interest rate of 12% for a period of 10 years. Calculate the total amount received at the end of this term as follows:
   a. Interest compounded annually.
   b. Interest compounded semiannually.
   c. Interest compounded monthly.
   d. Interest compounded continuously.
4. What is the present value of $1,000 paid at the end of one year if the appropriate interest rate is 15%?
5. CF_0 is initial outlay on an investment, and CF_1 and CF_2 are the cash flows at the end of the next two years. The notation r is the appropriate interest rate. Answer the following:
   a. What is the formula for the net present value?

Table 18.8 Present value of single period in terms of excel formula



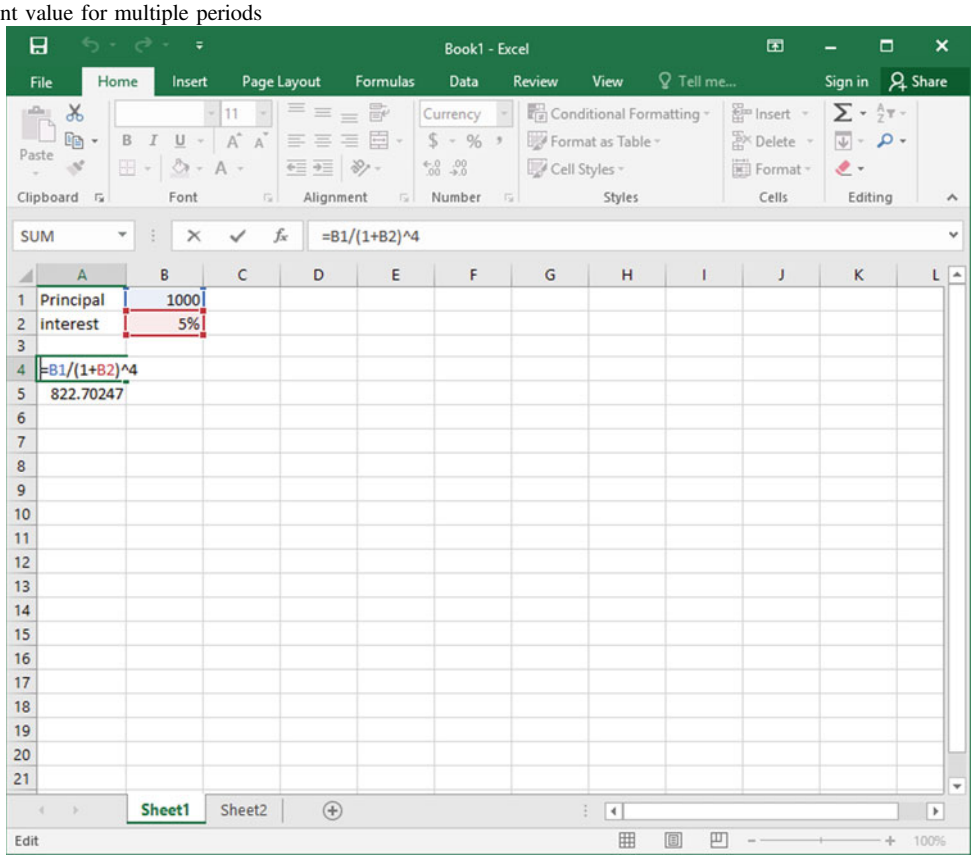
b. Find NPV when CF_0 = -$1,000, CF_1 = $600, CF_2 = $700, and r = 10%.
c. If the investment is risk-free, what rate is used as a proxy for r?

6. ABC Company is considering two projects for a new investment, as shown in table below (in dollars). Which is better if ABC uses the NPV rule to select between the projects? Suppose that the interest rate is 12%.

		Year 0	Year 1	Year 2	Year 3	Year 4
Project A	Costs	10,000	0	0	0	0
	Returns	0	0	0	1,000	20,000
Project B	Costs	5,000	5,000	0	0	0
	Returns	0	10,000	5,000	3,000	2,000

7. Suppose that C dollars is to be received at the end of each of the next N years, and that the annual interest rate is r over the N years.
   a. What is the formula for the present value of the payments?
   b. Calculate the present value of the payments when C = $1,000, r = 10%, and N = 50.
   c. Would you pay $10,000 now ($t$ = 0) for the annuity of $1,000 to be received every year for the next 50 years?
   d. If $1,000 per year is to be received forever, what is the present value of those cash flow streams?
8. Mr. Smith is 50 years old and his salary will be $40,000 next year. He thinks his salary will increase at an annual rate of 10% until his retirement at age 60.

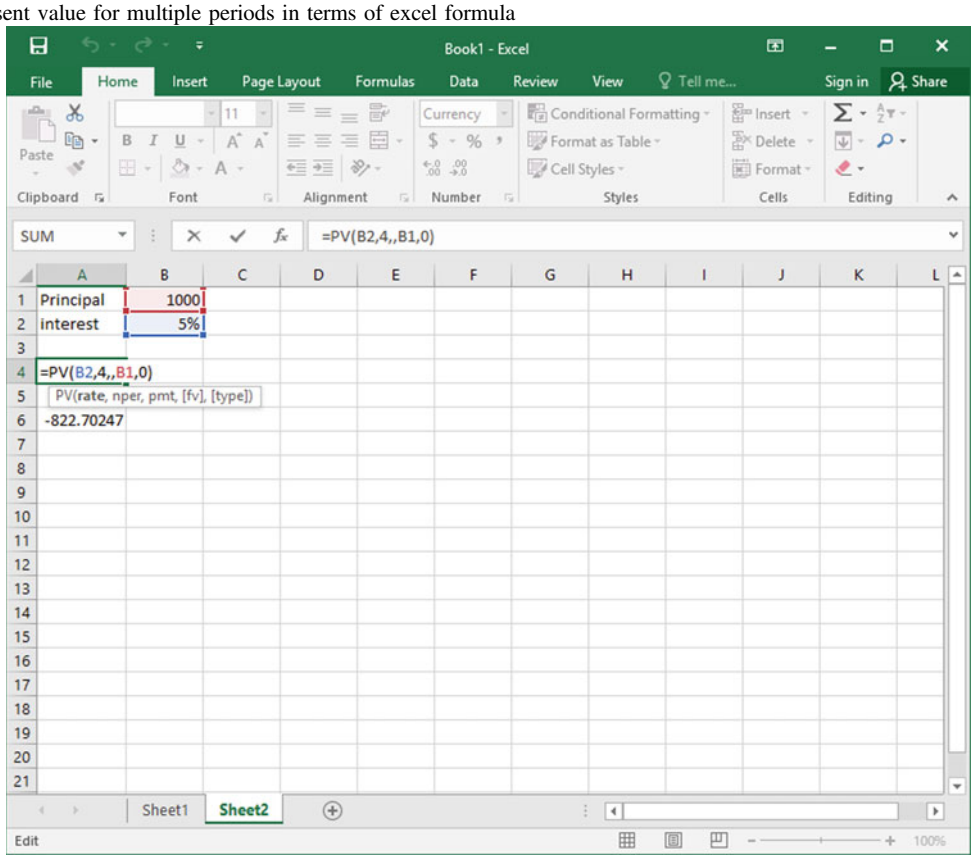
Table 18.9 Present value for multiple periods



a. If the appropriate interest rate is 8%, what is the present value of these future payments?
b. If Mr. Smith saves 50% of his salary each year and invests these savings at the annual interest rate of 12%, how much will he save by age 60?

9. Suppose someone pays you $10 at the beginning of each year for 10 years, expecting that you will pay back a fixed amount of money each year forever commencing at the beginning of Year 11. For a fair deal when annual interest rate is 10% how much should the annual fixed amount of money be?
10. ZZZ Bank agrees to lend ABC Company $10,000 today in return for the company's promise to pay back $25,000 five years from today. What annual rate of interest is the bank charging the company?
11. Which of the following would you choose if the current interest rate is 10%?
    a. $100 now.
    b. $12 at the end of each year for the next ten years.
    c. $10 at the end of each year forever.
    d. $200 at the end of the seventh year.
    e. $50 now and yearly payments decreasing by 50% a year forever.
    f. $5 now and yearly payments increasing by 5% a year forever.
12. You are given an opportunity to purchase an investment which pays no cash in years 0 through 5, but will pay $150 per year beginning in year 6 and continuing forever. Your required rate of return for this investment is 10%. Assume all cash flows occur at the end of each year.

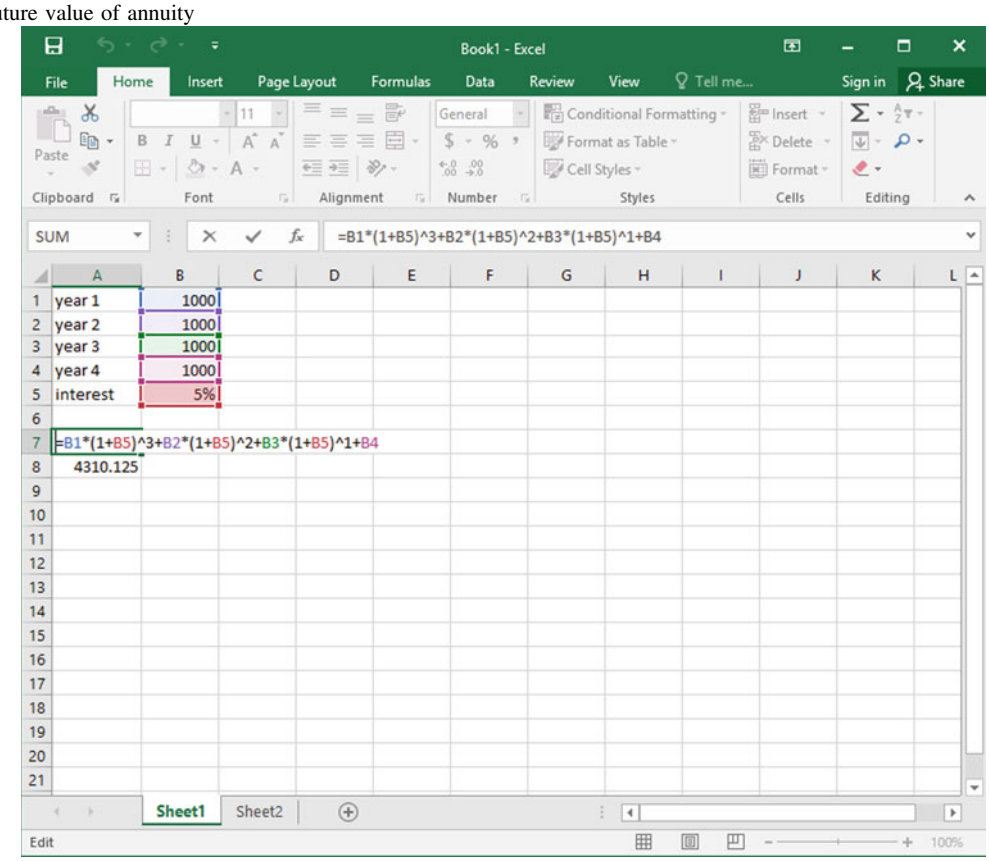
Table 18.10 Present value for multiple periods in terms of excel formula



a. Show how much you should be willing to pay for the investment at the end of year 5.
b. How much should you be willing to pay for the investment now?

13. If you deposit $100 at the end of each year for the next five years, how much will you have in your account at the end of five years if the bank pays 5% interest compounded annually?
14. If you deposit $100 at the beginning of each year for the next five years, how much will you have in your account at the end of five years if the bank pays 5% interest compounded annually?
15. If you deposit $200 at the end of each year for the next 10 years and interest is compounded continuously at an annual quoted rate of 5%, how much will you have in your account at the end of 10 years?
16. Your mother is about to retire. Her firm has given her the option of retiring with a lump sum of $50,000 now or an annuity of $5,200 per year for 20 years. Which is worth more if your mother can earn an annual rate of 6% on similar investments elsewhere?
17. You borrow $6145 now and agree to pay the loan off over the next ten years in ten equal annual payments, which include principal and 10% annually compounded interest on the unpaid balance. What will your annual payment be?

Table 18.11 Future value of annuity



18. Ms. Mira Jones plans to deposit a fixed amount at the end of each month so that she can have $1000 once year hence. How much money would she have to save every month if the annual rate of interest is 12%?
19. You are planning to buy an annuity at the end of five years from now. The annuity will pay $1500 per quarter for the next four years after you buy it (t = 6 thru 9). How much would you have to pay for this annuity in year 5 if the annual rate of interest is 8%?
20. Air Control Corporation wants to borrow $22,500. The loan is repayable in 12 equal monthly installments of $2,000. The corporate policy is to pay no more than an annual interest rate of 10%. Should Air Control accept this loan?

Table 18.12 Future value of annuity in terms of excel formula

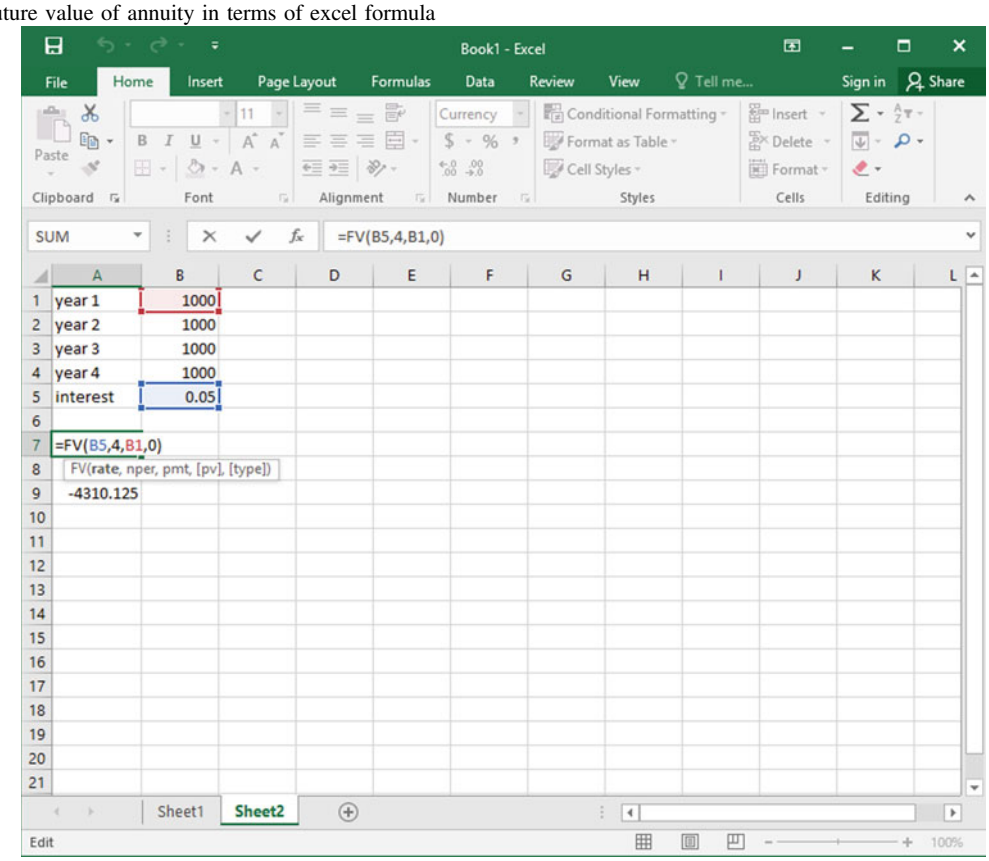


Table 18.13 Future value of annuity due

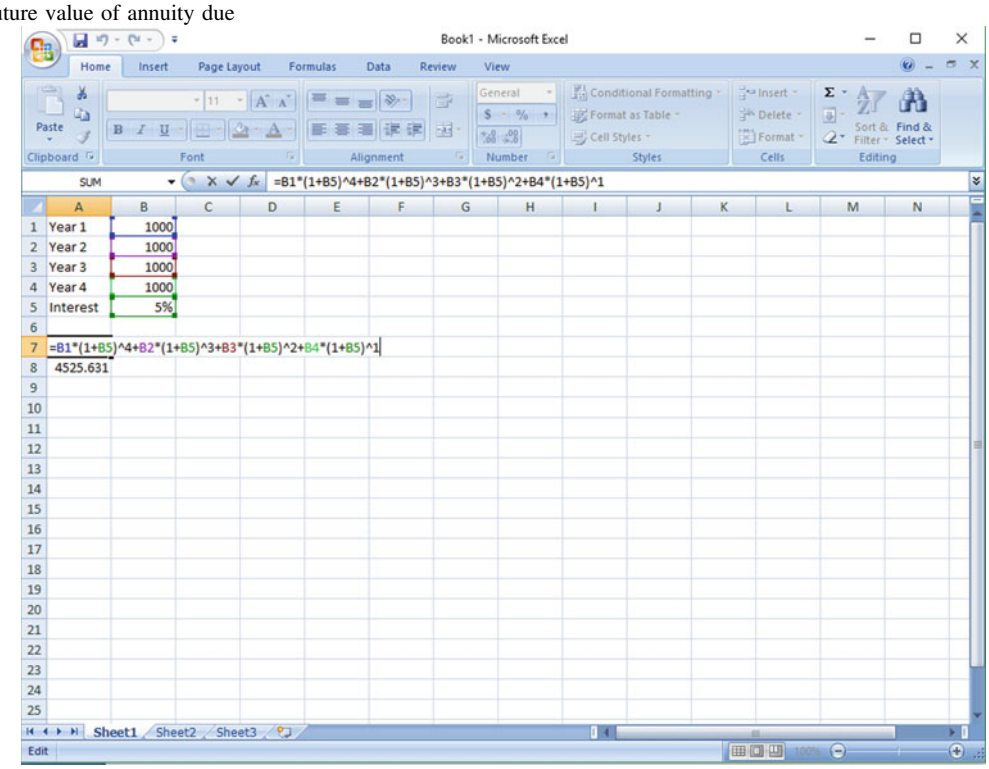


Table 18.14 Future value of annuity due in terms of excel formula

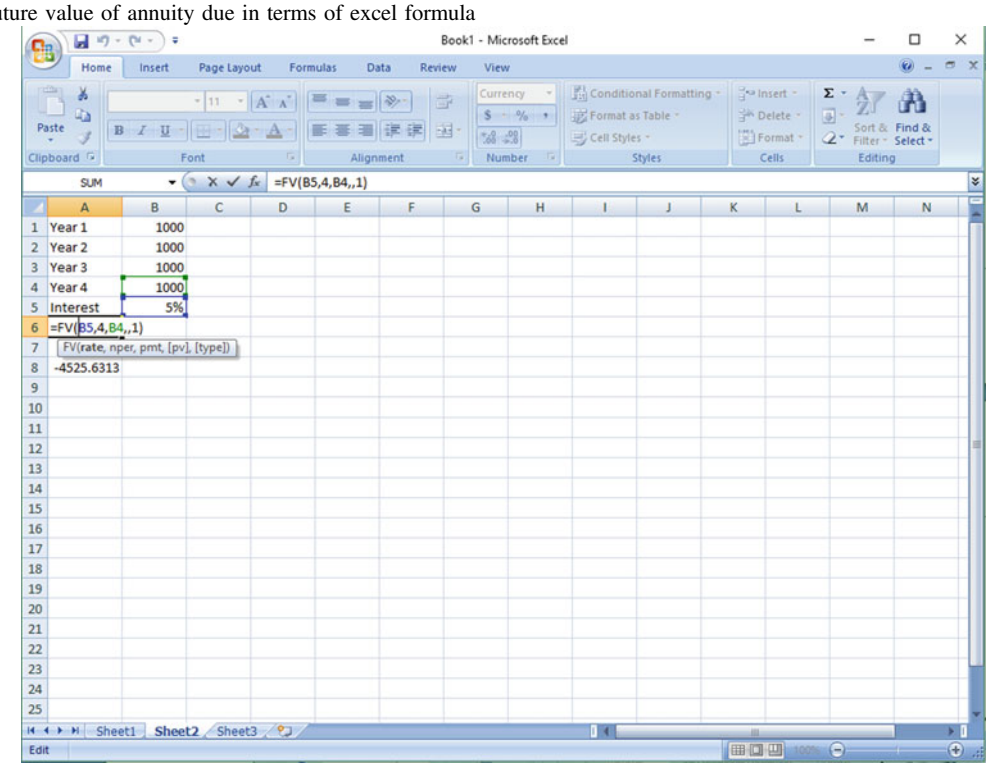


Table 18.15 Present value of annuity

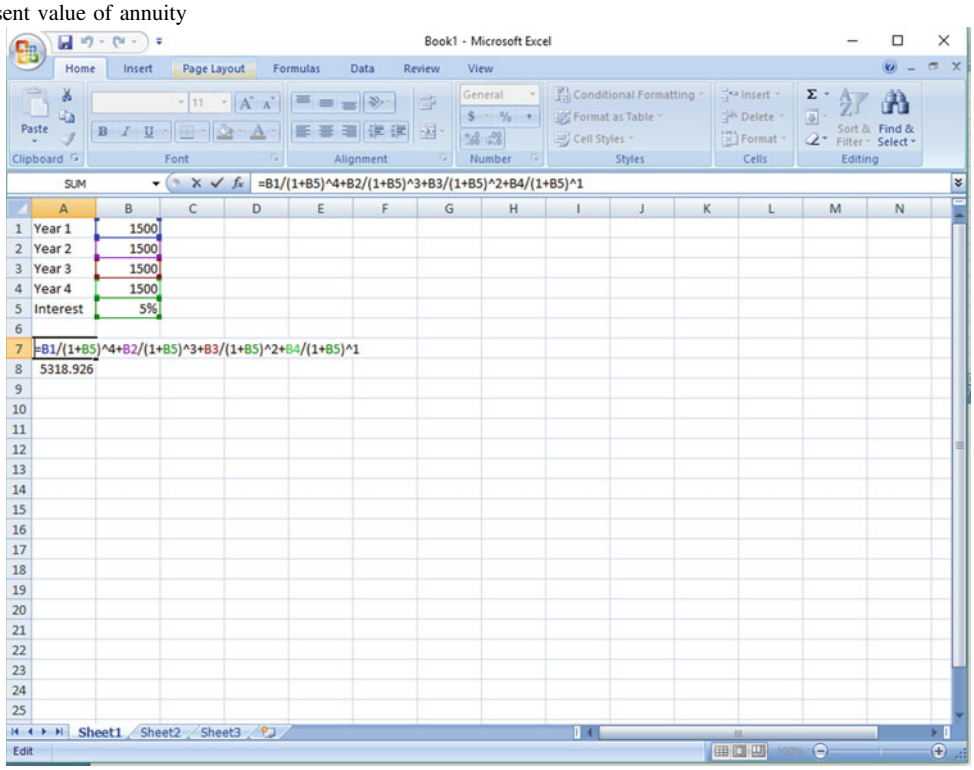


Table 18.16 Present value of annuity in terms of excel formula

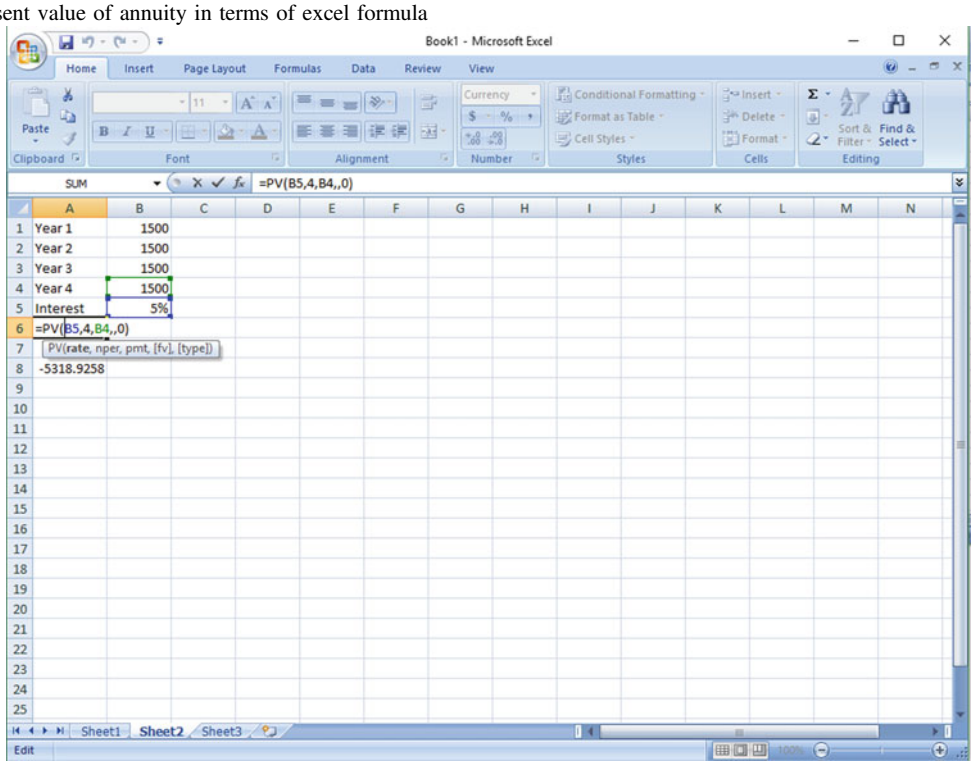


Table 18.17 Present value of annuity due

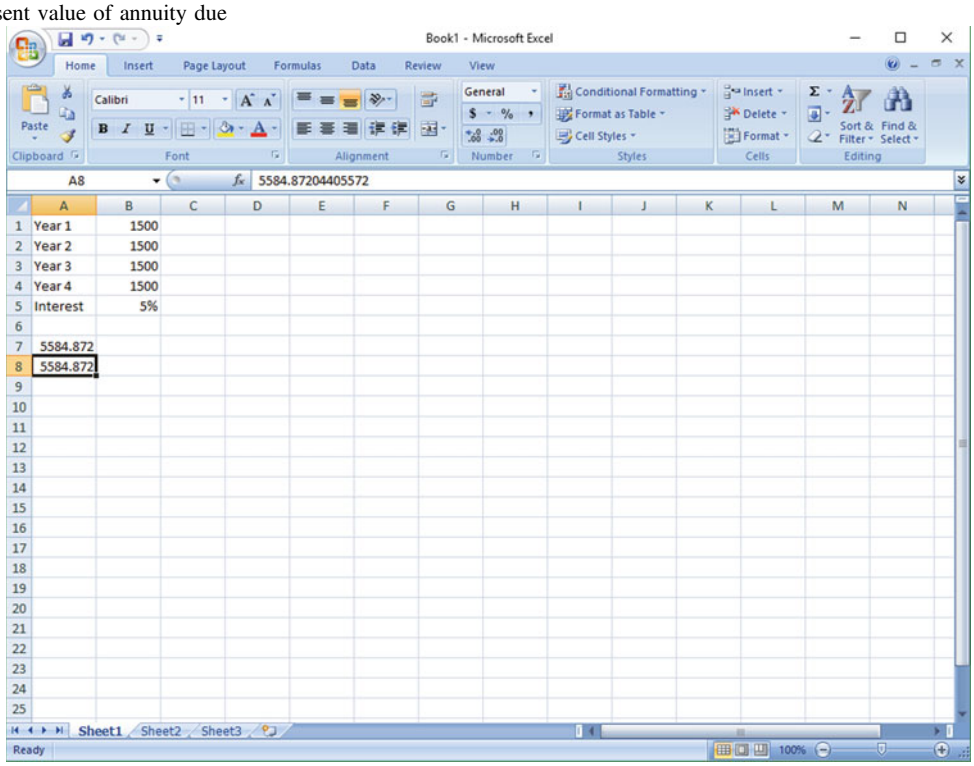


Table 18.18 Present value of annuity due in terms of excel formula

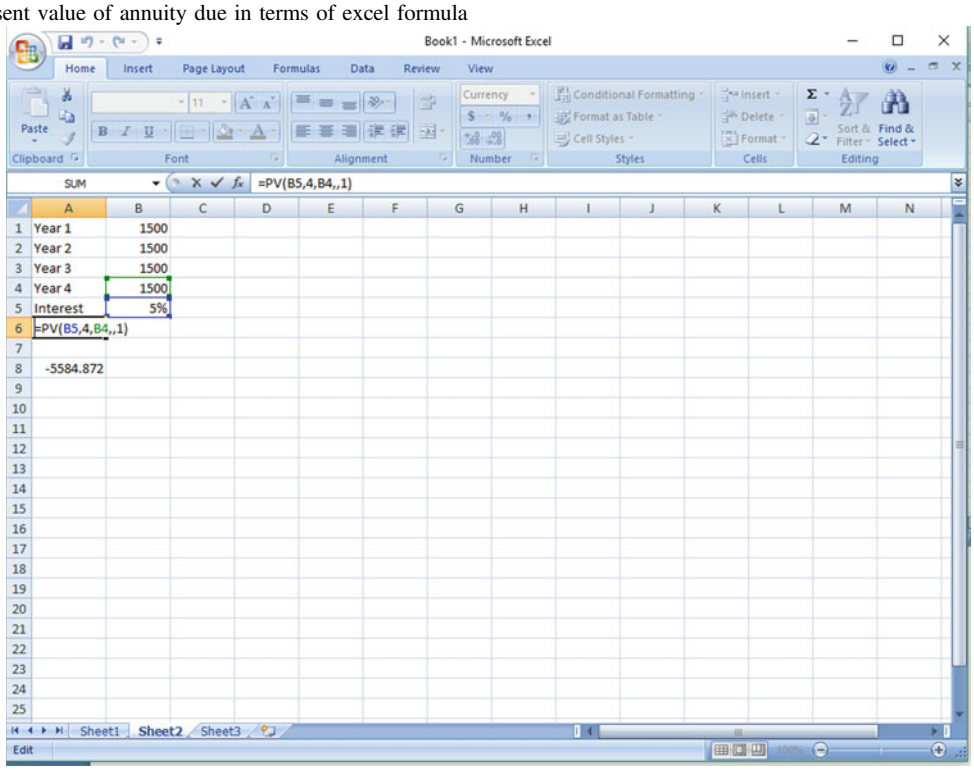


Table 18.19 Present value of perpetuity

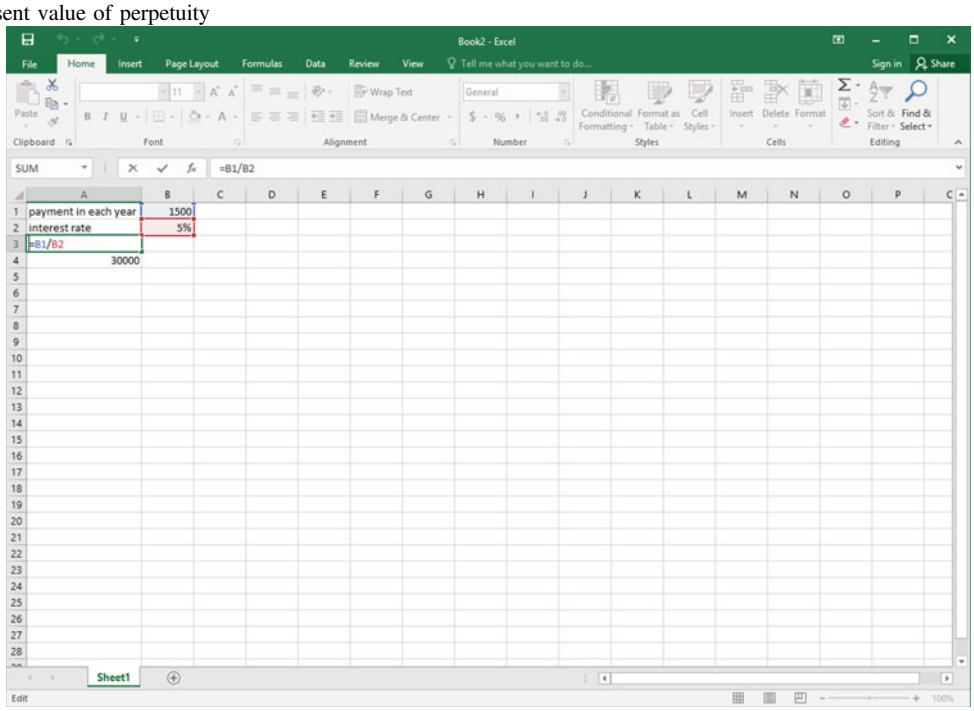


Table 18.20 Future value table (discrete annually compounded)

t/ r	2%	4%	6%	8%	10%	12%	14%	16%	18%	20%
1	1.0200	1.0400	1.0600	1.0800	1.1000	1.1200	1.1400	1.1600	1.1800	1.2000
2	1.0404	1.0816	1.1236	1.1664	1.2100	1.2544	1.2996	1.3456	1.3924	1.4400
3	1.0612	1.1249	1.1910	1.2597	1.3310	1.4049	1.4815	1.5609	1.6430	1.7280
4	1.0824	1.1699	1.2625	1.3605	1.4641	1.5735	1.6890	1.8106	1.9388	2.0736
5	1.1041	1.2167	1.3382	1.4693	1.6105	1.7623	1.9254	2.1003	2.2878	2.4883
6	1.1262	1.2653	1.4185	1.5869	1.7716	1.9738	2.1950	2.4364	2.6996	2.9860
7	1.1487	1.3159	1.5036	1.7138	1.9487	2.2107	2.5023	2.8262	3.1855	3.5832
8	1.1717	1.3686	1.5938	1.8509	2.1436	2.4760	2.8526	3.2784	3.7589	4.2998
9	1.1951	1.4233	1.6895	1.9990	2.3579	2.7731	3.2519	3.8030	4.4355	5.1598
10	1.2190	1.4802	1.7908	2.1589	2.5937	3.1058	3.7072	4.4114	5.2338	6.1917
11	1.2434	1.5395	1.8983	2.3316	2.8531	3.4785	4.2262	5.1173	6.1759	7.4301
12	1.2682	1.6010	2.0122	2.5182	3.1384	3.8960	4.8179	5.9360	7.2876	8.9161
13	1.2936	1.6651	2.1329	2.7196	3.4523	4.3635	5.4924	6.8858	8.5994	10.6993
14	1.3195	1.7317	2.2609	2.9372	3.7975	4.8871	6.2613	7.9875	10.1472	12.8392
15	1.3459	1.8009	2.3966	3.1722	4.1772	5.4736	7.1379	9.2655	11.9737	15.4070
16	1.3728	1.8730	2.5404	3.4259	4.5950	6.1304	8.1372	10.7480	14.1290	18.4884
17	1.4002	1.9479	2.6928	3.7000	5.0545	6.8660	9.2765	12.4677	16.6722	22.1861
18	1.4282	2.0258	2.8543	3.9960	5.5599	7.6900	10.5752	14.4625	19.6733	26.6233
19	1.4568	2.1068	3.0256	4.3157	6.1159	8.6128	12.0557	16.7765	23.2144	31.9480
20	1.4859	2.1911	3.2071	4.6610	6.7275	9.6463	13.7435	19.4608	27.3930	38.3376

Suppose that k dollar(s) is invested now at an interest rate of r per period, with interest compounded at the end of each period
This table gives the future value of k dollar(s) at the end of t periods for various interest rates, r, and the number of periods, t
Assume the amount of money in dollar(s) is $1

Table 18.21 Future value table (continuously compounded)

t/ r	2%	4%	6%	8%	10%	12%	14%	16%	18%	20%
1	1.0202	1.0408	1.0618	1.0833	1.1052	1.1275	1.1503	1.1735	1.1972	1.2214
2	1.0408	1.0833	1.1275	1.1735	1.2214	1.2712	1.3231	1.3771	1.4333	1.4918
3	1.0618	1.1275	1.1972	1.2712	1.3499	1.4333	1.5220	1.6161	1.7160	1.8221
4	1.0833	1.1735	1.2712	1.3771	1.4918	1.6161	1.7507	1.8965	2.0544	2.2255
5	1.1052	1.2214	1.3499	1.4918	1.6487	1.8221	2.0138	2.2255	2.4596	2.7183
6	1.1275	1.2712	1.4333	1.6161	1.8221	2.0544	2.3164	2.6117	2.9447	3.3201
7	1.1503	1.3231	1.5220	1.7507	2.0138	2.3164	2.6645	3.0649	3.5254	4.0552
8	1.1735	1.3771	1.6161	1.8965	2.2255	2.6117	3.0649	3.5966	4.2207	4.9530
9	1.1972	1.4333	1.7160	2.0544	2.4596	2.9447	3.5254	4.2207	5.0531	6.0496
10	1.2214	1.4918	1.8221	2.2255	2.7183	3.3201	4.0552	4.9530	6.0496	7.3891
11	1.2461	1.5527	1.9348	2.4109	3.0042	3.7434	4.6646	5.8124	7.2427	9.0250
12	1.2712	1.6161	2.0544	2.6117	3.3201	4.2207	5.3656	6.8210	8.6711	11.0232
13	1.2969	1.6820	2.1815	2.8292	3.6693	4.7588	6.1719	8.0045	10.3812	13.4637
14	1.3231	1.7507	2.3164	3.0649	4.0552	5.3656	7.0993	9.3933	12.4286	16.4446
15	1.3499	1.8221	2.4596	3.3201	4.4817	6.0496	8.1662	11.0232	14.8797	20.0855
16	1.3771	1.8965	2.6117	3.5966	4.9530	6.8210	9.3933	12.9358	17.8143	24.5325
17	1.4049	1.9739	2.7732	3.8962	5.4739	7.6906	10.8049	15.1803	21.3276	29.9641
18	1.4333	2.0544	2.9447	4.2207	6.0496	8.6711	12.4286	17.8143	25.5337	36.5982
19	1.4623	2.1383	3.1268	4.5722	6.6859	9.7767	14.2963	20.9052	30.5694	44.7012
20	1.4918	2.2255	3.3201	4.9530	7.3891	11.0232	16.4446	24.5325	36.5982	54.5982

Suppose that k dollar(s) is invested now at an interest rate of r per period, with interest continuously compounded
This table shows the future value of k dollar(s) invested for t periods at interest rate r per period, continuously compounded
Assume the amount of money in dollar(s) is $1

Table 18.22 Present value table—present value of a dollar received t periods in the future

t/ r	2%	4%	6%	8%	10%	12%	14%	16%	18%	20%
1	0.9804	0.9615	0.9434	0.9259	0.9091	0.8929	0.8772	0.8621	0.8475	0.8333
2	0.9612	0.9246	0.8900	0.8573	0.8264	0.7972	0.7695	0.7432	0.7182	0.6944
3	0.9423	0.8890	0.8396	0.7938	0.7513	0.7118	0.6750	0.6407	0.6086	0.5787
4	0.9238	0.8548	0.7921	0.7350	0.6830	0.6355	0.5921	0.5523	0.5158	0.4823
5	0.9057	0.8219	0.7473	0.6806	0.6209	0.5674	0.5194	0.4761	0.4371	0.4019
6	0.8880	0.7903	0.7050	0.6302	0.5645	0.5066	0.4556	0.4104	0.3704	0.3349
7	0.8706	0.7599	0.6651	0.5835	0.5132	0.4523	0.3996	0.3538	0.3139	0.2791
8	0.8535	0.7307	0.6274	0.5403	0.4665	0.4039	0.3506	0.3050	0.2660	0.2326
9	0.8368	0.7026	0.5919	0.5002	0.4241	0.3606	0.3075	0.2630	0.2255	0.1938
10	0.8203	0.6756	0.5584	0.4632	0.3855	0.3220	0.2697	0.2267	0.1911	0.1615
11	0.8043	0.6496	0.5268	0.4289	0.3505	0.2875	0.2366	0.1954	0.1619	0.1346
12	0.7885	0.6246	0.4970	0.3971	0.3186	0.2567	0.2076	0.1685	0.1372	0.1122
13	0.7730	0.6006	0.4688	0.3677	0.2897	0.2292	0.1821	0.1452	0.1163	0.0935
14	0.7579	0.5775	0.4423	0.3405	0.2633	0.2046	0.1597	0.1252	0.0985	0.0779
15	0.7430	0.5553	0.4173	0.3152	0.2394	0.1827	0.1401	0.1079	0.0835	0.0649
16	0.7284	0.5339	0.3936	0.2919	0.2176	0.1631	0.1229	0.0930	0.0708	0.0541
17	0.7142	0.5134	0.3714	0.2703	0.1978	0.1456	0.1078	0.0802	0.0600	0.0451
18	0.7002	0.4936	0.3503	0.2502	0.1799	0.1300	0.0946	0.0691	0.0508	0.0376
19	0.6864	0.4746	0.3305	0.2317	0.1635	0.1161	0.0829	0.0596	0.0431	0.0313
20	0.6730	0.4564	0.3118	0.2145	0.1486	0.1037	0.0728	0.0514	0.0365	0.0261

Suppose that k dollar(s) is to be received t periods in the future and that the rate of interest is r, with compounding at the end of each period
This table gives the present value of k dollar(s) collected at the end of t periods for various interest rates, r, and the number of periods, t
Assume the amount of money in dollar(s) is $1

Table 18.23 Present value table—present value of an annuity of a dollar per period

t/ r	2%	4%	6%	8%	10%	12%	14%	16%	18%	20%
1	0.9804	0.9615	0.9434	0.9259	0.9091	0.8929	0.8772	0.8621	0.8475	0.8333
2	1.9416	1.8861	1.8334	1.7833	1.7355	1.6901	1.6467	1.6052	1.5656	1.5278
3	2.8839	2.7751	2.6730	2.5771	2.4869	2.4018	2.3216	2.2459	2.1743	2.1065
4	3.8077	3.6299	3.4651	3.3121	3.1699	3.0373	2.9137	2.7982	2.6901	2.5887
5	4.7135	4.4518	4.2124	3.9927	3.7908	3.6048	3.4331	3.2743	3.1272	2.9906
6	5.6014	5.2421	4.9173	4.6229	4.3553	4.1114	3.8887	3.6847	3.4976	3.3255
7	6.4720	6.0021	5.5824	5.2064	4.8684	4.5638	4.2883	4.0386	3.8115	3.6046
8	7.3255	6.7327	6.2098	5.7466	5.3349	4.9676	4.6389	4.3436	4.0776	3.8372
9	8.1622	7.4353	6.8017	6.2469	5.7590	5.3282	4.9464	4.6065	4.3030	4.0310
10	8.9826	8.1109	7.3601	6.7101	6.1446	5.6502	5.2161	4.8332	4.4941	4.1925
11	9.7868	8.7605	7.8869	7.1390	6.4951	5.9377	5.4527	5.0286	4.6560	4.3271
12	10.5753	9.3851	8.3838	7.5361	6.8137	6.1944	5.6603	5.1971	4.7932	4.4392
13	11.3484	9.9856	8.8527	7.9038	7.1034	6.4235	5.8424	5.3423	4.9095	4.5327
14	12.1062	10.5631	9.2950	8.2442	7.3667	6.6282	6.0021	5.4675	5.0081	4.6106
15	12.8493	11.1184	9.7122	8.5595	7.6061	6.8109	6.1422	5.5755	5.0916	4.6755
16	13.5777	11.6523	10.1059	8.8514	7.8237	6.9740	6.2651	5.6685	5.1624	4.7296
17	14.2919	12.1657	10.4773	9.1216	8.0216	7.1196	6.3729	5.7487	5.2223	4.7746
18	14.9920	12.6593	10.8276	9.3719	8.2014	7.2497	6.4674	5.8178	5.2732	4.8122
19	15.6785	13.1339	11.1581	9.6036	8.3649	7.3658	6.5504	5.8775	5.3162	4.8435
20	16.3514	13.5903	11.4699	9.8181	8.5136	7.4694	6.6231	5.9288	5.3527	4.8696

Suppose that k dollar(s) is collected, and the interest is compounded at the end of each period
This table gives the value of k dollar(s) collected at the end of t periods for various interest rates, r, and the number of periods, t
Assume the amount of money in dollar(s) is $1

References

Feldstein, M. and L. Summers. "Inflation and the Taxation of Capital Income in the Corporate Sector," *National Tax Journal* (December 1979, pp. 445–47).

French, K., R. Ruback, and W. Schwert. "Effects of Nominal Contracting on Stock Returns," *Journal of Political Economy* 91 (February 1983, pp. 70–96).

Tobin, J. and W. C. Brainard. "Asset Markets and the Cost of Capital," *Economic Progress, Private Values, and Public Policy: Essays in Honor of William Fellner,* B. Balassa and R. Nelson, eds. (Amsterdam: North-Holland, 1977).

Van Horne, J. and W. Glassmire. "The Impact of Unanticipated Changes in Inflation on the Value of Common Stocks," *Journal of Finance* (December 1972, pp. 1083–92).

Capital Budgeting Method Under Certainty and Uncertainty

19

19.1 Introduction

Having examined some of the issues surrounding the cost of capital for a firm, it is time to address a closely related topic, the selection of investment projects for the firm.

To begin an examination of the issues in capital budgeting, we will assume certainty in both the cash flows and the cost of funds. Later, these assumptions will be relaxed to deal with uncertainty in estimation, and with the problems involved with inflation.

First, we will discuss a brief overview of the capital budgeting process in Sect. 19.2. Issues related to using cash flows to evaluate alternative projects will be discussed in Sect. 19.3. Alternative capital budgeting methods will be investigated in Sect. 19.4. A linear programming method for capital rationing will be discussed in detail in Sect. 19.5. In Sect. 19.6, we will discuss the statistical distribution method for capital budgeting under uncertainty. Simulation methods for capital budgeting under uncertainty will be discussed in Sect. 19.7. Finally, the results of this chapter will be summarized in Sect. 19.8. In Appendix 19A, linear programming method will be used to solve capital rationing. Decision tree method for investment decision will be discussed in Appendix 19B, In Appendix 19C, we will discuss Hillier's statistical distribution method for capital budgeting under uncertainty.

19.2 The Capital Budgeting Process

In his article "Myopia, Capital Budgeting and Decision Making," Pinches (1982) assessed capital budgeting from both the academic and the practitioner's point of view. He presented a framework for discussion of the capital budgeting process, which we use in this chapter.

Capital budgeting techniques can be used for very simple "operational" decisions concerning whether to replace existing equipment, or they may be used in larger, more "strategic" decisions concerning acquisition or divestiture of a firm or division, expansion into a new product line, or increasing capacity.

The dividing line between operational and strategic decisions varies greatly depending on the organization and its circumstances. The same analytical techniques can be used in either circumstance, but the amount of information required and the degree of confidence in the results of the analysis depend on whether an operational or a strategic decision is being made. Many firms do not require capital budgeting justification for small, routine, or "production" decisions. Even when capital budgeting techniques are used for operating decisions, the tendency is not to recommend projects unless upper-level management is ready to approve them. Hence, while operating decisions arc important and can be aided by capital budgeting analysis, the more important issue for most organizations is the use and applicability of capital budgeting techniques in strategic planning.

In a general sense, the capital budgeting framework of analysis can be used for many types of decisions, including such areas as acquisition, expansion, replacement, bond refinancing, lease versus buy, and working capital management. Each of these decisions can be approached from either of two perspectives: the *top-down* approach, or the *bottom-up* approach. By *top-down*, we mean the initiation of an idea or a concept at the highest management level, which then filters down to the lower levels of the organization. By *bottom-up*, we mean just the reverse.

For the sake of exposition, we will use a simple four-step process to present an overview of capital budgeting. The steps are (1) identification of areas of opportunity, (2) development of information and data for decisions regarding these opportunities, (3) selection of the best alternative or courses of action to be implemented, and (4) control or feedback of the degree of success or failure of both the project and the decision process itself. While we would expect these steps to occur sequentially, there are many circumstances where the order may be switched or the steps may occur simultaneously.

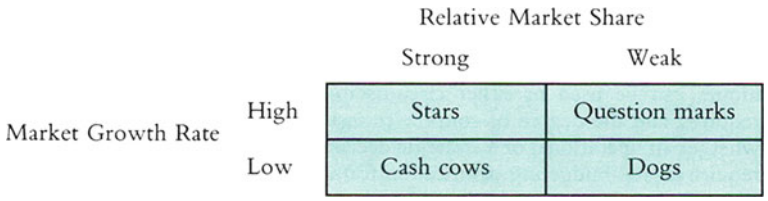
© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_19

19.2.1 Identification Phase

The identification of potential capital expenditures is directly linked to the firm's overall strategic objective; the firm's position within the various markets it serves; government fiscal, monetary, and tax policies; and the leadership of the firm's management. A widely used approach to strategic planning is based on the concept of viewing the firm as a collection, or portfolio, of assets grouped into strategic business units. This approach, called the Business Strategy Matrix, has been developed and used quite successfully by the Boston Consulting Group. It emphasizes market share and market growth rate in terms of stars, cash cows, question marks, and dogs, as shown in Exhibit 19.1.

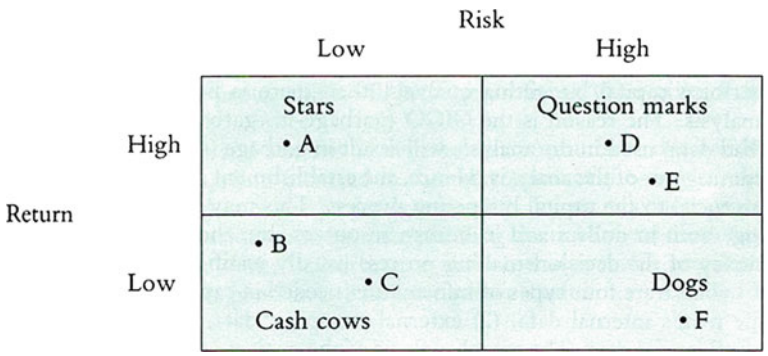
Exhibit 19.1: Boston Consulting Group, Business Strategy Matrix



Given an organization that follows some sort of strategic planning relative to the Business Strategy Matrix, the most common questions are, How does capital budgeting fit into this framework? and, Are the underlying factors of capital budgeting decisions consistent with the firm's objectives of managing market share?

There are various ways to relate the Business Strategy Matrix to capital budgeting. One of the more appealing is presented in Exhibit 19.3.

Exhibit 19.2: Capital Budgeting and the Business Strategy Matrix



This approach highlights the risk-and-return nature of both capital budgeting and business strategy. As presented, the inclusion of risk in the analysis focuses on the identification of projects such as A, which will add sufficient value (return) to the organization to justify the risk that the firm must take. Because of its high risk and low return, project F will not normally be sought after, nor will the extensive effort be made to evaluate its usefulness. Marginal projects such as B, C, D, and E require careful scrutiny. In the case of projects such as B, with low risk but also low return, there may be justification for acceptance based on capital budgeting considerations, but such projects may not fit into the firm's

strategic plans. On the other hand, projects such as E, which make strategic sense to the organization, may not offer sufficient return to justify the higher risk and so may be rejected by the capital budgeting decision-maker.

To properly identify appropriate projects for management consideration, both the firm's long-run strategic objectives and its financial objectives must be considered. One of the major problems facing the financial decision-maker today is the integration of long-run strategic goals with financial decision-making techniques that produce short-run gains. Perhaps the best way to handle this problem is in the project identification step by considering whether the investment makes sense in light of long-run corporate objectives. If the answer is no, look for more compatible projects. If the answer is yes, proceed to the next step, the development phase.

19.2.2 Development Phase

The development, or information generation, step of the capital budgeting process is probably the most difficult and most costly. The entire development phase rests largely on the type and availability of information about the investment under consideration. With limited data and an information system that cannot provide accurate, timely, and pertinent data, the usefulness of the capital budgeting process will be limited. If the firm does not have a functioning management information system (MIS) that provides the type of information needed to perform capital budgeting analysis, then there is little need to perform such analysis. The reason is the GIGO (garbage-in, garbage-out) problem; garbage (bad data) used in the analysis will result in garbage (bad or useless information) coming out of the analysis. Hence, the establishment and use of an effective MIS are crucial to the capital budgeting process. This may be an expensive undertaking, both in dollars and in human resources, but the improvement in the efficiency of the decision-making process usually justifies the cost.

There are four types of information needed in capital budgeting analysis: (1) the firm's internal data, (2) external economic data, (3) financial data, and (4) nonfinancial data. The actual analysis of the project will eventually rely on firm-specific financial data because of the emphasis on cash flow. However, in the development phase, different types of information are needed, especially when various options are being formulated and considered. Thus, economic data external to the firm such as general economic conditions, product market conditions, government regulation or deregulation, inflation, labor supply, and technological change—play an important role in developing the alternatives. Most of this initial screening data is nonfinancial. But even such nonfinancial considerations as the quality and quantity of the workforce, political activity, competitive reaction, regulation, and environmental concerns must be integrated into the process of selecting alternatives.

Depending on the nature of the firm's business, there are two other considerations. First, different levels of the firm's management require different types of information. Second, as Ackoff (1970) notes, "most managers using a management information system suffer more from an overabundance of irrelevant information than they do from a lack of relevant information."

In a world in which all information and analysis were free, we could conceive of management analyzing every possible investment idea. However, given the cost, in both dollars and time, of gathering and analyzing information, management is forced to eliminate many alternatives based on strategic considerations. This paring down of the number of feasible alternatives is crucial to the success of the overall capital budgeting program. Throughout this process, the manager faces critical questions, such as are excellent proposals being eliminated from consideration because of lack of information? and, Are excessive amounts of time and money being spent to generate information on projects that are only marginally acceptable? These questions must be addressed on a firm-by-firm basis. When considered in the global context of the firm's success, these questions are the most important considerations in the capital budgeting process.

After the appropriate alternatives have been determined during the development phase, we are ready to perform the detailed economic analysis, which occurs during the selection phase.

19.2.3 Selection Phase

Because managers want to maximize the firm's value for the shareholders, they need some guidance as to the potential value of the investment projects. The selection phase involves measuring the value, or the return, of the project as well as estimating the risk and weighing the costs and benefits of each alternative to be able to select the project or projects that will increase the firm's value given a risk target.

In most cases, the costs and benefits of an investment occur over an extended period, usually with costs being incurred in the early years of the project's life and benefits being realized over the project's entire life. In our selection procedures, we take this into consideration by incorporating the time value of money. The basic valuation framework, or

normative model, that we will use in the capital budgeting selection process is based on present value, as presented in Eq. 19.1:

$$PV = \sum_{t=1}^{N} \frac{CF_t}{(1+k)^t}, \tag{19.1}$$

where PV = the present value or current price of the investment; CF_t = the future value or cash flow that occurs in time t; N = the number of years that benefits accrue to the investor; and k = the time value of money or the firm's cost of capital.

By using this framework for the selection process, we are looking explicitly at the firm's value over time. We are not emphasizing short-run or long-run profits or benefits, but are recognizing that benefits are desirable whenever they occur. However, benefits in the near future are more highly valued than benefits far down the road.

The basic normative model (Eq. 19.1) will be expanded to fit various situations that managers encounter as they evaluate investment proposals and determine which proposals are best.

19.2.4 Control Phase

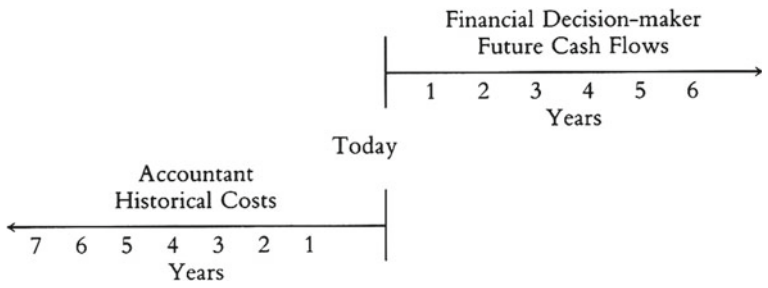
The control phase is the final step of the capital budgeting process. This phase involves placing an approved project on the appropriation budget and controlling the magnitude and timing of expenditures while the project is progressing. A major portion of this phase is the *postaudit* of the project, through which past decisions are evaluated for the benefit of future capital expenditures.

The firm's evaluation and control system are important not only to the postaudit procedure but also to the entire capital budgeting process. It is important to understand that the investment decision is based on cash flow and relevant costs, while the postaudit is based on accrued accounting and assigned overhead. Also, firms typically evaluate performance based on accounting net income for profit centers within the firm, which may be inaccurate because of the misspecification of depreciation and tax effects. The result is that, while managers make decisions based on cash flow, they are evaluated by an accounting-based system.

In addition to data and measurement problems, the control phase is even more complicated in practice because there is a growing concern that the evaluation, reward, and executive incentive system emphasizes a short-run, accounting-based return instead of the maximization of long-run value of cash flow. Thus, quarterly earnings per share, or revenue growth, are rewarded at the expense of longer-run profitability. This emphasis on short-run results may encourage management to forego investments in capital stock or research and development that have long-run benefits in exchange for short-run projects that improve earnings per share.

A brief discussion of the differences between accounting-based information and cash flow is appropriate at this point. The first major difference between the financial decision-maker who uses cash flow and the accountant who uses accounting information is one of time perspective. Exhibit 6.4 shows the differences in time perspective between financial decision-makers and accountants.

Exhibit 19.3: Relevant Time Perspective



As seen in Exhibit 19.3, the financial decision-maker is concerned with future cash flows and value, while the accountant is concerned with historical costs and revenue. The financial decision-maker faces the question, What will I do? while the accountant asks, How did I do?

The second problem is one of definition. The financial decision-maker is concerned with economic income, or a change in wealth. For example, if you purchase a share of stock for $10 and later sell the stock for $30, from a financial viewpoint you have gained $20 of value. It is easy to measure economic income in this case. However, when we look at a firm's actual operations, the measurement of economic income becomes quite complicated.

The accountant is concerned with accounting income, which is measured by the application of generally accepted accounting principles. Accounting income is the result of essential but arbitrary judgments concerning the matching of revenues and expenses during a particular period. For example, revenue may be recognized when goods are sold, shipped, or invoiced, or on receipt of the customer's check. A financial analyst and an accountant would likely differ on when revenue is recognized.

Clearly, over long periods economic value and accounting income converge and are equal because the problems of allocation to particular time periods disappear. However, over short periods, there can be significant differences between these two measures. The financial decision-maker should be concerned with the value added over the life of the project, even though the postaudit report of results is an accounting report based on only one quarter or one year of the project's life. To incorporate a long-run view of value creation, the firm must establish a relationship between its evaluation system, its reward or management incentive system, and the normative goals of the capital budgeting system.

Another area of importance in the control or postaudit phase is the decision to terminate or abandon a project once it has been accepted. Too often we consider capital budgeting as only the acquisition of investments for their entire economic life. The possibility of abandoning an investment prior to the end of its estimated useful or economic life has important implications for the capital budgeting decision. The possibility of abandonment expands the options available to management and reduces the risk associated with decisions based on holding an asset to the end of its economic life. This form of contingency planning gives the financial decision-maker and management a second chance to deal with the economic and political uncertainties of the future.

At any point, to justify the continuation of a project, the project's value from future operations must be greater than its current abandonment value. Given the recent increase in the number and frequency of divestitures, many firms now give greater consideration to abandonment questions in their capital budgeting decision-making. An ideal time to reassess the value of an ongoing investment is at regular intervals during the postaudit.

19.3 Cash-Flow Evaluation of Alternative Investment Projects

Investment should be undertaken by a firm only if it will increase the value of shareholders' wealth. Theoretically, Fama and Miller (1972) and Copeland et al. (2004) show that the investment decisions of the firm can be separated from the individual investor's consumption–investment decision in a perfect capital market. This is known as Fisher's (1930) separation theorem. With perfect capital markets, the manager will increase shareholder wealth if he or she chooses projects with a rate-of-return greater than the market-determined rate-of-return (cost of funds), regardless of the shape of individual shareholders' indifference curves. The ability to borrow or lend in perfect capital markets leads to a higher wealth level for investors than they would be able to achieve without capital markets. This ability also leads to optimal production decisions that do not depend on individual investors' resources and preferences. Thus, the investment decision of the firm is separated from the individual's decision concerning current consumption and investment. Investment decision will therefore depend only on equating the rate-of-return of production possibilities with the market rate-of-return.

This separation principle implies that the maximization of the shareholders' wealth is identical to maximizing the present value of their lifetime consumption. Under these circumstances, different shareholders of the same firm will be unanimous in their preference. This is known as the unanimity principle. It implies that the managers of a firm, in their capacity as agents for shareholders, need not worry about making decisions that reconcile differences of opinion among shareholders: All shareholders will have identical interests. In fact, the price system by which profit is measured conveys the shareholders' unanimously preferred production decisions to the firm.

Looked at in another way, the use of investment decision rules, or capital budgeting, is really an example of a firm attempting to realize the economic principle of operating at the point where marginal cost equals marginal revenue to maximize shareholder wealth. In terms of investment decisions, the "marginal revenue" is the rate-of-return on investment projects, which must be equated with the marginal cost, or the market-determined cost of capital.

Investment decision rules, or capital budgeting, involve the evaluation of the possible capital investments of a firm according to procedures that will ensure the proper

comparison of the cost of the project, that is, the initial and continuing outlays for the project, with the benefits, the expected cash flows accruing from the investment over time. To compare the two cash flows, future cash amounts must be discounted to the present by the firm's cost of capital. Only in this way will the cost of funds to the firm be equated with the benefits from the investment project.

The firm generally receives funds from creditors and shareholders. Both fund suppliers expect to receive a rate-of-return that will compensate them for the level of risk they take. Hence, the discount rate used to discount the cash flow should be the weighted-average cost of debt and equity. In Chap. 10, we will discuss the weighted cost of capital with tax effect in detail.

The weighted-average cost of capital is the same with the market-determined opportunity cost of funds provided to the firm. It is important to understand that projects undertaken by firms must earn enough cash for the creditors and shareholders to compensate their expected risk-adjusted rate-of-return. If the present value of annuity on the cash flow obtained from the weighted-average cost of capital is larger than the initial investment, then there are some gains in shareholders' wealth using this kind of concept. Copeland et al. (2004) demonstrated that maximizing the discount cash flows provided by the investment project.

Before any capital-budgeting techniques can be surveyed, a rigorous definition of cash flows to a firm from a project must be undertaken. First, the decision-maker must consider only those future cash flows that are incremental to the project; that is, only those cash flows accruing to the firm that are specifically caused by the project in question. In addition, any decrease in cash flows to the company by the project in question (i.e., the tax-depreciation benefit from a machine replaced by a new one) must be considered as well.

The main advantage of using the cash-flow procedure in capital-budgeting decisions is that it avoids the difficult problem underlying the measurement of corporate income associated with the accrual method of accounting, for example, the selection of depreciation methods and inventory-valuation methods.

It is well known that the equality between sources and uses of funds for an all-equity firm in period t can be defined as

$$R_t + N_t P_t = N_t d_t + WSMS_t + I_t, \tag{19.2}$$

where

R_t = Revenue in period t,

$N_t P_t$ = New equity in period t,

$N_t d_t$ = Total dividend payment in period t,

WSMS_t = Wages, salaries, materials, and service payment in period t, and

I_t = Investment in period t.

Equation (19.2) is the basic equation to be used to determine the cash flow for capital-budgeting determination.

Second, the definition of cash flow relevant to financial decision-making involves finance rather than accounting income. Accounting regulations attempt to adjust cash flows over several periods (e.g., the expense of an asset is depreciated over several time periods); finance cash flows are calculated as they occur to the firm. Thus, the cash outlay (I_t) to purchase a machine is considered a cash outflow in the finance sense when it occurs at acquisition.

To illustrate the actual calculations involved in defining the cash flows accruing to a firm from an investment project, we consider the following situation. A firm is faced with a decision to replace an old machine with a new and more efficient model. If the replacement is made, the firm will increase production sufficiently each year to generate \$10,000 in additional cash flows to the company over the life of the machine. Thus, the before-tax cash flow accruing to the firm is \$10,000.

The cash flow must be adjusted for the net increase in income taxes that the firm must now pay due to the increased net depreciation of the new machine. The annual straight line depreciation for the new machine over its 5-year life will be \$2,000, and we assume no terminal salvage value. The old machine has a current book value of \$5,000 and a remaining depreciable life of 5 years with no terminal salvage value. Thus, the incremental annual depreciation will be the annual depreciation charges of the new, \$2,000, less the annual depreciation of the old, or \$1,000. The additional income to the firm from the new machine is then the \$10,000 cash flow less the incremental depreciation, \$1,000. The increased tax outlay from the acquisition will then be (assuming a 50% corporate income tax rate) $0.50 \times \$9{,}000$, or \$4,500.

Adjusting the gross annual cash flow of \$10,000 by the incremental tax expense of \$4,500 gives \$5,500 as the net cash flow accruing to the firm from the new machine. It should be noted that corporate taxes are real outflow and must be taken into account when evaluating a project's desirability. However, the depreciation allowance (dep) is not a cash outflow and therefore should not be subtracted from the annual cash flow.

The calculations of post-tax cash flow mentioned above can be summarized in Eq. (19.3):

$$\begin{aligned}\text{Annual After} - \text{Tax Cash Flow} &= \text{ICFBT} - (\text{ICFBT} - \Delta\,\text{dep})\tau \\ &= \text{ICFBT}(1-\tau) + (\text{dep})\tau\end{aligned} \tag{19.3}$$

where

ICFBT = Annual incremental operating cash flows,

τ = Corporate tax rate, and

Δdep = Incremental annual depreciation charge, or the annual depreciation charges on the new machine less the annual depreciation on the old.

Following Eq. (19.3), ICFBT can be defined in Eq. (19.4) as

$$\text{ICFBT} = \Delta R_t - \Delta \text{WSMS}_t. \tag{19.4}$$

Note that ICFBT is an amount before interest and depreciation are deducted and Δ indicates the change of related variables. The reason is that when discounted at the weighted cost of capital, we are implicitly assuming that the project will return the expected interest payments to creditors and the expected dividends to shareholders.

Alternative depreciation methods will change the time pattern but not the total amount of the depreciation allowance. Hence, it is important to choose the optimal depreciation method. To do this, the net present value (NPV) of tax benefits due to the tax deductibility of the depreciation allowance can be defined as

$$NPV(\text{tax benefit}) = \tau \sum_{t=1}^{N} \frac{dep_t}{(1+k)^t},$$

where dep_t = depreciation allowance in period t and N = life of project; it will depend upon whether the straight-line, double declining balance, or sum-of-years'-digits method is used.

The net cash inflow in period t (C_t) used for capital budgeting decision can be defined as

$$C_t = CF_t - \tau_c(CF_t - dep_t - I_t), \tag{19.5}$$

where $CF_t = [Q_t(P_t - V_t)]$; Q_t = quantity produced and sold; P_t = price per unit; V_t = variable costs per unit; dep_t = depreciation; τ_c = tax rate; and I_t = interest expense.

19.4 Alternative Capital-Budgeting Methods

Several methods can be used by a manager to evaluate an investment decision. Some of the simplest methods, such as the accounting rate-of-return or net payback period, are useful in that they are easily and quickly calculated. However, other methods—the net present value, the profitability index, and the internal rate-of-return methods—are superior in that explicit consideration is given by them to both the cost of capital and the time value of money.

For illustrating these methods, we will use the data in Table 19.1, which shows the estimates of cash flows for four investment projects. Each project has an initial outlay of $100, and the project life for the four projects is 4 years.

Table 19.1 Initial cost and net cash inflow for four projects

Year	A	B	C	D
0	−100	−100	−100	−100
1	20	0	30	25
2	80	20	50	40
3	10	60	60	50
4	−20	160	80	115

Since they are mutually exclusive investment projects, only one project can be accepted, according to the following capital-budgeting methods.

19.4.1 Accounting Rate-of-Return

In this method, a rate-of-return for the project is computed by using average net income and average investment outlay. This method does not incorporate the time value of money and cash flow. The *ARR* takes the ratio of the investment's average annual net income after taxes to either total outlay or average outlay. The accounting rate-of-return method averages the after-tax profit from an investment for every period over the initial outlay:

$$ARR = \frac{\sum_{t=0}^{N} \frac{AP_t}{N}}{I}, \tag{19.6}$$

where

AP_t = After-tax profit in period t,
I = Initial investment, and
N = Life of the project.

By assuming that the data in Table 19.1 are accounting profits and the depreciation is $25, the accounting rates-of-return for the four projects are

Project A: −2.5%,
Project B: 35%,
Project C: 30%, and
Project D: 32.5%.

Project B shows the highest accounting rate-of-return; therefore, we will choose Project B as the best one.

The ARR, like the payback method, which will be investigated later in this section, ignores the timing of the cash flows by its failure to discount cash flows back to the present. In addition, the use of accounting cash flows rather than finance cash flows distorts the calculations through the artificial adjustment of some cash flows over several periods.

19.4.2 Internal Rate-of-Return Method

The internal rate-of-return (IRR, r) is the discount rate which equates the discounted cash flows from a project to its investment. Thus, one must solve iteratively for the r in Eq. (19.7):

$$\sum_{t=1}^{N} \frac{CF_t}{(1+r)^t} = I, \tag{19.7}$$

where

CF_t = Cash flow (positive or negative) in period t,
I = Initial investment, and
N = Life of the project.
The IRR for the four projects in Table 19.1 are

Project A: IRR does not exist (since the cash flows are less than the initial investment),
Project B: 28.158%,
Project C: 33.991%, and
Project D: 32.722%.

Since the four projects are mutually exclusive and Project C has the highest IRR, we will choose Project C.

The IRR is then compared to the cost of capital of the firm to determine whether the project will return benefits greater than its cost. A consideration of advantages and disadvantages of the IRR method will be undertaken when it is compared to the net present value method.

19.4.3 Payback Method

The payback method calculates the time period required for a firm to recover the cost of its investment. It is that point in time at which the cumulative new cash flow from the project equals the initial investment.

The payback periods for the four projects in Table 19.1 are

Project A: 2.0 years,
Project B: 3.125 years,
Project C: 2.33 years, and
Project D: 2.70 years.

If we use the payback method, we will choose Project A.

Several problems can arise if a decision-maker uses the payback method. First, any cash flows accruing to the firm after the payback period are ignored. Second, and most importantly, the method disregards the time value of money. That is, the cash flow returned in the later years of the project's life is weighted equally with more recent cash flows accruing to the firm.

Although there are several problems in using the payback method as a capital-budgeting method, the reciprocal or payback period is related to the internal rate-of-return of the project when the life of the project is very long. For example, assume an investment project that has an initial outlay of I and an annual cash flow of R. The payback period is I/R and its reciprocal is R/I. On the other hand, the internal rate-of-return (r) of a project can be written as follows:

$$r = \frac{R}{I} - \left(\frac{R}{I}\right)\left[\frac{1}{(1+r)^N}\right], \tag{19.8}$$

where r is the internal rate-of-return and N is the life of the project in years. Clearly, when N approaches infinity, the reciprocal of payback period R/I will approximate the annuity rate-of-return. The payback method provides a liquidity measure, i.e., sooner is better than later.

Equation (19.8) is the special case of the internal rate-of-return formula defined in Eq. (19.7). By assuming equal annual net receipts and zero semi-annual value, Eq. (19.7) can be rewritten as

$$I = \frac{R}{1+r}\left[1 + \frac{1}{(1+r)} + \frac{1}{(1+r)^2} + \ldots + \frac{1}{(1+r)^{N-1}}\right], \tag{19.7'}$$

where $R = CF_1 = CF_2 = \cdots = CF_n$. Summing the geometric series within the square brackets and reorganizing terms, we obtain Eq. (19.8).

19.4.4 Net Present Value Method

The net present value of a project is computed by discounting the project's cash flows to the present by the appropriate cost of capital. The net present value of the firm is

$$NPV = \sum_{t=1}^{N} \frac{CF_t}{(1+k)^t} - I, \tag{19.9}$$

where k = the appropriate discount rate, and all other terms are defined as above.

The NPV method can be applied to the cash flows of the four projects in Table 19.1. By assuming a 12% discount rate, the NPV for the four projects are as follows:

Project A: −23.95991,
Project B: 60.33358,
Project C: 60.19367, and
Project D: 62.88278.

Since Project D has the highest NPV, we will select Project D as the best one.

Clearly, the NPV method explicitly considers both time value of money and economic cash flows. It should be noted that this conclusion is based upon the discount rate which is 12%. However, if the discount rate is either higher or lower than 12%, this conclusion may not be entirely true. This issue can be resolved by crossover rate analysis, which can be found in Appendix 19.2. In Appendix 19.2, we analyzed projects A and B for different cash flows and different discount rates. The main conclusion for Appendix 19.2 can be summarized as follows.

NPV(B) is higher with low discount rates and NPV(A) is higher with high discount rates. This is because the cash flows of project A occur early and those of project B occur later. If we assume a high discount rate, we would favor project A; if a low discount rate is expected, project B will be chosen. In order to make the right choice, we can calculate the crossover rate. If the discount rate is higher than the crossover rate, we should choose project A; if otherwise, we should go for project B.

Based upon the concept of break-even analysis discussed in Eq. (2.6) of Chap. 2, we can determine the units of product that must be produced in order for NPV to be zero. If $CF_1 = CF_2 = \ldots = CF_N = CF$ and NPV = 0, then Eq. (19.9) can be rewritten as

$$CF[\sum_{t=1}^{N}\frac{1}{(1+k)^t}] = I. \qquad (19.9')$$

By substituting the definition of CF given in Eq. (19.5) into Eq. (19.9′), we can obtain the break-even point (Q*) for capital budgeting as

$$Q^* = \{\frac{[I-(dep)\tau]/(1-\tau)}{\sum_{t=1}^{N}1/[(1+k)^t]}\}(\frac{1}{p-v}). \qquad (19.10)$$

A real-world example of an application of the NPV method to breakeven analysis can be found in Reinhardt (1973) and Chap. 13 of Lee and Lee (2017).

19.4.5 Profitability Index

The profitability index is very similar to the NPV method. The PI is calculated by dividing the discounted cash flows by the initial investment to arrive at the present value per dollar outlay:

$$\text{PI} = \frac{\sum_{t=1}^{N}[CF_t/((1+k)^t)]}{I}. \qquad (19.11)$$

The project should be undertaken if the PI is greater than 1; the firm should be indifferent to its undertaking if PI equals one. The project with the highest PI greater than one should be accepted first. Obviously, PI considers the time value of money and the correct finance cash flows, as does the NPV method. Further, the PI and NPV methods will lead to identical decisions unless ranking mutually exclusive projects and/or under capital rationing. When considering mutually exclusive projects, the PI can lead to a decision different from that derived by the NPV method.

For example:

Project	Initial outlay	Present value of cash inflows	NPV	PI
A	100	200	100	2
B	1000	1300	300	1.3

Project A and B are mutually exclusive projects. Project A has a lower NPV and higher PI compared to Project B. This will lead to a decision to select Project A by using the PI method and select Project B by using the NPV method. In the case shown here, the NPV and PI rankings differ because of the differing scale of investment: The NPV subtracts the initial outlay while the PI method divides by the original cost. Thus, differing initial investments can cause a difference in ranking between the two methods.

The firm that desires to maximize its absolute present value rather than percentage return will prefer Project B, because the NPV of Project B ($300) is greater than the NPV of Project A ($100). Thus, the PI method should not be used as a measure of investment worth for projects of differing sizes where mutually exclusive choices have to be made. In other words, if there exist no other investment opportunities, then the NPV will be the superior method in this case because, under the NPV, the highest ranking investment project (the one with the largest NPV) will add the most value to shareholders' wealth. Since this is the objective of the firm's owners, the NPV will lead to a more accurate decision.

The manager's views on alternative capital budgeting methods and related practical issues will be presented in Appendix 19.1.

19.5 Capital-Rationing Decision

In this section, we will discuss a capital-budgeting problem that involves the allocation of scarce capital resources among competing economically desirable projects, not all of which can be carried out due to a capital (or other) constraint. This kind of problem is often called "capital rationing." In this section, we will show how linear programming can be used to make capital-rationing decisions.

19.5.1 Basic Concepts of Linear Programming

Linear programming is a mathematical technique used to find optimal solutions to problems of a firm involving the allocation of scarce resources among competing activities. Mathematically, the type of problem that linear programming can solve is one in which both the objective of the firm to be maximized (or minimized) and the constraints limiting the firm's actions are linear functions of the decision variables involved. Thus, the first step in using linear programming as a tool for financial decisions is to model the problem facing the firm in a linear programming form. To construct the linear programming model, one must take the following steps.

First, identify the controllable decision variables involved in the firm's problem. Second, define the objective or criterion to be maximized or minimized and represent it as a linear function of the controllable decision variables. In finance, the objective generally is to maximize the profit contribution or the market value of the firm or to minimize the cost of production.

Third, define the constraints and express them as linear equations or inequalities of the decision variables. This will usually involve (a) a determination of the capacities of the scarce resources involved in the constraints and (b) a derivation of a linear relationship between these capacities and the decision variables.

Symbolically, then, if X_1, X_2, …, X_n represent the quantities of output, the linear programming model takes the general form:

$$\text{Maximize (or minimize) } Z = c_1X_1 + c_2X_2 + \cdots + c_nX_n, \tag{19.12}$$

$$\begin{aligned} \text{Subject to:} \quad & a_{11}X_1 + a_{12}X_2 + \cdots + a_{1n}X_n \le b_1 \\ & a_{21}X_1 + a_{22}X_2 + \cdots + a_{2n}X_n \le b_2 \\ & a_{m1}X_1 + a_{m2}X_2 + \cdots + a_{mn}X_n \le b_m \\ & \vdots \qquad\qquad \vdots \\ & X_j \ge 0, (j = 1, 2, \ldots, n). \end{aligned}$$

Here, Z represents the objective to be maximized (or minimized), profit or market value (or cost), c_1, c_2, …, c_n and a_{11}, a_{12}, …, a_{mn} are constant coefficients relating to profit contribution and input, respectively; b_1, b_2, …, b_m are the firm's capacities of the constraining resources. The last constraint ensures that the decision variables to be determined are nonnegative.

Several points should be noted concerning the linear programming model. First, depending upon the problem, the constraints may also be stated with equal signs (=) or as greater-than-or-equal-to. Second, the solution values of the decision variables are divisible, that is, a solution would permit x(j) = 1/2, 1/4, etc. If such fractional values are not possible, the related technique of integer programming, yielding only whole numbers as solutions, can be applied. Third, the constant coefficients are assumed known and deterministic (fixed). If the coefficients have probabilistic distributions, one of the various methods of stochastic programming must be used. Examples will be given below of the application of linear programming to the areas of capital rationing and capital budgeting.

19.5.2 Capital Rationing

The XYZ Company produces products A, B, and C within the same product line, with sales totaling $37 million last year. Top management has adopted the goal of maximizing shareholder wealth, which to them is represented by gain in shareholder price. Wickwire plans to finance all future projects with internal or external equity; funds available from the equity market depend on share price in the stock market for the period.

Three new projects were proposed to the Finance Committee, for which the following net after-tax annual funds flows are forecast:

Project	Year					
	0	1	2	3	4	5
X	−100	30	30	60	60	60
Y	−200	70	70	70	70	70
Z	−100	−240	−200	400	300	300

All three projects involve financing cost-saving equipment for well-established product lines; adoption of any one project does not preclude adoption of any other. The following NPV formulations have been prepared by using a discount rate of 12%.

Investment	NPV
X	65.585
Y	52.334
Z	171.871

In addition, the finance start has calculated the maximum internally generated funds that will be available for the current year and succeeding 2 years, not counting any cash generated by the projects currently under consideration.

Year 0	Year 1	Year 2
$300	$70	$50

Assuming that the stock market is in a serious downturn, and thus no external financing is possible, the problem is which of the three projects should be selected, assuming that fractional projects are allowed.

The problem essentially involves the rationing of the capital available to the firm among the three competing projects such that share price will be maximized. Thus, assuming a risk-adjusted discount rate of 12%, the objective function becomes

$$\text{Maximize } V = 65.585X + 52.334Y + 171.871Z + 0C + 0D + 0E,$$

where V represents the total present value realized from the projects, and C, D, and E will represent idle funds in periods 0, 1, and 2, respectively. The constraint for period 0 must ensure that the funds used to finance the projects do not exceed the funds available. Thus,

$$100X + 200Y + 100Z + C + 0D + 0E = 300.$$

In this constraint, C represents any idle funds unused in period 0 after projects are paid for. Similarly, for periods 1 and 2,

$$-30X - 70Y + 240Z - C + D + 0E = 70,$$
$$-30X - 70Y + 200Z + 0C - D + E = 50.$$

Here, −D and −E are included in the second and third constraints, ensuring that idle funds unused from one period are carried over to the succeeding period. In addition, to prevent the program from repeatedly selecting only one project (the "best") until funds are exhausted, three additional constraints are needed:

$$X \leq 1, \quad Y \leq 1, \quad Z \leq 1.$$

The solution to the model if V = $208.424, is.

The process of solving this linear program with Excel is illustrated in Appendix 19A.

To give an indication of the value of relaxing the fund constraint in any period (the most the firm would be willing to pay for additional financing), the shadow price of the fund constraints is given below:

Funds constraint	Shadow price
1st period	0.4517
2nd period	0.4517
3rd period	0.0914

It should be noted that the constraints related to $X \leq 1$, $Y \leq 1$, and $Z \leq 1$ are required in solving capital-rationing problems. If these constraints are removed, then we will obtain X = 2.4074, Y = 0, and Z = 0.5926. This issue has been discussed by Copeland et al. (2004) and Weingartner (1963, 1977).

Thus, linear programming is a valuable mathematical tool with which to solve capital-budgeting problems when funds rationing is required. In addition, duality has been used by the banking industry to determine the cost of capital of funds. The relative advantages and disadvantages between the linear-programming method and other methods used to fund the cost of capital remain as a subject for further research. Appendix 19.1 shows how Excel program can be used to solve this kind of linear programming model for capital rationing.

We have discussed an alternative method for capital-budgeting decision under certainty; in addition, we show how linear programming model can use to perform capital rationing. By using NPV method, we will discuss two alternative capital budgeting under uncertainty in the next two sections.

19.6 The Statistical Distribution Method

Capital budgeting frequently incorporates the concept of probability theory. To illustrate, consider two projects—project *x* and project *y*—and three states of the economy—prosperity, normal, and recession—for any given time. For each of these states, we may calculate a probability of occurrence and estimate their respective returns, as indicated in Table 19.2.

The expected returns for projects *x* and *y* can be calculated by Eq. 19.13:

$$\bar{k} = \sum k_i p_i \tag{19.13}$$

$$\bar{k}_x = 6.25\% + 7.50\% + 1.25\% = 15.00\%$$
$$\bar{k}_y = 10\% + 7.50\% - 2.50\% = 15.00\%$$

and the standard deviation for these returns can be found through Eq. 19.14

$$\sigma = \sqrt{\sum_{i=1}^{n} (k_i - \bar{k})^2 p_i} \tag{19.14}$$

$$\sigma_x = \left[(.25 - .15)^2(.25) + (.15 - .15)^2(.50) + (.05 - .15)^2(.25)\right]^{\frac{1}{2}} = 7.07\%$$
$$\sigma_y = \left[(.40 - .15)^2(.25) + (.15 - .15)^2(.50) + (-.10 - .15)^2(.25)\right]^{\frac{1}{2}} = 17.68\%$$

Table 19.2 Means and standard deviation

State of economy	Probability of state (p_i)	Return (k_i)	$k_i p_i$
Project X			
Prosperity	.25	25%	6.25%
Normal	.50	15%	7.50%
Recession	.25 1.00	5%	1.25% 15.00%
Standard deviation = $\sigma_x = 7.07\%$			
Project Y			
Prosperity	.25	40%	10.00%
Normal	.50	15%	7.5%
Recession	.25 1.00	−10%	−2.5% 15.00%
Standard deviation = $\sigma_y = 17.68\%$			

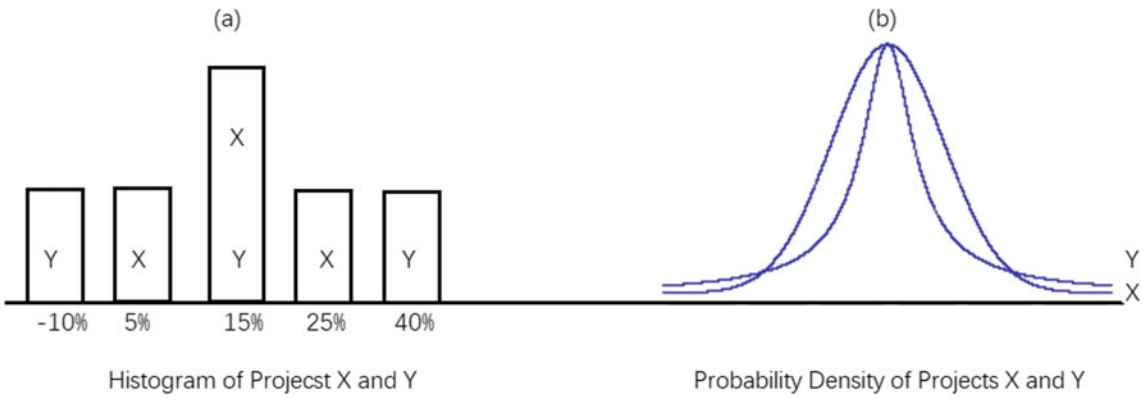


Fig. 19.1 Statistical distribution for Projects X and Y

Data in Table 19.2 can be used to draw histograms of projects *x* and *y,* as depicted in Fig. 19.1. If we assume that rates of return (*k*) are distributed continuously and normally, then Fig. 19.1a can be drawn as Fig. 19.1b.

The concept of statistical probability distribution can be combined with capital budgeting to derive the statistical distribution method for selecting risky investment projects.

The expected return for both projects is 15%, but because project *y* has a flatter distribution with a wider range of values, it is the riskier project. Project *x* has a normal distribution with a larger collection of values nearer the 15% expected rate-of-return and therefore is more stable.

19.6.1 Statistical Distribution of Cash Flow

From Eq. (19.2) of this chapter, the equation for net cash inflow can be explicitly defined as

$$C_t = CF_t - \tau_c(CF_t - dep_t - I_t)$$

where $CF_t = [Q_t(P_t - V_t)]$; C_t = net cash flow in period t; Q_t = quantity produced and sold; P_t = price; V_t = variable costs; dep_t = depreciation; τ_c = tax rate; and I_t = interest expense. For this equation, net cash flow is a random number because Q, P, and V are not known with certainty. We can assume that net C_t has a normal distribution.

If two projects have the same expected cash flow, or return, as determined by the expected value (Eq. 19.9), we may be indifferent between either project if we were to base our choice solely on return. However, if we also take risk into account, we get a more accurate picture of the type of distribution to expect, as shown in Fig. 19.1.

With the introduction of risk, a firm is not necessarily indifferent between two investment proposals having equal NPV. Both NPV and its standard deviation (σ_{NPV}) should be

Table 19.3 Cash flows are displayed in $ thousands

Year	Project A		Project B	
	Cash flow	Std. deviation	Cash flow	Std. deviation
0	($60)		($60)	
1	$20	4	$20	2
2	20	4	20	2
3	20	4	20	2
4	20	4	20	2
Salvage Value	$5		$5	
Assume a discount rate of 10%				

estimated in performing capital-budgeting analysis under uncertainty, NPV under uncertainty is defined as

$$NPV = \sum_{t=1}^{N} \frac{\tilde{C}_t}{(1+k)^t} + \frac{S}{(1+k)^N} - I_o \tag{19.15}$$

where $\tilde{C}_t$ = uncertain net cash flow in period t; k = risk-adjusted discounted rate; S_t = salvage value; and I_o = initial outlay.

The mean of the NPV distribution and its standard deviation is defined as

$$NPV = \sum_{t=1}^{N} \frac{\overline{C}_t}{(1+k)^t} + \frac{S}{(1+k)^N} - I_o \tag{19.16}$$

$$\sigma_{NPV} = \left[\sum_{t=1}^{N} \frac{\sigma_t^2}{(1+k)^{2t}} \right]^{\frac{1}{2}} \tag{19.17}$$

for cash flows that are mutually independent ($\rho = 0$) cash flows. The generalized case for both Eqs. 19.16 and 19.17 is explored in Appendix 19.2.

Example 19.1 A firm is considering two new product lines, projects A and B, with the same life, mean returns, and salvage flow, as indicated in Table 19.3. Under the certainty methods (this chapter), both projects would have the same NPV:

$$NPV_A = NPV_B = \sum_{t=1}^{5} \frac{\overline{C}_t}{(1+k)^t}$$

$$\begin{aligned} NPV &= 20(PVIF_{10\%,1}) + 20(PVIF_{10\%,2}) + 20(PVIF_{10\%,3}) \\ &\quad + 20(PVIF_{10\%,4}) + 20(PVIF_{10\%,5}) - 60 + 5(PVIF_{10\%,5}) \\ &= 20(.9091) + 20(.8264) + 20(.6830) + 20(.6209) - 60 + 5(.6209) \\ &= 19.90 \end{aligned}$$

However, because the standard deviations of project A's cash flows are greater than project B's, project A is riskier than project B. This difference can only be explicitly evaluated by using the statistical distribution method. To examine the riskiness between the two projects, we can calculate the standard deviation of their NPVs. If cash flows are perfectly positively correlated over time, then the standard deviation of NPV (σ_{NPV}) can be simplified as[1]

$$\sigma_{NPV} = \sum_{t=1}^{N} \frac{\sigma_t}{(1+k)^t} \tag{19.17a}$$

$$\begin{aligned} \sigma_{NPV(A)} &= (\$4)(PVIF_{10\%,1}) + (\$4)(PVIF_{10\%,2}) + \ldots + (\$4)(PVIF_{10\%,5}) \\ &= (4)(.9091) + (4)(.8264) + (4)(.7513) + (4)(.6830) + (4)(.6209) \\ &= 15.16 \, or \, \$15{,}160 \end{aligned}$$

$$\begin{aligned} \sigma_{NPV(B)} &= (\$2)(PVIF_{10\%,1}) + (\$2)(PVIF_{10\%,2}) + \ldots + (\$2)(PVIF_{10\%,5}) \\ &= (2)(.9091) + (2)(.8264) + (2)(.7513) + (2)(.6830) + (2)(.6209) \\ &= 7.58 \, or \, \$7{,}580 \end{aligned}$$

With the same NPV, project B's cash flows would fluctuate by $7,580 per year, while project A's would fluctuate by $15,160. Therefore, project B would be preferred, given the same returns, because it is less risky.

Lee and Wang (2010) provide the fuzzy real option valuation approach to solve the capital budgeting decision under an uncertainty environment. In Wang and Lee's model framework, the concept of probability is employed in describing fuzzy events under the estimated cash flow based on fuzzy numbers, which can better reflect the uncertainty in the project. By using a fuzzy real option valuation, the managers can select fuzzy projects and determine the optimal time to abandon the project under the assumption of limited capital budget. Lee and Lee (2017) has discussed this in detail in Chap. 14.

[1] Equation 19.17a is a special case of Eq. 19.19.

19.7 Simulation Methods

Simulation is another approach to capital budgeting decision-making under uncertainty. In cases of uncertainty, every variable relevant to the capital budgeting decision can be viewed as random. With so many random variables, it may be difficult or impossible to obtain an optimal solution with an economic or financial model.

Any model of a business decision problem can be used as a simulation model if it replicates or simulates business problems and conditions. However, true simulation models are designed to generate alternatives rather than find an optimal solution. A decision is then made through examination of these alternative results. Another aspect of the simulation model is that it focuses on an operation of the firm in detail, either physical or financial, and studies the operations of such a system over time. Simulation is also a useful tool for looking at how the real system operates and showing the effects of the important variables.

When uncertain, or random, variables play a key part in the operations of a system, the uncertainty will be included in the simulation and the model is referred to as a *probabilistic simulation*. Its counterpart, *deterministic simulation*, does not include uncertainty.

The easiest way to explain simulation is to present a simple simulation problem and discuss how it is used.

Example 19.2 A production manager of a small machine-manufacturing firm wants to evaluate the firm's weekly ordering policy for machine parts. The current method is to order the same amount demanded the previous week. However, the manager does not believe this is the most efficient or productive approach. The parts for assembly of one particular product cost $20 per machine, and each machine is sold for $60. The parts are ordered Friday morning and received Monday morning.

From experience, the manager knows that about 300–750 machines have been sold by its distributors per week and has tabulated this demand in Table 19.4.

The manager is considering two courses of action: (1) to order the amount that was demanded in the past or (2) to order the expected value based on past weekly demands for the product, which in this case is

$$(350)(.10) + (450)(.30) + (550)(.20) + (680)(.30) + (750)(.10) = 550 \text{ machines}$$

The manager would like to compare the results of these alternatives. The current procedure—ordering what was demanded the previous week—will be designated as alternative A and the second procedure as alternative B. These are defined as follows:

Table 19.4 Weekly demand information

Demand per week	Relative frequency
350 machines	0.10
450	0.30
550	0.20
650	0.30
750	0.10
	1.0

$$\text{Alternative } A\text{: } Q_n = D_{n-1}$$

$$\text{Alternative } B\text{: } Q_n = 550$$

where Q_n = amount ordered on day n and D_{n-1} = amount demanded the previous week. These alternatives can be compared to the firm's weekly profits on that particular machine as follows:

$$P_n = (S_n \times P_n) - (Q_n \times C) \tag{19.14}$$

where P_n = profit in week n; S_n = amount sold in week n; P = selling price per machine; Q_n = amount ordered at end of week n; and C = cost per machine.

To further prepare this problem for simulation, there must be a method to generate weekly demand to compare these two alternatives. For this purpose, we will use a probability distribution and a random number table. The relative-frequency values must be connected to probabilities. A specific number or numbers are then attached to each probability value to reflect the proportion of numbers from 00 to 99 that corresponds to each probability entry. In our example, the numbers from 00 to 09 represent 10% of the numbers, 10–30 represent 30% of these numbers, 40–59 represent 20%, and so on. Table 19.5 depicts the relative frequency, corresponding probability, and associated random numbers, for this problem.

Table 19.6 is a uniformly distributed table of random numbers.

We can easily carry out hand simulation to determine if alternative A or B is optimal for the firm's planning and production needs. The basic procedure is as follows:

Table 19.5 Weekly demands and their probabilities

Demand per week	Relative frequency	Probability	Random interval
350	.10	.10	00–09
450	.30	.30	10–39
550	.20	.20	40–59
650	.30	.30	60–89
750	.10	.10	90–99

Table 19.6 Uniformly distributed random numbers

06,433	80,674	24,520	18,222	10,610	05,794	37,515	48,619	02,866
39,208	47,829	72,648	37,414	75,755	01,717	29,899	78,817	03,500
89,884	59,051	67,533	08,123	17,730	95,862	08,034	19,473	03,071
61,512	32,155	51,906	61,662	64,130	16,688	37,275	51,262	11,569
99,653	47,635	12,506	88,535	36,553	23,757	34,209	55,803	96,275
95,913	11,045	13,772	76,638	48,423	25,018	99,041	77,529	81,360
55,804	44,004	13,112	44,115	01,691	50,541	00,147	77,685	58,788
35,334	82,410	91,601	40,617	72,876	33,967	73,830	15,405	96,554
59,729	88,646	76,487	11,622	96,297	24,160	09,903	14,041	22,917
57,383	89,317	63,677	70,119	94,739	25,875	38,829	68,377	43,918
30,574	06,039	07,967	32,422	76,791	39,725	53,711	93,385	13,421
81,307	13,314	83,580	79,974	45,929	85,113	72,208	09,858	52,104
02,410	96,385	79,007	54,039	21,410	86,980	91,772	93,307	34,116
18,969	87,444	52,233	62,319	08,598	09,066	95,288	04,794	01,534
87,803	80,514	66,800	62,297	80,198	19,347	73,234	86,265	49,096
68,397	10,538	15,438	62,311	72,844	60,203	46,412	05,943	79,232
28,520	54,247	58,729	10,854	99,058	18,260	38,765	90,038	94,200
44,285	09,452	15,867	70,418	57,012	72,122	36,634	97,283	95,943
80,299	22,510	33,517	23,309	57,040	29,285	07,870	21,913	72,958
84,842	05,748	90,894	61,658	15,001	94,055	36,308	41,161	37,341

1. Draw a random number from Table 19.6. It doesn't matter exactly where on the table numbers are picked, as long as the pattern for drawing numbers is consistent and unvaried; for example, the first two numbers of row 1, then row 2, then row 3, and so forth.
2. In Table 19.5, find the random number interval associated with the random number chosen from Table 19.6.
3. Find the weekly demand (D_n) in Table 19.5 that corresponds to the random number (RN).
4. Calculate the amount sold (S_n). If $D_n > Q_n$, then $S_n = Q_n$; if $D_n < Q_n$, $S_n = D_n$.
5. Calculate weekly profit $[P_n = (S_nP) - (Q_nC)]$.
6. Repeat steps 1 to 5 until 20 days have been simulated.

The results of the above procedures are summarized in Table 19.7. There are nine columns in Table 19.7. Column a represents the week, column b represents the random number, column c represents the weekly demand, column d represents the amount ordered for the nth week for alternative A, column e represents the sales for alternative A, column f represents the profit of nth week for alternative A, column g represents the amount ordered for the nth week for alternative B, column h represents the sales for alternative B, and column i represents the profit of nth week for alternative B.

We will now explain how the random numbers in column b were obtained. The first nine random numbers were taken from the first two digits of the random numbers in row 1 of Table 19.6. The second nine numbers were obtained from the first two digits of the random numbers in row 6 of Table 19.6. The last three random numbers are from the first two digits of the first three random numbers in row 11. Column c is a number demand for the nth week. The first number for number demand for Week 0 is 550, which is the average number of Table 19.5. The first random number, 06, is in the first random interval of Table 19.5; therefore, the demand is 350. The second random number, 80, is in the fourth random interval of Table 19.5; therefore, the demand is 650. Similarly, we can obtain other random numbers in column c. Column d represents the quantity order for alternative A, which represents the number demand of the previous week. Column e and column h represent the amount sold in week n for alternatives A and B, respectively. This sale number is determined in accordance with Procedure 4, which was mentioned above. Column g represents the weekly amount order for alternative B, which is the average number (550) of Table 19.5. Column f and column i represent the weekly profit for alternatives A and B, respectively, which was calculated using the formula in Eq. 19.14.

Through simulation, we can see that because there would be fewer machine parts in the inventory, the firm would earn, on average, an additional $667 per week using alternative B rather than alternative A. This is because an average of about 29 more machines are sold per week. Through the simulation of these two types of order techniques, we have found that alternative B is the better of the two, but not necessarily the optimal choice. We may run simulations for other types of decision alternatives and may choose among these.

A simulation model is a representation of a real system, wherein the system's elements are depicted by arithmetic or logical processes. These processes are then executed either manually, as illustrated in Example 19.2, or by using a computer, for more complicated models, to examine the dynamic properties of the system. Simulation of the actual operation of a system tests the performance of the specific system. For this reason, simulation models must be custom-made for each situation.

Table 19.7 Simulation results for alternative A and alternative B

			Alternative A			Alternative B		
Week	RN	D_n	Q_n	S_n	P_n (A)	Q_n	S_n	P_n (B)
(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
0		550	–	–	–	–	–	–
1	06	350	550	350	$10,000	550	350	$10,000
2	80	650	350	350	14,000	550	550	22,000
3	24	450	650	450	14,000	550	450	16,000
4	18	450	450	450	18,000	550	450	16,000
5	10	450	450	450	18,000	550	450	16,000
6	05	350	450	350	12,000	550	350	10,000
7	37	450	350	350	14,000	550	450	16,000
8	48	550	450	450	18,000	550	550	22,000
9	02	350	550	350	10,000	550	350	10,000
10	95	750	350	350	14,000	550	550	22,000
11	11	450	750	450	12,000	550	450	16,000
12	13	450	450	450	18,000	550	450	16,000
13	76	650	450	450	18,000	550	550	22,000
14	48	550	650	550	20,000	550	550	22,000
15	25	450	550	450	16,000	550	450	16,000
16	99	750	450	450	18,000	550	550	22,000
17	77	650	750	650	24,000	550	550	22,000
18	81	650	650	650	26,000	550	550	22,000
19	30	550	550	550	22,000	550	550	22,000
20	06	350	550	350	10,000	550	350	10,000
21	07	350	550	350	10,000	550	350	10,000
Total		11,200	11,050	9,250	$336,000	11,550	9,550	$350,000
Weekly average		533.3	526.2	440.5	$16,000	550	469	$16,667

Example 19.2 is a specific production management problem and serves as a learning tool on manual simulation. Simulation models have been developed for capital budgeting decisions, and by way of Example 19.2, we can see how such models can be utilized at the financial analysis and planning level.

19.7.1 Simulation Analysis and Capital Budgeting

The following example shows how the simulation model developed by Hertz (1964, 1979) can be used in capital budgeting. Here we consider a firm that intends to introduce a new product; the 11 input variables thought to determine project value are shown in Table 19.7. Of these inputs, variables 1–9 are specified as random variables (that is, there is no predetermined sequence or order for their occurrence) with ranges as listed in the table. We could add a random element to variables 10 and 11, but the computational complexity and the insights gained do not justify the effort. Also, for ease of modeling, we use a uniform distribution to describe the probability of any particular outcome in a specified range. By using a set range for each of the nine random variables, we are not actually allowing the probabilities of each possible outcome to vary, but the spirit of varying probabilities is imbedded in the simulation approach. One further qualification of our model is that the life of the facilities is restricted to an integer value with the range as specified at the bottom of Table 19.8.

The uniform distribution density function[2] can be written as

[2] For a more detailed discussion of the properties of the uniform density function, see Hamburg (1983, pp. 100–101). Other more realistic distributions, such as log-normal and normal distributions, can be used to improve the empirical results of this kind of simulation.

Table 19.8 Variables for simulation

Variables	Range
1. Market size (units)	2,500,000–3,000,000
2. Selling price ($/unit)	40–60
3. Market growth	0–5%
4. Market share	10–15%
5. Total investment required ($)	8,000,000–10,000,000
6. Useful life of facilities (years)	5–9
7. Reside value of investment ($)	1,000,000–2,000,000
8. Operating cost ($/unit)	30–045
9. Fixed costs ($)	400,000–500,000
10. Tax rate	40%
11. Discount rate	12%

Source Reprinted from Lee (1985, p. 359)
Notes (a) Random numbers from Wonnacott and Wonacott (1977) are used to determine the value of a variable for simulation

$$f_x = \begin{cases} \dfrac{1}{\beta - \alpha} \end{cases} \tag{19.18}$$

where β is the upper bound on the variable value and α is the lower bound. Over the range $\alpha < x < \beta$, the function $f_x = 1/\beta - \alpha$; over the range $\beta < x < \alpha$, $f_x = 0$. With this in mind, note the way the values are assigned. For each successive input variable, a random-number generator selects a value from 01 to 00 (where 00 is the proxy for 100 using a 2-digit random-number generator) and then translates that value into a variable value by taking account of the specified range and distribution of the variable in question.

For each simulation, nine random numbers are selected. From these random numbers, a set of values for the nine key factors is created. For example, the first set of random numbers, as shown in Table 19.9, is 39, 73, 72, 75, 37, 02, 87, 98, and 10. The procedure of selecting these numbers is similar to Example 9.2; however, these random numbers are not based upon the uniform distribution random number as presented in Table 19.6. If we use the random numbers from Table 19.6, we can use the first two digits of the first row of this random table, then the random numbers are 06, 80, 24, 18, 10, 05, 37, 48, and 02.

The value of the market size factor for the first simulation can be obtained as follows:

$$2{,}500{,}000 + \frac{39}{100}(3{,}000{,}000 - 2{,}500{,}000) = 2{,}695{,}000$$

The value of sale price factor for the first simulation can be obtained as follows:

$$40 + \frac{73}{100}(60 - 40) = 54.6$$

The operating cost for the first simulation can be obtained as follows:

$$30 + \frac{98}{100}(45 - 30) = 44.7$$

Similar computations can be used to calculate the values of all variables except the useful life of the facilities. Because useful life of facilities is restricted to integer values, we use the following correspondence between random numbers and useful life of facilities:

Random number	01–19	20–39	40–59	60–79	90–99	00
Useful life	5	6	7	8	9	10

Since the random number for useful life is 02, it is within the range of 01–19; therefore, the useful life is 5 years.

For each simulation, a series of cash flows and its net present value can be calculated by using the following formula:

$$\begin{aligned}(sales\ volume)_t = &\left[(market\ size) \times (1 + market\ growth\ rate)^t\right] \\ &\times (market\ share)\end{aligned}$$

$$\begin{aligned}EBIT = &(sales\ volume)_t \times (selling\ price - operating\ cost) \\ &- (fixed\ cost)\end{aligned}$$

$$(cash\ flow)_t = EBIT_t \times (1 - tax\ rate)$$

$$NPV = \sum_{t=1}^{N} \frac{(cash\ flow)_t}{(1 + discount\ rate)^t} - I_0$$

where t represents the tth year and N represents the useful life.

The results in terms of cash flow for each simulation are listed in Table 19.10, with each period's cash flows shown separately. Now, we will discuss how the cash flow for the first simulation is calculated. For example, the cash flow for the first three periods are 2,034,382.33, 2,116,529.56, and 2,201,525.23. 2,034,382.335, can be calculated as follows:

$$\begin{aligned}(sales\ volume)_1 &= \left[(market\ size) \times (1 + market\ growth\ rate)^t\right] \\ &\times (market\ share) \\ &= [(2{,}695{,}000) \times (1 + 0.036)] \times (13.75\%) \\ &= 383{,}902.75\end{aligned}$$

$$\begin{aligned}EBIT_1 &= (383{,}902.75) \times (54.6 - 44.7) - (410{,}000) \\ &= 3{,}390{,}637.22\end{aligned}$$

$$(cash\ flow)_1 = 3{,}390{,}637.22 \times (1 - 40\%) = 2{,}034{,}382.33$$

Table 19.9 Simulation

Variables	1	2	3	4	5	6
VMARK 1	(39)2,695,000	(47)2,735,000	(67)2,835,000	(12)2,580,000	(78)2,890,000	(89)2,945,000
PRICE 2	(73)$54.6	(93)$58.6	(59)$51.8	(78)$55.6	(61)$52.2	(18)$43.6
GROW 3	(72)3.6%	(21)1.05%	(63).0315	(03).0015	(42).021	(83).0415
SMARK 4	(75)13.75%	(95)14.75%	(78).139	(04).102	(77).1385	(08).104
TOINV 5	(37)8,740,000	(97)9,940,000	(87)9,740,000	(61)9,220,000	(65)9,300,000	(90)9,800,000
KUSE 6	(02)5 years	(68)8 years	(47)7 years	(23)6 years	(71)8 years	(05)5 years
RES 7	(87)1,870,000	(41)1,410,000	(56)1,560,000	(15)1,150,000	(20)1,200,000	(89)1,890,000
VAR 8	(98)$44.7	(91)$43.65	(22)$33.3	(58)$38.7	(17)$32.55	(18)$32.7
FIX 9	(10)$410,000	(80)$480,000	(19)$419,000	(93)$493,000	(48)$448,000	(08)$408,000
TAX 10	.4	.4	.4	.4	.4	.4
DIS 11	.12	.12	.12	.12	.12	.12
NPV	$197,847.561	$7,929,874.287	$12,146,989.579	$1,169,846.55	$15,306,345	$−1,513,820.475

Variables	7	8	9	10
VMARK 1	(26)2,630,000	(60)2,800,000	(68)2,840,000	(23)2,615,000
PRICE 2	(47)$49.4	(88)$57.6	(39)$47.8	(47)$49.4
GROW 3	(94).047	(17).0085	(71).0355	(25).0125
SMARK 4	(06).103	(36).118	(22).111	(79).1395
TOINV 5	(72)9,440,000	(77)9,540,000	(76)9,520,000	(08)8,160,000
KUSE 6	(40)7 years	(43) 7 years	(81) 9 years	(71)1,710,000
RES 7	(62)1,620,000	(28)1,280,000	(88)1,880,000	(71)1,710,000
VAR 8	(47)$37.05	(31)$34.65	(94)$44.1	(58)$38.7
FIX 9	(68)$468,000	(06)$406,000	(76)$476,000	(56)$456,000
TAX 10	.4	.4	.4	.4
DIS 11	.12	.12	.12	.12
NPV	$11,327,171.67	$839,650.211	$−6,021,018.052	$563,687.461

Source Reprinted from Lee and Lee (2017, p. 685)
Note Definitions of variables can be found in Table 19.8.

$$(sales\ volume)_2 = \left[(market\ size) \times (1 + market\ growth\ rate)^t\right] \times (market\ share)$$
$$= \left[(2{,}695{,}000) \times (1 + 0.036)^2\right] \times (13.75\%)$$
$$= 397{,}732.249$$
$$EBIT_2 = (397{,}732.249) \times (54.6 - 44.7) - (410{,}000)$$
$$= 3{,}527{,}549.27$$

$$(cash\ flow)_2 = 3{,}527{,}549.27 \times (1 - 40\%) = 2{,}116{,}529.56$$

$$(sales\ volume)_3 = \left[(market\ size) \times (1 + market\ growth\ rate)^t\right] \times (market\ share)$$
$$= \left[(2{,}695{,}000) \times (1 + 0.036)^3\right] \times (13.75\%)$$
$$= 412{,}041.285$$

$$EBIT_3 = (412{,}041.285) \times (54.6 - 44.7) - (410{,}000)$$
$$= 3{,}669{,}208.72$$

$$(cash\ flow)_3 = 3{,}669{,}208.72 \times (1 - 40\%) = 2{,}201{,}525.23$$

In Table 19.10 for the first, sixth, and tenth simulations, we calculate cash flow for five periods. For the second and fifth simulations, we calculate cash flow for eight periods. For the third, seventh, and eighth simulations, we calculate cash flow for seven periods. For the fourth simulation, we calculate cash flow for six periods. Finally, for the ninth simulation, we calculate cash flow for nine periods.

The NPVs for each simulation are given under the input values listed in Table 19.9. From these NPV figures, we can calculate a mean NPV figure and standard deviation, from which we can analyze the project's risk and return profile. As we can see, this project's NPV can range from −$6

Table 19.10 Cash flow estimation for each simulation

Period	1	2	3	4	5	6
1	2,034,382.335	3,368,605.531	4,260,506.327	2,376,645.064	4,549,425.961	1,841,398.655
2	2,116,529.563	3,406,999.889	4,402,631.377	2,380,653.731	4,650,608.707	1,927,975.899
3	2,201,525.239	3,445,797.388	4,549,233.365	2,384,668.412	4,753,916.289	2,018,146.099
4	2,289,636.147	3,485,002.261	4,700,453.316	2,388,689.114	4,859,393.331	2,112,058.362
5	2,380,919.049	3,524,618.785	4,856,436.695	2,392,715.848	4,967,085.391	2,209,867.984
6		3,564,651.282	5,017,333.551	2,396,748.622	5,077,038.985	
7		3,605,104.120	5,183,298.658		5,189,301.603	
8		3,645,981.714			5,303,921,737	
9						

Period	7	8	9	10
1	1,820,837.760	4,344,679.668	439,076.864	2,097,642.448
2	1,919,614.735	4,383,680.045	464,802.893	2,127,282.979
3	2,023,034.228	4,423,011.926	491,442.196	2,157,294.016
4	2,131,314.436	4,502,681.491	519,027.194	2,187,680.191
5	2,244.683.815	4,502,681.491	547,591.459	2,218,446.194
6	2,363,381.554	4,543,024.884	577,169.756	
7	2,487,658.087	4,583,711.195	607,798.082	
8			639,513.714	
9			672,355.251	

Source Reprinted from Lee and Lee (2017, p. 686)
Note NPVs are listed in Table 19.8

million to +$15 million, depending on the combinations of random events that could take place. The mean NPV is $4,194,647.409 with a standard deviation of $6,618,476.469. This indicates that there is a 70% chance that the NPV will be greater than 0. In addition, we can use this average NPV and its standard deviation to calculate interval estimate for NPV. In other words, by using simulation we can have interval estimate of NPV, which was used in both statistical distribution method and decision tree method.

Furthermore, if we change the range or distribution of the random variables, we can then perform sensitivity analysis to investigate the impact of a change of an input factor on the risk and return of the investment project.

Also, by using sensitivity analysis, we essentially break down the uncertainty involved in the undertaking of any project, thereby highlighting exactly what the decision-maker should be primarily concerned with in forecasting in terms of those variables critical to the analysis. The information obtained from simulation analysis is valuable in allowing the decision-maker to more accurately evaluate risky capital investments.

19.8 Summary

Important concepts and methods related to capital-budgeting decisions under certainty were explored in Sects. 19.3, 19.4, and 19.5. Cash-flow estimation methods were discussed before alternative capital-budgeting methods were explored. Capital-rationing decisions in terms of linear programming were also discussed in this chapter.

In this chapter, we have also discussed uncertainty and how capital-budgeting decisions are made under conditions of uncertainty. Presented were two methods of handling uncertainty: statistical distribution method and simulation method. Each method is based on the NPV approach, so that, in theory, using any of the methods should yield similar

results. However, in practice, the method used will depend on the availability of information and the reliability of that information.

Appendix 19.1: Solving the Linear Program Model for Capital Rationing

The first step is to choose the cells which represent the unknowns: X, Y, Z, C, D, and E.

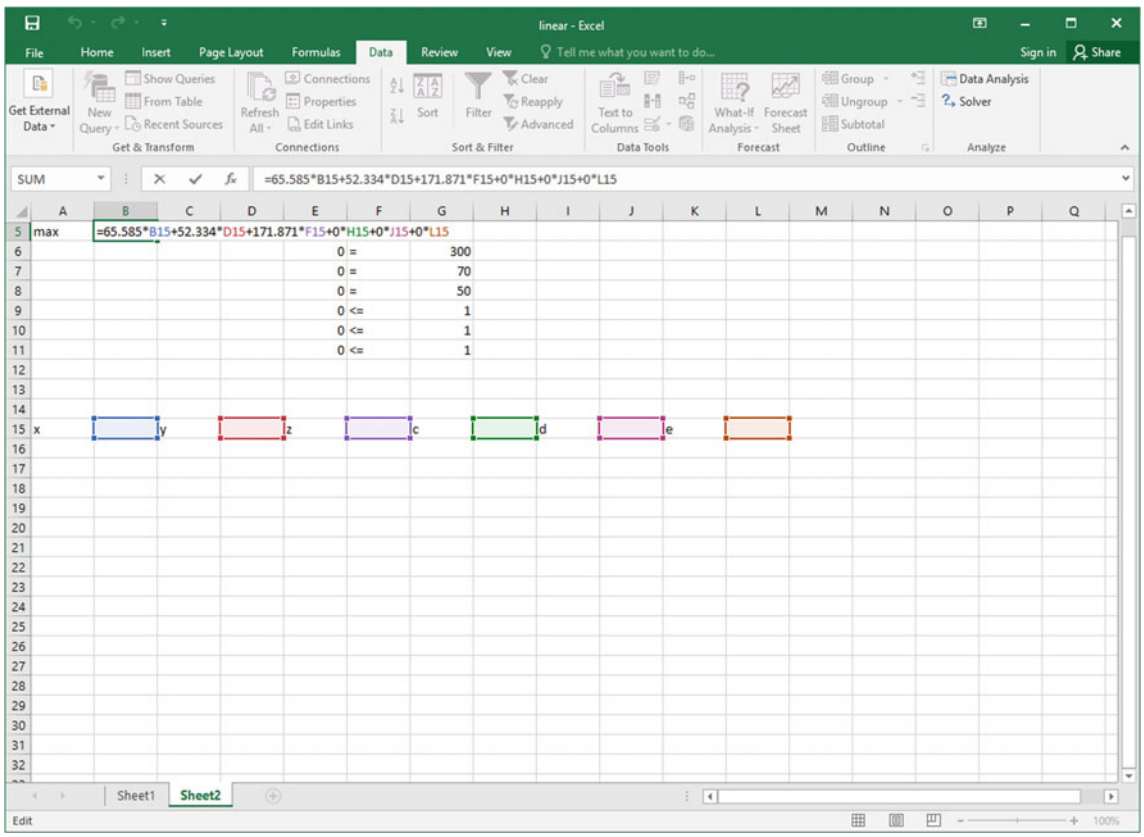
I use B15 to represent X, D15 represent Y, F15 represent Z, H15 represent C, J15 represent D, L15 represent E. Indeed, you can choose any cells to proxy for the unknowns based on your preference.

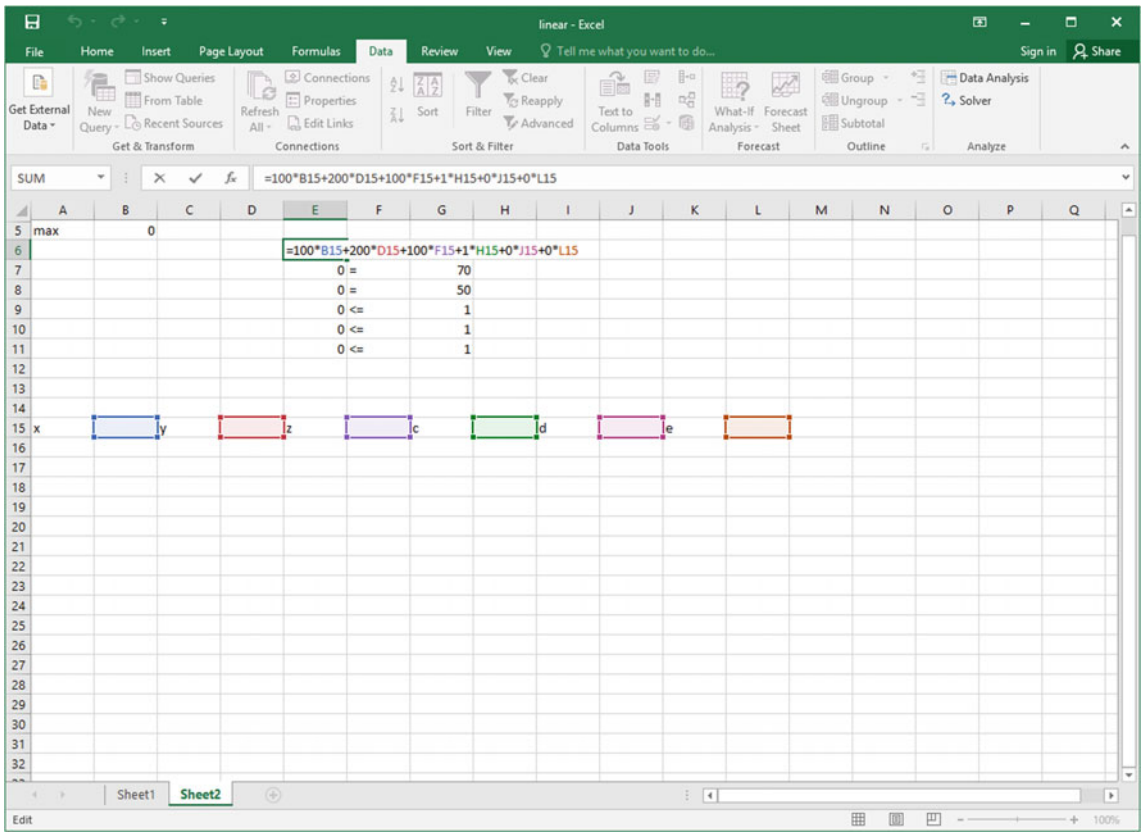
The second step is to express the objective function.

As our objective is to maximize $V = 65.585X + 52.334Y + 171.871Z + 0C + 0D + 0E$, V is our objective function. I then input the expression of the objective function in B5: "$= 65.585^{*}B15 + 52.334 * D15 + 171.871 * F15 + 0 * H15 + 0 * J15 + 0 * L15$".

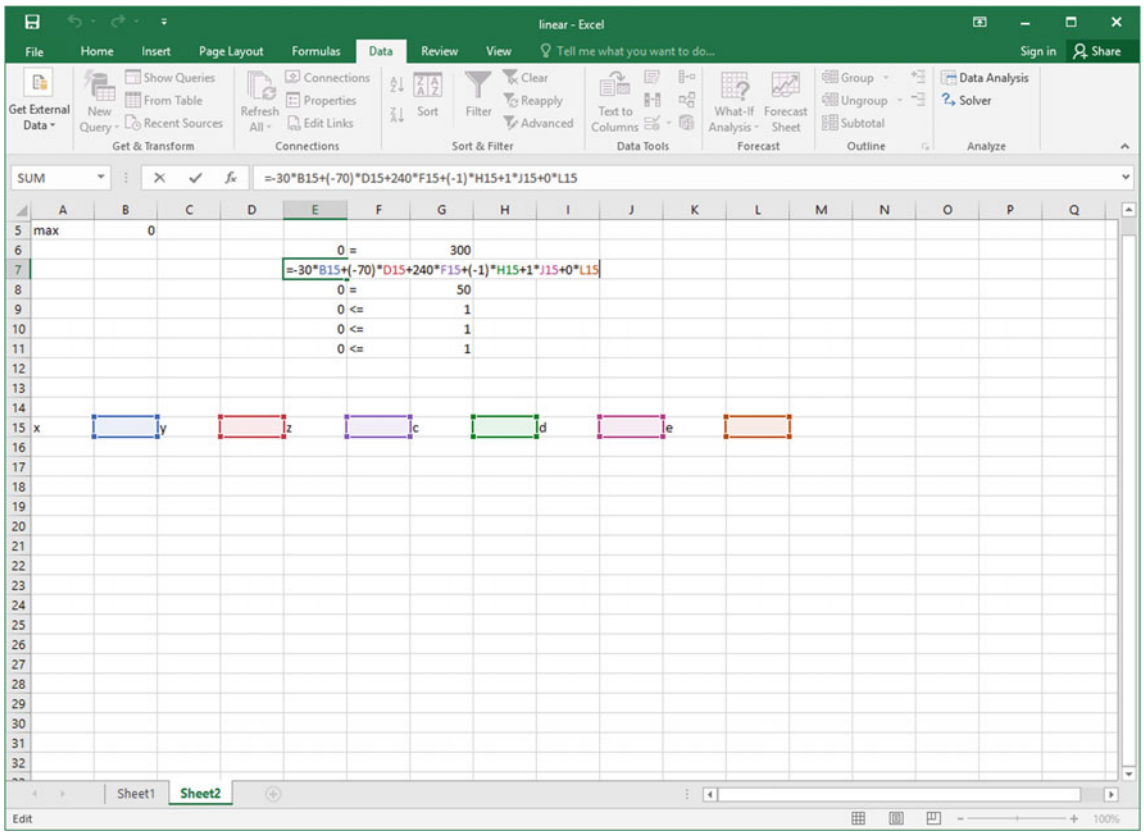
The third step is to input the expression of the constraint.

Our first constraint is $100X + 200Y + 100Z + C + 0D + 0E = 300$, so I input the left side of this equation "$= 100 * B15 + 200 * D15 + 100 * F15 + 1 * H15 + 0 * J15 + 0 * L15$" in E6.

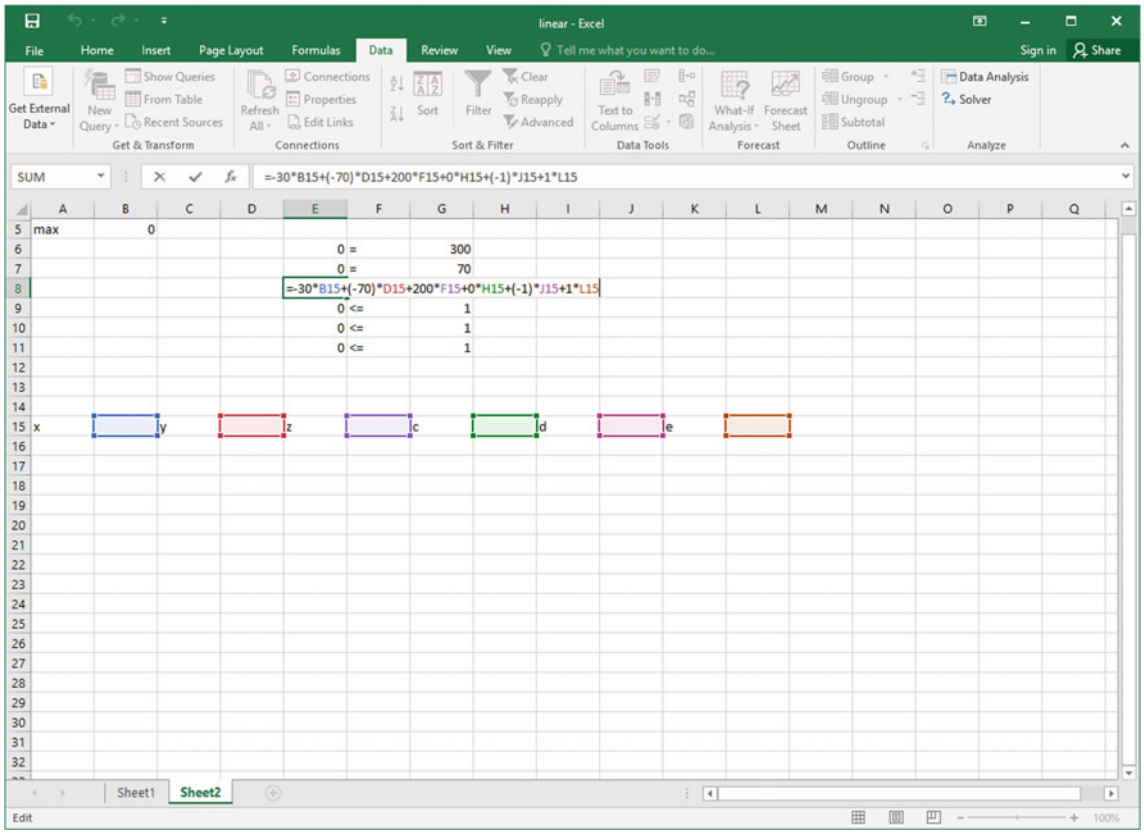




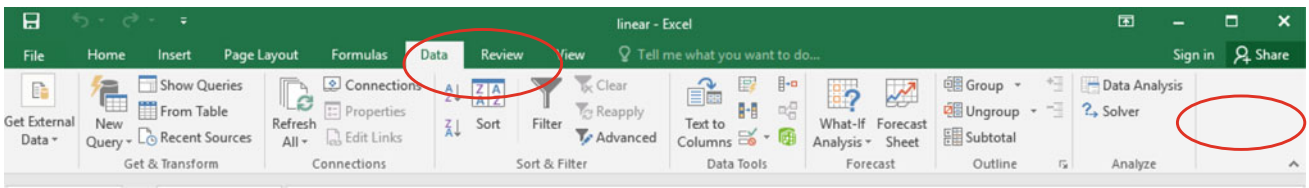
Our second constraint is $-30X - 70Y + 240Z - C + D + 0E = 70$, so I input the left side of this equation "$= -30 * B15 + (-70) * D15 + 240 * F15 + (-1) * H15 + 1 * J15 + 0 * L15$" in E7.



Our third constraint is $-30X - 70Y + 200Z + 0C - D + E = 50.$, so I input the left side of this equation "$= -30 * B15 + (-70) * D15 + 200 * F15 + 0 * H15 + (-1) * J15 + 1 * L15$" in E8.

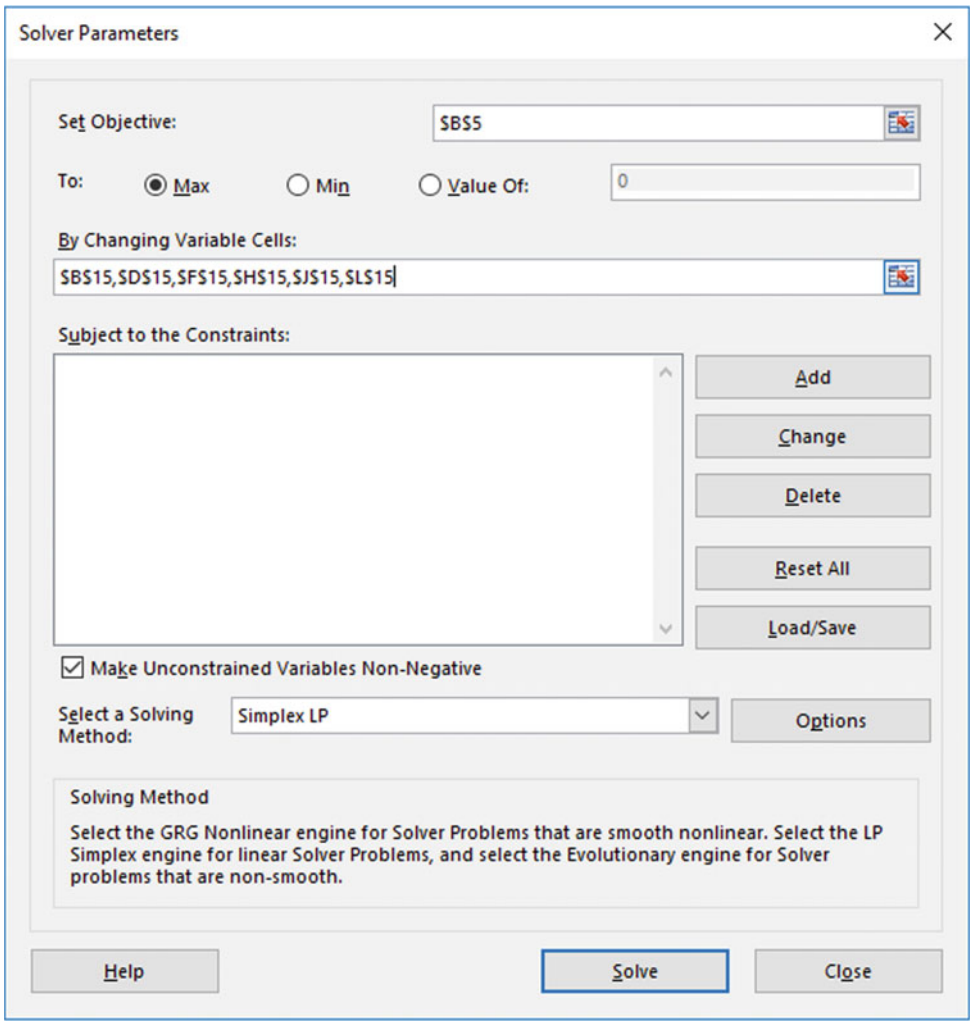


Additionally, we have constraints on X, Y, and Z: $X \leq 1$, $Y \leq 1$, $Z \leq 1$ and non-negative. We will deal with them later. *The fourth step is to click "data" and then open "Solver".*



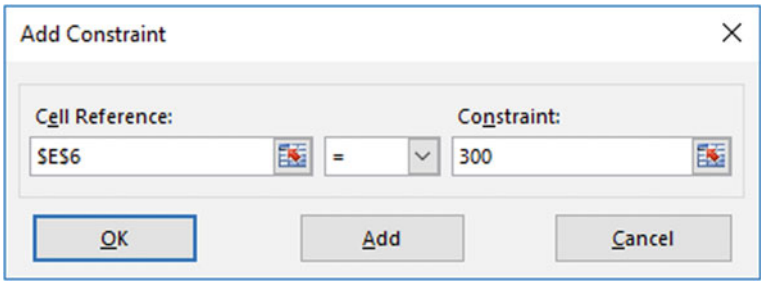
As our objective function is expressed in B5, we select "B5" in the place "set objective". Then we choose "Max" since we want to maximize the function.

Next, we select B15, D15, F15, H15, J15, and L15 in the place "By changing variable cells" since we use these cells to represent our unknowns X, Y, Z, C, D, and E.

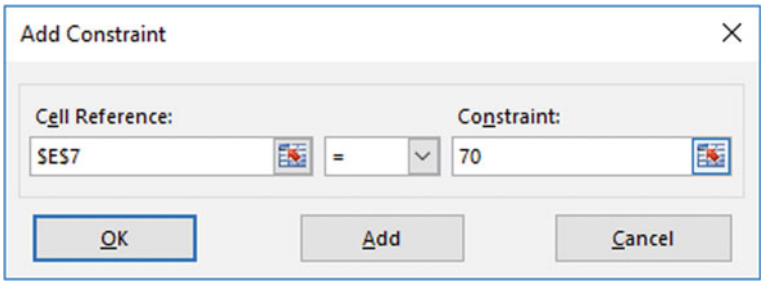


Next, we add constraints via clicking “Add”.

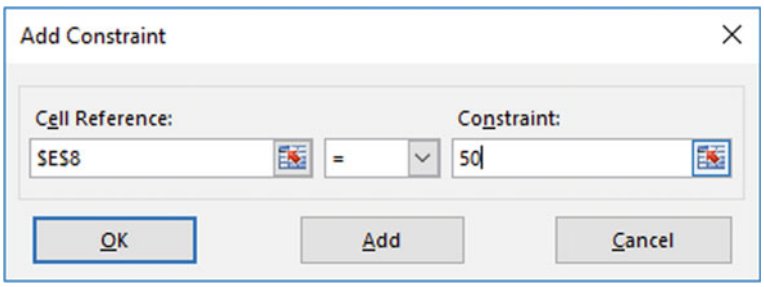
Our first constraint is expressed in E6, so we select E6 in cell reference. Then we let E6 “=300” and click “Add”.



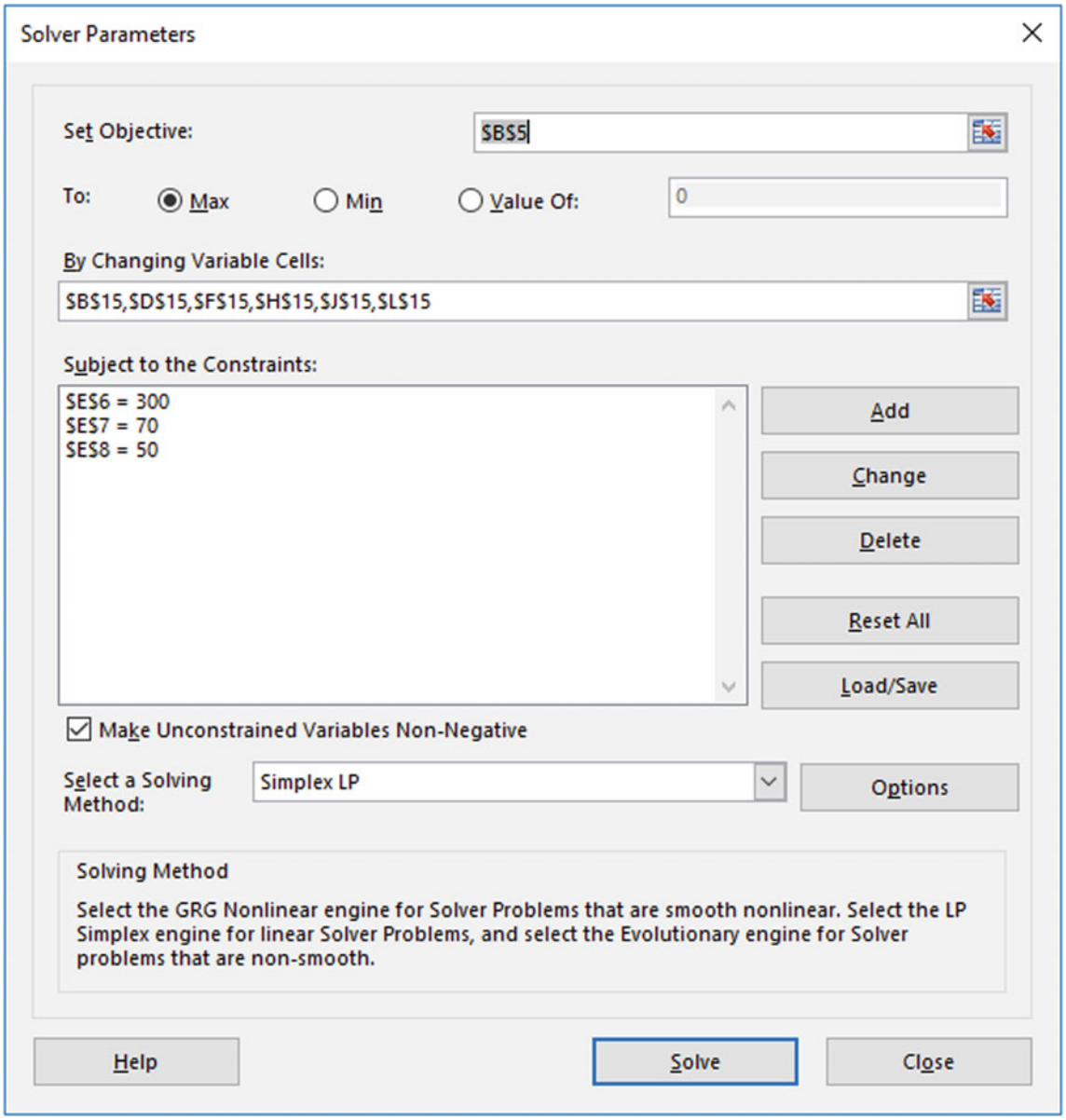
Our second constraint is expressed in E7, so we select E6 in cell reference. Then we let E7 “=70” and click “Add”.



Our third constraint is expressed in E8, so we select E6 in cell reference. Then we let E7 “=50” and click “Add”.

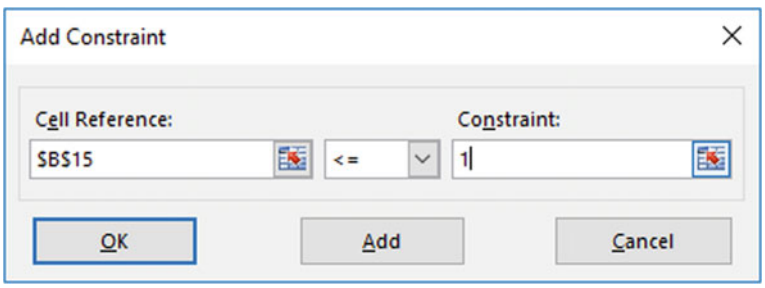


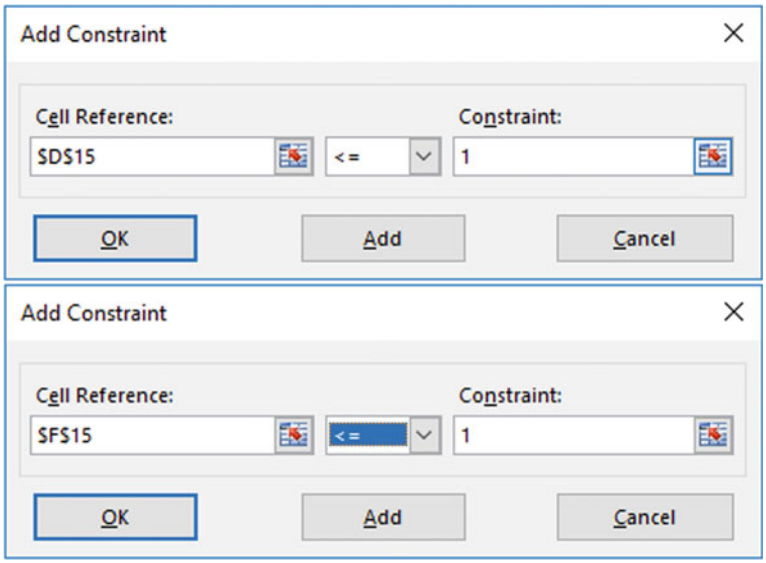
After we finish adding the three constraints, we have the following display:

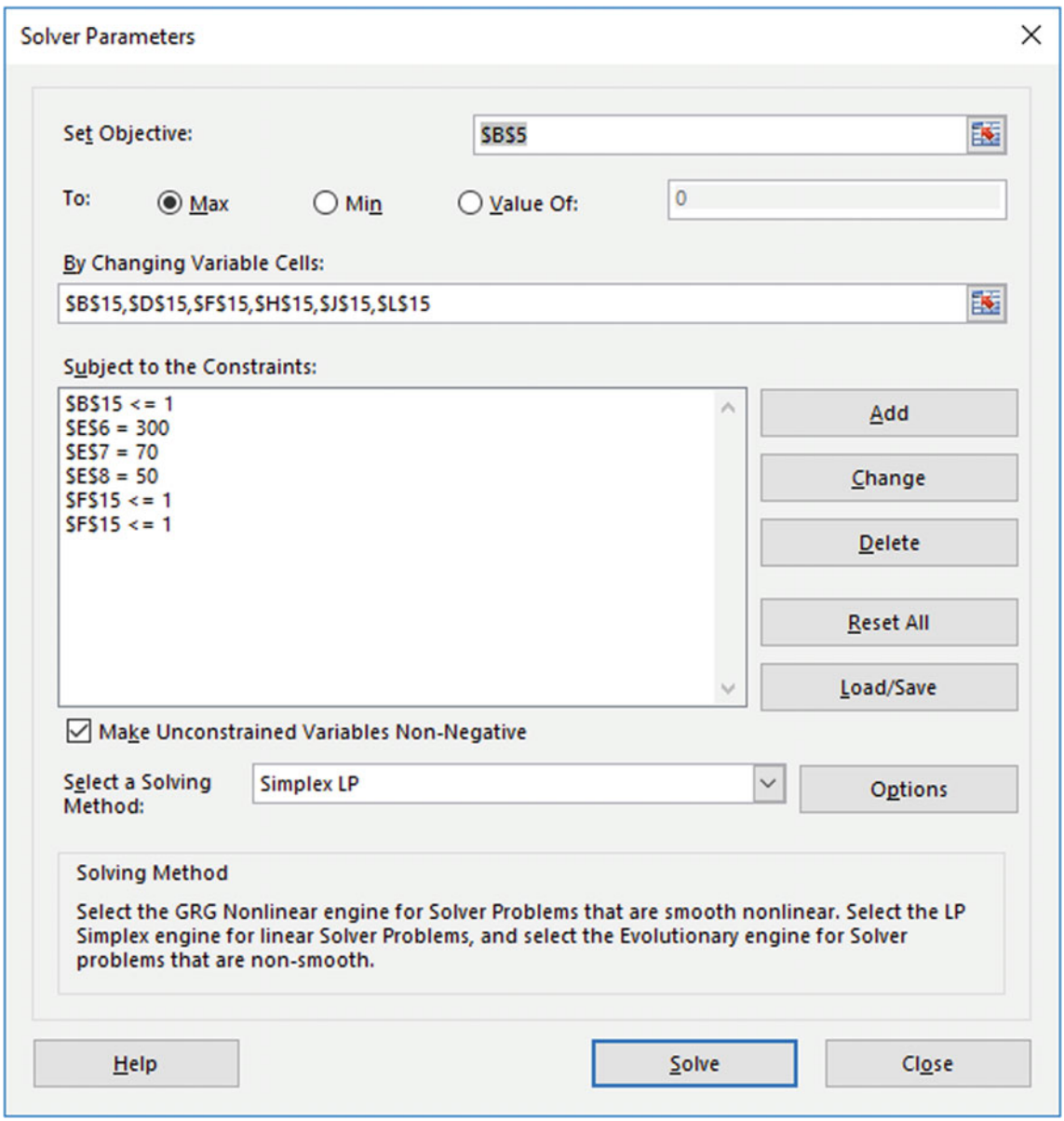


For additional constraints, $X \leq 1$, $Y \leq 1$, $Z \leq 1$ and non-negative, we continue clicking "add" and set them as follows:

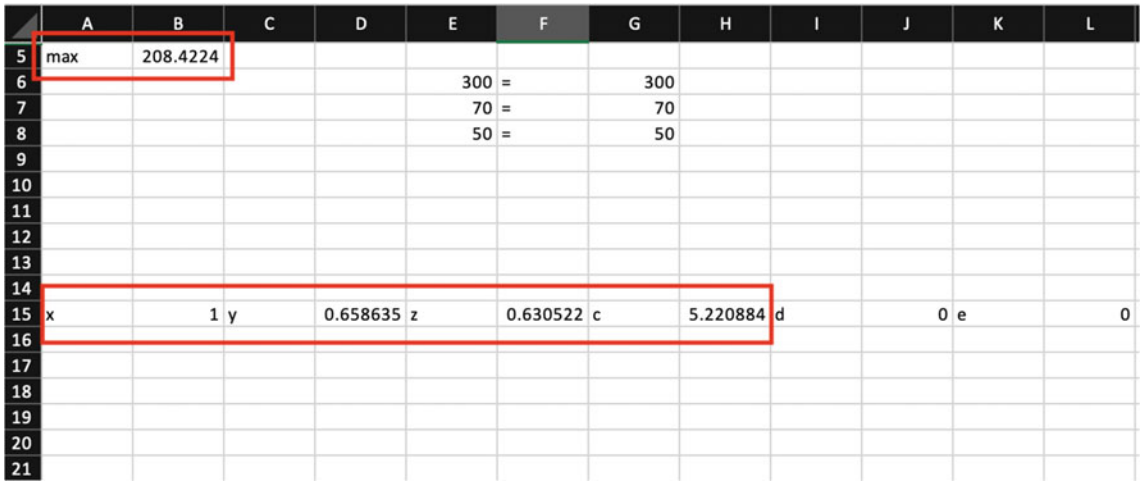
After adding all the constraints, we should select "Make Unconstrained variables Non-negative" because our X, Y, Z, C, D, and E are non-negative. The final display of setting the model is as follows:




Add Constraint
Cell Reference:
$D$15
<=
Constraint:
1
OK
Add
Cancel
Add Constraint
Cell Reference:
$F$15
<=
Constraint:
1
OK
Add
Cancel


Solver Parameters
Set Objective:
$B$5
To:
Max
Min
Value Of:
0
By Changing Variable Cells:
$B$15,$D$15,$F$15,$H$15,$J$15,$L$15
Subject to the Constraints:
$B$15 <= 1
$E$6 = 300
$E$7 = 70
$E$8 = 50
$F$15 <= 1
$F$15 <= 1
Add
Change
Delete
Reset All
Load/Save
Make Unconstrained Variables Non-Negative
Select a Solving Method:
Simplex LP
Options
Solving Method
Select the GRG Nonlinear engine for Solver Problems that are smooth nonlinear. Select the LP Simplex engine for linear Solver Problems, and select the Evolutionary engine for Solver problems that are non-smooth.
Help
Solve
Close

Now, we can click "solve" to get our final result. The Excel will give us the optimal weights X, Y, Z, C, D, and E in B15, D15, F15, H15, J15, and L15, respectively, and the maximum value of V in B5. The results are consistent with the solution shown in the example.



	A	B	C	D	E	F	G	H	I	J	K	L
5	max	208.4224										
6					300	=	300					
7					70	=	70					
8					50	=	50					
9												
10												
11												
12												
13												
14												
15	x	1	y	0.658635	z	0.630522	c	5.220884	d	0	e	0
16												
17												
18												
19												
20												
21												

Appendix 19.2: Decision Tree Method for Investment Decisions

A *decision tree* is a general approach to structuring complex decisions and helps direct the user to a solution. It is a graphical tool that describes the types of actions available to the decision-maker and their consequences.

In capital budgeting decision-making, the decision tree is used to analyze investment opportunities involving a sequence of investment decisions over time. To illustrate the basic ideas of the decision tree, we will develop a problem involving numerous decisions.

First, we must enumerate some of the basic rules to implement this methodology: (1) the decision-maker should try to include only important decisions or events to prevent the tree from becoming a "bush"; (2) the decision tree requires subjective estimates on the part of the decision-maker when assessing probabilities; and (3) the decision tree must be developed in chronological order to ensure the proper sequence of events and decisions.

A decision point is represented by a box, or *decision node*. The available alternatives are represented by branches out of this node. A circle represents an *event node*, and branches from this type of node represent types of possible events.

The *expected monetary value* (EMV) is calculated for each event node by multiplying probabilities by conditional profits and then summing them. The EMV is then placed in the event node and represents the expected value of all branches arising from that node.

Example 19.3

Figure 19.2 illustrates a decision tree for a packaging firm that sells paper and paperboard materials to customers for packaging such items as cans and bottles. The firm predicts that, with the advent of shrink-wrap packaging, their products may be obsolete within a decade. The firm must first decide on one of four short-term plans: (1) do nothing, (2) establish a tie-in with a company that manufactures plastics packaging, (3) acquire such a company, or (4) develop its own plastics packaging. These four alternatives are the first four branches extending from the event node in Fig. 19.2. If the firm does nothing it's short-term profits will be about the same as in previous years. If the firm decides to establish a tie-in with another firm, it foresees either a 90% successful introduction of its new plastics line or a 10% possibility of failure. If the firm decides on acquisition, it foresees a 10% chance of encountering legal barriers, such as problems with antitrust laws; a 30% possibility of an unsuccessful introduction of the plastics line; and a 60% chance of success. If the firm decides to manufacture a plastics line on its own, it foresees many more problems. The firm anticipates a 10% chance of having problems with suppliers in developing a total packaging system for customers, a 30% chance of customers not purchasing the new materials, and a 50% chance of success in the development and introduction of the plastics line.

The third column in Fig. 19.2 is *conditional profit*, the amount of profit the firm can expect to make with the advent of each preceding set of alternative and consequent events.

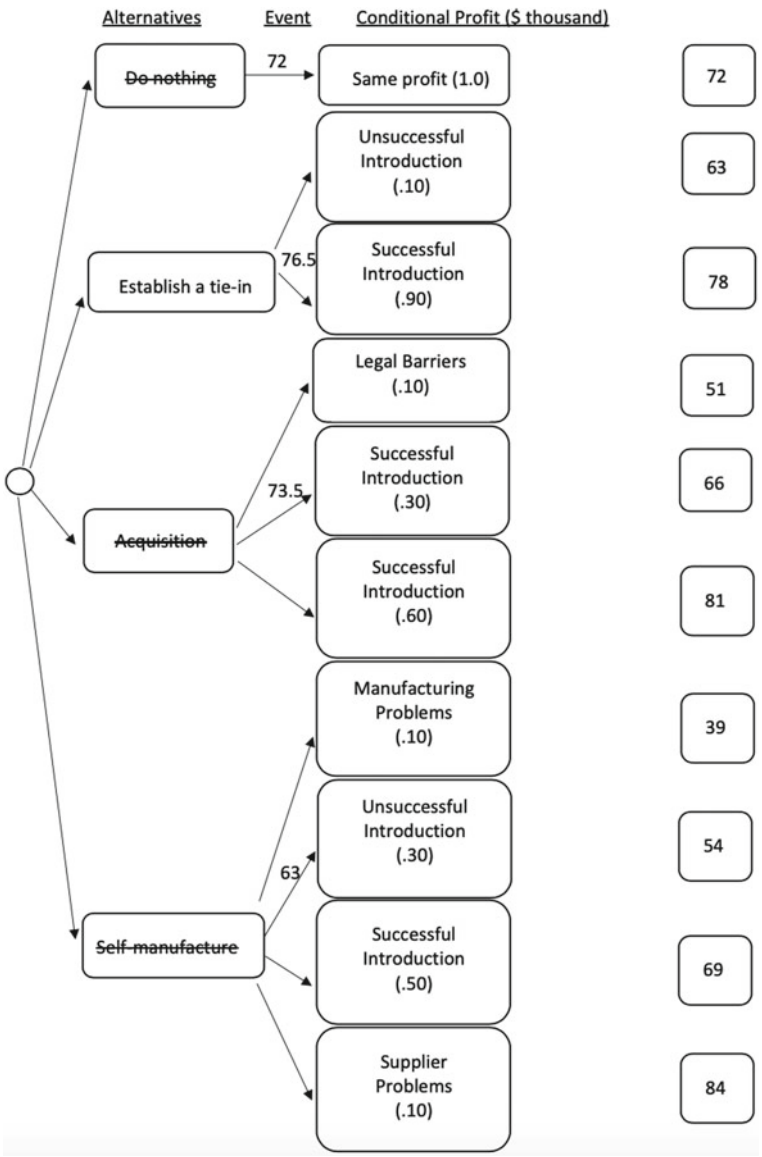


Fig. 19.2 Decision tree for capital-budgeting decision

In Fig. 19.2. the expected monetary values are shown in the event nodes. The financial planner decides which actions to take by selecting the highest EMV, which in this case is $76.5, as indicated in the decision node at the beginning of the tree. The parallel lines drawn across the nonoptimal decision branches indicate the elimination of these alternatives from consideration.

In Example 19.3, we have simplified the number of possible alternatives and events to provide a simpler view of the decision tree process. However, as we introduce more possibilities to this problem, and as it becomes more complex, the decision tree becomes more valuable in organizing the information necessary to make the decision. This is especially true when making a sequence of decisions rather than a single decision. A more detailed discussion of the decision tree method for capital budgeting decision can be found in Chap. 14 of Lee and Lee (2017).

Appendix 19.3: Hillier's Statistical Distribution Method for Capital Budgeting Under Uncertainty

In this chapter, we discussed the calculation of the standard deviation of NPV (1) where cash flows are independent of each other as presented in Eq. 19.17 and (2) where cash flows are perfectly positively correlated as presented in Eq. 19.17a. In either case, the covariance term drops out of the equation for the variance of the NPV. Now we develop a general formula for the standard deviation of NPV that can be used for all cash flow relationships.

The general equation for the standard deviation of NPV (σ_{NPV}) with a mean of

$$NPV = \sum_{t=1}^{N} \frac{\overline{C}_t}{(1+k)^t} + \frac{\overline{S}_t}{(1+k)^N} - I_0$$

is

$$\sigma_{NPV} = \left[\sum_{t=1}^{N} \frac{\sigma_t^2}{(1+k)^{2t}} + \sum_{t=1}^{N} \sum_{\tau=1}^{N} W_t W_\tau COV(C_\tau C_t) \right]^{\frac{1}{2}} (\tau \neq t) \tag{19.19}$$

where σ_t^2 = variance of cash flows in the tth period; W_t and W_τ = discount factors for the tth and τth period (that is, $W_t = 1(1+\mathrm{K})^{\mathrm{t}}$ and $W_\tau = 1/(1+\mathrm{K})^{\tau}$; and $\mathrm{COV}(\mathrm{C}_t, \mathrm{C}_\tau)$ = covariability between cash flows in t and τ (that is, $\mathrm{COV}(\mathrm{C}_t, \mathrm{C}_\tau) = \rho_{t\tau}\sigma_\tau\sigma_t$, where $\rho_{t\tau}$ = correlation coefficient between cash flow in tth and τth period).

Cash flows between periods t and τ are generally related. Therefore, $\mathrm{COV}(\mathrm{C_t}, \mathrm{C}_\tau)$ is an important factor in the estimation of σ_{NPV}. The magnitude, sign, and degree of the relationships of these cash flows depend on the economic operating conditions and the nature of the product or service produced.

Using portfolio theory to calculate the standard deviation of a set of securities, we have derived Eq. 19.19, which can be explained by an example. Suppose we have cash flows for a three-year period, C_1, C_2, C_3, with discount factors of W_1, W_2, W_3. Table 19.11 shows the calculation of σ_{NPV}.

The summation of the diagonal ($W^2{}_1\sigma_1^2$, $W^2{}_2\sigma_2$, $W^2{}_3\sigma_3^2$) results in the first part of Eq. 19.19, or

$$\sum_{t=1}^{N} \sum_{\substack{\tau=1 \\ t \neq \tau}}^{N} W_t W_\tau COV(C_\tau, C_t)$$

This calculation is similar to the calculation of portfolio variance, as discussed in Chap. 19. However, in portfolio analysis, W_t represents the percent of money invested in the ith security, and the summation of W_t equals 1. In the calculation of σ_{NPV}, W_t represents a discount factor. Therefore, the summation of W_t will not necessarily equal 1.

Table 19.11 Variance covariance matrix

	W_1C_1	W_2C_2	W_3C_3
W_1C_1	$W^2{}_1\,\sigma_1^2$	$W_1W_2COV(C_1, C_2)$	$W_1W_3COV(C_1, C_3)$
W_2C_2	$W_1W_2COV(C_2, C_1)$	$W^2{}_2\,\sigma_2^2$	$W_2W_3COV(C_2, C_3)$
W_3C_3	$W_1W_3COV(C_3, C_1)$	$W_2W_3COV(C_2, C_3)$	$W^2{}_3\,\sigma_3^2$

Equation 19.19 is the general equation for σ_{NPV}. Thus, Eq. 19.17 for σ_{NPV} under perfectly correlated cash flows or independent cash flows is a special case derived from the general Eq. 19.19.

Hillier (1963) combined the assumption of mutual independence and perfect correlation to develop a mode of σ_{NPV} to deal with mixed situations. This model is presented in Eq. 19.20, which analyzes investment proposals in which expected cash flows are a combination of correlated and independent flows.

$$\sigma = \left[\sum_{t=1}^{N} \frac{\sigma_{yt}^2}{(1+k)^{2t}} + \sum_{h=1}^{m} \left(\sum_{t=0}^{N} \frac{\sigma_{zt}^h}{(1+k)^t} \right)^2 \right] \tag{19.20}$$

where σ_{yt}^2 = variance for an independent net cash flow in period t and σ_{zt}^h = standard deviation for stream h of a perfectly correlated cash flow stream in t. If h = 1, then Eq. 19.20 is a combination of Eqs. 19.17 and 19.17a.

References

Ackoff, Russell. "A concept of corporate planning." Long Range Planning 3.1 (1970): 2–8.

Copeland, Thomas E, J. Fred Weston, Kuldeep Shastri, Financial Theory and Corporate Policy (4th Edition) Pearson, 2004

Fama, E.F. and Miller, M.H. (1972) The Theory of Finance. Holt, Rinehart and Winston, New York.

Fisher, I., **The Theory of Interest**, MacMillan, New York, 1930.

Hamburg, Morris. "Statistical Analysis for Decision Making. NY: Harccurt Brace Jovanovich." (1983).

Hertz, D. B. "Risk Analysis in Capital Investments," *Harvard Business Review*, 42 (1964, pp. 95–106).

Hertz, D. B. "Risk Analysis in Capital Investments," *Harvard Business Review*, 57 (1979, pp. 169–81).

Hillier, F. S. "The Derivation of Probabilistic Information for the Evaluation of Risky Investments," *Management Science*, 9 (1963, pp. 443–57).

Lee. C. F. and J. Lee *Financial Analysis, Planning & Forecasting: Theory Application* (Singapore: World Scientific, 2017).

Lee, C. F. and S. Y. Wang "A Fuzzy Real Option Valuation Approach to Capital Budgeting Under Uncertainty Environment," *International Journal of Information Technology & Decision Making*, Volume: 9, Issue: 5, pp. 695–713, 2010.

Pinches, G. E. "Myopic, Capital Budgeting and Decision Making," *Financial Management*, 11 (Autumn 1982, pp. 6–19).

Reinhardt, Uwe E. "BREAK-EVEN ANALYSIS FOR LOCKHEED'S TRI STAR: AN APPLICATION OF FINANCIAL THEORY." The Journal of Finance 28.4 (1973): 821–838.

Weingartner, H. Martin. "The excess present value index-A theoretical basis and critique." Journal of Accounting Research (1963): 213–224.

Weingartner, H. Martin. "Capital rationing: n authors in search of a plot." The Journal of Finance 32.5 (1977): 1403–1431.

Check for updates

Financial Analysis, Planning, and Forecasting

20

20.1 Introduction

This chapter covers alternative financial planning models and their use in financial analysis and decision-making. The approach taken in this chapter gives the student an opportunity to combine information (accounting, market, and economics), theory, (classical, M & M, CAPM, and OPM), and methodology (regression and linear programming).

We begin by presenting the procedure for financial planning and analysis in Sect.20.2. This is followed by a discussion of the Warren and Shelton algebraic simultaneous equations planning model in Sect. 20.3. Section 20.4 covers the application of linear programming (LP) to financial planning and analysis, Sect. 20.5 discusses the application of econometric approaches to financial planning and analysis, and Sect. 20.6 talks about the importance of sensitivity analysis and its application to Warren and Shelton's financial planning model. Finally, Sect. 20.7 summarizes the chapter. Appendix 20.1 shows how the simplex method is used in the capital rationing decision. Appendix 20.2 is a description of parameter inputs used to forecast Johnson & Johnson's financial statements and share price. Appendix 20.3 shows the procedure of how to use Excel to implement the FinPlan program.

20.2 Procedures for Financial Planning and Analysis

Before discussing the various financial planning models, we must first be sure of our understanding of what the financial planning process is all about. Otherwise, we run the risk of too narrowly defining financial planning as simply data gathering and running computer programs. In reality, financial planning involves a *process* of analyzing alternative dividend, financing, and investment strategies, forecasting their outcome and impact within various economic environments, and then deciding how much risk to take on and which projects to pursue. Thus, financial planning models are merely tools to improve forecasting as well as to help managers better understand the interactions of dividend, financing, and investment decisions.

More formally, we can outline the financial planning process as follows:

1. *Utilize* the existing set of economic, legal, accounting, marketing, and company policy information.
2. *Analyze* the interactions of the dividend, financing, and investment choices open to the firm.
3. *Forecast* the future consequences of present decisions to avoid unexpected events as well as to aid in understanding the interaction of present and future decisions.
4. *Decide* which alternatives the firm should undertake, the explicit outline for which is contained in the financial plan.
5. *Evaluate* the subsequent outcome of these decisions once they are implemented against the objectives set forth in the financial plan.

So where does the financial planning model come in? To clarify its role in this process, look at Fig. 20.1, which presents a flowchart of a financial planning model. The inputs to the model are economic and accounting information (discussed in Chap. 2) and market and policy information (discussed in Chaps. 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 and 20). Three alternative financial planning, analysis, and forecasting models are (1) the algebraic simultaneous equations model, (2) the linear programming model, and (3) the econometric model.[1] The outputs of the financial planning and forecasting model are pro forma financial statements, forecasted PPS, EPS, and DPS, new equity issued, and new debt issued. Essentially, the benefit of the

[1] This chapter discusses three alternative financial planning models. The simultaneous equation model can be found in Lee and Lee's (2017) Chapter 24. The linear programming model can be found in Chaps. 22 and 23. Finally, the econometric type of financial planning model can be found in Chap. 26. This chapter has discussed the simultaneous equation model in detail; however, the other two models have only been briefly discussed. For further information on these two models, see Lee and Lee (2017).

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_20

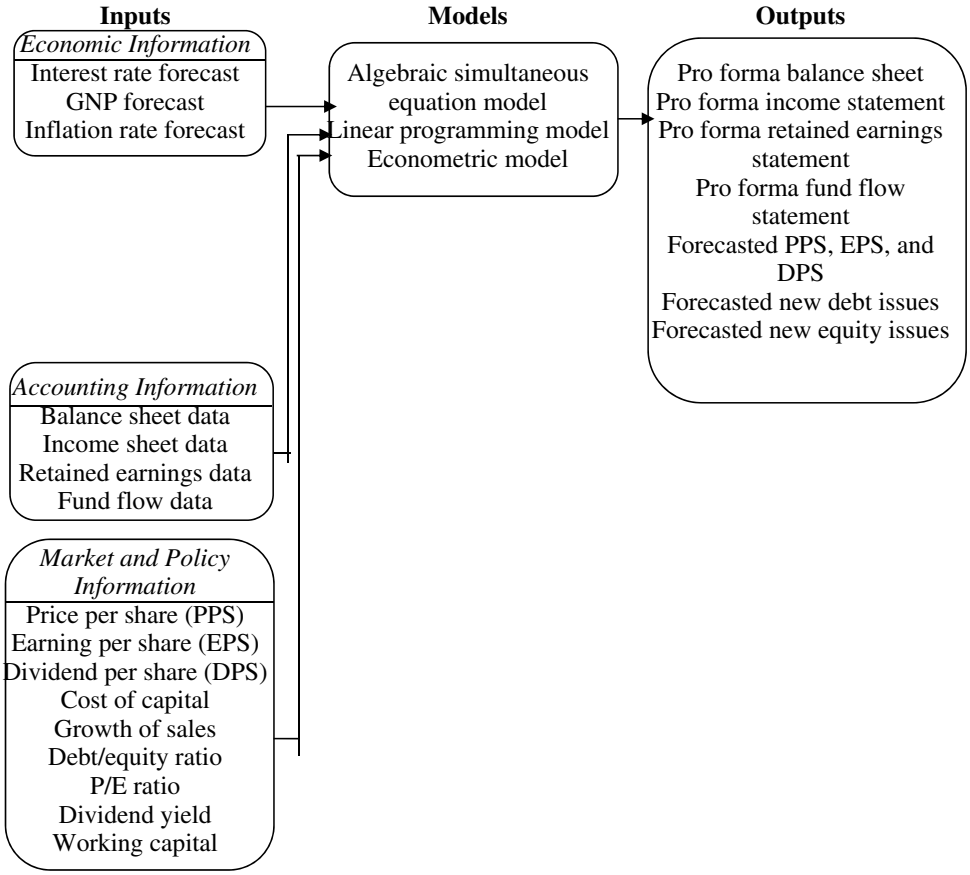


Fig. 20.1 Inputs, models, and outputs for financial planning and forecasting models

model is to efficiently and effectively handle the analysis of information and its interactions with the forecasting of future consequences within the planning process.

Hence, the financial planning model efficiently improves the depth and breadth of the information the financial manager uses in the decision-making process. Moreover, before the finalized plan is implemented, an evaluation of how well subsequent performance stands up to the financial plan provides additional input for future planning actions.

A key to the value of any financial planning model is how it is formulated and constructed. That is, the credibility of the model's output depends on the underlying assumptions and particular financial theory the model is based on, as well as its ease of use for the financial planner. Because of its potentially great impact on the financial planning process and, consequently, on the firm's future, the particular financial planning model to be used must be chosen carefully. Specifically, we can state that a useful financial planning model should have the following characteristics:

1. The model results and assumptions should be credible.
2. The model should be flexible so that it can be adapted and expanded to meet a variety of circumstances.
3. The model should improve on current practice in a technical or performance sense.
4. The model inputs and outputs should be comprehensible to the user without extensive additional knowledge or training.
5. The model should take into account the interrelated investment, financing, dividend, and production decisions and their effect on the firm's market value.
6. The model should be fairly simple for the user to operate without the extensive intervention of nonfinancial personnel and tedious formulation of the input.

On the basis of these guidelines, we now present and discuss the simultaneous equations, linear programming, and econometric financial planning models, which can be used for financial planning and analysis.

20.3 The Algebraic Simultaneous Equations Approach to Financial Planning and Analysis

In this section, we present the financial planning approach of Warren and Shelton (1971), which is based on a *simultaneous equations* concept. The model, called FINPLAN, deals with overall corporate financial planning as opposed to just some are of planning, such as capital budgeting. The objective of the FINPLAN model is not to optimize anything, but rather, to serve as a tool to provide relevant information to the decision-maker. One of the strengths of this planning model, in addition to its construction, is that it allows the user to simulate the financial impacts of changing assumptions regarding such variables as sales, operating ratios, price-to-earnings ratios, retention rates, and debt-to-equity ratios.

The advantage of utilizing a simultaneous equation structure to represent a firm's investment, financing, production, and dividend policies is the enhanced ability for the interaction of these decision-making areas. The Warren and Shelton (WS) model is a system of 20 equations which are listed in Table 20.1. These equations are segmented into distinct subgroups corresponding to sales, investment, financing, and per share (return to investors) data. The flowchart describing the interrelationships of the equations is shown in Fig. 20.2.

The key concepts of the interaction of investment, financing, and dividends, as explained in Chap. 13, are the basis of the FINPLAN model, which we now consider in some detail. First, we discuss the inputs to the model; second, we delve into the interaction of the equations in the model; and third, we look at the output of the FINPLAN model.

The inputs to the model are shown in Table 20.2B. The driving force of the WS model is the sales growth estimates ($GSALS^t$). Equation (20.1) in Table 20.1 shows that sales for period t is the product of sales in the prior period multiplied by the growth rate in sales for period t. *EBIT* is then derived, by expressing *EBIT* as a percentage of the sales ratio, as in Eq. (2) of Table 20.1. Current and fixed assets are then derived in Eqs. 3 and 4 of the table through the use of the *CA/SALES* and *FA/SALES* ratios. The sum of *CA* and *FA* is the total assets for the period.

Financing of the desired level of assets is undertaken in Sect. 3 of the table. In Eq. 6, current liabilities in period t are derived from the ratio of *CL/SALES* multiplied by *SALES*. Equation 20.7 represents the funds required (NF^t). FINPLAN assumes that the amount of preferred stock is constant over the planning horizon. In determining what funds are

Table 20.1 WS model

Section 1—Generation of sales and earnings before interest and taxes for period t.

(1) $\text{SALES}_t = \text{SALES}_{t-l}(1 + \text{GSALS}_t)$

(2) $EBIT_t = REBIT_t SALES_t$

Section 2—Generation of total assets required for period t

(3) $CA_t = RCA_t SALES_t$

(4) $FA_t = RFA_t SALES_t$

(5) $A_t = CA_t + FA_t$

Section 3—Financing the desired level of assets

(6) $CL_t = RCL_t SALES_t$

(7) $NF_t = (A_t - CL_t - PFDSK_t) - (L_{t-1} - LR_t) - S_{t-1} - R_{t-1} - b_r$
$\{(1 - T_t)[EBIT_t - i_{t-1}(L_{t-1} - LR_t)] - PFDIV_t\}$

(8) $NF_t + b_t(1 - T_t)\left[l_t^e NL_t + U_t^l NL_t\right] = NL_t + NS_t$

(9) $L_t = L_{t-1} - LR_t + NL_t$

(10) $S_t = S_{t-1} + NS_t$

(11) $R_t = R_{t-1} + b_t\left\{(1 - T_t)\left[EBIT_t - i_t L_t - U_t^l NL_t\right] - PFDIV_t\right\}$

(12) $i_t = i_{t-1}\left(\frac{L_{t-1} - LR_t}{L_t}\right) + i^e \frac{NL_t}{L_t}$

(13) $\frac{L_t}{S_t + R_t} = K_t$

Section 4—Generation of per share data for period t

(14) $EAFCD_t = (1 - T_t)\left[EBIT_t - i_t L_t - U_t^l NL_t\right] - PFDIV_t$

(15) $CMDIV_t = (1 - b_t)EAFCD_t$

(16) $NUMCS^t = NUMCS_{t-1} + NEWCS_t$

(17) $NEWCS_t = \frac{NS_t}{(1 - U_t^s)P_t}$

(18) $P_t = m_t EPS_t$

(19) $EPS_t = \frac{EAFCD_t}{NUMCS_t}$

(20) $DPS_t = \frac{CMDIV_t}{NUMCS_t}$

Source Adapted from Warren and Shelton (1971)

The above system is "complete" in 20 equations and 20 unknowns. The unknowns are listed and defined in this table along with the parameters (inputs) management is required to provide.

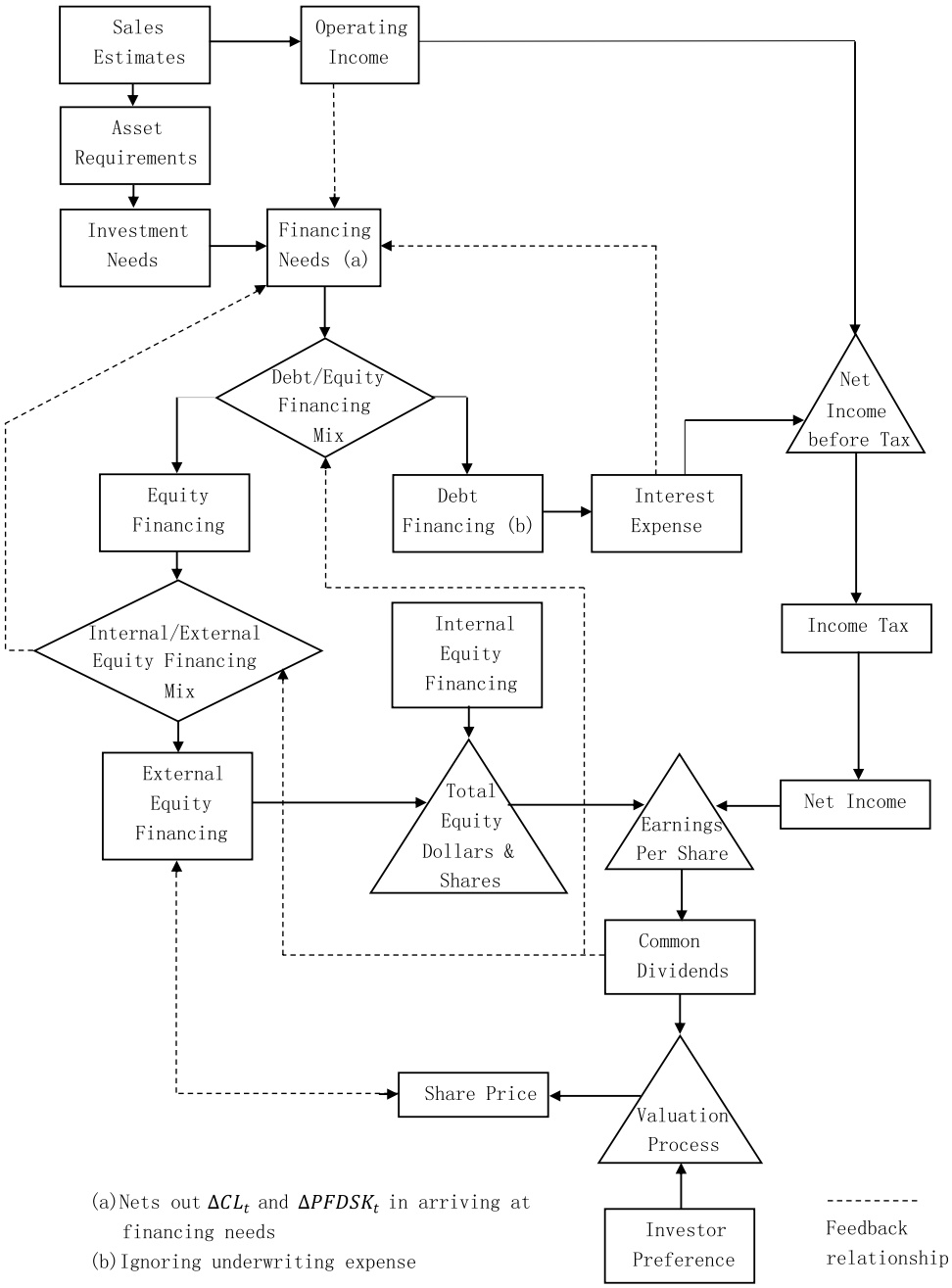


Fig. 20.2 Flow chart of a simplified financial planning model

needed and where they are to come from, FINPLAN uses a source-and-use-of-funds accounting identity. For instance, Eq. 20.7 shows that the assets for period t are the basis for the firm's financing needs. Current liabilities, as determined in the prior equation, are one source of funds and therefore are subtracted from asset levels. As mentioned above, preferred stock is a constant and therefore must be subtracted also. After the first term in Eq. 20.7, $(A_t - CL_t - PFDSK_t)$, we have the financing that must come from internal sources (retained earnings and operations) and long-term external sources (debt and stock issues). The term in the second parenthesis, $(L_{t-1} - LR_t)$, takes into account the remaining old debt outstanding, after retirements, in period t. Then the funds provided by existing stock and retained earnings are subtracted. The last quantity is the funds provided by operations during period t.

Once the funds needed for operations are defined, Eq. 8 specifies that new funds, after taking into account underwriting costs and additional interest costs from new debt, are to come from long-term debt and new stock issues. Equations 20.9 and 20.10 simply update the debt and equity accounts for the new issues. Equation 20.11 updates the

Table 20.2 List of unknowns and list of parameters provided by management

A. Unknowns	
1. $SALES_t$	Sales
2. CA_t	Current assets
3. FA_t	Fixed assets
4. A_t	Total assets
5. CL_t	Current payables
6. NF_t	Needed funds
7. $EBIT_t$	Earnings before interest and taxes
8. NL_t	New debt
9. NS_t	New stock
10. L_t	Total debt
11. S_t	Common stock
12. R_t	Retained earnings
13. I_t	Interest rate on debt
14. $EAFCD_t$	Earnings available for common dividends
15. $CMDIV_t$	Common dividends
16. $NUMCS_t$	Number of common shares outstanding
17. $NEWCS_t$	New common shares issued
18. P_t	Price per share
19. EPS_t	Earnings per share
20. DPS_t	Dividends per share
B. Provided by management	
21. $SALES_{t-1}$	Sales in previous period
22. $GSALS_t$	Sustainable growth rate
23. RCA_t	Current assets as a percent of sales
24. RFA_t	Fixed assets as a percent of sales
25. RCL_t	Current payables as a percent of sales
26. $PFDSK_t$	Preferred stock
27. $PFDIV_t$	Preferred dividends
28. L_{t-1}	Debt in previous period
29. LR_t	Debt repayment
30. S_{t-1}	Common stock in previous period
31. R_{t-1}	Retained earnings in previous period
32. b_t	Retention rate
33. T_t	Average tax rate
34. i_{t-1}	Average interest rate in previous period
35. i^e_t	Expected interest rate on new debt
36. $REBIT_t$	Operating income as a percent of sales
37. U^I_t	Underwriting cost of debt
38. U^s_t	Underwriting cost of equity
39. K_t	Ratio of debt to equity
40. $NUMSCS_{t-1}$	Number of common shares outstanding in previous period
41. m_t	Price-earnings ratio

Source Adapted from Warren and Shelton (1971)

retained-earnings account for the portion of earnings available to common stockholders from operations during period t. Specifically, b^t is the retention rate in period t, and $(1 - T^t)$ is the after-tax percentage, which is multiplied by the earnings from the period after netting out interest costs on both new and old debt. Since preferred stockholders must be paid before common stockholders, preferred dividends must be subtracted from funds available for common stockholders. Equation 20.12 calculates the new weighted-average interest rate for the firm's debt. Equation 20.13 is the new debt-to-equity ratio for period t.

Section 4 of Table 20.1 applies to the common stockholder; in particular, dividends and market value. Equation 14 represents the earnings available for common dividends and is simply the firm's after-tax earnings. Correspondingly, Eq. 15 computes the earnings to be paid to common stockholders. Equation 16 updates the number of common shares for new issues.

As Eq. 17 shows, the number of new common shares is determined by the total new stock issue divided by the stock price after discounting for issuance costs. Equation 18 determines the price of the stock through the use of a price-earnings ratio (m^t) of the stock purchase. Equation 19 determines *EPS*, as usual, by dividing earnings available to common stockholders by the number of common shares outstanding. Equation 20 determines dividends in a similar manner.

Tables 20.3, 20.4, and 20.5 illustrate the setup of the necessary input variables and the resulting output of the pro forma balance sheet and income statement for the Exxon Company. As mentioned, the WS equation system requires values for parameter inputs, which for this example are listed in Table 20.3. The first column represents the value of the input, while the second column corresponds to the variable number. The third and fourth columns pertain to the beginning and ending periods for the desired planning horizon.

From Tables 20.4 and 20.5 you can see the type of information the FINPLAN model generates. With 2016 as a base year, planning information is forecasted for the firm over the period 2017–2020. Based on the model's construction, its underlying assumptions, and the input data, the WS model reveals the following:

1. The amount of investment to be carried out
2. How this investment is to be financed
3. The amount of dividends to be paid
4. How alternative policies can affect the firm's market value

Even more important, as we will explore later in this chapter, this model's greatest value (particularly for FINPLAN) arises from the sensitivity analysis that can be performed. That is, by varying one or several of the input parameters, the financial manager can better understand how his or her decisions interact and, consequently, how they will affect the company's future. (Sensitivity analysis is discussed in greater detail later in this chapter.)

We have shown how we can use Excel to solve 20 simultaneous equation systems as presented in Table 20.1, and the results are presented in Table 20.4 and Table 20.5. Now, we will discuss how we can use the data from Table 20.3 to calculate the unknown variables for Sect. 1, Sect. 2, Sect. 3, and Sect. 4 in 2017.

Section 1: Generation of Sales and Earnings before Interest and Taxes for Period t

1. $\text{Sales}_t = \text{Sales}_{t-1} \times (1 + \text{GSALS}_t)$
   $= 71,890 \times 1.1267$
   $= 80,998.46$
2. $\text{EBIT}_t = \text{REBIT}_{t-1} \times \text{Sales}_t$
   $= 0.2872 \times 80998.463$
   $= 23,262.76$

Section 2: Generation of Total Assets Required for Period t

3. $CA_t = RCA_{t-1} \times \text{Sales}_t$
   $= 0.9046 \times 80998.463$
   $= 73,271.21$
4. $FA_t = RFA_{t-1} \times \text{Sales}_t$
   $= 1.0596 \times 80998.463$
   $= 85,825.97$
5. $A_t = CA_t + FA_t$
   $= 73271.21 + 85825.97$
   $= 159,097.18$

Section 3: Financing the Desired Level of Assets

6. $CL_t = RCL_{t-1} \times \text{Sales}_t$
   $= 0.3656 \times 80998.463$
   $= 29,613.00$

7. $NF_t = (A_t - CL_t - PFDSK_t) \quad -(L_{t-1} - LR_t) - S_{t-1} - R_{t-1} - b_t \times \{(1 - T_t)[EBIT_t - i_t - 1(L_t - 1 - LR_t) - PFDIV_t]\} = (159097.181 - 29,613.00 - 0) - (22,442 - 2,223) - 3,120.0 - 110,551 - 0.4788 \times \{(1 - 0.18) \times [23262.76 - 0.0332 \times (22,442 - 2,223)] - 0\} = -13,275.64$

12. $i_t L_t = i_{t-1}(L_{t-1} - LR_t) + i_{t-1}^{e_t} \times NL_t$
    $= 0.0332 \times (22,442 - 2,223) + 0.0368 \times NL_t$
    $= 671.2708 + 0.0368 \times NL_t$

Table 20.3 FINPLAN inputs

	Value of data	Variable number	Beginning period	Last period
The number of years to be simulated	4	1	0	0
Net Sales at t-1 = 2016	71,890	2	0	0
Growth in SALES	0.1267	3	1	4
Current assets as a percent of sales	0.9046	4	1	4
Fixed assets as a percent of sales	1.0596	5	1	4
Current payables as a percent of sales	0.3656	6	1	4
Preferred stock	0	7	1	4
Preferred dividends	0	8	1	4
Long term debt in 2016	22,442	9	0	0
Long term debt repayment (Reduction)	2223	10	1	4
Common stock in 2016	3120	11	0	0
Retained earnings in 2016	110,551	12	0	0
Retention rate	0.4788	13	1	4
Average tax rate (Income Taxes/Pretax Income)	0.18	14	1	4
Average interest rate in 2016	0.0332	15	0	0
Expected interest rate on new debt	0.0368	16	1	4
Operating income as a percentage of sales	0.2872	17	1	4
Underwriting cost of debt	0.02	18	1	4
Underwriting cost of equity	0.01	19	1	4
Ratio of long-term debt to equity	0.3187	20	1	4
Number of common shares outstanding	2737.3	21	0	0
Price-earnings ratio	19.075	22	1	4

Table 20.4 Pro forma balance sheet (2016–2020)

	2016	2017	2018	2019	2020
Assets					
Current assets	0.00	73,271.6	82,555.56	93,015.84	104,801.5
Fixed assets	0.00	85,826.43	96,701.16	108,953.8	122,758.9
Total assets	0.00	159,098	179,256.7	201,969.6	227,560.4
Liabilities and net worth					
Current liabilities	0.00	29,613.2	33,365.37	37,592.96	42,356.21
Long term debt	22,442.00	31,293.56	35,258.64	39,726.12	44,759.66
Preferred stock	0.00	0	0	0	0
Common stock	3120.00	−21,298.1	−18,972.8	−16,350.1	−13,392.6
Retained earnings	110,551.00	119,489.3	129,605.5	141,000.6	153,837.1
Total liabilities and net worth	0.00	159,098	179,256.7	201,969.6	227,560.4
Computed DBT/EQ	0.0000	0.3187	0.3187	0.3187	0.3187
Int. rate on total debt	0.0332	0.034474	0.034882	0.035205	0.035464
Per share data					
Earnings	0.0000	7.292306	8.205176	9.188508	10.29033
Dividends	0.0000	3.80075	4.276538	4.78905	5.363322
Price	0.0000	139.1007	156.5137	175.2708	196.2881

Table 20.5 Pro forma income statement (2016–2020)

	2016	2017	2018	2019	2020
Sales	71,890.00	80,998.90	91,261.94	102,825.38	115,853.98
Operating income	0.00	23,262.88	26,210.43	29,531.45	33,273.26
Interest expense	0.00	1078.81	1229.90	1398.57	1587.35
Underwriting commission—debt	0.00	221.49	123.76	133.81	145.13
Income before taxes	0.00	21,962.58	24,856.77	27,999.07	31,540.79
Taxes	0.00	3953.26	4474.22	5039.83	5677.34
Net income	0.00	18,009.31	20,382.55	22,959.24	25,863.44
Preferred dividends	0.00	0.00	0.00	0.00	0.00
Available for common dividends	0.00	18,009.31	20,382.55	22,959.24	25,863.44
Common dividends	0.00	9386.45	10,623.39	11,966.36	13,480.03
Debt repayments	0.00	2223.00	2223.00	2223.00	2223.00
Actl funds needed for investment	0.00	−13,028.02	8870.34	9715.43	10,667.10

8. $NF_t + b_t(1-T)\left[i\psi_1 \times NL_t + U_t^{L_t} \times NL_t\right] = NL_t + NS_t \quad -13275.64 + 0.4788 \times (1-0.18) \times (0.0332NL_t + 0.02 \times NL_t) = NL_t + NS_t$

$-13275.64 + 0.02089 \times NL_t = NL_t + NS_t$

(a) $NS_t + 0.97911NL_t = -24,337.4104$

9. $L_t = L_{t-1} - LR_t + NL_t$

(b) $L_t = 22,442 - 2223 + NL_t$

$L_t - NL_t = 20,219$

10. $S_t = S_{t-1} + NS_t$

(c) $-NS_t + S_t = 3,120.0$

11. $R_t = R_{t-1} + b_t\left\{(1-T_t)\left[EBIT_t - i_tL_t - U_t^L NL_t\right] - PFDIV_t\right\} = 110,551 + 0.4778 \times \{(1-0.18) \times [23,262.76 - i_tL_t - 0.02 \times NL_t] - 0\}$

Substitute (12) into (11)

$R_t = 110,551 + 0.4778 \times \{0.82[23,262.76 - (671.2708 + 0.0368 \times NL_t) - 0.02 \times NL_t]\} = 119,420.7796 - 0.0223NL_t$

(d) $119,420.7796 = R_t + 0.0223NL_t$

13. $L_t = (S_t + R_t)K_t$

$L_t = 0.3187S_t + 0.3187R_t$

(e) $L_t - 0.3187S_t - 0.3187R_t = 0$

(b) − (e) = (f)

$20,219 = 0.3187S_t + 0.3187R_t - NL_t$

(f) -0.3187(c) = (g)

$19,224.656 = 0.3187NS_t - NL_t + 0.3187R_t$

(g) -0.3187(d)

$-\mathrm{NL_t} + 0.3187\mathrm{NS_t} - 0.0071\mathrm{NL_t} = -18834.74646$

$-18834.74646 = 0.3187NS_t - 1.0071NL_t$

(h) -0.3187(a) = (i)

$-1.0071N_t - 0.3120NL_t = -14603.81$

$NL_t = -14603.81/1.31915 = 11070.62$

Substitute NL_t in (a)
NS_t = −24114.98745
Substitute NL_t in (b)
L_t = 31289.62094
Substitute NS_t in (c)
S_t = −20994.98745
Substitute NL_t in (d)
R_t = 119173.9047
Substitute NL_tL_t in (12)…
i_t = 0.03447
Section 4: Generation of Per Share Data for Period t

14. $EAFCD_t = (1-T_t)\left(EBIT_t - i_tL_t - U^L tNL_t\right) - PFDIV_t = (1-0.18) \times [23,262.75857 - 0.03447 \times 31289.62 - 0.02 \times (11070.62)] - 0 = 18009.49019$

15. $CMDIV_t = (1 - b_t)EAFCD_t$
$= (1 - 0.4788)(18009.49019)$
$= 9386.546287$

16. NUMCS $_t = X_1 = NUMCS_{t-1} + NEWCS_t$
$X_1 = 2737.3 + NEWCS_t$

17. $NEWCS_t = X_2 = NS_t/(1 - U^E)P_t$
$X_1 = 2737.3 +$ NEWCS $_t$

18. $P_t = X_3 = m_t EPS_t$
$X_3 = 19.075(EPS_t)$

19. $EPS_t = X_4 = EAFCD_t/NUMCS_t$
$X_4 = 18009.49019/$ NUMCS $_t$

20. $DPS_t = X_5 = CMDIV_t/$ NUMCS $_t$
$X_5 = 9386.546287/$ NUMCS $_t$

(A) = For (18) and (19), we obtain X_3 = 19.075 (18009.49019)/$NUMCS_t$ = 343,531.0254/X_1

Substitute (A) into Equation (17) to calculate (B)

$$(B) = X_2 = -24114.98745/[(1 - 0.01) \times 343,531.0254/X_1]$$
$$(B) = X_2 = -0.0709X_1$$

Substitute (B) into Equation (16) to calculate (C)

$$(C) = X_1 = 2,737.3 - 0.0709X_1$$
$$(C) = X_1 = 2556.058882 = \text{NUMCS}_t$$

Substitute (C) into (B)…

(B) = X_2 = −181.2411175=NEWCS$_t$

From Equation (19) and (20) we obtain X_4, X_5

$$X_4 = 7.04 = \text{EPS}_t$$
$$X_5 = 3.67 = \text{DPS}_t$$

From Equation (18) we obtain X_3

X_3 = 134.40 = P$_t$

Now we summarize the forecasted variables for 2017 as follows:

- Sales = $80,998.46
- Current Assets = $73,271.21
- Fixed Assets = $85,825.97
- Total Assets = $159,097.18
- Current Payables = $29,613.00
- Needed Funds = ($13,275.64)
- Earnings before Interest and Taxes = $23,262.76
- New Debt = $8393.78
- New Stock = ($24,114.99)
- Total Debt = $31,289.62094
- Common Stock = ($20,994.98745)
- Retained Earnings = $119,173.9047
- Interest Rate on Debt = 3.43%
- Earnings Available for Common Dividends = $18,009.49019
- Common Dividends = $9386.546287
- Number of Common Shares Outstanding = 2556.058882
- New Common Shares Issued = (181.2411175)
- Price per Share = $134.40
- Earnings per Share = $7.04
- Dividends per Share = $3.67

The above-forecasted variables are almost identical to the numbers for 2017 presented in Tables 20.4 and 20.5.

20.4 The Linear Programming Approach to Financial Planning and Analysis

In this section, we will discuss how linear programming techniques can be used to (i) solve profit maximization problems, (ii) to perform capital rationing problems, and (iii) to perform financial planning and forecasting.

An alternative approach to financial planning is based on using the optimization technique of linear programming. Using linear programming to do financial planning, the decision-maker sets up an objective function, such as to maximize firm value based on some financial theory. Hence, the model optimizes this objective function subject to certain constraints, such as maximum allowable debt/equity and payout ratios.

To use the linear programming approach for financial decisions, the problem must be formulated using the following three steps:

1. Identify the controllable decision variable of the problem.
2. Define the objective to be maximized or minimized, and define this function in terms of the controllable decision variables. In general, the objective is usually to maximize profit or minimize cost.
3. Define the constraints, either as linear equations or inequalities of the decision variables.

Several points need to be noted concerning the linear programming model. The variables representing the decision variables are divisible; that is, a workable solution would permit the variable to have a value of ½, ¾, etc. If such a fractional value is not realistic (that is, you cannot produce ½

of a product), then a related technique called *integer programming* can be used.[2]

In this section, we apply linear programming to profit maximization, capital rationing, and financial planning and forecasting.

20.4.1 Profit Maximization

XYZ, a toy manufacturer, produces three types of toys: King Kobra (KK), Pistol Pete (PP), and Rock Coolies (RC). To produce each toy, the plastic parts must be molded by machine and then assembled. The machine and assembly times for each type of toy are shown in Table 20.6.

Variable costs, selling prices, and profit contributions for each type of toy are presented in Table 20.7.

XYZ finances its operations through bank loans. The covenants of the loans require that XYZ maintain a current ratio of 1 or more; otherwise, the full amount of the loan must be immediately repaid. The balance sheet of XYZ is presented in Table 20.8.

For this case, the objective function is to maximize the profit contribution for each product. From Table 20.7, we see that the profit contribution for each product is KK = $1, PP = $4, and RC = $3. We can multiply this contribution per unit times the number of units sold to identify the firm's total operating income. Thus, the objective function is

$$MAXP = X_1 + 4X_2 + 3X_3 \qquad (20.1)$$

where X_1, X_2, X_3 are the number of units of KK, PP, and RC.

We can now identify the constraints of the linear programming problem. The firm's capacities for producing KK, PP, and RC depend on the number of hours of available machine time and assembly time. Using the information from Table 20.6, we can identify the following capacity constraints:

$$5X_1 + 4X_2 + 5X_3 \leq 150\, hours (machine\ time\ constraint) \qquad (20.2)$$

$$5X_1 + 3X_2 + 4X_3 \leq 100\, hours (assembly\ time\ constraints) \qquad (20.3)$$

There is also a constraint on the number of Pistol Petes (PP) and Rock Coolies (RC) that can be produced. The firm's marketing department has determined that 10 units of PPs and RCs are the maximum amount that can be sold; hence

[2] Both linear programming and integer programming are generally taught in the MBA or undergraduate operation-analysis course. See Hillier and Lieberman, *Introduction to Operation Research*, for discussion of these methods.

Table 20.6 Production information for XYZ toys

Toy	Machine time (h)	Assembly time (h)
KK	5	5
PP	4	3
RC	5	4
Total hours available	150	100

Table 20.7 Financial information for XYZ toys

Toy	Selling price ($/unit)	Variable cost ($/unit)	Profit contribution ($/unit)
KK	11	10	1
PP	8	4	4
RC	8	5	3

Table 20.8 Balance sheet of XYZ toys

Assets		Liabilities and equity	
Cash	$100	Bank loan	$130
Marketable securities	100	Long-term debt	300
Accounts receivable	50	Equity	70
Plant and equipment	250 $500		$500

$$X_2 + X_3 \leq 10 \qquad (marketing\ constraint) \qquad (20.4)$$

Finally, the bank covenant requiring a current ratio greater than 1 must be met. Thus,

$$\frac{cash + marketable\ securities + \frac{A}{R} - cost\ of\ production}{bankloan} \geq 1$$

$$\frac{100 + 100 + 50 - 10X_1 - 4X_2 - 5X_2}{130} \geq 1$$

$$10X_1 + 4X_2 + 5X_3 \leq 120 (current\ ratio\ constraint) \qquad (20.5)$$

Since the production of each toy must, at minimum, be 0, three nonnegative constraints complete the formulation of the problem:

$$X_1, X_2, X_3 \geq 0 \qquad (nonnegative\ constraint) \qquad (20.6)$$

Combining the objective functions and constraints yields

$$MAXX_1 + 4X_2 + 3X_3 \qquad (20.7)$$

subject to $5X_1 + 4X_2 + 5X_3 \leq 150$; $5X_1 + 3X_2 + 4X_3 \leq 100$; $X_2 + X_3 \leq 10$; $10X_1 + 4X_2 + 5X_3 \leq 120$; and $X_1 \geq 0$, $X_2 \geq 0$, $X_3 \geq 0$.

Using the simplex method to solve this linear programming problem, we derive the three simplex method tableaus in Table 20.9. Tableau 1 presents the information of

Table 20.9 Simplex method tableaus for solving Eq. 20.7

Tableau 1								
	Real variables			Slack variables				
	X_1	X_2	X_3	S_1	S_2	S_3	S_4	
S_1	5	4	5	1	0	0	0	150
S_2	5	3	4	0	1	0	0	100
S_3	0	1	1	0	0	1	0	10
S_4	10	4	5	0	0	0	1	120
Objective function coefficients								
Profit	1	4	3	0	0	0	0	0
Total profit: 0								
Tableau 2								
	Real variables			Slack variables				
	X_1	X_2	X_3	S_1	S_2	S_3	S_4	
S_1	5	0	1	1	0	−4	0	110
S_2	5	0	1	0	1	−3	0	70
X_2	0	1	1	0	0	1	0	10
S_4	10	0	1	0	0	−4	1	80
Objective function coefficients								
Profit	1	0	−1	0	0	−4	0	−40
Total profit: 40								
Tableau 3								
	Real variables			Slack variables				
	X_1	X_2	X_3	S_1	S_2	S_3	S_4	
S_1	0	0	.5	1	0	−2	.5	70
S_2	0	0	.5	0	1	−1	.5	30
X_2	0	1	1	0	0	1	0	10
X_1	1	0	.1	0	0	−0.4	.1	8
Objective function coefficients								
Profit	0	0	−1.1	0	0	−3.6	−.1	−48
Total profit: 48								

objective function and constraints as derived in Eq. 20.7. Since there are constraints for four resources, there are four slack variables: S_1, S_2, S_3, and S_4. The initial tableau implies that we produce neither KK, PP, or RC. Therefore, the total profit is 0, a result that is not optimal because all objective coefficients are positive. In the second tableau, the firm produces ten units of PP and generates a $40 profit. But this result also is not optimal because one of the objective function coefficients is positive. Tableau 3 presents the optimal situation because none of the objective function coefficients is positive. (Appendix 20.1 presents the method and procedure for specifying tableau 1 and solving tableaus 2, and 3 in terms of a capital rationing example.)

In tableau 3, the solution values for variables X_1 and X_2 are found in the right-hand column. Thus, $X_1 = 8$ units and $X_2 = 10$ units. Since X_3 doesn't appear in the final solution, it has a value of 0. The slack variables indicate the amount of XYZ's unused resources. For example, $S_1 = 70$ indicates that the firm has 70 h of unused machine time. To produce 8 units of X_1 requires 40 h, and to produce 10 units of X_2 requires 40 h, so our total usage of machine time is 80 h. This is 70 h less than the total hours of machine time the firm has available. $S_2 = 30$ indicates that there are additional assembly hours available. $S_3 = 0$ (it is not in the solution) implies that the constraint to make 10 units of $X_2 + X_3$ is satisfied. $S_4 = 0$ implies that the current ratio constraint is also satisfied and that financing, or, more precisely, the lack of financing, is limiting the amount of production. If the firm can change the bank loan covenant or increase the amount of available funds, it will be able to produce more. The maximum total profit contribution is $48 given the current production level.

20.4.2 Linear Programming and Capital Rationing

Linear programming is a mathematical technique that can be used to find the optimal solution to problems involving the allocation of scarce resources among competing activities. Mathematically, linear programming can best solve problems in which both the firm's objective is to be maximized and the constraints limiting the firm's actions are linear functions of the decision variables involved. Thus, the first step in using linear programming as a tool for financial decision-making is to model the problem facing the firm into a linear-programming form. To construct the programming model involves the following steps.

First, identify the controllable decision variables. Second, define the objective to be maximized or minimized and formulate that objective into a linear function with controllable decision variables. In finance, the objective generally is to maximize profit and market value or to minimize production costs. Third, the constraints must be defined and expressed as linear equations (equalities or inequalities) of the decision variables. This usually involves determining the capacities of the scarce resources involved in the constraints and then deriving a linear relationship between these capacities and the decision variables.

For example, suppose that X_1, X_2, …, X_N represents output quantities. Then the linear programming model takes the general form:

Maximize (or minimize)

$$Z = c_1X_1 + c_2X_2 + \ldots + c_NX_N$$

Subject to

$$\begin{aligned} a_{11}X_1 + a_{12}X_2 + \cdots + a_{1N}X_N &\le b_1 \\ a_{21}X_1 + a_{22}X_2 + \cdots + a_{2N}X_N &\le b_2 \\ \cdot \quad \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \quad \cdot \\ a_MX_1 + a_{N2}X_2 + \cdots + a_{MN}X_N &\le b_M \\ X_j \ge 0, (j = 1, 2, \ldots, N) & \end{aligned}$$

Z represents the objective to be maximized or minimized (that is, profit, market value, or (cost)); c_1, c_2, …, c_N and a_1, a_2, …, a_{MN} are constant coefficients relating to profit contribution and input, respectively; and b_1, b_2, …, b_N are the firm's capacities of the constraining resources. The last constraint ensures that the decision variables to be determined are positive or zero.

Several points should be noted concerning the linear programming model. First, depending on the problem, the constraints may also be stated with equal (=) signs or greater than ($\ge$) or less than ($\le$) signs. Second, the solution values of the decision variables are divisible, such that a solution would permit X(1) = ½, ¼, etc. If such fractional values are not possible, the related technique of integer programming (yielding only whole numbers as solutions) can be applied. Third, the constant coefficients are assumed known and deterministic (fixed). If the coefficients have probabilistic distributions, then one of the stochastic programming methods must be used.

As an example of the application of linear programming to the areas of capital rationing and capital budgeting, assume that a firm has a 12 percent cost of capital and $15 million in resources for investment opportunities. Management is considering four investment projects, with financial information as listed in Table 20.10.

Restating the problem in linear programming equations, the objective is to select the projects that yield the largest total net present value; that is, to invest in optimal amounts of alternative projects such that

$$NPV = 34.72X_A + 41.34X_B + 27.81X_C + 60.88X_D$$

is maximized, where X_A, X_B, X_C, and X_D represent amounts to be invested in project A, project B, project C, and project D.

The projects are subject to several constraints. For one, the total cash outflow in period 0 cannot be more than the $15 million ceiling. That is

$$15X_A + 7.5X_B + 7.5X_C + 0X_D < 15$$

Table 20.10 .

Cash flow ($ millions)			NPV at 12% ($ millions)		
Project	C_0	C_1	C_2	C_3	
A	−15	+45	+7.5	+5	+34.72
B	−7.5	+7.5	+35	+20	+41.34
C	−7.5	+7.5	+22.5	+15	+27.81
D	0	−60	+90	+60	+60.88

Another constraint is that not more than one project can be purchased or can a negative amount be purchased:

$$\begin{aligned} 0 \le X_A \le 1 \\ 0 \le X_B \le 1 \\ 0 \le X_C \le 1 \\ 0 \le X_D \le 1 \end{aligned}$$

Collecting all these equations together forms the linear program:

Maximize

$$34.72X_A + 41.34X_B + 27.81X_C + 60.88X_D \qquad (20.8)$$

Subject to

$$\begin{aligned} 15X_A + 7.5X_B + 7.5X_C + 0X_D \le 15 \\ 0 \le X_A \le 10 \le X_B \le 10 \le X_C \le 10 \le X_D \le 1 \end{aligned}$$

To obtain a solution, we can use either linear or integer (zero-one) programming. Integer programming is a linear program that limits X's to whole integers. This is especially important in this type of business decision because we might not accept a fraction of a project, which is what the constraint $0 \le X \le 1$ is likely to produce.

The best integer solution is to accept projects B and C ($X_B = 1$ and $X_C = 1$), which yields the maximum *NPV* of $69.15.[3]

20.4.3 Linear Programming Approach to Financial Planning

Carleton (1970) and Carleton, Dick, and Downes (CDD 1973) have formulated a financial planning model within a linear programming framework. Their objective function is based on the dividend stream model as expressed in Eq. 20.9:

[3] The best linear programming solution for this problem is to accept only project B ($X_B = 2$), which yields the maximum *NPV* of $82.68. The procedure of solving this problem can be found in the Appendix 21.1 of Chap. 21.

$$\frac{P_0}{N_0} = \sum_{t=0}^{T-1} \frac{D_t}{N_t(1+k)^t} + \frac{P_T}{N_T(1+k)^T} \tag{20.9}$$

where N_0 = total common shares in period 0; P_0 = total equity value in period 0; P_T = aggregate market value of the firm's equity at the end of period T; N_t = number of common shares outstanding at the beginning of period t; D_t = total dividends paid by the firm in period t; k = cost of equity capital, assuming constant risk and a constant k; and N_T = number of common shares outstanding in period T.

This objective function attempts to maximize the present value of the owners' equity, which includes all future dividends and long-term growth opportunities. (This model formulation is simply a rearranged version of the Gordon theory discussed in Chap. 5.)

Equation 20.9 is a nonlinear function in terms of N_t. To apply the linear programming method to this objective function, the function should be linearized. Following Lee (1985), a three-period linearized objective function for Eq. 20.9 can be defined as

$$\frac{D_0}{P_0} = \frac{D_0}{N_0} + \frac{D_1}{N_0(1+k)} - \frac{\Delta E_1^n}{N_0(1+k)(1-c)} + \frac{D_2}{N_0(1+k)^2} - \frac{\Delta E_2^n}{N_0(1+k)^2(1-c)} - \frac{\Delta E_3^n}{N_0(1+k)^3(1-c)} + \frac{P_3}{N_0(1+k)^3} \tag{20.10}$$

where D_0, P_0, N_0, and k are as defined in Eq. 20.9; ΔE_1^n, ΔE_2^n, ΔE_3^n represent the new equity issued in periods 1, 2, and 3; D_1 and D_2 represent dividend payments in periods 1 and 2; c is an estimate of the portion of equity lost to underpricing and transaction costs; and P_3 is the total market value of equity in the third period. To use this model, P_3 should be forecasted first. Since both D_0/N_0 and P_3 are predetermined, they can be omitted from the objective function without affecting the optimization results. If N_0 = 49.69, c = .10, and k = 16.5 percent, then the objective function without $D_0//N_0$ and P_3 can be written as

Table 20.11 Constraints involved in the linear programming model

Definition constraints
Available earnings for common equity holders
Sources and uses of funds
Policy constraints
Leverage-ratio related
Dividend-payment related

$$MAX.018D_1 - .020\Delta E_1^n + .015D_2 - .017\Delta E_2^n - .014\Delta E_3^n$$

Using this objective function and the constraints listed in Table 21.11, this model can be used to forecast important variables related to key pro forma financial statements. In Table 20.11, the constraint of available earnings for the common equity holders pertains to the amount of net income available to common equity holders. The constraint of sources and uses of funds involves the relationship among the investments, dividend payments, new equity issued, and new debt issued.

Policy constraints pertain to financing policy and dividend policy as described in Chaps. 3, 9, 12, and 13. Financing policy can be classified into interest coverage and maximum leverage limitation. The dividend-related constraints can be classified into prefinancing limitations to avoid accumulating excess cash, minimum dividend growth, and payout restrictions. (More detailed discussion of these constraints can be found in Lee (1985, Chap. 16).

The maximization of the Carleton or CDD objective function of the linear programming planning model is subject to legal, economic, and policy constraints. Thus, the LP approach blends financial theory with the idiosyncrasies of market imperfections and company preferences. The objective function and the constraints are inputs to the planning model. The rest of the input information for the CDD financial planning model includes base information and forecasts of the economic environment. *Base information* is simply the most recent fiscal-year results.

Figure 20.3 is a flowchart of Carleton's long-term financial planning model. This flowchart implies that the results of financial plans should be carefully evaluated before they are

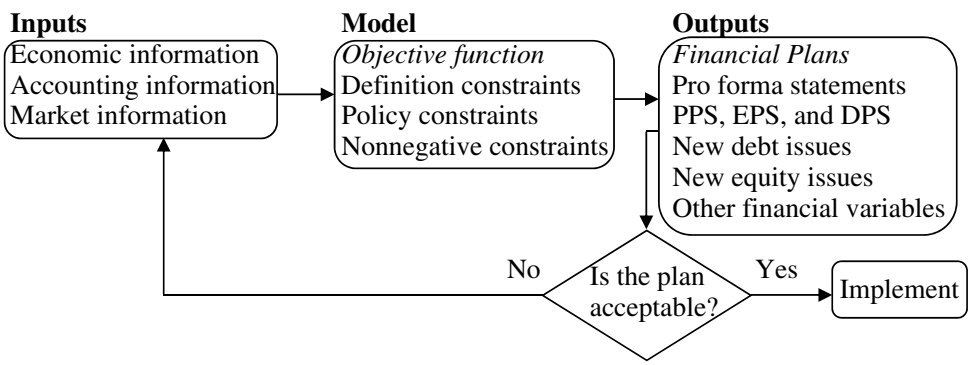


Fig. 20.3 Flowchart of Carleton's long-term planning and forecasting model

implemented. If the outputs are not satisfactory, both the inputs and the model should be reconsidered and modified.

Output from the LP model consists of the firm's major financial planning decisions (dividends, working capital, financing). The use of linear programming techniques allows these decisions to be determined simultaneously.

Carleton and CDD emphasize the importance of the degree of detail included in their model's forecasted balance sheets and income and funds-flow statements. That is, these statements are broken down into the minimum number of accounts consistent with making meaningful financial decisions: capital investment, working capital, capital structure, and dividends. Complicating the interpretations of the results with myriad details can diminish the effectiveness of any financial planning model.

In comparing the LP and simultaneous equations approaches to financial planning, the main difference between the two is that the linear programming method optimizes the plan based on classical finance theory while the simultaneous equations approach does not. However, in terms of ease of use, particularly for doing sensitivity analysis, the simultaneous equations model has the upper hand.

20.5 The Econometric Approach to Financial Planning and Analysis

The econometric approach to financial planning and analysis combines the simultaneous equations technique with regression analysis. The econometric approach models the firm in terms of a series of predictive regression equations and then proceeds to estimate the model parameters simultaneously, thereby taking account of the interactions among various policies and decisions.

To investigate the interrelationship between investment, financing, and dividend decisions, Spies (1974) developed five multiple regressions to describe the behavior of five alternative financial management decisions. Spies used a simultaneous equations technique to estimate all the equations at once.[4] He then used this model to demonstrate that investment, financing, and dividend policies generally are jointly determined within an individual industry.

Through the partial-adjustment model, the five endogenous variables (dividend payments, net short-term investment, gross long-term investment, new debt issued, and new equity issued), as defined in Table 20.12, are determined simultaneously through the use of the "uses equals sources" accounting identity. This identity ensures that the adjustment of each component of the budgeting process (the endogenous variables) depends not only on the component's distance from its target but also on the simultaneous adjustment of the other four decision variables.[5]

Table 20.12 Endogenous and exogenous variables

Endogenous variables
(a) $X_{1,t} = D/V_t$ = cash dividends paid in period t
(b) $X_{2,t} = IST_t$ = net investment in short-term assets during period t
(c) $X_{3,t} = ILT_t$ = gross investment in long-term assets during period t
(d) $X_{4,t} = -DF_t$ = minus the net proceeds from new debt issued during period t
(e) $X_{5,t} = -EQF_t$ = minus the net proceeds from new equity issued during period t
Exogenous variables
$Y_t = \sum_{i=1}^{5} X_{i,t} = \sum_{i=1}^{5} X^*_{i,t}$ where Y = net profits + depreciation allowance (a reformulation of the sources = uses identity)
(b) RCB = corporate bond rate
(c) RDP_t = average dividend-price ratio (or dividend yield)
(d) DEL_t = debt-equity ratio
(e) R_t = the rates of return the corporation could expect to earn on its future long-term investment (or internal rate of return)
(f) CU_t = rates of capacity utilization (used by Francis and Rowell (1978) to lag capital requirements behind changes in percent sales; used here to define the R_t expected)

Source Adapted from Spies (1974)

20.5.1 A Dynamic Adjustment of the Capital Budgeting Model

The capital budgeting decision affects the entire structure of the corporation. By its nature, the capital budgeting decision determines the firm's very essence and thus has been discussed at great length in both finance literature in general and in this book. In Chap. 13, we recognized that the components of the capital budget are determined jointly. The investment, dividend, and financing decisions are tied together by the "uses equals sources" identity, a simple accounting identity that requires all capital invested or distributed to stockholders to be accounted for.[6] However, despite the obviousness of this relationship, few attempts have been made to incorporate it into an econometric model. In this section, we describe Spies' (1974) econometric capital budgeting model, which explicitly recognizes the "uses equals sources" identity.

In his empirical work, Spies divided the capital budgeting decision into five basic components: dividends, net short-term investment, gross long-term investment, new debt financing, and new equity financing. The first three

[4] This technique takes into account the interaction relationship among investment, financing, and dividend policies (discussed in Chap. 13).

[5] It is assumed that there are targets for all five decision variables. In Table 21–11, $X^*_{1,t}$, $X^*_{2,t}$, $X^*_{3,t}$, $X^*_{4,t}$, $X^*_{5,t}$ represent the targets of $X_{1,t}$, $X_{2,t}$, $X_{3,t}$, $X_{4,t}$, $X_{5,t}$.

[6] This constraint also plays an important role in both Warren and Shelton's model and Carleton's model, as discussed previously.

components are uses of funds, while the latter two components are sources of funds. The dividends component includes all cash payments to stockholders and must be nonnegative. Net short-term investment is the net change in the corporation's holdings of short-term financial assets, such as cash, government securities, and accounts receivable. This component of the capital budget can be either positive or negative. Gross long-term investment is the change in gross long-term assets during the period. For example, the replacement of old equipment is considered a positive long-term investment. Long-term investment can be negative, but only if the sale of long-term assets exceeds replacement plus new investment.

As for sources of funds, the debt-financing component is simply the net change in the corporation's liabilities, such as corporate bonds, bank loans, taxes owed, and other accounts payable. Since a corporation can either increase its liabilities or retire existing liabilities, this variable can be either positive or negative. Finally, new equity financing is the change in stockholder equity minus the amount due to retained earnings. This should represent the capital raised by the sale of new shares of common stock. Although corporations frequently repurchase stock already sold, this variable is almost always positive when aggregated.

The first step is to develop a theoretical model that describes the optimal capital budget as a set of predetermined economic and financial variables. The first of these variables is a measure of cash flow: net profits plus depreciation allowances. This variable, denoted by Y, is exogenous as long as the policies determining production, pricing, advertising, taxes, and the like cannot be changed quickly enough to affect the current period's earnings. Since quarterly data are used in this work, this seems a reasonable assumption. It should also be noted that the "uses equals sources" identity ensures the following:

$$\sum_{i=1}^{5} X_{i,t} = \sum_{i=1}^{5} X^{*}_{i,t} = Y_t \tag{20.11}$$

where $X_{1,t}$, $X_{2,t}$, $X_{3,t}$, $X_{4,t}$, $X_{5,t}$, $X^{*}_{1,t}$, and Y_t are defined in Table 20.12.[7]

The second exogenous variable in the model is the corporate bond rate, RCD_t, which was used as a measure of the corporations' borrowing rate. In addition, the debt-equity ratio at the start of the period, DEL_t, was included to allow for the increase in the cost of financing due to leverage. The average dividend-price ratio for all stocks, RDP_t, was used as a measure of the rate of return demanded by investors in a no-growth, unlevered corporation for the average-risk class. The last two exogenous variables, R and CU_t, describe the rate of return the corporation could expect to earn on its future long-term investment. The ratio of the change in earnings to invest in the previous quarter should provide a rough measure of the rate of return on that investment. Spies used a four-quarter average of that ratio, R_t, to smooth out the normal fluctuations in earnings. The rate of capacity utilization, CU_t, was also included to improve this measure of the expected rate of return. Finally, a constant and three seasonal dummy variables were included. The exogenous variables are summarized in Table 20.12.

20.5.2 Simplified Spies Model

The simplified Spies model[8] for dividend payments ($X_{1,t}$), net short-term investments ($X_{2,t}$), gross long-term investments ($X_{3,t}$), new debt issues ($X_{4,t}$) and new equity issues ($X_{5,t}$) is defined as

$$X_{i,t} = a_{0i} + a_{1t}Y_t + a_{2i}RCB_t + a_{3i}RDP_t + a_{4i}DEL_t + a_{5i}R_t + a_{6i}CU_t + a_{7i}X_{i,t-1} \tag{20.12}$$

where $i = 1, 2, 3, 4, 5$, etc. Equation 20.12 implies that dividend payments, net short-term investments, gross long-term investments, new debt issues, and new equity issues all can be affected by new cash inflow (Y_t), the corporate bond rate (RCB_t), average dividend yield (RDP_t), debt-equity ratio (DEL_t), rates of return on long-term investment (R_t), rates of capacity utilization (CU_t), and $X_{i,t-1}$ (the last period's dividend payment, net short-term investment, etc.). These empirical models simultaneously take into account theory, information, and methodologies, and they can be used to forecast cash payments, net short-term investment, gross long-term investment, new debt issues, and new equity issues.

20.6 Sensitivity Analysis

So far, we have covered three types of financial planning models and discussed their strengths, weaknesses, and functional procedures. The efficiency of these models will depend solely on how they are employed. This section looks at alternative uses of financial planning models to improve their information dissemination. One of the most

[7] Expanding Eq. 21.11, we obtain. $X_{1,t} + X_{2,t} + X_{3,t} + X_{4,t} + X_{5,t} = X^{*}_{1,t} + X^{*}_{2,t} + X^{*}_{3,t} + X^{*}_{4,t} + X^{*}_{5,t} = Y_t$.

[8] The original Spies model and its application can be found in Lee and Lee (2017). In addition, Tagart (1977) has proposed an alternative econometric model for financial planning and analysis. Readers who are interested in this model, please see Lee and Lee (2017) Chapter 26 for further detail.

advantageous ways to use these financial planning models is to perform *sensitivity analysis*. The purpose of sensitivity analysis is to hold all but one or perhaps a couple of variables constant and then analyze the impact of their change on the predicted outcome.

As mentioned earlier, financial planning models are merely forecasting tools to help the financial manager analyze the interactions of important company decisions with uncertain economic elements. Since we can never be precisely sure what the future holds, sensitivity analysis stands out as a desirable manner of examining the impact of the unexpected as well as of the expected.

Of the three types of financial planning models presented in this chapter, the simultaneous equations approach, as embodied in Warren and Shelton's FINPLAN, offers the best method for performing sensitivity analysis. By changing the parameter values, we can compare new outputs of the financial statements with those such as in Tables 20.4 and 20.5. The difference between the new statement and the statements in Tables 20.4 and 20.5 reflects the impact of potential changes in such areas as *economic conditions* (reflected in the interest rate, tax rate, and sales growth estimates) and *company policy decisions* (reflected in the maximum and minimum limits specified for the maturity and amount of debt and in the dividend policy as reflected in the specified payout ratio).

To perform sensitivity analysis, we change growth in sales (variable 3), operating income as a percentage of sales (variable 17), the P/E ratio (variable 22), the expected interest rate on new debt (variable 16), and long-term debt-to-equity ratio (variable 20). The new parameters are listed in Table 20.13. Summary results of the alternative sensitivity analyses for EPS, DPS, and price per share (*PPS*) are listed in Table 20.14. The results indicate that changes in key financial decision variables will generally affect *EPS, DPS,* and *PPS.*

Table 20.13 Sensitivity analysis parameters

Model variable number	Parameter	Alternative values	Sensitivity analysis number
3	Growth in sales	.20 −.15	1 2
20	Long-term debt-to-equity ratio	.10 .5	3 4
17	Operating income as a percentage of sales	.20 .50	5 6
22	Price-to-earnings ratio	5 30	7 8
16	Expected interest rate on new debt	.005 .05	9 10

Table 20.14 Summary results of sensitivity analysis for EPS, DPS, and PPS (2017–2020)

Original analysis	2017	2018	2019	2020
EPS	6.73	7.18	7.63	8.10
DPS	3.51	3.74	3.97	4.22
PPS	128.29	136.96	145.45	154.48
Sensitivity analysis #1				
EPS	7.35	8.66	10.15	11.91
DPS	3.83	4.51	5.29	6.21
PPS	140.23	165.17	193.68	227.12
Sensitivity analysis #2				
EPS	5.89	5.40	4.94	4.52
DPS	3.07	2.81	2.58	2.36
PPS	112.38	103.00	94.25	86.23
Sensitivity analysis #3				
EPS	6.90	7.71	8.58	9.55
DPS	3.60	4.02	4.47	4.98
PPS	131.58	147.03	163.62	182.10
Sensitivity analysis #4				
EPS	7.07	8.05	9.03	10.14
DPS	3.68	4.20	4.71	5.29
PPS	134.82	153.56	172.34	193.42
Sensitivity analysis #5				
EPS	4.88	5.41	5.96	6.56
DPS	2.54	2.82	3.10	3.42
PPS	93.01	103.17	113.62	125.14
Sensitivity analysis #6				
EPS	12.34	14.04	15.93	18.08
DPS	6.43	7.32	8.30	9.42
PPS	235.39	267.81	303.88	344.82
Sensitivity analysis #7				
EPS	8.36	9.22	10.11	8.36
DPS	4.36	4.80	5.27	4.36
PPS	41.82	46.08	50.53	41.82
Sensitivity analysis #8				
EPS	6.88	7.75	8.69	9.74
DPS	3.58	4.04	4.53	5.08
PPS	206.26	232.42	260.65	292.32
Sensitivity analysis #9				
EPS	7.15	8.09	9.09	10.21
DPS	3.73	4.22	4.74	5.32
PPS	136.48	154.27	173.38	194.74
Sensitivity analysis #10				
EPS	7.00	7.85	8.76	9.79
DPS	3.65	4.09	4.57	5.10
PPS	133.53	149.73	167.15	186.65

EPS Earning per share; *DPS* Dividend per share; *PPS* price per share

20.7 Summary

This chapter has examined three types of financial planning models available to the financial manager for use in analyzing the interactions of company decisions: the algebraic simultaneous equations model, the linear programming model, and the econometric model. We also have discussed the benefits of sensitivity analysis for determining the impact on the company from changes (expected and unexpected) in economic conditions.

The student should understand the basic functioning of all three models, along with the underlying financial theory. Moreover, it is essential to understand that a financial planning model is an aid or tool to be used in the decision-making process and is not an end in and of itself.

The computer-based financial modeling discussed in this chapter can be performed on either a mainframe computer or a PC. An additional dimension is the development of electronic spreadsheets. These programs simulate the matrix or spreadsheet format used in accounting and financial statements. Their growing acceptance and popularity are due to the ease with which users can make changes in the spreadsheet. This flexibility greatly facilitates the use of these programs for sensitivity analysis.

Appendix 20.1: The Simplex Algorithm for Capital Rationing

The procedure of using the simplex method in capital rationing to solve Eq. 20.8 is as follows:

Step 1: Convert equality constraints into a system of equalities through the introduction of slack variables S_1 and S_2, as follows:

$$\begin{aligned} 15X_1 + 7.5X_2 + 7.5X_3 + S_1 &= 15 \\ -45X_1 - 7.5X_2 - 7.5X_3 + 60X_4 + S_2 &= 20 \end{aligned} \tag{20.13}$$

where $X_1 = X_A$; $X_2 = X_B$; $X_3 = X_C$; and $X_4 = X_D$ (each of these is a separate investment project)

Step 2: Construct a tableau or tableaus for representing the objective function and equality constraints. This has been done for four tableaus in Table 20.A1. In tableau 1, the figures in columns 2 through 6 are the coefficients of X_1, X_2, X_3, X_4, S_1, and S_2, as specified in the two equalities in Eq. 20.13. Below these figures are the objective function coefficients. Note that only S_1 and S_2 are listed in the first column of tableau 1. This indicates that S_1 and S_2 are basic variables in tableau 1 and that remaining variables X_1, X_2, X_3, and X_4 have been arbitrarily set equal to 0.

With X_1, X_2, X_3, and X_4 all equal to 0, the remaining variables assume the values in the last column of the tableau; that is, $S_1 = 15$ and $S_2 = 20$. The numbers in the last column represent the values of basic variables in a particular basic-feasible solution.

Step 3: Obtain a new feasible solution. The basic-feasible solution of tableau 1 indicates zero profits for the firm. Clearly, this basic-feasible solution can be bettered because it shows no profit, and profit should be expected from the adoption of any project.

The fact that X_4 has the largest incremental NPV indicates that the value of X_4 should be increased from its present level of 0. If we divide the column of figures under X_4 into the corresponding figures in the last column, we obtain quotients ∞ and 1/3. Since the smallest positive quotient is associated with S_2, then S_2 should be replaced by X_4 in tableau 2.

The figures in tableau 2 are computed by setting the value of S_1 to 0, S_2 to 1, and *NPV* to 0. The steps in the derivation are as follows: To eliminate the nonzero terms, we first divide the second row in tableau 1 by 60 and thus obtain the coefficients indicated in the second row of tableau 2. We then multiply this row by -60.88 and combine this result with the third row, as follows:

$$\begin{aligned} &[34.72 + (-60.88) \times (-.75)]X_1 \\ &\quad + [41.34 + (-60.88) \times (-.125)]X_2 \\ &\quad + [27.81 + (-60.88) \times (-.125)]X_3 \\ &\quad + [60.88 - (60.88) \times 1]X_4 + [0 + (-60.88)(0)]S_1 \\ &\quad + [0 + (-60.88)(.017)]S_2 \\ &\quad = (-60.88)\left(\frac{1}{3}\right)(20\boldsymbol{A} - 2) \end{aligned}$$

The objective function coefficients of Eq. 20A-2 are listed in the third row of tableau 2. Tableau 2 implies that the company will undertake 1/3 units of project 4 (X_4) and that the total NPV of X_4 is $20.2933. All coefficients associated with the objective function are positive, which implies that the NPV can be improved by replacing S_1 with either X_1, X_2, X_3, X_4.

Using the same procedure mentioned above, we can now obtain tableau 3. In tableau 3, the only positive objective function coefficient is X_2. Therefore, X_2 can replace either X_1 or X_4 to increase the NPV.

Table 20.15 Simplex method tableaus

Tableau 1							
	Real variables				Slack variables		
	X_1	X_2	X_3	X_4	S_1	S_2	
S_1	15	7.5	7.5	0	1	0	15
S_2	−45	−7.5	−7.5	60	0	1	20
Objective function coefficients							
NPV	34.72	41.34	27.81	60.88	0	0	
Total *NPV*: 0							
Tableau 2							
	Real variables				Slack variables		
	X_1	X_2	X_3	X_4	S_1	S_2	
S_1	15	7.5	7.5	0	1	0	15
X_4	−.75	−.125	−.125	1	0	.017	.333
Objective function coefficients							
NPV	80.38	48.95	35.42	0	0	−1.015	−20.2933
Total *NPV*: 20.2933							
Tableau 3							
	Real variables				Slack variables		
	X_1	X_2	X_3	X_4	S_1	S_2	
X_1	1	.5	.5	0	.067	0	1
X_4	0	.25	.25	1	.05	.017	1.083
Objective function coefficients							
NPV	0	8.76	−4.77	0	−5.359	−1.015	−100.673
Total *NPV*: 100.673							
Tableau 4							
	Real variables				Slack variables		
	X_1	X_2	X_3	X_4	S_1	S_2	
X_2	2	1	1	0	.133	0	2
X_4	−.5	0	0	1	.017	.017	.583
Objective function coefficients							
NPV	−17.52	0	−13.53	0	−6.527	−1.015	−118.193
Total *NPV*: 118.193							

Once again, using the procedure discussed above, we now obtain tableau 4. In tableau 4, none of the coefficients associated with the objective function are positive. Therefore, the solution in this tableau is optimal. Tableau 4 implies that the company will undertake 2 units of project 2 (X_2) and .583 units of project 4 (X_4) to maximize its total NPV.

From tableau 4, we obtain the best feasible solution:

$$X_1 = 0, X_2 = 2, X_3 = 0, \text{ and } X_4 = 0.583$$

Total NPV is now equal to (2)(41.34) + (60.88)(.583) = $118.193.

Although there are computer packages that can be used for linear programming, we can use the simplex method to hand-calculate the optimal number of projects and the maximum NPV in order to understand and appreciate the basic technique of derivation.

Appendix 20.2: Description of Parameter Inputs Used to Forecast Johnson & Johnson's Financial Statements and Share Price

In our financial planning plan program, there are 20 equations and 20 unknowns. To use this program, we need to input 21 parameters. These 20 unknowns and 21 parameters can be found in Table 20.2.

We use 2016 as the initial reference year and input the 21 parameters, the bulk of which can be obtained or derived

from the historical financial statements of JNJ. The first input is SALE t-1 ($71,890), defined as fiscal 2016 net sales and can be obtained from the income statement of JNJ. The second input is GCALSt-1. This parameter can be calculated by either the percentage change method: $\frac{Sales_{t-1}-Sales_{t-2}}{Sales_{t-2}} = 2.59\%$ or sustainability growth rate: $\frac{ROE_{t-1}*b_{t-1}}{1-ROE_{t-1}*b_{t-1}} = 12.7\%$

The third input is RCAt-1 (90.46%), defined as current assets divided by total sales, and the fourth input is RLA t-1 (1.0596), defined as total asset minus current asset divided by net sales. The next parameter is RCLt-1 (36.57%), defined as current liabilities as a percentage of net sales. The sixth parameter is preferred stock issued (PKV), with a value of 0, as JNJ does not currently have any preferred stock outstanding. The inputs for the aforementioned three parameters are all obtained from JNJ's fiscal 2016 balance sheet. The seventh input is JNJ's preferred stock dividends, and since there is no preferred stock outstanding, it is correspondingly 0. The eighth input is LR t-1 ($22,442), defined as long-term debt, coming from the balance sheet of JNJ for the fiscal year 2016, and the ninth input is LR t-1 ($-2,223), defined as long-term debt retirement, from the 2016 statement of cash flows.

The tenth input is St-1 ($3,120), which represents common stock issued, and the eleventh input is retained earnings (Rt-1 = $110,551). Both of these two variables can be found in the balance sheet for JNJ's fiscal year 2016. The twelfth input is the retention rate (bt-1 = 47.88%), defined as $1 - \frac{Dividendpayout_{t-1}}{Netincome_{t-1}}$. The thirteenth input, the average tax rate (Tt-1), is assumed to be 15%. The fourteenth input is the weighted average effective interest rate (It-1 = 3.33%), which JNJ provides in its annual report (page 53 of the respective 10-K filing). The fifteenth input is expected interest on new debt (iet-1 = 3.68%), calculated as the average of the weighted average interest rates in the previous two periods.

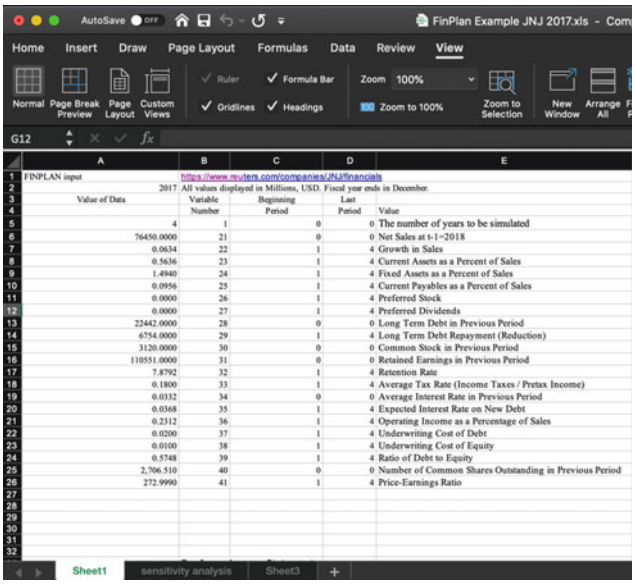
The next input is REBITt-1 (28.71%), defined as operating income as a percentage of sales. However, JNJ does not list explicitly list operating income in its income statements. Thus, we defined operating income as JNJ's earnings before provision for taxes on income, with interest expense added back and interest income subtracted out. We also adjusted for non-recurring expenses and added back other income/losses (related primarily to hedging activities, write-downs, and restructuring charges) to get to an adjusted and normalized operating income figure.

The seventeenth input is the underwriting cost of debt (UL) that we assume to be 2%, and the eighteenth parameter is the underwriting cost of equity (UE = 1%). The nineteen input is the ratio of long-term debt to equity (K t-1 = 31.87%), defined as long-term debt divided by total equity. The twentieth input is the number of common shares outstanding (NUMCSt-1 = 2,737.3) listed in the JNJ's Balance Sheet for the fiscal year 2016. The last input is the P/E ratio(mt-1 = 19.075) which is calculated as JNJ's closing share price on the last trading day of 2016 divided by fiscal 2016 net income.

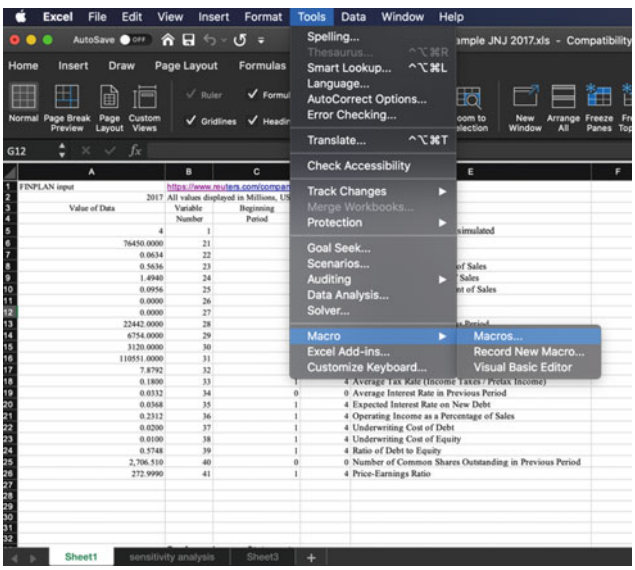
Appendix 20.3: Procedure of Using Excel to Implement the FinPlan Program

This appendix describes the detailed procedure of using Excel to implement the FinPlan program. There are four steps to use the FinPlan program.

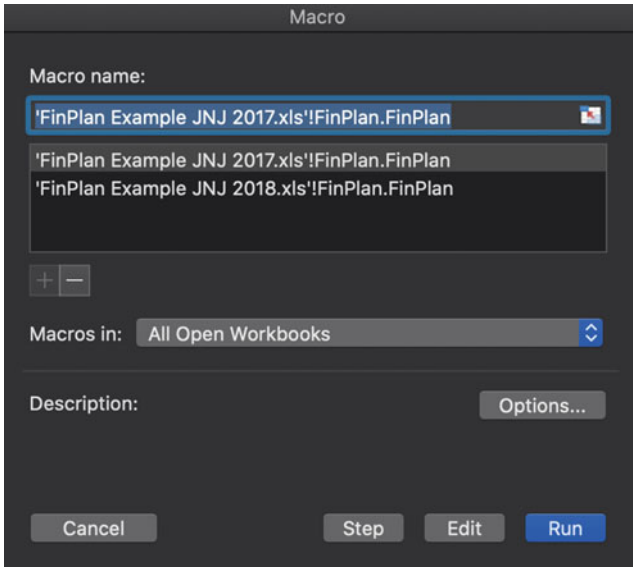
Step 1. Open the Excel file of FinPlan Example.



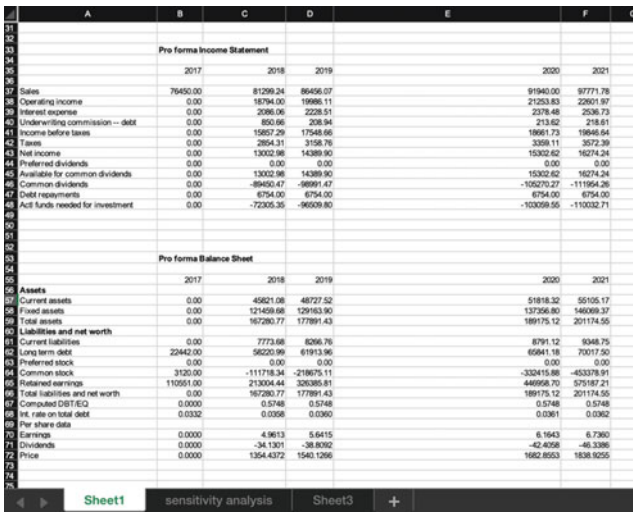
Step 2. Click the "Tools" and see "Macros".



Step 3. Choose "Macros" and then click "Run".



Step 4. Excel will show the solutions of the simultaneous equations.



After we obtain the forecasted values from the model, we compare them with the actual data of JNJ in 2018 via calculating the absolute percentage change of error. The following table shows the results.

	Forecast	Actual	Error
Sales	81,299.24	81,581.00	0.35%
Operating income	18,794.00	17,999.00	4.42%
Interest expense	2086.06	394.00	429.46%
Income before taxes	15,857.29	17,999.00	11.90%
Taxes	2854.31	2702.00	5.64%
Net income	13,002.98	15,297.00	15.00%
Preferred dividends	0	0	0.00%
Common dividends	−89,450.47	−9494.00	−842.18%
Debt repayments	6754.00	−3949.00	−271.03%
Assets			
Current assets	45,821.08	46,033.00	0.46%
Fixed assets	121,459.68	106,921.00	13.60%
Total assets	167,280.77	152,954.00	9.37%
Liabilities and net worth			
Current liabilities	7773.68	31,230.00	75.11%
Long term debt	58,220.99	27,684.00	110.31%
Preferred stock	0	0	0.00%
Common stock	−111,718.34	3120	3680.72%
Retained earnings	213,004.44	106,216.00	100.54%
Total liabilities and net worth	167,280.77	152,954.00	9.37%
Computed DBT/EQ	0.57	0.51	11.76%
Int. rate on total debt	0.04	0.03	33.33%
Per share data			
Earnings	4.96	6.92	28.32%
Dividends	−34.13	3.54	1064.12%
Price	1354.44	125.51	979.15%

Questions and Problems

1. According to Warren and Shelton (1971), what are the characteristics of a good financial planning model?
2. Briefly discuss the Warren and Shelton model of using a simultaneous equations approach to financial planning. How does this model compare with the Carleton model?
3. Discuss the basic concepts of simultaneous econometric models. How can accounting information be used in the econometric approach to do financial planning and forecasting?
4. Briefly discuss the use of econometric models to deal with dynamic capital budgeting decisions. How are these kinds of capital budgeting decisions useful to the financial manager?
5. Briefly compare programming models, simultaneous models, and econometric models. Which type of model seems better for use in financial planning?
6. Discuss and justify the WS model.
7. Discuss how linear programming can be used in financial planning and forecasting.

8. How can investment, financing, and dividend policies be integrated in terms of either linear programming or econometric financial planning and forecasting??
9. Using information in Tables 21.3, 21.12, use the FINPLAN program enclosed in the instructor's manual to solve empirical results as listed in Tables 21.4, 21.5, and Table 21.14
10. a. Identify the input variables in the Warren and Shelton model which require forecasted values and those which are obtained directly from current financial statements.

b. Discuss how the analyst can obtain values for the forecasted values.

c. Why is sensitivity analysis so important and beneficial in this model?

11. a. List and define the five basic components of the capital budgeting decision of the Spies model.

b. Identify which of the components are sources of funds and which are uses.

c. Identify the exogenous variables in this model.

12.a Please use the 21 inputs indicated in Table 20.16 to solve Warren and Shelton model presented in this chapter.

12.a Please interpret the results which you have obtained from 12a.

Solutions for 12a:

1. $SALES_t = 47348(1 + 0.0687) = 50600.81$
2. $EBIT_t = 50600.81(0.2754) = 13935.46$
3. $CA_t = 0.577(50600.81) = 29196.67$
4. $CA_t = 0.2204(50600.81) = 11152.42$
5. $A_t = 29196.67 + 11152.42 = 40349.08$
6. $CL_t = 0.2941(50600.81) = 14881.7$
7. $$NF_t = (40349.08 - 14881.7) - (2565 - 395) - 3120 - 35223 - 0.6179\{0.6628[13935.46 - 0.0729(2565 - 395)]\} = -20688.02$$
8. $-20688.02 + 0.6179\{0.6628[0.0729(NL_t) + 0.05(NL_t)]\} = NL_t + NS_t$ (a)

$$0.9497NL_t + NS_t = -20688.02$$

9. $L_t = 2565 - 395 + NL_t = 2170 + NL_t$ (b)

10. $S_t = 3120 + NS_t$ (c)

11. $R_t = 35223 + 0.6179\{0.6628[13935.46 - i_tL_t - 0.05NL_t]\}$

12. $i_tL_t = 0.0729(2565 - 395) + 0.0729NL_t = 158.193 + 0.0729NL_t$

Table 20.16 Inputs for Warren and Shelton model

Data	Variable	Description
47,348.0	$SALE_{t-1}$	The net sales (revenues) of the firm at the beginning
		of the simulation. t-1= 2004
0.0687	$GCALS_t$	Growth rate in sales during period t.
0.5770	RCA_{t-1}	Expected ratio of current assets (CA) to sales in t.
0.2204	RFA_{t-1}	Expected ratio of fixed assets (FA) to sales in t.
0.2941	RCL_{t-1}	Current Payables as a Percent of Sales
0.0	$PFDSK_{t-1}$	Preferred Stock
0.0	$PFDIV_{t-1}$	Preferred Dividends
2,565.0	L_{t-1}	Debt in Previous Period
395.0	LR_{t-1}	Debt Repayment
3,120.0	S_{t-1}	Common Stock in Previous Period
35,223.0	R_{t-1}	Retained Earnings in Previous Period
0.6179	b_{t-1}	Retention Rate
0.3372	T_{t-1}	Average Tax Rate
0.0729	i_{t-1}	Average Interest Rate in Previous Period
0.0729	i^e_{t-1}	Expected Interest Rate on New Debt
0.2754	$REBIT_{t-1}$	Operating Income as a Percentage of Sales
0.05	U^L	Underwriting Cost of Debt
0.05	U^E	Underwriting Cost of Equity
0.6464	K_t	Ratio of Debt to Equity
2,971.0	$NUMCS_{t-1}$	Number of Common Shares Outstanding in Previous Period
19.9	m_{t-1}	Price-Earnings Ratio

Substituting (12) into (11) yields

$$R_t = 40865.4 - 0.05NL_t \quad (d)$$

13. $L_t = (S_t + R_t)0.6464$ (e)

(b)–(e) yields

$$(S_t + R_t)0.6464 - NL_t = 2170 \quad (f)$$

(f)-0.6464(c) yields

$$0.6464R_t + 0.6464NS_t - NL_t = 153.232 \quad (g)$$

And (g)-0.6464(d) yields

$$0.6464NS_t - 1.0323NL_t = -26262.16 \quad (h)$$

Finally, (h)-0.6464(a) yields

$$NL_t = 7829.756$$

Substitute NL_t in (a) yields

$$NS_t = -28123.94$$

Substitute NL_t in (b) yields

$$L_t = 9999.756$$

Substitute NS_t in (c) yields

$$S_t = -25003.94$$

Substitute NL_t in (d) yields

$$R_t = 40473.91$$

Substitute NL_t in (12) yields

$$i_t L_t = 158.193 + 0.0729NL_t = 211.39$$

14. $EAFCD_t = 0.6628[13935.46 - 211.39 - 0.05(7829.756)] = 8836.84$

15. $CMDIV_t = 0.3821(8836.84) = 3376.56$

16. $NUMCS_t = 2971 + NEWCS_t$

17. $NEWCS_t = \frac{-28123.94}{(1-0.05)P_t} = \frac{-29604.15}{P_t}$

18. $P_t = 19.9(EPS_t)$

19. $EPS_t = EAFCD_t / NUMCS_t = 8836.84/ NUMCS_t$

From (18) and (19) we know that

$$P_t = 175853.12/NUMCS_t$$

Substitute P_t in (17) yields

$$NEWCS_t = \frac{-29604.15}{P_t} = -0.1684NUMCS_t$$

Substitute $NEWCS_t$ in (16) yields

$$NUMCS_t = 2971 - 0.1684NUMCS_t \Rightarrow NUMCS_t = 2542.79$$

Consequently we know that

$$NEWCS_t = -0.1684(NUMCS_t) = -428.21$$

And

EPS_t = 8836.84/ 2542.79=3.475

(20) DPS_t = $CMDIV_t$/ $NUMCS_t$= 3376.56/ 2542.79=1.328

Finally the price per share is equal to

$$P_t = 175853.12/NUMCS_t = 69.158$$

Solutions for 13b to be completed

Alternative Policies Analysis and Share Price Forecasting: XYZ Company as a Case Study

A. Introduction
   The main purpose of this paper is to use XYZ Company as a case study to analyze alternative policies. In Section B, we use the cash flow statement of XYZ Company to analyze alternative policies. In Section C discuss Warren and Shelton model in terms of four different sections, especially we will discuss the 20 unknowns and 21 parameters. In Section D, we calculate 21 input parameters. In Section E, we perform the calculation of this equation system in terms of both manual approach and Excel approach. For the manual approach we use data from 2017 to forecast 2018. For the Excel approach we will forecast 2018, 2019, and 2020. In Section F, we will perform sensitivity analysis by changing growth rate, debt equity ratio, and P/E ratio.
B. Investment, Financing, Dividend, and Production Policy for XYZ Company
   In this section students should use the information from the cash flow statement which contains information about all four policies. In addition, students should use these policies which have been learned in the class, which include Chaps. 7, 13, 14, 17, And 18 to do some meaningful analysis.

C. Warren and Shelton Model
   Warren and Shelton Model is a 20-equation model with 20 unknowns and 21 parameters to be input into the model. This model includes the following four sections:
   1. Generating of Sales and Earnings Before Interest and Taxes for Period t
   2. Generating of Total Assets Required for Period t
   3. Financing the Desired Level of Assets
   4. Generation of Per Share Data for Period t

D. Calculate 21 Input Parameters (Definitions of these variables can be found on page 1168 of the textbook)
   It should be noted that most of the parameters have already been calculated in the first project. In addition, for students to calculate these parameters, they should extensively search for information from the four financial statements.
E. Perform the calculation of 20 Unknown Variables
   1. Manual approach.
   2. Excel approach.
F. Sensitivity Analysis of Forecasting Stock Price Per Share and Important Financial Statement ItemsIn this section you should change growth rate, debt equity ratio, and P/E ratio.
G. Summary and Concluding Remarks

References

Carleton, W. T. "An Analytical Model for Long-range Planning," *Journal of Finance*, 25 (1970, pp. 291-315).

Carleton, W. T., C. L. Dick, Jr., and David H. Downes. "Financial Policy Models: Theory and Practice," *Journal of Financial and Quantitative Analysis*, 8 (1973, pp. 691–709).

Francis, J. C. and D. R. Rowell. "A Simultaneous Equation Model of the Firm for Financial Analysis and Planning," *Financial Management* 7 (Spring 1978, pp. 29–44).

Harrington, D. R. *Case Studies in Financial Decision-Making* (Chicago, IL: the Dryden Press, 1985).

Hillier, F. S. and G. J. Lieberman. *Introduction to Operation Research* (Oakland, CA: Holden-Day, 1986).

Lee, C. F. and J. Lee. *Financial Analysis and Planning: Theory and Application* 3rd Ed. (Singapore, World Scientific, 2017).

McLaughlin, H. S. and J. R. Boulding. *Financial Management with Lotus 1-2-3* (Englewood Cliffs, NJ: Prentice-Hall, 1986).

Myers, S. C. "Interaction of Corporate Financing and Investment Decisions," *Journal of Finance*, 29 (March 1974, pp. 1-25).

Myers, S. C. and G. A. Pogue. "A Programming Approach to Corporate Financial Management," *Journal of Finance*, 29 (May 1974, pp. 579-99).

Spies, R. "The Dynamics of Corporate Capital Budgeting," *Journal of Finance,* 29 (June 1974, pp. 829-45).

Stern, J. M. "The Dynamics of Financial Planning," *Analytical Methods in Financial Planning*, (1980, pp. 29–41).

Taggart, R. A., Jr. "A Model of Corporate Financing Decisions," *Journal of Finance*, 32 (December 1977, pp. 1467-84).

Warren, J. and J. Shelton. "A Simultaneous Equations Approach to Financial Planning." *Journal of Finance*, 26 (September 1971, pp. 1123-42).

Part V

Applications of R Programs for Financial Analysis and Derivatives

Check for updates

Hedge Ratio Estimation Methods and Their Applications

21

21.1 Introduction

One of the best uses of derivative securities such as futures contracts is in hedging. In the past, both academicians and practitioners have shown great interest in the issue of hedging with futures. This is quite evident from the large number of articles written in this area.

One of the main theoretical issues in hedging involves the determination of the optimal hedge ratio. However, the optimal hedge ratio depends on the particular objective function to be optimized. Many different objective functions are currently being used. For example, one of the most widely used hedging strategies is based on the minimization of the variance of the hedged portfolio (e.g., see Johnson 1960; Ederington 1979; Myers and Thompson 1989). This so-called minimum-variance (MV) hedge ratio is simple to understand and estimate. However, the MV hedge ratio completely ignores the expected return of the hedged portfolio. Therefore, this strategy is in general inconsistent with the mean–variance framework unless the individuals are infinitely risk-averse or the futures price follows a pure martingale process (i.e., expected futures price change is zero).

Other strategies that incorporate both the expected return and risk (variance) of the hedged portfolio have been recently proposed (e.g., see Howard and D'Antonio 1984; Cecchetti et al. 1988; Hsin et al. 1994). These strategies are consistent with the mean–variance framework. However, it can be shown that if the futures price follows a pure martingale process, then the optimal mean–variance hedge ratio will be the same as the MV hedge ratio.

Another aspect of the mean–variance based strategies is that even though they are an improvement over the MV strategy, for them to be consistent with the expected utility maximization principle, either the utility function needs to be quadratic or the returns should be jointly normal. If neither of these assumptions is valid, then the hedge ratio may not be optimal with respect to the expected utility maximization principle. Some researchers have solved this problem by deriving the optimal hedge ratio based on the maximization of the expected utility (e.g., see Cecchetti et al. 1988; Lence 1995, 1996). However, this approach requires the use of specific utility function and specific return distribution.

Attempts have been made to eliminate these specific assumptions regarding the utility function and return distributions. Some of them involve the minimization of the mean extended-Gini (MEG) coefficient, which is consistent with the concept of stochastic dominance (e.g., see Cheung et al. 1990; Kolb and Okunev 1992, 1993; Lien and Luo 1993a; Shalit 1995; Lien and Shaffer 1999). Shalit (1995) shows that if the prices are normally distributed, then the MEG-based hedge ratio will be the same as the MV hedge ratio.

Recently, hedge ratios based on the generalized semivariance (GSV) or lower partial moments have been proposed (e.g., see De Jong et al. 1997; Lien and Tse 1998, 2000; Chen et al. 2001). These hedge ratios are also consistent with the concept of stochastic dominance. Furthermore, these GSV-based hedge ratios have another attractive feature whereby they measure portfolio risk by the GSV, which is consistent with the risk perceived by managers, because of its emphasis on the returns below the target return (see Crum et al. 1981; Lien and Tse 2000). Lien and Tse (1998) show that if the futures and spot returns are jointly normally distributed and if the futures price follows a pure martingale process, then the minimum-GSV hedge ratio will be equal to the MV hedge ratio. Finally, Hung et al. (2006) has proposed a related hedge ratio that minimizes the Value-at-Risk associated with the hedged portfolio when choosing hedge ratio. This hedge ratio will also be equal to MV hedge ratio if the futures price follows a pure martingale process.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_21

Most of the studies mentioned above (except Lence 1995, 1996) ignore transaction costs as well as investments in other securities. Lence (1995, 1996) derives the optimal hedge ratio where transaction costs and investments in other securities are incorporated in the model. Using a CARA utility function, Lence finds that under certain circumstances the optimal hedge ratio is zero; i.e., the optimal hedging strategy is not to hedge at all.

In addition to the use of different objective functions in the derivation of the optimal hedge ratio, previous studies also differ in terms of the dynamic nature of the hedge ratio. For example, some studies assume that the hedge ratio is constant over time. Consequently, these static hedge ratios are estimated using unconditional probability distributions (e.g., see Ederington 1979; Howard and D'Antonio 1984; Benet 1992; Kolb and Okunev 1992, 1993; Ghosh 1993). On the other hand, several studies allow the hedge ratio to change over time. In some cases, these dynamic hedge ratios are estimated using conditional distributions associated with models such as ARCH (Autoregressive conditional heteroscedasticity) and GARCH (Generalized Autoregressive conditional heteroscedasticity) (e.g., see Cecchetti et al. 1988; Baillie and Myers 1991; Kroner and Sultan 1993; Sephton 1993a). The GARCH-based method has recently been extended by Lee and Yoder (2007) where regime-switching model is used. Alternatively, the hedge ratios can be made dynamic by considering a multi-period model where the hedge ratios are allowed to vary for different periods. This is the method used by Lien and Luo (1993b).

When it comes to estimating the hedge ratios, many different techniques are currently being employed, ranging from simple to complex ones. For example, some of them use such a simple method as the ordinary least squares (OLS) technique (e.g., see Ederington 1979; Malliaris and Urrutia 1991; and Benet 1992). However, others use more complex methods such as the conditional heteroscedastic (ARCH or GARCH) method (e.g., see Cecchetti et al. 1988; Baillie and Myers 1991; Sephton 1993a), the random coefficient method (e.g., see Grammatikos and Saunders 1983), the cointegration method (e.g., see Ghosh 1993; Lien and Luo 1993b; and Chou et al. 1996), or the cointegration-heteroscedastic method (e.g., see Kroner and Sultan 1993). Recently, Lien and Shrestha (2007) has suggested the use of wavelet analysis to match the data frequency with the hedging horizon. Finally, Lien and Shrestha (2010) also suggest the use of multivariate skew-normal distribution in estimating the minimum variance hedge ratio.

It is quite clear that there are several different ways of deriving and estimating hedge ratios. In the chapter, we review these different techniques and approaches and examine their relations.

The chapter is divided into six sections. In Sect. 21.2 alternative theories for deriving the optimal hedge ratios are discussed. Various estimation methods are presented in Sect. 21.3. Section 21.4 presents applications of OLS, GARCH, CECM models to estimate the optimal hedge ratio. Section 21.5 presents a discussion on the relationship among lengths of hedging horizon, maturity of futures contract, data frequency, and hedging effectiveness. Finally, in Sect. 21.6 we provide the summary and conclusion.

21.2 Alternative Theories for Deriving the Optimal Hedge Ratio

The basic concept of hedging is to combine investments in the spot market and futures market to form a portfolio that will eliminate (or reduce) fluctuations in its value. Specifically, consider a portfolio consisting of C_s units of a long spot position and C_f units of a short futures position.[1] Let S_t and F_t denote the spot and futures prices at time t, respectively. Since the futures contracts are used to reduce the fluctuations in spot positions, the resulting portfolio is known as the hedged portfolio. The return on the hedged portfolio, R_h, is given by:

$$R_h = \frac{C_s S_t R_s - C_f F_t R_f}{C_s S_t} = R_s - hR_f, \tag{21.1a}$$

where $h = \frac{C_f F_t}{C_s S_t}$ is the so-called hedge ratio, and $R_s = \frac{S_{t+1} - S_t}{S_t}$ and $R_f = \frac{F_{t+1} - F_t}{F_t}$ are so-called one-period returns on the spot and futures positions, respectively. Sometimes, the hedge ratio is discussed in terms of price changes (profits) instead of returns. In this case the profit on the hedged portfolio, ΔV_H, and the hedge ratio, H, are respectively given by:

$$\Delta V_H = C_s \Delta S_t - C_f \Delta F_t \quad \text{and} \quad H = \frac{C_f}{C_s}, \tag{21.1b}$$

where $\Delta S_t = S_{t+1} - S_t$ and $\Delta F_t = F_{t+1} - F_t$.

The main objective of hedging is to choose the optimal hedge ratio (either h or H). As mentioned above, the optimal hedge ratio will depend on a particular objective function to be optimized. Furthermore, the hedge ratio can be static or dynamic. In subsections A and B, we will discuss the static hedge ratio and then the dynamic hedge ratio.

It is important to note that in the above setup, the cash position is assumed to be fixed and we only look for the optimum futures position. Most of the hedging literature assumes that the cash position is fixed, a setup that is suitable for financial futures. However, when we are dealing

[1] Without loss of generality, we assume that the size of the future contract is 1.

with commodity futures, the initial cash position becomes an important decision variable that is tied to the production decision. One such setup considered by Lence (1995, 1996) will be discussed in subsection *C*.

21.2.1 Static Case

We consider here that the hedge ratio is static if it remains the same over time. The static hedge ratios reviewed in this chapter can be divided into eight categories, as shown in Table 21.1. We will discuss each of them in the chapter.

21.2.1.1 Minimum-Variance Hedge Ratio

The most widely-used static hedge ratio is the minimum-variance (MV) hedge ratio. Johnson (1960) derives this hedge ratio by minimizing the portfolio risk, where the risk is given by the variance of changes in the value of the hedged portfolio as follows:

$$Var(\Delta V_H) = C_s^2 Var(\Delta S) + C_f^2 Var(\Delta F) - 2C_s C_f Cov(\Delta S, \Delta F).$$

The MV hedge ratio, in this case, is given by:

$$H_J^* = \frac{C_f}{C_s} = \frac{Cov(\Delta S, \Delta F)}{Var(\Delta F)}. \tag{21.2a}$$

Alternatively, if we use definition (21.1a) and use $Var(R_h)$ to represent the portfolio risk, then the MV hedge ratio is obtained by minimizing $Var(R_h)$ which is given by:

$$Var(R_h) = Var(R_s) + h^2 Var(R_f) - 2hCov(R_s, R_f).$$

In this case, the MV hedge ratio is given by:

$$h_J^* = \frac{Cov(R_s, R_f)}{Var(R_f)} = \rho \frac{\sigma_s}{\sigma_f}, \tag{21.2b}$$

where ρ is the correlation coefficient between R_s and R_f, and σ_s and σ_f are standard deviations of R_s and R_f, respectively.

The attractive features of the MV hedge ratio are that it is easy to understand and simple to compute. However, in general the MV hedge ratio is not consistent with the mean–variance framework since it ignores the expected return on the hedged portfolio. For the MV hedge ratio to be consistent with the mean–variance framework, either the investors need to be infinitely risk-averse or the expected return on the futures contract needs to be zero.

21.2.1.2 Optimum Mean–Variance Hedge Ratio

Various studies have incorporated both risk and return in the derivation of the hedge ratio. For example, Hsin et al. (1994) derive the optimal hedge ratio that maximizes the following utility function:

$$\underset{C_f}{Max}\ V(E(R_h), \sigma; A) = E(R_h) - 0.5A\sigma_h^2, \tag{21.3}$$

Table 21.1 A list of different static hedge ratios

Hedge ratio	Objective function
Minimum-variance (MV) hedge ratio	Minimize variance of R_h
Optimum mean–variance hedge ratio	Maximize $E(R_h) - \frac{A}{2} Var(R_h)$
Sharpe hedge ratio	Maximize $\frac{E(R_h) - R_F}{\sqrt{Var(R_h)}}$
Maximum expected utility hedge ratio	Maximize $E[U(W_1)]$
Minimum mean extended-Gini (MEG) coefficient hedge ratio	Minimize $\Gamma_v(R_h v)$
Optimum mean-MEG hedge ratio	Maximize $E[R_h] - \Gamma_v(R_h v)$
Minimum generalized semivariance (GSV) hedge ratio	Minimize $V_{\delta,\alpha}(R_h)$
Maximum mean-GSV hedge ratio	Maximize $E[R_h] - V_{\delta,\alpha}(R_h)$
Minimum VaR hedge ratio over a given time period τ	Minimize $Z_\alpha \sigma_h \sqrt{\tau} - E[R_h]\tau$

Notes
1. R_h = return on the hedged portfolio
$E(R_h)$ = expected return on the hedged portfolio
$Var(R_h)$ = variance of return on the hedged portfolio
σ_h = standard deviation of return on the hedged portfolio
Z_α = negative of left percentile at α for the standard normal distribution
A = risk aversion parameter
R_F = return on the risk-free security
$E(U(W_1))$ = expected utility of end-of-period wealth
$\Gamma_v(R_h v)$ = mean extended-Gini coefficient of R_h
$V_{\delta,\alpha}(R_h)$ = generalized semivariance of R_h
2. With W_1 given by Eq. (21.17), the maximum expected utility hedge ratio includes the hedge ratio considered by Lence (1995, 1996)

where A represents the risk aversion parameter. It is clear that this utility function incorporates both risk and return. Therefore, the hedge ratio based on this utility function would be consistent with the mean–variance framework. The optimal number of futures contract and the optimal hedge ratio are respectively given by:

$$h_2 = -\frac{C_f^* F}{C_s S} = -\left[\frac{E(R_f)}{A\sigma_f^2} - \rho\frac{\sigma_s}{\sigma_f}\right]. \tag{21.4}$$

One problem associated with this type of hedge ratio is that in order to derive the optimum hedge ratio, we need to know the individual's risk aversion parameter. Furthermore, different individuals will choose different optimal hedge ratios, depending on the values of their risk aversion parameter.

Since the MV hedge ratio is easy to understand and simple to compute, it will be interesting and useful to know under what condition the above hedge ratio would be the same as the MV hedge ratio. It can be seen from Eqs. (21.2b) and (21.4) that if $A \to \infty$ or $E(R_f) = 0$, then h_2 would be equal to the MV hedge ratio h_J^*. The first condition is simply a restatement of the infinitely risk-averse individuals. However, the second condition does not impose any condition on the risk-averseness, and this is important. It implies that even if the individuals are not infinitely risk averse, then the MV hedge ratio would be the same as the optimal mean–variance hedge ratio if the expected return on the futures contract is zero (i.e. futures prices follow a simple martingale process). Therefore, if futures prices follow a simple martingale process, then we do not need to know the risk aversion parameter of the investor to find the optimal hedge ratio.

21.2.1.3 Sharpe Hedge Ratio

Another way of incorporating the portfolio return in the hedging strategy is to use the risk-return tradeoff (Sharpe measure) criteria. Howard and D'Antonio (1984) consider the optimal level of futures contracts by maximizing the ratio of the portfolio's excess return to its volatility:

$$\underset{C_f}{Max}\ \theta = \frac{E(R_h) - R_F}{\sigma_h}, \tag{21.5}$$

where $\sigma_h^2 = Var(R_h)$ and R_F represent the risk-free interest rate. In this case, the optimal number of futures positions, C_f^*, is given by:

$$C_f^* = -C_s \frac{\left(\frac{S}{F}\right)\left(\frac{\sigma_s}{\sigma_f}\right)\left[\frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)}{E(R_s)-R_F}\right) - \rho\right]}{\left[1 - \frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)\rho}{E(R_s)-R_F}\right)\right]}. \tag{21.6}$$

From the optimal futures position, we can obtain the following optimal hedge ratio:

$$h_3 = -\frac{\left(\frac{\sigma_s}{\sigma_f}\right)\left[\frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)}{E(R_s)-R_F}\right) - \rho\right]}{\left[1 - \frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)\rho}{E(R_s)-R_F}\right)\right]}. \tag{21.7}$$

Again, if $E(R_f) = 0$, then h_3 reduces to:

$$h_3 = \left(\frac{\sigma_s}{\sigma_f}\right)\rho, \tag{21.8}$$

which is the same as the MV hedge ratio h_J^*.

As pointed out by Chen et al. (2001), the Sharpe ratio is a highly non-linear function of the hedge ratio. Therefore, it is possible that Eq. (21.7), which is derived by equating the first derivative to zero, may lead to the hedge ratio that would minimize, instead of maximizing, the Sharpe ratio. This would be true if the second derivative of the Sharpe ratio with respect to the hedge ratio is positive instead of negative. Furthermore, it is possible that the optimal hedge ratio may be undefined as in the case encountered by Chen et al. (2001), where the Sharpe ratio monotonically increases with the hedge ratio.

21.2.1.4 Maximum Expected Utility Hedge Ratio

So far we have discussed the hedge ratios that incorporate only risk as well as the ones that incorporate both risk and return. The methods, which incorporate both the expected return and risk in the derivation of the optimal hedge ratio, are consistent with the mean–variance framework. However, these methods may not be consistent with the expected utility maximization principle unless either the utility function is quadratic or the returns are jointly normally distributed. Therefore, in order to make the hedge ratio consistent with the expected utility maximization principle, we need to derive the hedge ratio that maximizes the expected utility. However, in order to maximize the expected utility we need to assume a specific utility function. For example, Cecchetti et al. (1988) derive the hedge ratio that maximizes the expected utility where the utility function is assumed to be the logarithm of terminal wealth. Specifically, they derive the optimal hedge ratio that maximizes the following expected utility function:

$$\int_{R_s}\int_{R_f} \log\left[1 + R_s - hR_f\right] f(R_s, R_f)\, dR_s dR_f,$$

where the density function $f(R_s, R_f)$ is assumed to be bivariate normal. A third-order linear bivariate ARCH model is used to get the conditional variance and covariance matrix, and a numerical procedure is used to maximize the objective function with respect to the hedge ratio.[2]

21.2.1.5 Minimum Mean Extended-Gini Coefficient Hedge Ratio

This approach of deriving the optimal hedge ratio is consistent with the concept of stochastic dominance and involves the use of the mean extended-Gini (MEG) coefficient. Cheung et al. (1990), Kolb and Okunev (1992), Lien and Luo (1993a), Shalit (1995), and Lien and Shaffer (1999) all consider this approach. It minimizes the MEG coefficient $\Gamma_v(R_h)$ defined as follows:

$$\Gamma_v(R_h) = -vCov\Big(R_h, (1 - G(R_h))^{v-1}\Big), \tag{21.9}$$

where G is the cumulative probability distribution and v is the risk aversion parameter. Note that $0 \leq v < 1$ implies risk seekers, $v = 1$ implies risk-neutral investors, and $v > 1$ implies risk-averse investors. Shalit (1995) has shown that if the futures and spot returns are jointly normally distributed, then the minimum-MEG hedge ratio would be the same as the MV hedge ratio.

21.2.1.6 Optimum Mean-MEG Hedge Ratio

Instead of minimizing the MEG coefficient, Kolb and Okunev (1993) alternatively consider maximizing the utility function defined as follows:

$$U(R_h) = E(R_h) - \Gamma_v(R_h). \tag{21.10}$$

The hedge ratio based on the utility function defined by Eq. (21.10) is denoted as the M-MEG hedge ratio. The difference between the MEG and M-MEG hedge ratios is that the MEG hedge ratio ignores the expected return on the hedged portfolio. Again, if the futures price follows a martingale process (i.e., $E(R_f) = 0$), then the MEG hedge ratio would be the same as the M-MEG hedge ratio.

21.2.1.7 Minimum Generalized Semivariance Hedge Ratio

In recent years, a new approach for determining the hedge ratio has been suggested (see De Jong et al. 1997; Lien and Tse 1998, 2000; Chen et al. 2001). This new approach is based on the relationship between the generalized semivariance (GSV) and expected utility as discussed by Fishburn (1977) and Bawa (1978). In this case, the optimal hedge ratio is obtained by minimizing the GSV given below:

$$V_{\delta,\alpha}(R_h) = \int_{-\infty}^{\delta} (\delta - R_h)^{\alpha} dG(R_h), \quad \alpha > 0, \tag{21.11}$$

where $G(R_h)$ is the probability distribution function of the return on the hedged portfolio R_h. The parameters δ and α (which are both real numbers) represent the target return and risk aversion, respectively. The risk is defined in such a way that the investors consider only the returns below the target return (δ) to be risky. It can be shown (see Fishburn 1977) that $\alpha < 1$ represents a risk-seeking investor and $\alpha > 1$ represents a risk-averse investor.

The GSV, due to its emphasis on the returns below the target return, is consistent with the risk perceived by managers (see Crum et al. 1981; Lien and Tse 2000). Furthermore, as shown by Fishburn (1977) and Bawa (1978), the GSV is consistent with the concept of stochastic dominance. Lien and Tse (1998) show that the GSV hedge ratio, which is obtained by minimizing the GSV, would be the same as the MV hedge ratio if the futures and spot returns are jointly normally distributed and if the futures price follows a pure martingale process.

21.2.1.8 Optimum Mean-Generalized Semivariance Hedge Ratio

Chen et al. (2001) extend the GSV hedge ratio to a Mean-GSV (M-GSV) hedge ratio by incorporating the mean return in the derivation of the optimal hedge ratio. The M-GSV hedge ratio is obtained by maximizing the following mean-risk utility function, which is similar to the conventional mean–variance based utility function (see Eq. (21.3)):

$$U(R_h) = E[R_h] - V_{\delta,\alpha}(R_h). \tag{21.12}$$

This approach to the hedge ratio does not use the risk aversion parameter to multiply the GSV as done in conventional mean-risk models (see Hsin et al. 1994, and Eq. (21.3)). This is because the risk aversion parameter is already included in the definition of the GSV, $V_{\delta,\alpha}(R_h)$. As before, the M-GSV hedge ratio would be the same as the GSV hedge ratio if the futures price follows a pure martingale process.

21.2.1.9 Minimum Value-at-Risk Hedge Ratio

Hung et al. (2006) suggest a new hedge ratio that minimizes the Value-at-Risk of the hedged portfolio. Specifically, the hedge ratio h is derived by minimizing the following Value-at-Risk of the hedged portfolio over a given time period τ:

$$VaR(R_h) = Z_\alpha \sigma_h \sqrt{\tau} - E[R_h]\tau. \tag{21.13}$$

The resulting optimal hedge ratio, which Hung et al. (2006) refer to as zero-VaR hedge ratio, is given by

$$h^{VaR} = \rho \frac{\sigma_s}{\sigma_f} - E\big[R_f\big] \frac{\sigma_s}{\sigma_f} \sqrt{\frac{1 - \rho^2}{Z_\alpha^2 \sigma_f^2 - E\big[R_f\big]^2}} \tag{21.14}$$

It is clear that, if the futures price follows martingale process, the zero-VaR hedge ratio would be the same as the MV hedge ratio.

21.2.2 Dynamic Case

We have up to now examined the situations in which the hedge ratio is fixed at the optimum level and is not revised during the hedging period. However, it could be beneficial to change the hedge ratio over time. One way to allow the hedge ratio to change is by recalculating the hedge ratio based on the current (or conditional) information on the covariance $\left(\sigma_{sf}\right)$ and variance $\left(\sigma_f^2\right)$. This involves calculating the hedge ratio based on conditional information (i.e., $\sigma_{sf}|\Omega_{t-1}$ and $\sigma_f^2|\Omega_{t-1}$) instead of unconditional information. In this case, the MV hedge ratio is given by:

$$h_1|\Omega_{t-1} = -\frac{\sigma_{sf}|\Omega_{t-1}}{\sigma_f^2|\Omega_{t-1}}.$$

The adjustment to the hedge ratio based on new information can be implemented using such conditional models as ARCH and GARCH (to be discussed later) or using the moving window estimation method.

Another way of making the hedge ratio dynamic is by using the regime switching GARCH model (to be discussed later) as suggested by Lee and Yoder (2007). This model assumes two different regimes where each regime is associated with different set of parameters and the probabilities of regime switching must also be estimated when implementing such methods. Alternatively, we can allow the hedge ratio to change during the hedging period by considering multi-period models, which is the approach used by Lien and Luo (1993b).

Lien and Luo (1993b) consider hedging with T periods' planning horizon and minimize the variance of the wealth at the end of the planning horizon, W_T. Consider the situation where $C_{s,t}$ is the spot position at the beginning of period t and the corresponding futures position is given by $C_{f,t} = -b_t C_{s,t}$. The wealth at the end of the planning horizon, W_T, is then given by:

$$\begin{aligned} W_T &= W_0 + \sum_{t=0}^{T-1} C_{s,t}[S_{t+1} - S_t - b_t(F_{t+1} - F_t)] \\ &= W_0 + \sum_{t=0}^{T-1} C_{s,t}[\Delta S_{t+1} - b_t \Delta F_{t+1}]. \end{aligned} \tag{21.15}$$

The optimal b_t's are given by the following recursive formula:

$$b_t = \frac{Cov(\Delta S_{t+1}, \Delta F_{t+1})}{Var(\Delta F_{t+1})} + \sum_{i=t+1}^{T-1} \left(\frac{C_{s,i}}{C_{s,t}}\right) \frac{Cov(\Delta F_{t+1}, \Delta S_{i+1} + b_i \Delta F_{t+i})}{Var(\Delta F_{t+1})}. \tag{21.16}$$

It is clear from Eq. (21.16) that the optimal hedge ratio b_t will change over time. The multi-period hedge ratio will differ from the single-period hedge ratio due to the second term on the right-hand side of Eq. (21.16). However, it is interesting to note that the multi-period hedge ratio would be different from the single-period one if the changes in current futures prices are correlated with the changes in future futures prices or with the changes in future spot prices.

21.2.3 Case with Production and Alternative Investment Opportunities

All the models considered in subsections *A* and *B* assume that the spot position is fixed or predetermined, and thus production is ignored. As mentioned earlier, such an assumption may be appropriate for financial futures. However, when we consider commodity futures, production should be considered in which case the spot position becomes one of the decision variables. In an important chapter, Lence (1995) extends the model with a fixed or predetermined spot position to a model where production is included. In his model, Lence (1995) also incorporates the possibility of investing in a risk-free asset and other risky assets, borrowing, as well as transaction costs. We will briefly discuss the model considered by Lence (1995) below.

Lence (1995) considers a decision maker whose utility is a function of terminal wealth $U(W_1)$, such that $U' > 0$ and $U'' < 0$. At the decision date $(t = 0)$, the decision maker will engage in the production of Q commodity units for sale at terminal date $(t = 1)$ at the random cash price P_1. At the decision date, the decision maker can lend L dollars at the risk-free lending rate $(R_L - 1)$ and borrow B dollars at the borrowing rate $(R_B - 1)$, invest I dollars in a different activity that yields a random rate of return $(R_I - 1)$ and sell X futures at futures price F_0. The transaction cost for the futures trade is f dollars per unit of the commodity traded to be paid at the terminal date. The terminal wealth (W_1) is, therefore, given by:

$$\begin{aligned} W_1 &= W_0 R \\ &= P_1 Q + (F_0 - F_1)X - f|X| - R_B B + R_L L + R_I I, \end{aligned} \tag{21.17}$$

where R is the return on the diversified portfolio. The decision maker will maximize the expected utility subject to the following restrictions:

$$\begin{gathered} W_0 + B \geq v(Q)Q + L + I, 0 \leq B \leq k_B v(Q)Q, \ k_B \geq 0, \\ L \geq k_L F_0|X|, \ k_L \geq 0, I \geq 0, \end{gathered}$$

where $v(Q)$ is the average cost function, k_B is the maximum amount (expressed as a proportion of his initial wealth) that the agent can borrow, and k_L is the safety margin for the futures contract.

Using this framework, Lence (1995) introduces two opportunity costs: opportunity cost of alternative

(sub-optimal) investment (c_{alt}) and opportunity cost of estimation risk (e^{Bayes}).[3] Let R_{opt} be the return of the expected-utility maximizing strategy and let R_{alt} be the return on a particular alternative (sub-optimal) investment strategy. The opportunity cost of alternative investment strategy c_{alt} is then given by:

$$E\left[U\left(W_0 R_{opt}\right)\right] = E[U(W_0 R_{alt} + c_{alt})]. \tag{21.18}$$

In other words, c_{alt} is the minimum certain net return required by the agent to invest in the alternative (sub-optimal hedging) strategy rather than in the optimum strategy. Using the CARA utility function and some simulation results, Lence (1995) finds that the expected-utility maximizing hedge ratios are substantially different from the minimum-variance hedge ratios. He also shows that under certain conditions, the optimal hedge ratio is zero; i.e., the optimal strategy is not to hedge at all.

Similarly, the opportunity cost of the estimation risk (e^{Bayes}) is defined as follows:

$$\begin{aligned} &E_\rho\left[E\left(U\left\{W_0\left[R_{opt}(\rho) - e_\rho^{Bayes}\right]\right\}\right)\right] \\ &\quad = E_\rho\left[E\left(U\left(W_0 R_{opt}^{Bayes}\right)\right)\right], \end{aligned} \tag{21.19}$$

where $R_{opt}(\rho)$ is the expected-utility maximizing return where the agent knows with certainty the value of the correlation between the futures and spot prices (ρ), R_{opt}^{Bayes} is the expected-utility maximizing return where the agent only knows the distribution of the correlation ρ, and $E_\rho[.]$ is the expectation with respect to ρ. Using simulation results, Lence (1995) finds that the opportunity cost of the estimation risk is negligible and thus the value of the use of sophisticated estimation methods is negligible.

21.3 Alternative Methods for Estimating the Optimal Hedge Ratio

In Sect. 21.2, we discussed different approaches to deriving the optimum hedge ratios. However, in order to apply these optimum hedge ratios in practice, we need to estimate these hedge ratios. There are various ways of estimating them. In this section we briefly discuss these estimation methods.

21.3.1 Estimation of the Minimum-Variance (MV) Hedge Ratio

21.3.1.1 OLS Method

The conventional approach to estimating the MV hedge ratio involves the regression of the changes in spot prices on the changes in futures price using the OLS technique (e.g., see Junkus and Lee 1985). Specifically, the regression equation can be written as:

$$\Delta S_t = a_0 + a_1 \Delta F_t + e_t, \tag{21.20}$$

where the estimate of the MV hedge ratio, H_j, is given by a_1. The OLS technique is quite robust and simple to use. However, for the OLS technique to be valid and efficient, assumptions associated with the OLS regression must be satisfied. One case where the assumptions are not completely satisfied is that the error term in the regression is heteroscedastic. This situation will be discussed later.

Another problem with the OLS method, as pointed out by Myers and Thompson (1989), is the fact that it uses unconditional sample moments instead of conditional sample moments, which use currently available information. They suggest the use of the conditional covariance and conditional variance in Eq. (21.2a). In this case, the conditional version of the optimal hedge ratio (Eq. (21.2a)) will take the following form:

$$H_J^* = \frac{C_f}{C_s} = \frac{Cov(\Delta S, \Delta F)|\Omega_{t-1}}{Var(\Delta F)|\Omega_{t-1}}. \tag{21.2a*}$$

Suppose that the current information (Ω_{t-1}) includes a vector of variables (X_{t-1}) and the spot and futures price changes are generated by the following equilibrium model:

$$\Delta S_t = X_{t-1}\alpha + u_t,$$

$$\Delta F_t = X_{t-1}\beta + v_t.$$

In this case the maximum likelihood estimator of the MV hedge ratio is given by (see Myers and Thompson 1989):

$$\hat{h}|X_{t-1} = \frac{\hat{\sigma}_{uv}}{\hat{\sigma}_v^2}, \tag{21.21}$$

where $\hat{\sigma}_{uv}$ is the sample covariance between the residuals u_t and v_t, and $\hat{\sigma}_v^2$ is the sample variance of the residual v_t. In general, the OLS estimator obtained from Eq. (21.20) would be different from the one given by Eq. (21.21). For the two estimators to be the same, the spot and futures prices must be generated by the following model:

$$\Delta S_t = \alpha_0 + u_t, \quad \Delta F_t = \beta_0 + v_t.$$

In other words, if the spot and futures prices follow a random walk, then with or without drift, the two estimators will be the same. Otherwise, the hedge ratio estimated from the OLS regression (21.18) will not be optimal. Now we show how SAS can be used to estimate the hedge ratio in terms of OLS method.

21.3.1.2 Multivariate Skew-Normal Distribution Method

An alternative way of estimating the MV hedge ratio involves the assumption that the spot price and futures price follow a multivariate skew-normal distribution as suggested by Lien and Shrestha (2010). The estimate of covariance matrix under skew-normal distribution can be different from the estimate of covariance matrix under the usual normal distribution resulting in different estimates of the MV hedge ratio. Let Y be a k-dimensional random vector. Then Y is said to have skew-normal distribution if its probability density function is given as follows:

$$f_Y(y) = 2\phi_k(y, \Omega_Y)\Phi(\alpha^t y)$$

where α is a k-dimensional column vector, $\phi_k(y, \Omega_Y)$ is the probability density function of a k-dimensional standard normal random variable with zero mean and correlation matrix Ω_Y and $\Phi(\alpha^t y)$ is the probability distribution function of a one-dimensional standard normal random variable evaluated at $\alpha^t y$.

21.3.1.3 ARCH and GARCH Methods

Ever since the development of ARCH and GARCH models, the OLS method of estimating the hedge ratio has been generalized to take into account the heteroscedastic nature of the error term in Eq. (21.20). In this case, rather than using the unconditional sample variance and covariance, the conditional variance and covariance from the GARCH model are used in the estimation of the hedge ratio. As mentioned above, such a technique allows an update of the hedge ratio over the hedging period.

Consider the following bivariate GARCH model (see Cecchetti et al. 1988; Baillie and Myers 1991):

$$\begin{bmatrix} \Delta S_t \\ \Delta F_t \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \begin{bmatrix} e_{1t} \\ e_{2t} \end{bmatrix} \quad \Leftrightarrow \quad \Delta Y_t = \mu + e_t,$$

$$e_t|\Omega_{t-1}\ N(0, H_t), \quad H_t = \begin{bmatrix} H_{11,t} & H_{12,t} \\ H_{12,t} & H_{22,t} \end{bmatrix},$$

$$vec(H_t) = C + A\ vec\left(e_{t-1}e'_{t-1}\right) + B\ vec(H_{t-1}). \quad (21.22)$$

The conditional MV hedge ratio at time t is given by $h_{t-1} = H_{12,t}/H_{22,t}$. This model allows the hedge ratio to change over time, resulting in a series of hedge ratios instead of a single hedge ratio for the entire hedging horizon. Equation (21.22) represents a GARCH model. This GARCH model will reduce to ARCH if B is equal to zero.

The model can be extended to include more than one type of cash and futures contracts (see Sephton 1993a). For example, consider a portfolio that consists of spot wheat $(S_{1,t})$, spot canola (S_{2t}), wheat futures (F_{1t}), and canola futures (F_{2t}). We then have the following multi-variate GARCH model:

$$\begin{bmatrix} \Delta S_{1t} \\ \Delta S_{2t} \\ \Delta F_{1t} \\ \Delta F_{2t} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix} + \begin{bmatrix} e_{1t} \\ e_{2t} \\ e_{3t} \\ e_{4t} \end{bmatrix} \quad \Leftrightarrow \quad \Delta Y_t = \mu + e_t,$$

$$e_t|\Omega_{t-1}\ N(0, H_t).$$

The MV hedge ratio can be estimated using a similar technique as described above. For example, the conditional MV hedge ratio is given by the conditional covariance between the spot and futures price changes divided by the conditional variance of the futures price change. Now we show how SAS can be used to estimate ratio in terms of ARCH and GARCH models.

21.3.1.4 Regime-Switching GARCH Model

The GARCH model discussed above can be further extended by allowing regime switching as suggested by Lee and Yoder (2007). Under this model, the data generating process can be in one of the two states or regime denoted by the state variable $s_t = \{1, 2\}$, which is assumed to follow a first-order Markov process. The state transition probabilities are assumed to follow a logistic distribution where the transition probabilities are given by

$$\Pr(s_t = 1|s_{t-1} = 1) = \frac{e^{p_0}}{1 + e^{p_0}} \quad \&$$
$$\Pr(s_t = 2|s_{t-1} = 2) = \frac{e^{q_0}}{1 + e^{q_0}}.$$

The conditional covariance matrix is given by

$$H_{t,s_t} = \begin{bmatrix} h_{1,t,s_t} & 0 \\ 0 & h_{2,t,s_t} \end{bmatrix} \begin{bmatrix} 1 & \rho_{t,s_t} \\ \rho_{t,s_t} & 1 \end{bmatrix} \begin{bmatrix} h_{1,t,s_t} & 0 \\ 0 & h_{2,t,s_t} \end{bmatrix}$$

where

$$h^2_{1,t,s_t} = \gamma_{1,s_t} + \alpha_{1,s_t} e^2_{1.t-1} + \beta_{1,s_t} h^2_{1,t-1}$$
$$h^2_{2,t,s_t} = \gamma_{2,s_t} + \alpha_{2,s_t} e^2_{2.t-1} + \beta_{2,s_t} h^2_{2,t-1}$$
$$\rho_{t,s_t} = \left(1 - \theta_{1,s_t} - \theta_{2,s_t}\right)\rho + \theta_{1,s_t}\rho_{t-1} + \theta_{2,s_t}\phi_{t-1}$$
$$\phi_{t-1} = \frac{\sum_{j=1}^{2} \varepsilon_{1,t-j}\varepsilon_{2,t-j}}{\sqrt{\left(\sum_{j=1}^{2} \varepsilon^2_{1,t-j}\right)\left(\sum_{j=1}^{2} \varepsilon^2_{2,t-j}\right)}},$$
$$\varepsilon_{i,t} = \frac{e_{i,t}}{h_{it}}, \quad \theta_1, \theta_2 \geq 0 \quad \& \quad \theta_1 + \theta_2 \leq 1$$

Once the conditional covariance matrix is estimated, the time varying conditional MV hedge ratio is given by the ratio of the covariance between the spot and futures returns to the variance of the futures return.

21.3.1.5 Random Coefficient Method

There is another way to deal with heteroscedasticity. This involves use of the random coefficient model as suggested by Grammatikos and Saunders (1983). This model employs the following variation of Eq. (21.20):

$$\Delta S_t = \beta_0 + \beta_t \Delta F_t + e_t, \tag{21.23}$$

where the hedge ratio $\beta_t = \overline{\beta} + v_t$ is assumed to be random. This random coefficient model can, in some cases, improve the effectiveness of the hedging strategy. However, this technique does not allow for the update of the hedge ratio over time even though the correction for the randomness can be made in the estimation of the hedge ratio.

21.3.1.6 Cointegration and Error Correction Method

The techniques described so far do not take into consideration the possibility that spot price and futures price series could be non-stationary. If these series have unit roots, then this will raise a different issue. If the two series are cointegrated as defined by Engle and Granger (1987), then the regression Eq. (21.20) will be mis-specified and an error-correction term must be included in the equation. Since the arbitrage condition ties the spot and futures prices, they cannot drift far apart in the long run. Therefore, if both series follow a random walk, then we expect the two series to be cointegrated in which case we need to estimate the error correction model. This calls for the use of the cointegration analysis.

The cointegration analysis involves two steps. First, each series must be tested for a unit root (e.g., see Dickey and Fuller 1981; Phillips and Perron 1988). Second, if both series are found to have a single unit root, then the cointegration test must be performed (e.g., see Engle and Granger 1987; Johansen and Juselius 1990; and Osterwald-Lenum 1992).

If the spot price and futures price series are found to be cointegrated, then the hedge ratio can be estimated in two steps (see Ghosh 1993; Chou et al. 1996). The first step involves the estimation of the following cointegrating regression:

$$S_t = a + bF_t + u_t. \tag{21.24}$$

The second step involves the estimation of the following error correction model:

$$\Delta S_t = \rho u_{t-1} + \beta \Delta F_t + \sum_{i=1}^{m} \delta_i \Delta F_{t-i} + \sum_{j=1}^{n} \theta_i \Delta S_{t-j} + e_j, \tag{21.25}$$

where u_t is the residual series from the cointegrating regression. The estimate of the hedge ratio is given by the estimate of β. Some researchers (e.g., see Lien and Luo 1993b) assume that the long-run cointegrating relationship is $(S_t - F_t)$, and estimate the following error correction model:

$$\Delta S_t = \rho(S_{t-1} - F_{t-1}) + \beta \Delta F_t + \sum_{i=1}^{m} \delta_i \Delta F_{t-i} + \sum_{j=1}^{n} \theta_i \Delta S_{t-j} + e_j. \tag{21.26}$$

Alternatively, Chou et al. (1996) suggest the estimation of the error correction model as follows:

$$\Delta S_t = \alpha \hat{u}_{t-1} + \beta \Delta F_t + \sum_{i=1}^{m} \delta_i \Delta F_{t-i} + \sum_{j=1}^{n} \theta_i \Delta S_{t-j} + e_j, \tag{21.27}$$

where $\hat{u}_{t-1} = S_{t-1} - (a + bF_{t-1})$; i.e., the series $\hat{u}_t$ is the estimated residual series from Eq. (21.24). The hedge ratio is given by β in Eq. (21.26).

Kroner and Sultan (1993) combine the error-correction model with the GARCH model considered by Cecchetti et al. (1988) and Baillie and Myers (1991) in order to estimate the optimum hedge ratio. Specifically, they use the following model:

$$\begin{bmatrix} \Delta \log_e(S_t) \\ \Delta \log_e(F_t) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \begin{bmatrix} \alpha_s(\log_e(S_{t-1}) - \log_e(F_{t-1})) \\ \alpha_f(\log_e(S_{t-1}) - \log_e(F_{t-1})) \end{bmatrix} + \begin{bmatrix} e_{1t} \\ e_{2t} \end{bmatrix}, \tag{21.28}$$

where the error processes follow a GARCH process. As before, the hedge ratio at time $(t-1)$ is given by $h_{t-1} = H_{12,t}/H_{22,t}$.

21.3.2 Estimation of the Optimum Mean–Variance and Sharpe Hedge Ratios

The optimum mean–variance and Sharpe hedge ratios are given by Eqs. (21.4) and (21.7), respectively. These hedge ratios can be estimated simply by replacing the theoretical moments by their sample moments. For example, the expected returns can be replaced by sample average returns, the standard deviations can be replaced by the sample standard deviations, and the correlation can be replaced by sample correlation.

21.3.3 Estimation of the Maximum Expected Utility Hedge Ratio

The maximum expected utility hedge ratio involves the maximization of the expected utility. This requires the estimation of distributions of the changes in spot and futures prices. Once the distributions are estimated, one needs to use a numerical technique to get the optimum hedge ratio. One such method is described in Cecchetti et al. (1988) where an ARCH model is used to estimate the required distributions.

21.3.4 Estimation of Mean Extended-Gini (MEG) Coefficient Based Hedge Ratios

The MEG hedge ratio involves the minimization of the following MEG coefficient:

$$\Gamma_v(R_h) = -vCov\Big(R_h, (1 - G(R_h))^{v-1}\Big).$$

In order to estimate the MEG coefficient, we need to estimate the cumulative probability density function $G(R_h)$. The cumulative probability density function is usually estimated by ranking the observed return on the hedged portfolio. A detailed description of the process can be found in Kolb and Okunev (1992), and we briefly describe the process here.

The cumulative probability distribution is estimated by using the rank as follows:

$$G(R_{h,i}) = \frac{Rank(R_{h,i})}{N},$$

where N is the sample size. Once we have the series for the probability distribution function, the MEG is estimated by replacing the theoretical covariance by the sample covariance as follows:

$$\Gamma_v^{sample}(R_h) = -\frac{v}{N}\sum_{i=1}^{N}(R_{h,i} - \overline{R}_h)\Big((1 - G(R_{h,i}))^{v-1} - \Theta\Big), \quad (21.29)$$

where

$$\overline{R}_h = \frac{1}{N}\sum_{i=1}^{N} R_{h,i} \quad \text{and} \quad \Theta = \frac{1}{N}\sum_{i=1}^{N}\big(1 - G(R_{h,i})\big)^{v-1}.$$

The optimal hedge ratio is now given by the hedge ratio that minimizes the estimated MEG. Since there is no analytical solution, the numerical method needs to be applied in order to get the optimal hedge ratio. This method is sometimes referred to as the empirical distribution method.

Alternatively, the instrumental variable (IV) method suggested by Shalit (1995) can be used to find the MEG hedge ratio. Shalit's method provides the following analytical solution for the MEG hedge ratio:

$$h^{IV} = \frac{Cov\Big(S_{t+1}, [1 - G(F_{t+1})]^{v-1}\Big)}{Cov\Big(F_{t+1}, [1 - G(F_{t+1})]^{v-1}\Big)}.$$

It is important to note that for the IV method to be valid, the cumulative distribution function of the terminal wealth (W_{t+1}) should be similar to the cumulative distribution of the futures price (F_{t+1}); i.e., $G(W_{t+1}) = G(F_{t+1})$. Lien and Shaffer (1999) find that the IV-based hedge ratio (h^{IV}) is significantly different from the minimum MEG hedge ratio.

Lien and Luo (1993a) suggest an alternative method of estimating the MEG hedge ratio. This method involves the estimation of the cumulative distribution function using a non-parametric kernel function instead of using a rank function as suggested above.

Regarding the estimation of the M-MEG hedge ratio, one can follow either the empirical distribution method or the non-parametric kernel method to estimate the MEG coefficient. A numerical method can then be used to estimate the hedge ratio that maximizes the objective function given by Eq. (21.10).

21.3.5 Estimation of Generalized Semivariance (GSV) Based Hedge Ratios

The GSV can be estimated from the sample by using the following sample counterpart:

$$V_{\delta,\alpha}^{sample}(R_h) = \frac{1}{N}\sum_{i=1}^{N}\big(\delta - R_{h,i}\big)^{\alpha}U\big(\delta - R_{h,i}\big), \quad (21.30)$$

where

$$U\big(\delta - R_{h,i}\big) = \begin{cases} 1 \text{ for } \delta \geq R_{h,i} \\ 0 \text{ for } \delta < R_{h,i} \end{cases}.$$

Similar to the MEG technique, the optimal GSV hedge ratio can be estimated by choosing the hedge ratio that minimizes the sample GSV, $V_{\delta,\alpha}^{sample}(R_h)$. Numerical methods can be used to search for the optimum hedge ratio. Similarly, the M-GSV hedge ratio can be obtained by minimizing the mean-risk function given by Eq. (21.12), where the expected return on the hedged portfolio is replaced by the sample average return and the GSV is replaced by the sample GSV.

One can instead use the kernel density estimation method suggested by Lien and Tse (2000) to estimate the GSV, and numerical techniques can be used to find the optimum GSV hedge ratio. Instead of using the kernel method, one can also employ the conditional heteroscedastic model to estimate the density function. This is the method used by Lien and Tse (1998).

21.4 Applications of OLS, GARCH, and CECM Models to Estimate Optimal Hedge Ratio[2]

In this section, we apply OLS, GARCH, and CECM models to estimate optimal hedge ratios through R language. Monthly data for S&P 500 index and its futures were

[2] R programs that are used to estimate the empirical results in this section can be found in Appendix 21.4.

Table 21.2 Hedge ratio coefficient using the conventional regression model

Variable	Estimate	Std. error	t-ratio	p-value
Intercept	0.1984	0.2729	0.73	0.4680
ΔF_t	0.9851	0.0034	292.53	<0.0001

collected from Datastream database, the sample consisted of 188 observations from January 31, 2005, to August 31, 2020. First, we use OLS method by regressing the changes in spot prices on the changes in futures prices to estimate the optimal hedge ratio. The estimate of hedge ratio obtained from the OLS technique are reported in Table 21.2. As shown in Table 21.2, we can see that the hedge ratio of S&P 500 index is significantly different from zero, at a 1% significance level. Moreover, the estimated hedge ratio, denoted by the coefficient of ΔF_t, is generally less than unity.

Secondly, we apply a conventional regression model with heteroscedastic error terms to estimate the hedge ratio. Here, an AR(2)-GARCH(1, 1) model for the changes in spot prices regressed on the changes in futures prices is specified as follows,

$$\Delta S_t = a_0 + a_1 \Delta F_t + e_t, e_t = \varepsilon_t - \varphi_1 e_{t-1} - \varphi_2 e_{t-2}$$

$$\varepsilon_t = \sqrt{h_t}\epsilon_t, h_t = \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 h_{t-1}$$

where $\epsilon_t \sim N(0,1)$. The estimated result of AR(2)-GARCH (1, 1) model is shown in Table 21.3. The coefficient estimates of the AR(2)-GARCH(1, 1) model, as shown in Table 21.3, are all significantly different from zero, at a 1% significance level. This finding suggests that the importance of capturing the heteroscedastic error structures in conventional regression model. In addition, the hedge ratio of conventional regression with AR(2)-GARCH(1, 1) model is higher than the OLS hedge ratio for S&P 500 futures contract.

Next, we will apply the CECM model to estimate the optimal hedge ratio. Here, standard augmented Dickey-Fuller (ADF) unit roots and Phillips and Ouliaris (1990) residual cointegration tests are performed and the optimal hedge ratios estimated by error correction model (ECM) will be presented. Here, we apply the augmented Dickey- Fuller (ADF) regression to test for the presence of unit roots. The ADF test statistics, as shown in Panel A of Table 21.4, indicate that the null hypothesis of a unit root cannot be rejected for the levels of the variables. Using differenced data, the computed ADF test statistics shown in Panel B of Table 21.4 suggested that the null hypothesis is rejected, at the 1% significance level. As differencing one produces stationarity, we may conclude that each series is integrated of order one, I(1), process which is necessary for testing the existence of cointegration. We then apply Phillips and Ouliaris (1990) residual cointegration test to examine the presence of cointegration. The result of Phillips–Ouliaris cointegration test shown is reported in Panel C of Table 21.4. The null hypothesis of the Phillips–Ouliaris cointegration test is that there is no cointegration present. The result of Phillips–Ouliaris cointegration test indicates the null hypothesis of no cointegration is rejected, at 1% significance level. This suggests that the spot S&P 500 index is cointegrated with the S&P 500 index futures.

Finally, we apply the ECM model in terms of Eq. (21.17) to estimate the optimal hedge ratio. Table 21.5 shows that the coefficient on the error-correction term, $\hat{u}_{t-1}$, is significantly different from zero, at a 1% significance level. This suggests that the importance of estimating the error correction model, and in particular the long-run equilibrium error term cannot be ignored in the conventional regression model. In addition, the ECM hedge ratio is higher than the conventional OLS hedge ratio for S&P 500 futures contract. This finding is consistent with the results in Lien (1996, 2004) who argued that the MV hedge ratio will be smaller if the cointegration relationship is not considered.

Table 21.3 Hedge ratio coefficient using the conventional regression model with heteroscedastic errors

Variable	Estimate	Std. error	t-ratio	p-value
Intercept	0.0490	0.0144	3.41	0.0007
ΔF_t	0.9994	0.0008	1179.59	<0.0001
e_{t-1}	−0.9873	0.0109	−90.29	<0.0001
e_{t-2}	−0.9959	0.0145	−68.83	<0.0001
ω	0.0167	0.0098	1.71	0.0866
ε_{t-1}^2	0.3135	0.0543	5.78	<0.0001
h_{t-1}	0.6855	0.0530	12.94	<0.0001

Table 21.4 Unit roots and residual cointegration tests results

Variable	ADF statistics	Lag parameter	p-value
Panel A. Level data			
Spot	−1.3353	1	0.8542
Futures	−1.3458	1	0.8498
Panel B. First-order differenced data			
Spot	−10.104	1	<0.01
Futures	−10.150	1	<0.01
Panel C. Phillips–Ouliaris cointegration test			
Phillips–Ouliaris demeaned	−60.783	1	<0.01

Table 21.5 Error correction estimates of hedge ratio coefficient

Variable	Estimate	Std. error	t-ratio	p-value
ΔF_t	0.9892	0.0031	316.60	<0.001
$\hat{u}_{t-1}$	−0.3423	0.0571	−5.99	<0.001

21.5 Hedging Horizon, Maturity of Futures Contract, Data Frequency, and Hedging Effectiveness

In this section, we discuss the relationship among the length of hedging horizon (hedging period), maturity of futures contracts, data frequency (e.g., daily, weekly, monthly, or quarterly), and hedging effectiveness.

Since there are many futures contracts (with different maturities) that can be used in hedging, the question is whether the minimum-variance (MV) hedge ratio depends on the time to maturity of the futures contract being used for hedging. Lee et al. (1987) find that the MV hedge ratio increases as the maturity is approached. This means that if we use the nearest to maturity futures contracts to hedge, then the MV hedge ratio will be larger compared to the one obtained using futures contracts with a longer maturity.

Aside from using futures contracts with different maturities, we can estimate the MV hedge ratio using data with different frequencies. For example, the data used in the estimation of the optimum hedge ratio can be daily, weekly, monthly, or quarterly. At the same time, the hedging horizon could be from a few hours to more than a month. The question is whether a relationship exists between the data frequency used and the length of the hedging horizon.

Malliaris and Urrutia (1991) and Benet (1992) utilize Eq. (21.20) and weekly data to estimate the optimal hedge ratio. According to Malliaris and Urrutia (1991), the ex ante hedging is more effective when the hedging horizon is one week compared to a hedging horizon of four weeks. Benet (1992) finds that a shorter hedging horizon (four-weeks) is more effective (in ex ante test) compared to a longer hedging horizon (eight-weeks and twelve-weeks). These empirical results seem to be consistent with the argument that when estimating the MV hedge ratio, the hedging horizon's length must match the data frequency being used.

There is a potential problem associated with matching the length of the hedging horizon and the data frequency. For example, consider the case where the hedging horizon is three months (one quarter). In this case we need to use quarterly data to match the length of the hedging horizon. In other words, when estimating Eq. (21.20) we must employ quarterly changes in spot and futures prices. Therefore, if we have five years' worth of data, then we will have 19 non-overlapping price changes, resulting in a sample size of 19. However, if the hedging horizon is one week, instead of three months, then we will end up with approximately 260 non-overlapping price changes (sample size of 260) for the same five years' worth of data. Therefore, the matching method is associated with a reduction in sample size for a longer hedging horizon.

One way to get around this problem is to use overlapping price changes. For example, Geppert (1995) utilizes k-period differencing for a k-period hedging horizon in estimating the regression-based MV hedge ratio. Since Geppert (1995) uses approximately 13 months of data for estimating the hedge ratio, he employs overlapping differencing in order to eliminate the reduction in sample size caused by differencing. However, this will lead to correlated observations instead of independent observations and will require the use of a regression with autocorrelated errors in the estimation of the hedge ratio.

In order to eliminate the autocorrelated errors problem, Geppert (1995) suggests a method based on cointegration and unit-root processes. We will briefly describe his method.

Suppose that the spot and futures prices, which are both unit-root processes, are cointegrated. In this case the futures and spot prices can be described by the following processes (see Stock and Watson 1988; Hylleberg and Mizon 1989):

$$S_t = A_1 P_t + A_2 \tau_t, \tag{21.31a}$$

$$F_t = B_1 P_t + B_2 \tau_t, \tag{21.31b}$$

$$P_t = P_{t-1} + w_t, \tag{21.31c}$$

$$\tau_t = \alpha_1 \tau_{t-1} + v_t, \quad 0 \le |\alpha_1| < 1, \tag{21.31d}$$

where P_t and τ_t are permanent and transitory factors that drive the spot and futures prices and w_t and v_t are white noise processes. Note that P_t follows a pure random walk process and τ_t follows a stationary process. The MV hedge ratio for a k-period hedging horizon is then given by (see Geppert 1995):

$$H_J^* = \frac{A_1 B_1 k\sigma_w^2 + 2A_2 B_2 \left(\frac{(1-\alpha^k)}{1-\alpha^2}\right)\sigma_v^2}{B_1^2 k\sigma_w^2 + 2B_2^2 \left(\frac{(1-\alpha^k)}{1-\alpha^2}\right)\sigma_v^2}. \tag{21.32}$$

One advantage of using Eq. (21.32) instead of a regression with non-overlapping price changes is that it avoids the problem of a reduction in sample size associated with non-overlapping differencing.

An alternative way of matching the data frequency with the hedging horizon is by using the wavelet to decompose the time series into different frequencies as suggested by Lien and Shrestha (2007). The decomposition can be done without the loss of sample size (see Lien and Shrestha (2007) for detail). For example, the daily spot and future returns series can be decomposed using the maximal overlap discrete wavelet transform (MODWT) as follows:

$$R_{s,t} = B_{J,t}^s + D_{J,t}^s + D_{J-1,t}^s + \cdots + D_{1,t}^s$$

$$R_{f,t} = B_{J,t}^f + D_{J,t}^f + D_{J-1,t}^f + \cdots + D_{1,t}^f$$

where $D_{j,t}^s$ and $D_{j,t}^f$ are the spot and futures returns series with changes on the time scale of length 2^{j-1} days, respectively.[4] Similarly, $B_{J,t}^s$ and $B_{J,t}^2$ represent spot and futures returns series corresponding to time scale of 2^J days and longer. Now, we can run the following regression to find the hedge ratio corresponding to hedging horizon equal to 2^{j-1} days:

$$D_{j,t}^s = \theta_{j,0} + \theta_{j,1} D_{j,t}^f + e_j \tag{21.33}$$

where the estimate of the hedge ratio is given by the estimate of $\theta_{j,1}$.

21.6 Summary and Conclusions

In this chapter, we have reviewed various approaches to deriving the optimal hedge ratio, as summarized in Appendix 21.1. These approaches can be divided into the mean–variance-based approach, the expected utility maximizing approach, the mean extended-Gini coefficient-based approach, and the generalized semivariance-based approach. All these approaches will lead to the same hedge ratio as the conventional minimum-variance (MV) hedge ratio if the futures price follows a pure martingale process and if the futures and spot prices are jointly normal. However, if these conditions do not hold, then the hedge ratios-based on the various approaches will be different.

The MV hedge ratio is the most understood and most widely used hedge ratio. Since the statistical properties of the MV hedge ratio are well known, statistical hypothesis testing can be performed with the MV hedge ratio. For example, we can test whether the optimal MV hedge ratio is the same as the naïve hedge ratio. Since the MV hedge ratio ignores the expected return, it will not be consistent with the mean–variance analysis unless the futures price follows a pure martingale process. Furthermore, if the martingale and normality condition do not hold, then the MV hedge ratio will not be consistent with the expected utility maximization principle. Following the MV hedge ratio is the mean–variance hedge ratio. Even if this hedge ratio incorporates the expected return in the derivation of the optimal hedge ratio, it will not be consistent with the expected maximization principle unless either the normality condition holds or the utility function is quadratic.

In order to make the hedge ratio consistent with the expected utility maximization principle, we can derive the optimal hedge ratio by maximizing the expected utility. However, to implement such approach, we need to assume a

specific utility function and we need to make an assumption regarding the return distribution. Therefore, different utility functions will lead to different optimal hedge ratios. Furthermore, analytic solutions for such hedge ratios are not known and numerical methods need to be applied.

New approaches have recently been suggested in deriving optimal hedge ratios. These include the mean-Gini coefficient-based hedge ratio, semivariance-based hedge ratios and Value-at-Risk-based hedge ratios. These hedge ratios are consistent with the second-order stochastic dominance principle. Therefore, such hedge ratios are very general in the sense that they are consistent with the expected utility maximization principle and make very few assumptions on the utility function. The only requirement is that the marginal utility be positive and the second derivative of the utility function be negative. However, both of these hedge ratios do not lead to a unique hedge ratio. For example, the mean-Gini coefficient-based hedge ratio depends on the risk aversion parameter (v) and the semivariance-based hedge ratio depends on the risk aversion parameter (α) and target return (δ). It is important to note, however, that the semivariance-based hedge ratio has some appeal in the sense that the semivariance as a measure of risk is consistent with the risk perceived by individuals. The same argument can be applied to Value-at-Risk-based hedge ratio.

So far as the derivation of the optimal hedge ratio is concerned, almost all of the derivations do not incorporate transaction costs. Furthermore, these derivations do not allow investments in securities other than the spot and corresponding futures contracts. As shown by Lence (1995), once we relax these conventional assumptions, the resulting optimal hedge ratio can be quite different from the ones obtained under the conventional assumptions. Lence's (1995) results are based on a specific utility function and some other assumption regarding the return distributions. It remains to be seen if such results hold for the mean extended-Gini coefficient-based as well as semivariance-based hedge ratios.

In this chapter, we have also reviewed various ways of estimating the optimum hedge ratio, as summarized in Appendix 21.2. As far as the estimation of the conventional MV hedge ratio is concerned, there are a large number of methods that have been proposed in the literature. These methods range from a simple regression method to complex cointegrated heteroscedastic methods with regime-switching, and some of the estimation methods include a kernel density function method as well as an empirical distribution method. Except for many of mean–variance-based hedge ratios, the estimation involves the use of a numerical technique. This has to do with the fact that most of the optimal hedge ratio formulae do not have a closed-form analytic expression. Again, it is important to mention that based on his specific model, Lence (1995) finds that the value of complicated and sophisticated estimation methods is negligible. It remains to be seen if such a result holds for the mean extended-Gini coefficient-based as well as semivariance-based hedge ratios.

In this chapter, we have also discussed about the relationship between the optimal MV hedge ratio and the hedging horizon. We feel that this relationship has not been fully explored and can be further developed in the future. For example, we would like to know if the optimal hedge ratio approaches the naïve hedge ratio when the hedging horizon becomes longer.

The main thing we learn from this review is that if the futures price follows a pure martingale process and if the returns are jointly normally distributed, then all different hedge ratios are the same as the conventional MV hedge ratio, which is simple to compute and easy to understand. However, if these two conditions do not hold, then there are many optimal hedge ratios (depending on which objective function one is trying to optimize) and there is no single optimal hedge ratio that is distinctly superior to the remaining ones. Therefore, further research needs to be done to unify these different approaches to the hedge ratio.

For those who are interested in research in this area, we would like to finally point out that one requires a good understanding of financial economic theories and econometric methodologies. In addition, a good background in data analysis and computer programming would also be helpful.

Appendix 21.1: Theoretical Models

References	Return definition and objective function	Summary
Johnson (1960)	Ret_1 O_1	The chapter derives the minimum-variance hedge ratio. The hedging effectiveness is defined as E_1, but no empirical analysis is done
Hsin et al. (1994)	Ret_2 O_2	The chapter derives the utility function-based hedge ratio. A new measure of hedging effectiveness E_2 based on a certainty equivalent is proposed. The new measure of hedging effectiveness is used to compare the effectiveness of futures and options as hedging instruments
Howard and D'Antonio (1984)	Ret_2 O_3	The chapter derives the optimal hedge ratio based on maximizing the Sharpe ratio. The proposed hedging effectiveness E_3 is based on the Sharpe ratio
Cecchetti et al. (1988)	Ret_2 O_4	The chapter derives the optimal hedge ratio that maximizes the expected utility function: $\int_{R_s}\int_{R_f}\log\left[1+R_s(t)-h(t)R_f(t)\right]f_t\left(R_s,R_f\right)dR_s dR_f$, where the density function is assumed to be bivariate normal. A third-order linear bivariate ARCH model is used to get the conditional variance and covariance matrix. A numerical procedure is used to maximize the objective function with respect to the hedge ratio. Due to ARCH, the hedge ratio changes over time. The chapter uses certainty equivalent (E_2) to measure the hedging effectiveness
Cheung et al. (1990)	Ret_2 O_5	The chapter uses mean-Gini ($v = 2$, not mean extended-Gini coefficient) and mean–variance approaches to analyze the effectiveness of options and futures as hedging instruments
Kolb and Okunev (1992)	Ret_2 O_5	The chapter uses mean extended-Gini coefficient in the derivation of the optimal hedge ratio. Therefore, it can be considered as a generalization of the mean-Gini coefficient method used by Cheung et al. (1990)
Kolb and Okunev (1993)	Ret_2 O_6	The chapter defines the objective function as O_6, but in terms of wealth (W) $U(W) = E[W] - \Gamma_v(W)$ and compares with the quadratic utility function $U(W) = E[W] - m\sigma^2$. The chapter plots the EMG efficient frontier in W and $\Gamma_v(W)$ space for various values of risk aversion parameters (v)
Lien and Luo (1993b)	Ret_1 O_9	The chapter derives the multi-period hedge ratios where the hedge ratios are allowed to change over the hedging period. The method suggested in the chapter still falls under the minimum-variance hedge ratio
Lence (1995)	O_4	This chapter derives the expected utility maximizing hedge ratio where the terminal wealth depends on the return on a diversified portfolio that consists of the production of a spot commodity, investment in a risk-free asset, investment in a risky asset, as well as borrowing. It also incorporates the transaction costs
De Jong et al. (1997)	Ret_2 O_7 (also uses O_1 and O_3)	The chapter derives the optimal hedge ratio that minimizes the generalized semivariance (GSV). The chapter compares the GSV hedge ratio with the minimum-variance (MV) hedge ratio as well as the Sharpe hedge ratio. The chapter uses E_1 (for the MV hedge ratio), E_3 (for the Sharpe hedge ratio), and E_4 (for the GSV hedge ratio) as the measures of hedging effectiveness
Chen et al. (2001)	Ret_1 O_8	The chapter derives the optimal hedge ratio that maximizes the risk-return function given by $U(R_h) = E[R_h] - V_{\delta,\alpha}(R_h)$. The method can be considered as an extension of the GSV method used by De Jong et al. (1997)
Hung et al. (2006)	Ret_2 O_{10}	The chapter derives the optimal hedge ratio that minimizes the Value-at-Risk for a hedging horizon of length τ given by $Z_\alpha\sigma_h\sqrt{\tau} - E[R_h]\tau$

Notes

A. *Return Model*

(Ret_1)	$\Delta V_H = C_s \Delta P_s + C_f \Delta P_f \quad \Rightarrow \quad$ hedge ratio $= H = \frac{C_f}{C_s}$, $C_s =$ units of spot commodity and $C_f =$ units of futures contract	
(Ret_2)	$R_h = R_s + hR_f, \qquad R_s = \frac{S_t - S_{t-1}}{S_{t-1}}$	(a) $R_f = \frac{F_t - F_{t-1}}{F_{t-1}} \Rightarrow$ hedge ratio : $h = \frac{C_f F_{t-1}}{C_s S_{t-1}}$ (b) $R_f = \frac{F_t - F_{t-1}}{S_{t-1}} \Rightarrow$ hedge ratio : $h = \frac{C_f}{C_s}$

B. *Objective Function*:

(O_1)	Minimize	$Var(R_h) = C_s^2\sigma_s^2 + C_f^2\sigma_f^2 + 2C_sC_f\sigma_{sf} \quad$ or $\quad Var(R_h) = \sigma_s^2 + h^2\sigma_f^2 + 2h\sigma_{sf}$
(O_2)	Maximize	$E(R_h) - \frac{A}{2}Var(R_h)$
(O_3)	Maximize	$\frac{E(R_h) - R_F}{Var(R_h)}$ (Sharpe ratio), $\mathrm{R_F} =$ risk $-$ free interest rate
(O_4)	Maximize	$E[U(W)], \quad U(.) =$ utility function, $\quad \mathrm{W} =$ terminal wealth
(O_5)	Minimize	$\Gamma_v(R_h), \quad \Gamma_v(R_h) = -vCov\left(R_h, (1 - F(R_h))^{v-1}\right)$
(O_6)	Maximize	$E[R_h] - \Gamma_v(R_h v)$
(O_7)	Minimize	$V_{\delta,\alpha}(R_h) = \int_{-\infty}^{\delta} (\delta - R_h)^{\alpha} dG(R_h), \quad \alpha > 0$
(O_8)	Maximize	$U(R_h) = E[R_h] - V_{\delta,\alpha}(R_h)$
(O_9)	Minimize	$Var(W_t) = Var\left(\sum_{t=1}^{T} C_{st}\Delta S_t + C_{ft}\Delta F_t\right)$
(O_{10})	Minimize	$Z_\alpha \sigma_h \sqrt{\tau} - E[R_h]\tau$

C. *Hedging Effectiveness*

(E_1)	$e = \left(1 - \frac{Var(R_h)}{Var(R_s)}\right)$
(E_2)	$e = R_h^{ce} - R_{ss}^{ce}, \quad R_h^{ce}(R_s^{ce}) =$ certainty equivalent return of hedged (unhedged) portfolio
(E_3)	$e = \frac{\frac{(E[R_h] - R_F)}{Var(R_h)}}{\frac{(E[R_s] - R_F)}{Var(R_s)}}$ or $e = \frac{(E[R_h] - R_F)}{Var(R_h)} - \frac{(E[R_s] - R_F)}{Var(R_s)}$
(E_4)	$e = 1 - \frac{V_{\delta,\alpha}(R_h)}{V_{\delta,\alpha}(R_s)}$

Appendix 21.2: Empirical Models

References	Commodity	Summary
Ederington (1979)	GNMA futures (1/1976–12/1977), Wheat (1/1976–12/1977), Corn (1/1976–12/1977), T-bill futures (3/1976–12/1977) [weekly data]	The chapter uses the Ret_1 definition of return and estimates the minimum-variance hedge ratio (O_1). E_1 is used as a hedging effectiveness measure. The chapter uses nearby contracts (3–6 months, 6–9 months and 9–12 months) and a hedging period of 2 weeks and 4 weeks. OLS (M_1) is used to estimate the parameters. Some of the hedge ratios are found not to be different from zero and the hedging effectiveness increases with the length of the hedging period. The hedge ratio also increases (closer to unity) with the length of the hedging period
Grammatikos and Saunders (1983)	Swiss franc, Canadian dollar, British pound, DM, Yen (1/1974–6/1980) [weekly data]	The chapter estimates the hedge ratio for the whole period and moving window (2-year data). It is found that the hedge ratio changes over time. Dummy variables for various sub-periods are used, and shifts are found. The chapter uses a random coefficient (M_3) model to estimate the hedge ratio. The hedge ratio for Swiss franc is found to follow a random coefficient model. However, there is no improvement in effectiveness when the hedge ratio is calculated by correcting for the randomness
Junkus and Lee (1985)	Three stock index futures for Kansas City Board of Trade, New York Futures Exchange, and Chicago Mercantile Exchange (5/82–3/83) [daily data]	The chapter tests the applicability of four futures hedging models: a variance-minimizing model introduced by Johnson (1960), the traditional one to one hedge, a utility maximization model developed by Rutledge (1972), and a basis arbitrage model suggested by Working (1953). An optimal ratio or decision rule is estimated for each model, and measures for the effectiveness of each hedge are devised. Each hedge strategy performed best according to its own criterion. The Working decision rule appeared to be easy to use and satisfactory in most cases. Although the maturity of the futures contract used affected the size of the optimal hedge ratio, there was no consistent maturity effect on performance. Use of a particular ratio depends on how closely the assumptions underlying the model approach a hedger's real situation
Lee et al. (1987)	S&P 500, NYSE, Value Line (1983) [daily data]	The chapter tests for the temporal stability of the minimum-variance hedge ratio. It is found that the hedge ratio increases as maturity of the futures contract nears. The chapter also performs a functional form test and finds support for the regression of rate of change for discrete as well as continuous rates of change in prices
Cecchetti et al. (1988)	Treasury bond, Treasury bond futures (1/1978–5/1986) [monthly data]	The chapter derives the hedge ratio by maximizing the expected utility. A third-order linear bivariate ARCH model is used to get the conditional variance and covariance matrix. A numerical procedure is used to maximize the objective function with respect to the hedge ratio. Due to ARCH, the hedge ratio changes over time. It is found that the hedge ratio changes over time and is significantly less (in

(continued)

References	Commodity	Summary
		absolute value) than the minimum-variance (MV) hedge ratio (which also changes over time). E_2 (certainty equivalent) is used to measure the performance effectiveness. The proposed utility-maximizing hedge ratio performs better than the MV hedge ratio
Cheung et al. (1990)	Swiss franc, Canadian dollar, British pound, German mark, Japanese yen (9/1983–12/1984) [daily data]	The chapter uses mean-Gini coefficient ($v = 2$) and mean–variance approaches to analyze the effectiveness of options and futures as hedging instruments. It considers both mean–variance and expected-return mean-Gini coefficient frontiers. It also considers the minimum-variance (MV) and minimum mean-Gini coefficient hedge ratios. The MV and minimum mean-Gini approaches indicate that futures is a better hedging instrument. However, the mean–variance frontier indicates futures to be a better hedging instrument, whereas the mean-Gini frontier indicates options to be a better hedging instrument
Baillie and Myers (1991)	Beef, Coffee, Corn, Cotton, Gold, Soybean (contracts maturing in 1982 and 1986) [daily data]	The chapter uses a bivariate GARCH model (M_2) in estimating the minimum-variance (MV) hedge ratios. Since the models used are conditional models, the time series of hedge ratios are estimated. The MV hedge ratios are found to follow a unit root process. The hedge ratio for beef is found to be centered around zero. E_1 is used as a hedging effectiveness measure. Both in-sample and out-of-sample effectiveness of the GARCH-based hedge ratios is compared with a constant hedge ratio. The GARCH-based hedge ratios are found to be significantly better compared to the constant hedge ratio
Malliaris and Urrutia (1991)	British pound, German mark, Japanese yen, Swill franc, Canadian dollar (3/1980–12/1988) [weekly data]	The chapter uses regression autocorrelated errors model to estimate the minimum-variance (MV) hedge ratio for the five currencies. Using overlapping moving windows, the time series of the MV hedge ratio and hedging effectiveness are estimated for both ex post (in-sample) and ex ante (out-of-sample) cases. E_1 is used to measure the hedging effectiveness for the ex post case, whereas average return is used to measure the hedging effectiveness. Specifically, the average return close to zero is used to indicate a better performing hedging strategy. In the ex post case, the four-week hedging horizon is more effective compared to the one-week hedging horizon. However, for the ex ante case the opposite is found to be true
Benet (1992)	Australian dollar, Brazilian cruzeiro, Mexican peso, South African rand, Chinese yuan, Finish markka, Irish pound, Japanese yen (8/1973–12/1985) [weekly data]	This chapter considers direct and cross hedging, using multiple futures contracts. For minor currencies, the cross hedging exhibits a significant decrease in performance from ex post to ex ante. The minimum-variance hedge ratios are found to change from one period to the other except for the direct hedging of Japanese yen. On the ex ante case, the hedging effectiveness does not appear to be related to the estimation period length. However, the effectiveness decreases as the hedging period length increases

(continued)

References	Commodity	Summary
Kolb and Okunev (1992)	Corn, Copper, Gold, German mark, S&P 500 (1989) [daily data]	The chapter estimates the mean extended-Gini (MEG) hedge ratio (M_9) with v ranging from 2 to 200. The MEG hedge ratios are found to be close to the minimum-variance hedge ratios for a lower level of risk parameter v (for v from 2 to 5). For higher values of v, the two hedge ratios are found to be quite different. The hedge ratios are found to increase with the risk aversion parameter for S&P 500, Corn, and Gold. However, for Copper and German mark, the hedge ratios are found to decrease with the risk aversion parameter. The hedge ratio tends to be more stable for higher levels of risk
Kolb and Okunev (1993)	Cocoa (3/1952 to 1976) for four cocoa-producing countries (Ghana, Nigeria, Ivory Coast, and Brazil) [March and September data]	The chapter estimates the Mean-MEG (M-MEG) hedge ratio (M_{12}). The chapter compares the M-MEG hedge ratio, minimum-variance hedge ratio, and optimum mean–variance hedge ratio for various values of risk aversion parameters. The chapter finds that the M-MEG hedge ratio leads to reverse hedging (buy futures instead of selling) for v less than 1.24 (Ghana case). For high-risk aversion parameter values (high v) all hedge ratios are found to converge to the same value
Lien and Luo (1993a)	S&P 500 (1/1984–12/1988) [weekly data]	The chapter points out that the mean extended-Gini (MEG) hedge ratio can be calculated either by numerically optimizing the MEG coefficient or by numerically solving the first-order condition. For $v = 9$ the hedge ratio of −0.8182 is close to the minimum-variance (MV) hedge ratio of −0.8171. Using the first-order condition, the chapter shows that for a large v the MEG hedge ratio converges to a constant. The empirical result shows that the hedge ratio decreases with the risk aversion parameter v. The chapter finds that the MV and MEG hedge ratio (for low v) series (obtained by using a moving window) are more stable compared to the MEG hedge ratio for a large v. The chapter also uses a non-parametric Kernel estimator to estimate the cumulative density function. However, the kernel estimator does not change the result significantly
Lien and Luo (1993b)	British pound, Canadian dollar, German mark, Japanese yen, Swiss franc (3/1980–12/1988), MMI, NYSE, S&P (1/1984–12/1988) [weekly data]	This chapter proposes a multi-period model to estimate the optimal hedge ratio. The hedge ratios are estimated using an error-correction model. The spot and futures prices are found to be cointegrated. The optimal multi-period hedge ratios are found to exhibit a cyclical pattern with a tendency for the amplitude of the cycles to decrease. Finally, the possibility of spreading among different market contracts is analyzed. It is shown that hedging in a single market may be much less effective than the optimal spreading strategy
Ghosh (1993)	S&P futures, S&P index, Dow Jones Industrial average, NYSE composite index (1/1990–12/1991) [daily data]	All the variables are found to have a unit root. For all three indices the same S&P 500 futures contracts are used (cross hedging). Using the Engle-Granger two-step test, the S&P 500 futures price is found to be cointegrated with each of the three spot prices: S&P 500, DJIA, and NYSE. The hedge ratio is estimated using the error-correction model (ECM) (M_4). Out-of-sample performance is better for the hedge ratio from the ECM compared to the Ederington model

(continued)

References	Commodity	Summary
Sephton (1993a)	Feed wheat, Canola futures (1981–82 crop year) [daily data]	The chapter finds unit roots on each of the cash and futures (log) prices, but no cointegration between futures and spot (log) prices. The hedge ratios are computed using a four-variable GARCH(1, 1) model. The time series of hedge ratios are found to be stationary. Reduction in portfolio variance is used as a measure of hedging effectiveness. It is found that the GARCH-based hedge ratio performs better compared to the conventional minimum-variance hedge ratio
Sephton (1993b)	Feed wheat, Feed barley, Canola futures (1988/89) [daily data]	The chapter finds unit roots on each of the cash and futures (log) prices, but no cointegration between futures and spot (log) prices. A univariate GARCH model shows that the mean returns on the futures are not significantly different from zero. However, from the bivariate GARCH canola is found to have a significant mean return. For canola the mean variance utility function is used to find the optimal hedge ratio for various values of the risk aversion parameter. The time series of the hedge ratio (based on bivariate GARCH model) is found to be stationary. The benefit in terms of utility gained from using a multivariate GARCH decreases as the degree of risk aversion increases
Kroner and Sultan (1993)	British pound, Canadian dollar, German mark, Japanese yen, Swiss franc (2/1985–2/1990) [weekly data]	The chapter uses the error-correction model with a GARCH error (M_5) to estimate the minimum-variance (MV) hedge ratio for the five currencies. Due to the use of conditional models, the time series of the MV hedge ratios are estimated. Both within-sample and out-of-sample evidence show that the hedging strategy proposed in the chapter is potentially superior to the conventional strategies
Hsin et al. (1994)	British pound, German mark, Yen, Swiss franc (1/1986–12/1989) [daily data]	The chapter derives the optimum mean–variance hedge ratio by maximizing the objective function O_2. The hedging horizons of 14, 30, 60, 90, and 120 calendar days are considered to compare the hedging effectiveness of options and futures contracts. It is found that the futures contracts perform better than the options contracts
Shalit (1995)	Gold, Silver, Copper, Aluminum (1/1977–12/1990) [daily data]	The chapter shows that if the prices are jointly normally distributed, the mean extended-Gini (MEG) hedge ratio will be same as the minimum-variance (MV) hedge ratio. The MEG hedge ratio is estimated using the instrumental variable method. The chapter performs normality tests as well as the tests to see if the MEG hedge ratios are different from the MV hedge ratios. The chapter finds that for a significant number of futures contracts the normality does not hold and the MEG hedge ratios are different from the MV hedge ratios
Geppert (1995)	German mark, Swiss franc, Japanese yen, S&P 500, Municipal Bond Index (1/1990–1/1993) [weekly data]	The chapter estimates the minimum-variance hedge ratio using the OLS as well as the cointegration methods for various lengths of hedging horizon. The in-sample results indicate that for both methods the hedging effectiveness increases with the length of the hedging horizon. The out-of-sample results indicate that in general the effectiveness (based on the method suggested by Malliaris and Urrutia (1991)) decreases as the length of the hedging horizon decreases. This

(continued)

References	Commodity	Summary
		is true for both the regression method and the decomposition method proposed in the chapter. However, the decomposition method seems to perform better than the regression method in terms of both mean and variance
De Jong et al. (1997)	British pound (12/1976–10/1993), German mark (12/1976–10/1993), Japanese yen (4/1977–10/1993) [daily data]	The chapter compares the minimum-variance, generalized semivariance and Sharpe hedge ratios for the three currencies. The chapter computes the out-of-sample hedging effectiveness using non-overlapping 90-day periods where the first 60 days are used to estimate the hedge ratio and the remaining 30 days are used to compute the out-of-sample hedging effectiveness. The chapter finds that the naïve hedge ratio performs better than the model-based hedge ratios
Lien and Tse (1998)	Nikkei Stock Average (1/1989–8/1996) [daily data]	The chapter shows that if the rates of change in spot and futures prices are bivariate normal and if the futures price follows a martingale process, then the generalized semivariance (GSV) (referred to as lower partial moment) hedge ratio will be same as the minimum-variance (MV) hedge ratio. A version of the bivariate asymmetric power ARCH model is used to estimate the conditional joint distribution, which is then used to estimate the time varying GSV hedge ratios. The chapter finds that the GSV hedge ratio significantly varies over time and is different from the MV hedge ratio
Lien and Shaffer (1999)	Nikkei (9/86–9/89), S&P (4/82–4/85), TOPIX (4/90–12/93), KOSPI (5/96–12/96), Hang Seng (1/87–12,189), IBEX (4/93–3/95) [daily data]	This chapter empirically tests the ranking assumption used by Shalit (1995). The ranking assumption assumes that the ranking of futures prices is the same as the ranking of the wealth. The chapter estimates the mean extended-Gini (MEG) hedge ratio based on the instrumental variable (IV) method used by Shalit (1995) and the true MEG hedge ratio. The true MEG hedge ratio is computed using the cumulative probability distribution estimated employing the kernel method instead of the rank method. The chapter finds that the MEG hedge ratio obtained from the IV method to be different from the true MEG hedge ratio. Furthermore, the true MEG hedge ratio leads to a significantly smaller MEG coefficient compared to the IV-based MEG hedge ratio
Lien and Tse (2000)	Nikkei Stock Average (1/1988–8/996) [daily data]	The chapter estimates the generalized semivariance (GSV) hedge ratios for different values of parameters using a non-parametric kernel estimation method. The kernel method is compared with the empirical distribution method. It is found that the hedge ratio from one method is not different from the hedge ratio from another. The Jarque–Bera (1987) test indicates that the changes in spot and futures prices do not follow normal distribution
Chen et al. (2001)	S&P 500 (4/1982–12/1991) [weekly data]	The chapter proposes the use of the M-GSV hedge ratio. The chapter estimates the minimum-variance (MV), optimum mean–variance, Sharpe, mean extended-Gini (MEG), generalized semivariance (GSV), mean-MEG (M-MEG), and mean-GSV (M-GSV) hedge ratios. The Jarque–Bera (1987) Test and D'Agostino (1971) D Statistic indicate that the price changes are not normally distributed. Furthermore, the expected value of the futures price

(continued)

References	Commodity	Summary
		change is found to be significantly different from zero. It is also found that for a high level of risk aversion, the M-MEG hedge ratio converges to the MV hedge ratio whereas the M-GSV hedge ratio converges to a lower value
Hung et al. (2006)	S&P 500 (01/1997–12/1999) [daily data]	The chapter proposes minimization of Value-at-Risk in deriving the optimum hedge ratio. The chapter finds cointegrating relationship between the spot and futures returns and uses bivariate constant correlation GARCH(1, 1) model with error correction term. The chapter compares the proposed hedge ratio with MV hedge ratio and hedge ratio (HKL hedge ratio) proposed by Hsin et al. (1994). The chapter finds the performance of the proposed hedge ratio to be similar to the HKL hedge ratio. Finally, the proposed hedge ratio converges to the MV hedge ratio for high risk-averse levels
Lee and Yoder (2007)	Nikkei 225 and Hang Send index futures (01/1989–12/2003) [weekly data]	The chapter proposes regime-switching time varying correlation GARCH model and compares the resulting hedge ratio with constant correlation GARCH and time-varying correlation GARCH. The proposed model is found to outperform the other two hedge ratio in both in-sample and out-of-sample for both contracts
Lien and Shrestha (2007)	23 different futures contracts (sample period depends on contracts) [daily data]	This chapter proposes wavelet base hedge ratio to compute the hedge ratios for different hedging horizons (1-day, 2-day, 4-day, 8-day, 16 day, 32-day, 64-day, 128-day; and 256-day and longer). It is found that the wavelet-based hedge ratio and the error-correction-based hedge ratio are larger than MV hedge ratio. The performance of wavelet-based hedge ratio improves with the length of the hedging horizon
Lien and Shrestha (2010)	22 different futures contracts (sample period depends on contracts) [daily data]	The chapter proposes the hedge ratio based on skew-normal distribution (SKN hedge ratio). The chapter also estimates the semi-variance (lower partial moment (LPM)) hedge ratio and MV hedge ratio among other hedge ratios. SKN hedge ratios are found to be different from the MV hedge ratio based on normal distribution. SKN hedge ratio performs better than LPM hedge ratio for long hedger especially for the out-of-sample cases

Notes

A. *Minimum-Variance Hedge Ratio*

A.1. *OLS*

(M_1):	$\Delta S_t = a_0 + a_1 \Delta F_t + e_t$:	*Hedge ratio* = a_1
	$R_s = a_0 + a_1 R_f + e_t$:	*Hedge ratio* = a_1

A.2. *Multivariate Skew-Normal*

(M_2):	The return vector $Y = \begin{bmatrix} R_s \\ R_f \end{bmatrix}$ is assumed to have skew-normal distribution with covariance matrix V: *Hedge ration* $= H_{skn} = \frac{V(1,2)}{V(2,2)}$

A.3. *ARCH/GARCH*

(M_3):	$\begin{bmatrix} \Delta S_t \\ \Delta F_t \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \begin{bmatrix} e_{1t} \\ e_{2t} \end{bmatrix}$, $e_t \vert \Omega_{t-1}\ N(0, H_t)$, $\quad H_t = \begin{bmatrix} H_{11,t} & H_{12,t} \\ H_{12,t} & H_{22,t} \end{bmatrix}$, *Hedge ratio* $= H_{12,t}/H_{22,t}$

A.4. *Regime-Switching GARCH*

(M_4):	The transition probabilities are given by: $\Pr(s_t = 1 \vert s_{t-1} = 1) = \frac{e^{p_0}}{1+e^{p_0}}$ & $\Pr(s_t = 2 \vert s_{t-1} = 2) = \frac{e^{q_0}}{1+e^{q_0}}$ The GARCH model: Two-series GARCH model with first series as return on futures. $H_{t,s_t} = \begin{bmatrix} h_{1,t,s_t} & 0 \\ 0 & h_{2,t,s_t} \end{bmatrix} \begin{bmatrix} 1 & \rho_{t,s_t} \\ \rho_{t,s_t} & 1 \end{bmatrix} \begin{bmatrix} h_{1,t,s_t} & 0 \\ 0 & h_{2,t,s_t} \end{bmatrix}$ $h^2_{1,t,s_t} = \gamma_{1,s_t} + \alpha_{1,s_t} e^2_{1.t-1} + \beta_{1,s_t} h^2_{1,t-1}, \quad h^2_{2,t,s_t} = \gamma_{2,s_t} + \alpha_{2,s_t} e^2_{2.t-1} + \beta_{2,s_t} h^2_{2,t-1}$ $\rho_{t,s_t} = (1 - \theta_{1,s_t} - \theta_{2,s_t})\rho + \theta_{1,s_t}\rho_{t-1} + \theta_{2,s_t}\phi_{t-1}$ $\phi_{t-1} = \frac{\sum_{j=1}^{2} \varepsilon_{1,t-j}\varepsilon_{2,t-j}}{\sqrt{\left(\sum_{j=1}^{2} \varepsilon^2_{1,t-j}\right)\left(\sum_{j=1}^{2} \varepsilon^2_{2,t-j}\right)}}$, $\varepsilon_{i,t} = \frac{e_{i,t}}{h_{it}}$, $\theta_1, \theta_2 \geq 0$ & $\theta_1 + \theta_2 \leq 1$, *Hedge ratio* $= \frac{H_{t,s_t}(1,2)}{H_{t,s_t}(2,2)}$

A.5. *Random Coefficient*

(M_5):	$\Delta S_t = \beta_0 + \beta_t \Delta F_t + e_t$ $\beta_t = \overline{\beta} + v_t$, *Hedge ratio* $= \overline{\beta}$

A.6. *Cointegration and Error-Correction*

(M_6):	$S_t = a + bF_t + u_t$ $\Delta S_t = \rho u_{t-1} + \beta \Delta F_t + \sum_{i=1}^{m} \delta_i \Delta F_{t-i} + \sum_{j=1}^{n} \theta_i \Delta S_{t-j} + e_j$, *EC Hedge ratio* $= \beta$

A.7. *Error-Correction with GARCH*

(M_7):	$\begin{bmatrix} \Delta \log_e(S_t) \\ \Delta \log_e(F_t) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \begin{bmatrix} \alpha_s(\log_e(S_{t-1}) - \log_e(F_{t-1})) \\ \alpha_f(\log_e(S_{t-1}) - \log_e(F_{t-1})) \end{bmatrix} + \begin{bmatrix} e_{1t} \\ e_{2t} \end{bmatrix}$, $e_t \vert \Omega_{t-1}\ N(0, H_t)$, $\quad H_t = \begin{bmatrix} H_{11,t} & H_{12,t} \\ H_{12,t} & H_{22,t} \end{bmatrix}$ *Hedge ratio* $= h_{t-1} = H_{12,t}/H_{22,t}$

A.8. *Common Stochastic Trend*

(M_8):	$S_t = A_1 P_t + A_2 \tau_t,\ F_t = B_1 P_t + B_2 \tau_t,\ P_t = P_{t-1} + w_t,\ \tau_t = \alpha_1 \tau_{t-1} + v_t, \quad 0 \le \lvert\alpha_1\rvert < 1,$ $\text{Hedge ratio for } k-\text{period investment horizon} = H_J^* = \frac{A_1 B_1 k\sigma_w^2 + 2A_2 B_2 \left(\frac{(1-\alpha^k)}{1-\alpha^2}\right)\sigma_v^2}{B_1^2 k\sigma_w^2 + 2B_2^2 \left(\frac{(1-\alpha^k)}{1-\alpha^2}\right)\sigma_v^2}.$

B. *Optimum Mean–Variance Hedge Ratio*

(M_9):	$\text{Hedge ratio} = h_2 = -\frac{C_f^* F}{C_s S} = -\left[\frac{E(R_f)}{A\sigma_f^2} - \rho\frac{\sigma_s}{\sigma_f}\right]$, where the moments $E[R_f], \sigma_s$ and σ_f are estimated by sample moments

C. *Sharpe Hedge Ratio*

(M_{10}):	$\text{Hedge ratio} = h_3 = -\frac{\left(\frac{\sigma_s}{\sigma_f}\right)\left[\frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)}{E(R_s)-i}\right) - \rho\right]}{\left[1 - \frac{\sigma_s}{\sigma_f}\left(\frac{E(R_f)\rho}{E(R_s)-i}\right)\right]}$, where the moments and correlation are estimated by their sample counterparts

D. *Mean-Gini Coefficient Based Hedge Ratios*

(M_{11}):	The hedge ratio is estimated by numerically minimizing the following mean extended-Gini coefficient, where the cumulative probability distribution function is estimated using the rank function: $\hat{\Gamma}_v(R_h) = -\frac{v}{N}\sum_{i=1}^{N}\left(R_{h,i} - \overline{R}_h\right)\left(\left(1 - G(R_{h,i})\right)^{v-1} - \Theta\right)$
(M_{12}):	The hedge ratio is estimated by numerically solving the first-order condition, where the cumulative probability distribution function is estimated using the rank function
(M_{13}):	The hedge ratio is estimated by numerically solving the first-order condition, where the cumulative probability distribution function is estimated using the kernel-based estimates
(M_{14}):	The hedge ratio is estimated by numerically maximizing the following function: $U(R_h) = E(R_h) - \Gamma_v(R_h)$, where the expected values and the mean extended-Gini coefficient are replaced by their sample counterparts and the cumulative probability distribution function is estimated using the rank function

E. *Generalized Semivariance Based Hedge Ratios*

(M_{15}):	The hedge ratio is estimated by numerically minimizing the following sample generalized hedge ratio: $V_{\delta,\alpha}^{sample}(R_h) = \frac{1}{N}\sum_{i=1}^{N}\left(\delta - R_{h,i}\right)^{\alpha} U\left(\delta - R_{h,i}\right)$, where $U\left(\delta - R_{h,i}\right) = \begin{cases} 1 \text{ for } \delta \ge R_{h,i} \\ 0 \text{ for } \delta < R_{h,i} \end{cases}$
(M_{16}):	The hedge ratio is estimated by numerically maximizing the following function: $U(R_h) = R_h - V_{\delta,\alpha}^{sample}(R_h)$

F. *Minimum Value-at-Risk Hedge Ratio*

(M_{17}):	The hedge ratio is estimated by minimizing the following Value-at-Risk: $VaR(R_h) = Z_\alpha \sigma_h \sqrt{\tau} - E[R_h]\tau$ The resulting hedge ratio is given by $h^{VaR} = \rho\frac{\sigma_s}{\sigma_f} - E[R_f]\frac{\sigma_s}{\sigma_f}\sqrt{\frac{1-\rho^2}{Z_\alpha^2\sigma_f^2 - E[R_f]^2}}$

Appendix 21.3: Monthly Data of S&P500 Index and Its Futures (January 2005–August 2020)

Date	Spot	Futures	C_spot	C_futures
1/31/2005	1181.27	1181.7	−30.65	−32
2/28/2005	1203.6	1204.1	22.33	22.4
3/31/2005	1180.59	1183.9	−23.01	−20.2
4/29/2005	1156.85	1158.5	−23.74	−25.4
5/31/2005	1191.5	1192.3	34.65	33.8
6/30/2005	1191.33	1195.5	−0.17	3.2
7/29/2005	1234.18	1236.8	42.85	41.3
8/31/2005	1220.33	1221.4	−13.85	−15.4
9/30/2005	1228.81	1234.3	8.48	12.9
10/31/2005	1207.01	1209.8	−21.8	−24.5
11/30/2005	1249.48	1251.1	42.47	41.3
12/30/2005	1248.29	1254.8	−1.19	3.7
1/31/2006	1280.08	1283.6	31.79	28.8
2/28/2006	1280.66	1282.4	0.58	−1.2
3/31/2006	1294.83	1303.3	14.17	20.9
4/28/2006	1310.61	1315.9	15.78	12.6
5/31/2006	1270.09	1272.1	−40.52	−43.8
6/30/2006	1270.2	1279.4	0.11	7.3
7/31/2006	1276.66	1281.8	6.46	2.4
8/31/2006	1303.82	1305.6	27.16	23.8
9/29/2006	1335.85	1345.4	32.03	39.8
10/31/2006	1377.94	1383.2	42.09	37.8
11/30/2006	1400.63	1402.9	22.69	19.7
12/29/2006	1418.3	1428.4	17.67	25.5
1/31/2007	1438.24	1443	19.94	14.6
2/28/2007	1406.82	1408.9	−31.42	−34.1
3/30/2007	1420.86	1431.2	14.04	22.3
4/30/2007	1482.37	1488.4	61.51	57.2
5/31/2007	1530.62	1532.9	48.25	44.5
6/29/2007	1503.35	1515.4	−27.27	−17.5
7/31/2007	1455.27	1461.9	−48.08	−53.5
8/31/2007	1473.99	1476.7	18.72	14.8
9/28/2007	1526.75	1538.1	52.76	61.4
10/31/2007	1549.38	1554.9	22.63	16.8
11/30/2007	1481.14	1483.7	−68.24	−71.2
12/31/2007	1468.35	1477.2	−12.79	−6.5
1/31/2008	1378.55	1379.6	−89.8	−97.6
2/29/2008	1330.63	1331.3	−47.92	−48.3
3/31/2008	1322.7	1324	−7.93	−7.3
4/30/2008	1385.59	1386	62.89	62
5/30/2008	1400.38	1400.6	14.79	14.6

(continued)

Date	Spot	Futures	C_spot	C_futures
6/30/2008	1280	1281.1	−120.38	−119.5
7/31/2008	1267.38	1267.1	−12.62	−14
8/29/2008	1282.83	1282.6	15.45	15.5
9/30/2008	1166.36	1169	−116.47	−113.6
10/31/2008	968.75	967.3	−197.61	−201.7
11/28/2008	896.24	895.3	−72.51	−72
12/31/2008	903.25	900.1	7.01	4.8
1/30/2009	825.88	822.5	−77.37	−77.6
2/27/2009	735.09	734.2	−90.79	−88.3
3/31/2009	797.87	794.8	62.78	60.6
4/30/2009	872.81	870	74.94	75.2
5/29/2009	919.14	918.1	46.33	48.1
6/30/2009	919.32	915.5	0.18	−2.6
7/31/2009	987.48	984.4	68.16	68.9
8/31/2009	1020.62	1019.7	33.14	35.3
9/30/2009	1057.08	1052.9	36.46	33.2
10/30/2009	1036.19	1033	−20.89	−19.9
11/30/2009	1095.63	1094.8	59.44	61.8
12/31/2009	1115.1	1110.7	19.47	15.9
1/29/2010	1073.87	1070.4	−41.23	−40.3
2/26/2010	1104.49	1103.4	30.62	33
3/31/2010	1169.43	1165.2	64.94	61.8
4/30/2010	1186.69	1183.4	17.26	18.2
5/31/2010	1089.41	1088.5	−97.28	−94.9
6/30/2010	1030.71	1026.6	−58.7	−61.9
7/30/2010	1101.6	1098.3	70.89	71.7
8/31/2010	1049.33	1048.3	−52.27	−50
9/30/2010	1141.2	1136.7	91.87	88.4
10/29/2010	1183.26	1179.7	42.06	43
11/30/2010	1180.55	1179.6	−2.71	−0.1
12/31/2010	1257.64	1253	77.09	73.4
1/31/2011	1286.12	1282.4	28.48	29.4
2/28/2011	1327.22	1326.1	41.1	43.7
3/31/2011	1325.83	1321	−1.39	−5.1
4/29/2011	1363.61	1359.7	37.78	38.7
5/31/2011	1345.2	1343.9	−18.41	−15.8
6/30/2011	1320.64	1315.5	−24.56	−28.4
7/29/2011	1292.28	1288.4	−28.36	−27.1
8/31/2011	1218.89	1217.7	−73.39	−70.7
9/30/2011	1131.42	1126	−87.47	−91.7
10/31/2011	1253.3	1249.3	121.88	123.3
11/30/2011	1246.96	1246	−6.34	−3.3
12/30/2011	1257.6	1252.6	10.64	6.6
1/31/2012	1312.41	1308.2	54.81	55.6

(continued)

Date	Spot	Futures	C_spot	C_futures
2/29/2012	1365.68	1364.4	53.27	56.2
3/30/2012	1408.47	1403.2	42.79	38.8
4/30/2012	1397.91	1393.6	−10.56	−9.6
5/31/2012	1310.33	1309.2	−87.58	−84.4
6/29/2012	1362.16	1356.4	51.83	47.2
7/31/2012	1379.32	1374.6	17.16	18.2
8/31/2012	1406.58	1405.1	27.26	30.5
9/28/2012	1440.67	1434.2	34.09	29.1
10/31/2012	1412.16	1406.8	−28.51	−27.4
11/30/2012	1416.18	1414.4	4.02	7.6
12/31/2012	1426.19	1420.1	10.01	5.7
1/31/2013	1498.11	1493.3	71.92	73.2
2/28/2013	1514.68	1513.3	16.57	20
3/29/2013	1569.19	1562.7	54.51	49.4
4/30/2013	1597.57	1592.2	28.38	29.5
5/31/2013	1630.74	1629	33.17	36.8
6/28/2013	1606.28	1599.3	−24.46	−29.7
7/31/2013	1685.73	1680.5	79.45	81.2
8/30/2013	1632.97	1631.3	−52.76	−49.2
9/30/2013	1681.55	1674.3	48.58	43
10/31/2013	1756.54	1751	74.99	76.7
11/29/2013	1805.81	1804.1	49.27	53.1
12/31/2013	1848.36	1841.1	42.55	37
1/31/2014	1782.59	1776.6	−65.77	−64.5
2/28/2014	1859.45	1857.6	76.86	81
3/31/2014	1872.34	1864.6	12.89	7
4/30/2014	1883.95	1877.9	11.61	13.3
5/30/2014	1923.57	1921.5	39.62	43.6
6/30/2014	1960.23	1952.4	36.66	30.9
7/31/2014	1930.67	1924.8	−29.56	−27.6
8/29/2014	2003.37	2001.4	72.7	76.6
9/30/2014	1972.29	1965.5	−31.08	−35.9
10/31/2014	2018.05	2011.4	45.76	45.9
11/28/2014	2067.56	2066.3	49.51	54.9
12/31/2014	2058.9	2052.4	−8.66	−13.9
1/30/2015	1994.99	1988.4	−63.91	−64
2/27/2015	2104.5	2102.8	109.51	114.4
3/31/2015	2067.89	2060.8	−36.61	−42
4/30/2015	2085.51	2078.9	17.62	18.1
5/29/2015	2107.39	2106	21.88	27.1
6/30/2015	2063.11	2054.4	−44.28	−51.6
7/31/2015	2103.84	2098.4	40.73	44
8/31/2015	1972.18	1969.2	−131.66	−129.2
9/30/2015	1920.03	1908.7	−52.15	−60.5

(continued)

Date	Spot	Futures	C_spot	C_futures
10/30/2015	2079.36	2073.7	159.33	165
11/30/2015	2080.41	2079.8	1.05	6.1
12/31/2015	2043.94	2035.4	−36.47	−44.4
1/29/2016	1940.24	1930.1	−103.7	−105.3
2/29/2016	1932.23	1929.5	−8.01	−0.6
3/31/2016	2059.74	2051.5	127.51	122
4/29/2016	2065.3	2059.1	5.56	7.6
5/31/2016	2096.96	2094.9	31.66	35.8
6/30/2016	2098.86	2090.2	1.9	−4.7
7/29/2016	2173.6	2168.2	74.74	78
8/31/2016	2170.95	2169.5	−2.65	1.3
9/30/2016	2168.27	2160.4	−2.68	−9.1
10/31/2016	2126.15	2120.1	−42.12	−40.3
11/30/2016	2198.81	2198.8	72.66	78.7
12/30/2016	2238.83	2236.2	40.02	37.4
1/31/2017	2278.87	2274.5	40.04	38.3
2/28/2017	2363.64	2362.8	84.77	88.3
3/31/2017	2362.72	2359.2	−0.92	−3.6
4/28/2017	2384.2	2380.5	21.48	21.3
5/31/2017	2411.8	2411.1	27.6	30.6
6/30/2017	2423.41	2420.9	11.61	9.8
7/31/2017	2470.3	2468	46.89	47.1
8/31/2017	2471.65	2470.1	1.35	2.1
9/29/2017	2519.36	2516.1	47.71	46
10/31/2017	2575.26	2572.7	55.9	56.6
11/30/2017	2647.58	2647.9	72.32	75.2
12/29/2017	2673.61	2676	26.03	28.1
1/31/2018	2823.81	2825.8	150.2	149.8
2/28/2018	2713.83	2714.4	−109.98	−111.4
3/30/2018	2640.87	2643	−72.96	−71.4
4/30/2018	2648.05	2647	7.18	4
5/31/2018	2705.27	2705.5	57.22	58.5
6/29/2018	2718.37	2721.6	13.1	16.1
7/31/2018	2816.29	2817.1	97.92	95.5
8/31/2018	2901.52	2902.1	85.23	85
9/28/2018	2913.98	2919	12.46	16.9
10/31/2018	2711.74	2711.1	−202.24	−207.9
11/30/2018	2760.17	2758.3	48.43	47.2
12/31/2018	2506.85	2505.2	−253.32	−253.1
1/31/2019	2704.1	2704.5	197.25	199.3
2/28/2019	2784.49	2784.7	80.39	80.2
3/29/2019	2834.4	2837.8	49.91	53.1
4/30/2019	2945.83	2948.5	111.43	110.7
5/31/2019	2752.06	2752.6	−193.77	−195.9

(continued)

Date	Spot	Futures	C_spot	C_futures
6/28/2019	2941.76	2944.2	189.7	191.6
7/31/2019	2980.38	2982.3	38.62	38.1
8/30/2019	2926.46	2924.8	−53.92	−57.5
9/30/2019	2976.74	2978.5	50.28	53.7
10/31/2019	3037.56	3035.8	60.82	57.3
11/29/2019	3140.98	3143.7	103.42	107.9
12/31/2019	3230.78	3231.1	89.8	87.4
1/31/2020	3225.52	3224	−5.26	−7.1
2/28/2020	2954.22	2951.1	−271.3	−272.9
3/31/2020	2584.59	2569.7	−369.63	−381.4
4/30/2020	2912.43	2902.4	327.84	332.7
5/29/2020	3044.31	3042	131.88	139.6
6/30/2020	3100.29	3090.2	55.98	48.2
7/31/2020	3271.12	3263.5	170.83	173.3
8/31/2020	3500.31	3498.9	229.19	235.4

Appendix 21.4: Applications of R Language in Estimating the Optimal Hedge Ratio

In this appendix, we show the estimation procedure on how to apply OLS, GARCH, and CECM models to estimate optimal hedge ratios through R language. R language is a high-level computer language that is designed for statistics and graphics. Compared to alternatives, SAS, Matlab or Stata, R is completely free. Another benefit is that it is open source. Users could head to http://cran.r-project.org/ to download and install R language. Based upon monthly S&P 500 index and its futures as presented in Appendix 21.1, the estimation procedures of applying R language to estimate hedge ratio are provided as follows.

First, we use OLS method in term of Eq. (74.11) to estimate minimum variance hedge ratio. By using linear model (lm) function in R language, we obtain the following program code.

```
SP500= read.csv(file="SP500.csv")
OLS.fit <- lm(C_spot~C_futures, data=SP500)
summary(OLS.fit)
```

Next, we apply a conventional regression model with an AR(2)-GARCH(1, 1) error terms to estimate minimum variance hedge ratio. By using rugarch package in R language, we obtain the following program.

Third, we apply the ECM model to estimate minimum variance hedge ratio. We begin by applying an augmented Dickey-Fuller (ADF) test for the presence of unit roots. The Phillips and Ouliaris (1990) residual cointegration test is applied to examine the presence of cointegration. Finally, the minimum variance hedge ratio is estimated by the error correction model. By using tseries package in R language, we obtain the following program.

```
library(rugarch)
fit.spec <- ugarchspec(
variance.model = list(model = "sGARCH",
garchOrder = c(1, 1)),
mean.model = list(armaOrder = c(2, 0),include.mean = TR
                  external.regressors= cbind(SP500$C_futures)),
                  distribution.model = "norm")

GARCH.fit <- ugarchfit(data = cbind(SP500$C_spot),
spec = fit.spec)
GARCH.fit
```

```
library(tseries)
# Augmented Dickey-Fuller Test
# Level data
adf.test(SP500$SPOT, k = 1)
adf.test(SP500$FUTURES, k = 1)
# First-order differenced data
adf.test(diff(SP500$SPOT), k = 1)
adf.test(diff(SP500$FUTURES), k = 1)

# Phillips and Ouliaris (1990) residual cointegration test
po.test(cbind(SP500$FUTURES,SP500$SPOT))

# Engle-Granger two-step procedure
## 1.Estimate cointegrating relationship
reg <- lm(SPOT~FUTURES, data=SP500)
## 2. Compute error term
Resid <- reg$resid
# Estimate optimal hedge ratio using the error correction model
ECM.fit <-lm(diff(SPOT) ~ -1 + diff(FUTURES) + Resid[-1], data=SP500)
summary(ECM.fit)
```

References

Baillie, R.T., & Myers, R.J. (1991). Bivariate Garch estimation of the optimal commodity futures hedge. *Journal of Applied Econometrics*, 6, 109–124.

Bawa, V.S. (1978). Safety-first, stochastic dominance, and optimal portfolio choice. *Journal of Financial and Quantitative Analysis*, 13, 255–271.

Benet, B.A. (1992). Hedge period length and ex-ante futures hedging effectiveness: the case of foreign-exchange risk cross hedges. *Journal of Futures Markets*, 12, 163–175.

Cecchetti, S.G., Cumby, R.E., & Figlewski, S. (1988). Estimation of the optimal futures hedge. *Review of Economics and Statistics*, 70, 623–630.

Chen, S.S., Lee, C.F., & Shrestha, K. (2001). On a mean-generalized semivariance approach to determining the hedge ratio. *Journal of Futures Markets*, 21, 581–598.

Cheung, C.S., Kwan, C.C.Y., & Yip, P.C.Y. (1990). The hedging effectiveness of options and futures: a mean-Gini approach. *Journal of Futures Markets*, 10, 61–74.

Chou, W.L., Fan, K.K., & Lee, C.F. (1996). Hedging with the Nikkei index futures: the conventional model versus the error correction model. *Quarterly Review of Economics and Finance*, 36, 495–505.

Crum, R.L., Laughhunn, D.L., & Payne, J.W. (1981). Risk-seeking behavior and its implications for financial models. *Financial Management*, 10, 20–27.

D'Agostino, R.B. (1971). An omnibus test of normality for moderate and large size samples. *Biometrika*, 58, 341–348.

De Jong, A., De Roon, F., & Veld, C. (1997). Out-of-sample hedging effectiveness of currency futures for alternative models and hedging strategies. *Journal of Futures Markets*, 17, 817–837.

Dickey, D.A., & Fuller, W.A. (1981). Likelihood ratio statistics for autoregressive time series with a unit root. *Econometrica*, 49, 1057–1072.

Ederington, L.H. (1979). The hedging performance of the new futures markets. *Journal of Finance*, 34, 157–170.

Engle, R.F., & Granger, C.W. (1987). Co-integration and error correction: representation, estimation and testing. *Econometrica*, 55, 251–276.

Fishburn, P.C. (1977). Mean-risk analysis with risk associated with below-target returns. *American Economic Review*, 67, 116–126.

Geppert, J.M. (1995). A statistical model for the relationship between futures contract hedging effectiveness and investment horizon length. *Journal of Futures Markets*, 15, 507–536.

Ghosh, A. (1993). Hedging with stock index futures: estimation and forecasting with error correction model. *Journal of Futures Markets*, 13, 743–752.

Grammatikos, T., & Saunders, A. (1983). Stability and the hedging performance of foreign currency futures. *Journal of Futures Markets*, 3, 295–305.

Howard, C.T., & D'Antonio, L.J. (1984). A risk-return measure of hedging effectiveness. *Journal of Financial and Quantitative Analysis*, 19, 101–112.

Hsin, C.W., Kuo, J., & Lee, C.F. (1994). A new measure to compare the hedging effectiveness of foreign currency futures versus options. *Journal of Futures Markets*, 14, 685–707.

Hung, J.C., Chiu, C.L. & Lee, M.C. (2006). Hedging with zero-value at risk hedge ratio, *Applied Financial Economics,* 16, 259–269.

Hylleberg, S., & Mizon, G.E. (1989). Cointegration and error correction mechanisms. *Economic Journal*, 99, 113–125.

Jarque, C.M., & Bera, A.K. (1987). A test for normality of observations and regression residuals. *International Statistical Review*, 55, 163–172.

Johansen, S., & Juselius, K. (1990). Maximum likelihood estimation and inference on cointegration—with applications to the demand for money. *Oxford Bulletin of Economics and Statistics*, 52, 169–210.

Johnson, L.L. (1960). The theory of hedging and speculation in commodity futures. *Review of Economic Studies*, 27, 139–151.

Junkus, J.C., & Lee, C.F. (1985). Use of three index futures in hedging decisions. *Journal of Futures Markets*, 5, 201–222.

Kolb, R.W., & Okunev, J. (1992). An empirical evaluation of the extended mean-Gini coefficient for futures hedging. *Journal of Futures Markets*, 12, 177–186.

Kolb, R.W., & Okunev, J. (1993). Utility maximizing hedge ratios in the extended mean Gini framework. *Journal of Futures Markets*, 13, 597–609.

Kroner, K.F., & Sultan, J. (1993). Time-varying distributions and dynamic hedging with foreign currency futures. *Journal of Financial and Quantitative Analysis*, 28, 535–551.

Lee, H.T. & Yoder J. (2007). Optimal hedging with a regime-switching time-varying correlation GARCH model. *Journal of Futures Markets,* 27, 495–516.

Lee, C.F., Bubnys, E.L., & Lin, Y. (1987). Stock index futures hedge ratios: test on horizon effects and functional form. *Advances in Futures and Options Research*, 2, 291–311.

Lence, S. H. (1995). The economic value of minimum-variance hedges. *American Journal of Agricultural Economics*, 77, 353–364.

Lence, S. H. (1996). Relaxing the assumptions of minimum variance hedging. *Journal of Agricultural and Resource Economics*, 21, 39–55.

Lien, D. (1996). The effect of the cointegration relationship on futures hedging:A note. The Journal of Futures Markets, 16, 773–780.

Lien, Donald. "Cointegration and the optimal hedge ratio: the general case." The Quarterly review of economics and finance 44.5 (2004): 654–658.

Lien, D., & Luo, X. (1993a). Estimating the extended mean-Gini coefficient for futures hedging. *Journal of Futures Markets*, 13, 665–676.

Lien, D., & Luo, X. (1993b). Estimating multiperiod hedge ratios in cointegrated markets. *Journal of Futures Markets*, 13, 909–920.

Lien, D., & Shaffer, D.R. (1999). Note on estimating the minimum extended Gini hedge ratio. *Journal of Futures Markets*, 19, 101–113.

Lien, D. & Shrestha, K. (2007). An empirical analysis of the relationship between hedge ratio and hedging horizon using wavelet analysis. *Journal of Futures Markets,* 27, 127–150.

Lien, D. & Shrestha, K. (2010). Estimating optimal hedge ratio: a multivariate skew-normal distribution. *Applied Financial Economics,* 20, 627–636.

Lien, D., & Tse, Y.K. (1998). Hedging time-varying downside risk. *Journal of Futures Markets*, 18, 705–722.

Lien, D., & Tse, Y.K. (2000). Hedging downside risk with futures contracts. *Applied Financial Economics*, 10, 163–170.

Malliaris, A.G., & Urrutia, J.L. (1991). The impact of the lengths of estimation periods and hedging horizons on the effectiveness of a hedge: evidence from foreign currency futures. *Journal of Futures Markets*, 3, 271–289.

Myers, R.J., & Thompson, S.R. (1989) Generalized optimal hedge ratio estimation. *American Journal of Agricultural Economics*, 71, 858–868.

Osterwald-Lenum, M. (1992). A note with quantiles of the asymptotic distribution of the maximum likelihood cointegration rank test statistics. *Oxford Bulletin of Economics and Statistics*, 54, 461–471.

Phillips, P.C.B., & Perron, P. (1988). Testing unit roots in time series regression. *Biometrika*, 75, 335–46.

Phillips, Peter CB, and Sam Ouliaris. "Asymptotic properties of residual based tests for cointegration." Econometrica: journal of the Econometric Society (1990): 165–193.

Rutledge, D.J.S. (1972). Hedgers' demand for futures contracts: a theoretical framework with applications to the United States soybean complex. *Food Research Institute Studies*, 11, 237–256.

Sephton, P.S. (1993a). Hedging wheat and canola at the Winnipeg commodity exchange.*Applied Financial Economics*, 3, 67–72.

Sephton, P.S. (1993b). Optimal hedge ratios at the Winnipeg commodity exchange. *Canadian Journal of Economics*, 26, 175–193.

Shalit, H. (1995). Mean-Gini hedging in futures markets. *Journal of Futures Markets*, 15, 617–635.

Stock, J.H., & Watson, M.W. (1988). Testing for common trends. *Journal of the American Statistical Association*, 83, 1097–1107.

Working, H. (1953). Hedging reconsidered. *Journal of Farm Economics*, 35, 544–561.

22 Application of Simultaneous Equation in Finance Research: Methods and Empirical Results

By Fu-Lai Lin, Da-Yeh University, Taiwan

22.1 Introduction

Simultaneous equation models have been widely adopted in finance literature. It is suggested that the relation, particularly the interaction, among corporate decisions, firm characteristics, and firm performance should be contemporaneously determined. In Chapter 4 of Lee et al. (2019), they discuss the concept of a simultaneous equation system, including a basic definition, specification, identification, and estimation methods. The applications of such a system in finance research are also provided. Some papers study the interrelationship among a firm's capital structure, investment, and payout policy (e.g., Grabowski and Mueller 1972; Higgins 1972; Fama 1974; McCabe 1979; Peterson and Benesh 1983; Switzer 1984; Fama and French 2002; Gugler 2003; MacKay and Phillips 2005; Aggarwal and Kyaw 2010; Harford et al. 2014), given the fact that these decisions are simultaneously determined. Moreover, the interrelationship between board composition (or ownership) and firm performance is often investigated in simultaneous equations (e.g., Loderer and Martin 1997; Demsetz and Villalonga 2001; Bhagat and Black 2002; Prevost et al. 2002; Woidtke 2002; Boone et al. 2007; Fich and Shivdasani 2007; Ferreira and Matos 2008; Ye 2012). In addition to the above-mentioned studies, many other issues of research also apply the simultaneous equations model in their papers because firm decisions, characteristics, and performance may be jointly determined.

Empirically, the utilization of ordinary least squares (OLS) estimation on simultaneous equations yields biased and inconsistent estimates since the assumption of no correlation between the regressors and the disturbance terms is violated. The instrumental variable (IV) class estimators, such as two-stage least squares (2SLS) and three-stage least squares (3SLS) estimations, are commonly used to deal with this endogeneity problem. Wang (2015) reviews the instrumental variables approach to correct for endogeneity in finance. The GMM estimator proposed by Hansen (1982) is also based on orthogonality conditions and provides an alternative solution. In contrast to traditional IV class estimators, the GMM estimator uses a weighting matrix taking account of temporal dependence, heteroskedasticity, or autocorrelation. Although many finance studies acknowledge the existence of endogeneity problems caused by omitted variables, measurement errors, and/or simultaneity, few of them provide the reason for the selected estimation methods (e.g., 2SLS, 3SLS, and/or GMM). Lee and Lee (2020) have several chapters, which discuss how different methodologies can be applied to the topics of finance and accounting research. In fact, different estimation methods for the simultaneous equations are not perfect substitutions under different assumptions. Thus, we need a detailed examination of which method is best for the model selection by some relevant statistical tests. In addition, the instrumental variables are usually chosen arbitrarily in finance studies. Thus, we compare the differences among 2SLS, 3SLS, and GMM methods under different conditions and present the related test for the validity of instruments.

The chapter proceeds as follows. Section 22.2 presents the literature reviews about applications of the simultaneous equations model in capital structure decisions. Section 22.3 discusses the 2SLS, 3SLS, and GMM methods applied in estimating simultaneous equations models. Section 22.4 illustrates the application of simultaneous equations to investigate the interaction among investment, financing, and dividend decisions. Conclusions are presented in Sect. 22.5.

22.2 Literature Review

The simultaneous equations models are applied in the capital structure decisions. Harvey et al. (2004) address the potentially endogenous relation among debt, ownership structure, and firm value by estimating a 3SLS regression model. They find that debt can mitigate the agency and information problem for emerging market firms. Billett et al. (2007) suggest that the corporate financial policies, which include the choices of leverage, debt maturity, and covenants, are

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_22

jointly determined, and thereby apply GMM in the estimation of simultaneous equations. They find that covenants can mitigate the agency costs of debt for high-growth firms. Berger and Bonaccorsi di Patti (2006) argue that an agency costs hypothesis predicts that leverage affects firm performance, yet firm performance also affects the choice of capital structure. To address this problem of reverse causality between firm performance and capital structure, they use 2SLS to estimate the simultaneous equations model. They also estimate by 3SLS and do not change the main findings that higher leverage is associated with higher profit efficiency. In the similar reason, Ruland and Zhou (2005) consider the potential endogeneity between firms' excess value and leverage and find that compared to specialized firms, the values of diversified firms increase with leverage by using 2SLS. Aggarwal and Kyaw (2010) recognize the interdependence between capital structure and dividend payout policy by using 2SLS and find that multinational companies have significantly lower debt ratios and pay higher dividends than domestic companies. MacKay and Phillips (2005) use GMM and find that financial structure, technology, and risk are jointly determined within industries.

In addition, simultaneous equations models are applied in studies considering the interrelationship among a firm's major policies. Higgins (1972), Fama (1974), and Morgan and Saint-Pierre (1978) investigate the relationship between investment decision and dividend decision. Grabowski and Mueller (1972) examine the interrelationship among investment, dividend, and research and development (R&D). Fama and French (2002) consider the interaction between dividend and financing decisions. Dhrymes and Kurz (1967), McDonald et al. (1975), McCabe (1979), Peterson and Benesh (1983), and Switzer (1984) argue that investment decision is related to financing decision and dividend decision. Lee et al. (2016) empirically investigate the interrelationship among investment, financing, and dividend decisions using the GMM method. Harford et al. (2014) consider the interdependence of a firm's cash holdings and the maturity of its debt by using a simultaneous equation framework and performing a 2SLS estimation. Moreover, Lee and Lin (2020) theoretically investigate how the unknown variance of measurement error estimation in dividend and investment decisions can be identified by the over-identified information in a simultaneous equation system.

The above literature review of finance shows many studies acknowledge the existence of endogeneity problems caused by omitted variables, measurement errors, and/or simultaneity, however, seldom studies provide the reason for the selected estimation method (e.g., 2SLS, 3SLS, and/or GMM). In fact, different methods of estimating the simultaneous equations have different assumptions and thereby cause them to be not perfect substitutions. For example, the parameters estimated by 3SLS, which is a full information estimation method, are asymptotically more efficient than the limited information method (e.g., 2SLS), although 3SLS is vulnerable to model specification errors. Thus, a comprehensive analysis of which method is best for the model selection would require some contemplation and relevant statistical tests. Moreover, the instrumental variables used in finance studies are usually chosen arbitrarily. Thus, in Sect. 22.3, we will discuss the difference among 2SLS, 3SLS, and GMM methods, present the applicable method under different conditions, and also present the related test for the validity of instruments.

22.3 Methodology

In this section, we review the discusses the 2SLS, 3SLS, and GMM methods applied in estimating simultaneous equations models. Suppose that a set of observations on a variable y is drawn independently from probability distribution depends on an unknown vector of parameters $\boldsymbol{\beta}$ of interest. One general approach for estimating parameters $\boldsymbol{\beta}$ is based on maximum likelihood (ML) estimation. The intuition behind ML estimation is to specify a probability distribution for it, and then find an estimate $\hat{\beta}$ in which the data would be most likely to have been observed. The drawback with maximum likelihood methods is that we have to specify a full probability distribution for the data. Here, we introduce an alternative approach for parameter estimation known as the generalized method of moments (GMM). The GMM estimation is formalized by Hansen (1982) and is one of the most widely used methods of estimation in economics and finance. In contrast to ML estimation, the GMM estimation only requires the specification of certain moment conditions rather than the form of the likelihood function.

The idea behind GMM estimation is to choose a parameter estimate so as to make the sample moment conditions as close as possible to the population moment of zero according to the measure of Euclidean distance. The GMM estimation proposes a weighting matrix reflecting the importance given to matching each of the moments. The alternative weighting matrix is associated with the alternative estimator. Many standard estimators, including ordinary least squares (OLS), method of moments (MM), ML, instrumental variable (IV), two-stage least squares (2SLS), and three-stage least squares (3SLS) can be seen as special cases of GMM estimators. For example, when the number of moment conditions and unknown parameters is the same, solving the quadratic criterion yields the GMM estimator, which is the same as MM estimator that sets the sample moment condition exactly equal to zero. The weighting matrix does not matter in this case. In particular, in models for which there are more

moment conditions than model parameters, GMM estimation provides a straightforward way to test the specification of the proposed model. This is an important feature that is unique to GMM estimation.

Recently, the endogeneity concern has received much attention in empirical corporate finance research. There are at least three generally recognized sources of endogeneity: omitted explanatory variables, simultaneity bias, and errors in variables. Whenever there is endogeneity, the application of OLS estimation yields biased and inconsistent estimates. In literature, the IV methods are commonly used to deal with this endogeneity problem. The basic motivation for the IV method is to deal with equations that exhibited both simultaneity and measurement errors in exogenous variables. The idea behind IV estimation is to select suitable instruments that are orthogonal to the disturbance while sufficiently correlated with the regressors. The IV estimator makes the linear combinations of sample orthogonality conditions close to zeros. The GMM estimator proposed by Hansen (1982) is also based on orthogonality conditions and provides an alternative solution. Hansen's (1982) GMM estimator generalizes Sargan's (1958, 1959) linear and nonlinear IV estimators based on optimal weighting matrix for the moment conditions. In contrast to traditional IV class estimators such as 2SLS and 3SLS estimators, the GMM estimator uses a weighting matrix considering temporal dependence, heteroskedasticity, or autocorrelation.

Here, we review the application of GMM estimation in the linear regression model and further survey the GMM estimation applied in estimating simultaneous equations models.

22.3.1 Application of GMM Estimation in the Linear Regression Model

Consider the following linear regression model:

$$y_t = \mathbf{x}_t\boldsymbol{\beta} + \varepsilon_t, \quad t = 1, \ldots, T \tag{22.1}$$

where y is the endogenous variable, $\mathbf{x}_t$ is a $1 \times K$ regressor vector that includes constant term, and ε_t is the error term. Here, $\boldsymbol{\beta}$ denotes a $K \times 1$ parameter vector of interest. The critical assumption made for the OLS estimation is that the disturbance ε_t is uncorrelated with the regressors $\mathbf{x}_t$, $E(\mathbf{x}_t'\varepsilon_t) = \mathbf{0}$. The T observations in the model (22.1) can be written in matrix form as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{22.2}$$

here $\mathbf{Y}$ denotes the $T \times 1$ data vector for the endogenous variable and $\mathbf{X}$ is a $T \times K$ data matrix for all regressors. In this matrix notation, the OLS estimator for $\boldsymbol{\beta}$ is as follows:

$$\widehat{\boldsymbol{\beta}}_{\mathrm{OLS}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} \tag{22.3}$$

If the disturbance term is correlated with at least some components of regressors, we say that the regressors are endogenous. Whenever there is endogeneity, the application of ordinary least squares (OLS) estimation to equation (22.2) yields biased and inconsistent estimates. The instrumental variable (IV) methods are commonly used to deal with this endogeneity problem. In a typical IV application, the researcher first chooses a set of variables as instruments that are exogenous and applies two-stage least squares (2SLS) methods to estimate the parameter $\boldsymbol{\beta}$. A good instrument should be highly correlated with the endogenous regressors while uncorrelated with the disturbance in the structural equation. The IV estimator for $\boldsymbol{\beta}$ can be regarded as the solution to following moment conditions of the form

$$E[\mathbf{z}_t'\varepsilon_t] = E[\mathbf{z}_t'(y_t - \mathbf{x}_t'\boldsymbol{\beta})] = \mathbf{0} \tag{22.4}$$

where $\mathbf{z}_t$ is a $1 \times L$ vector of instrumental variables which are uncorrelated with disturbance but correlated with $\mathbf{x}_t$, and the sample moment conditions are

$$\frac{1}{T}\sum_{t=1}^{T}\mathbf{z}_t'(y_t - \mathbf{x}_t\widehat{\boldsymbol{\beta}}) = \mathbf{0} \tag{22.5}$$

Assume $\mathbf{Z}$ denotes a $T \times L$ instrument matrix. If the system is just identified ($L = K$) and $\mathbf{Z}'\mathbf{X}$ is invertible, the system of sample moment conditions in (22.5) has a unique solution. We have an IV estimator $\widehat{\boldsymbol{\beta}}_{IV}$ as follows:

$$\widehat{\boldsymbol{\beta}}_{\mathrm{IV}} = (\mathbf{Z}'\mathbf{X})^{-1}\mathbf{Z}'\mathbf{Y} \tag{22.6}$$

Suppose that the number of instruments exceeds the number of explanatory variables ($L > K$), the system in (22.5) is over-identified. Then there the question arises that how to select or combine more than enough moment conditions to get K equations. Here, the two-stage least squares (2SLS) estimator which is the most efficient IV estimator out of all possible linear combinations of the valid instruments under homoscedasticity, is employed in this case. The first stage of the 2SLS estimator is regressing each endogenous regressor on all instruments to get its OLS prediction, expressed in matrix notation as $\hat{\boldsymbol{X}} = \boldsymbol{Z}(\boldsymbol{Z}'\boldsymbol{Z})^{-1}\boldsymbol{Z}$. The second stage is regressing the dependent variable on $\hat{\boldsymbol{X}}$ to obtain the 2SLS estimator for $\boldsymbol{\beta}$, $\hat{\beta}_{2SLS} = \left(\hat{\boldsymbol{X}}'\hat{\boldsymbol{X}}\right)^{-1}\hat{\boldsymbol{X}}'\boldsymbol{Y}$. Substitute

$\mathbf{Z}(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{X}$ for $\hat{X}$, the 2SLS estimator $\widehat{\boldsymbol{\beta}}_{2SLS}$ can be written as

$$\widehat{\boldsymbol{\beta}}_{2SLS} = \left[(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{X}\right]^{-1}(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{Y} \quad (22.7)$$

Hansen (1982)'s (GMM) estimation provides an alternative approach for parameter estimation in this over-identified model. The idea behind GMM estimation is to choose a parameter estimate to make the sample moment conditions in (22.5) as close as possible to the population moment of zero. The GMM estimator is constructed based on the moment conditions (22.5) and minimizes the following quadratic function:

$$\left[\sum_{t=1}^{T}\mathbf{z}_t'(y_t - \mathbf{x}_t\boldsymbol{\beta})\right]' \mathbf{W}_T^{-1}\left[\sum_{t=1}^{T}\mathbf{z}_t'(y_t - \mathbf{x}_t\boldsymbol{\beta})\right] \quad (22.8)$$

for some $L \times L$ positive definite weighting matrix $\mathbf{W}_T^{-1}$. If the system is just identified and $\mathbf{Z}'\mathbf{X}$ is invertible, we can solve for the parameter vector which makes the sample moment conditions of zero in (22.5). In this case, the weighting matrix is irrelevant. The corresponding GMM estimator is just as the IV estimator $\widehat{\boldsymbol{\beta}}_{IV}$ in (22.6). If the model is over-identified, we cannot set the sample moment conditions in (22.5) exactly equal to zero. The GMM estimator for $\boldsymbol{\beta}$ can be obtained by minimizing the quadratic function in (22.8) as follows:

$$\widehat{\boldsymbol{\beta}}_{GMM} = \left[(\mathbf{X}'\mathbf{Z})\mathbf{W}_T^{-1}\mathbf{Z}'\mathbf{X}\right]^{-1}(\mathbf{X}'\mathbf{Z})\mathbf{W}_T^{-1}\mathbf{Z}'\mathbf{Y} \quad (22.9)$$

Alternative weighting matrices $\mathbf{W}_T$ are associated with alternative estimators. The question in GMM estimation is which $\mathbf{W}_T$ to use in (22.8). Hansen (1982) shows that the optimal weighting matrix $\mathbf{W}_T$ for the resulting estimator is

$$\mathbf{W}_T = \mathrm{Var}[\mathbf{z}'\varepsilon] \equiv E[\mathbf{z}\mathbf{z}'\varepsilon^2] = E_{\mathbf{z}}\{\mathbf{z}\mathbf{z}'[E(\varepsilon^2|\mathbf{z})]\} \quad (22.10)$$

Under conditional homoscedasticity $E(\varepsilon^2|\mathbf{z}) = \sigma^2$, the optimal weighting matrix in which case is

$$\mathbf{W}_T = \left(\frac{\mathbf{Z}'\mathbf{Z}}{T}\right)\sigma^2 \quad (22.11)$$

Hence, any scalar in $\mathbf{W}_N$ will be canceled in this case yields

$$\widehat{\boldsymbol{\beta}}_{GMM} = \left[(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{X}\right]^{-1}(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{Y} \quad (22.12)$$

Thus, the GMM estimator is simply the 2SLS estimator under conditional homoscedasticity. However, if the conditional variance of ε_t the given $\mathbf{z}_t$ depends on $\mathbf{z}_t$, the optimal weighting matrix $\mathbf{W}_T$ should be estimated by

$$\mathbf{W}_T = \frac{1}{T}\sum_{t=1}^{T}\mathbf{z}_t'\mathbf{z}_t\hat{\varepsilon}_t^2 = \frac{1}{T}\mathbf{Z}'\mathbf{D}\mathbf{Z} \quad (22.13)$$

where $\hat{\varepsilon}_t$ is sample residuals and $\mathbf{D} = \mathrm{diag}(\hat{\varepsilon}_1^2, \ldots, \hat{\varepsilon}_T^2)$. Here, we can apply the two-stage least-squares (2SLS) estimator in equation (22.7) to obtain the sample residuals by $\hat{\varepsilon}_t = y_t - \mathbf{x}_t\widehat{\boldsymbol{\beta}}_{2SLS}$, then the GMM estimator $\widehat{\boldsymbol{\beta}}_{GMM}$ is

$$\widehat{\boldsymbol{\beta}}_{GMM} = \left[(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{X}\right]^{-1}(\mathbf{X}'\mathbf{Z})(\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{Y} \quad (22.14)$$

Note that the GMM estimator is obtained by the two-step procedure under heteroskedasticity. First, use the 2SLS estimator as an initial estimator since it is consistent to get the residuals by $\hat{\varepsilon}_t = y_t - \mathbf{x}_t\widehat{\boldsymbol{\beta}}_{2SLS}$. Then substitute $\sum_{t=1}^{T}\mathbf{z}_t'\mathbf{z}_t\hat{\varepsilon}_t^2$ into $\mathbf{W}_T$ as the weighting matrix to obtain the GMM estimator. For this reason, the GMM estimator is sometimes called a two-stage instrumental variables estimator.

22.3.2 Applications of GMM Estimation in the Simultaneous Equations Model

Consider the following linear simultaneous equations model:

$$\begin{aligned}
y_{1t} &= \delta_{12}y_{2t} + \delta_{13}y_{3t} + \cdots + \delta_{1J}y_{Jt} + x_{1t}\gamma_1 + \varepsilon_{1t}\\
y_{2t} &= \delta_{21}y_{1t} + \delta_{23}y_{3t} + \cdots + \delta_{2J}y_{Jt} + x_{2t}\gamma_2 + \varepsilon_{2t}\\
&\vdots\\
y_{Jt} &= \delta_{J1}y_{1t} + \delta_{J2}y_{2t} + \cdots + \delta_{J(J-1)}y_{(J-1)t} + x_{Jt}\gamma_J + \varepsilon_{Jt}
\end{aligned} \quad (22.15)$$

Here t=1,2,...,T. Define that $\mathrm{y_t}=[\mathrm{y_{1t}}\ \mathrm{y_{2t}}\cdots\ \mathrm{y_{Jt}}]'$ is a $J\times1$ vector for endogenous variables, $\mathrm{x_t}=[\mathrm{x_{1t}}\ \mathrm{x_{2t}}\cdots\ \mathrm{x_{Jt}}]$ is a vector for all exogenous variables in this system includes the constant term. $\varepsilon_t = [\varepsilon_{1t}\quad \varepsilon_{2t}\cdots\varepsilon_{Jt}]'$ is a $J\times1$ vector for the disturbances. Here, $\boldsymbol{\delta}$ and $\boldsymbol{\gamma}$ are the parameters matrices of interest defined as

$$\delta = \begin{bmatrix} \delta_{12} & \delta_{13} & \cdots & \delta_{1J} \\ \delta_{21} & \delta_{23} & \cdots & \delta_{2J} \\ \vdots & \vdots & \vdots & \vdots \\ \delta_{J1} & \delta_{J2} & \cdots & \delta_{J(J-1)} \end{bmatrix} = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_J \end{bmatrix} \quad \text{and} \quad \gamma = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_J \end{bmatrix}. \quad (22.16)$$

There are two approaches to estimate the structural parameters $\boldsymbol{\delta}$ and $\boldsymbol{\gamma}$ of the system, one is the single equation

estimation and the other is the system estimation. First, we introduce the single equation estimation shown below. We can rewrite the j-th equation in our simultaneous equations model in terms of the full set of T observations:

$$\mathbf{y}_j = \mathbf{Y}_j\boldsymbol{\delta}_j + \mathbf{X}_j\boldsymbol{\gamma}_j + \boldsymbol{\varepsilon}_j = \mathbf{Z}_j\boldsymbol{\beta}_j + \boldsymbol{\varepsilon}_j, \quad j = 1, 2, \ldots, J, \tag{22.17}$$

where $\mathbf{y}_j$ denotes the $T\times1$ vector of observations for the endogenous variables on the left-hand side of j-th equation. $\mathbf{Y}_j$ denotes the $T\times(J\text{-}1)$ data matrix for the endogenous variables on the right-hand side of this equation. $\mathbf{X}_j$ is a data matrix for all exogenous variables in this equation. Since these jointly determined variables $\mathbf{y}_j$ and $\mathbf{Y}_j$ are determined within the system, they are correlated with the disturbance terms. This correlation usually creates estimation difficulties because the OLS estimator would be biased and inconsistent (e.g., Johnston and DiNardo 1997; Greene 2011).

As discussed above, the application of OLS estimation to equation (22.17) yields biased and inconsistent estimates because of the correlation of $\mathbf{Z}_j$ and $\boldsymbol{\varepsilon}_j$. The 2SLS approach is the most common method used to deal with this endogeneity problem resulting from the correlation of $\mathbf{Z}_j$ and $\boldsymbol{\varepsilon}_j$. The 2SLS estimation uses all the exogenous variables in this system as instruments to obtain the predictions of $\mathbf{Y}_j$. In the first stage, we regress $\mathbf{Y}_j$ on all exogenous variables in the system to receive the predictions of the endogenous variables on the right-hand side of this equation, $\hat{\mathrm{Y}}_\mathrm{j}$. In the second stage, we regress $\mathbf{y}_j$ on $\hat{\mathrm{Y}}_\mathrm{j}$ and $\mathbf{X}_j$ to obtain the estimator of $\boldsymbol{\beta}_j$ in equation (22.17). Thus, the 2SLS estimator for $\boldsymbol{\beta}_j$ in Eq. (22.17) is

$$\hat{\beta}_{j,2SLS} = \left[(\mathbf{Z}_j'\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Z}_j\right]^{-1}(\mathbf{Z}_j'\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}_j, \tag{22.18}$$

where $\mathbf{X}=[\mathbf{X}_1\,\mathbf{X}_2\cdots\mathbf{X}_J]$ is a matrix for all exogenous variables in this system.

The GMM estimation provides an alternative approach to deal with this simultaneity bias problem. As for the GMM estimator with instruments $\mathbf{X}$, the moment conditions in the equation (22.17) is

$$\mathrm{E}_t(\mathbf{x}_t'\varepsilon_{jt}) = \mathrm{E}_t\left[\mathbf{x}_t'(\mathrm{y}_{jt} - \mathbf{Z}_{jt}\boldsymbol{\beta}_j)\right] = 0. \tag{22.19}$$

We can apply the 2SLS estimator in equation (22.18) with instruments $\mathbf{X}$ to estimate $\boldsymbol{\beta}_j$ and obtain the sample residuals $\hat{\varepsilon}_j = \mathbf{y}_j - \mathbf{Z}_j\hat{\beta}_{j,2SLS}$. Then, compute the weighting matrix $\mathbf{W}_j$ for the GMM estimator based on those residuals as follows:

$$\mathbf{W}_j = \left[\frac{1}{\mathrm{T}^2}\left(\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'\hat{\varepsilon}_{jt}\hat{\varepsilon}_{jt}\mathbf{x}_t\right)\right]. \tag{22.20}$$

The GMM estimator based on the moment conditions (22.19) minimizes the following quadratic function:

$$\left[\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'(\mathrm{y}_{jt} - \mathbf{Z}_{jt}\boldsymbol{\beta}_j)\right]\mathbf{W}_j^{-1}\left[\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'(\mathrm{y}_{jt} - \mathbf{Z}_{jt}\boldsymbol{\beta}_j)\right]. \tag{22.21}$$

The GMM estimator that minimizes this quadratic function (22.21) is obtained as

$$\hat{\beta}_{GMM} = \left[(\mathbf{Z}_j'\mathbf{X})\widehat{\mathbf{W}}_j^{-1}(\mathbf{X}'\mathbf{Z}_j)\right]^{-1}\left[(\mathbf{Z}_j'\mathbf{X})\widehat{\mathbf{W}}_j^{-1}(\mathbf{X}'\mathbf{y}_j)\right]. \tag{22.22}$$

In the homoscedastic and serially independent case, a good estimate of the weighting matrix $\widehat{\mathbf{W}}_j$ would be

$$\widehat{\mathbf{W}} = \left[\frac{\hat{\sigma}^2}{\mathrm{T}}(\mathbf{X}'\mathbf{X})\right]. \tag{22.23}$$

Given the estimate of $\hat{\sigma}^2$ is obtained, then rearrange terms in equation (22.22), which yields

$$\hat{\beta}_{GMM} = \left[(\mathbf{Z}_j'\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Z}_j)\right]^{-1}(\mathbf{Z}_j'\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{y}_j). \tag{22.24}$$

Thus the 2SLS estimator is a special case of the GMM estimator.

As Chen and Lee (2010) pointed out, the 2SLS estimation is a limited information method. The 3SLS estimation is a full information method. The 3SLS estimation takes into account the information from a full system of equations. Thus, it is more efficient than the 2SLS estimation. The 3SLS method estimates all structural parameters of this system jointly. This allows the possibility of a contemporaneous correlation between the disturbances in different structural equations. We introduce the 3SLS estimation below. We rewrite our full system of equations in equation (22.17) as

$$\mathbf{Y} = \mathbf{Z}\beta + \boldsymbol{\varepsilon}, \tag{22.25}$$

where $\mathbf{Y}$ is a vector defined as $[\mathbf{y}_1\,\mathbf{y}_2\cdots\mathbf{y}_J]'$. $\mathbf{Z} = \mathrm{diag}[\mathbf{Z}_1\,\mathbf{Z}_2\cdots\mathbf{Z}_J]$ is a block diagonal data matrix for all variables on the right-hand side of this system with the form $\mathbf{Z}_j = [\mathbf{Y}_j \quad \mathbf{X}_j]$ as defined in equation (22.17). $\boldsymbol{\beta}$ is a vector of interest parameters defined as $[\boldsymbol{\beta}_1 \quad \boldsymbol{\beta}_2\cdots\boldsymbol{\beta}_J]'$. $\boldsymbol{\varepsilon}$ is a vector of disturbances defined as $[\boldsymbol{\varepsilon}_1 \quad \boldsymbol{\varepsilon}_2\cdots\boldsymbol{\varepsilon}_J]'$ with $\mathrm{E}(\boldsymbol{\varepsilon})=\mathbf{0}$ and

$E(\varepsilon\varepsilon') = \boldsymbol{\Sigma} \otimes \mathbf{I}_T$ where $\otimes$ signifies the Kroneker product. Here, $\boldsymbol{\Sigma}$ is defined as

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1J} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{J1} & \sigma_{J2} & \cdots & \sigma_{JJ} \end{bmatrix}. \tag{22.26}$$

The 3SLS approach is the most common method used to estimate the structural parameters of this system simultaneously. Basically, the 3SLS estimator is a generalized least square (GLS) estimator in the entire system taking account of the covariance matrix in equation (22.26). The 3SLS estimator is equivalent to using all exogenous variables as instruments and estimating the entire system using GLS estimation (Intriligator et al. 1996). The 3SLS estimation uses all exogenous variables $\mathbf{X} = [\mathbf{X}_1 \quad \mathbf{X}_2 \cdots \mathbf{X}_J]$ as instruments in each equation of this system, pre-multiplying the model (22.25) by $\mathbf{X}_I' = \text{diag}[\mathbf{X}' \cdots \mathbf{X}'] = \mathbf{X} \otimes \mathbf{I}_J$ yields the model

$$\mathbf{X}_I'\mathbf{Y} = \mathbf{X}_I'\mathbf{Z}\beta + \mathbf{X}_I'\boldsymbol{\varepsilon}. \tag{22.27}$$

The covariance matrix from (22.26) is

$$\text{Cov}(\mathbf{X}_I'\boldsymbol{\varepsilon}) = \mathbf{X}_I'\text{Cov}(\boldsymbol{\varepsilon})\mathbf{X}_I = \mathbf{X}_I'(\Sigma \otimes \mathbf{I}_T)\mathbf{X}_I. \tag{22.28}$$

The GLS estimator of the equation (22.27) is the 3SLS estimator. Thus the 3SLS estimator is given as follows:

$$\hat{\beta}_{3SLS} = \{\mathbf{Z}'\mathbf{X}_I\left[\mathbf{X}_I'(\boldsymbol{\Sigma} \otimes \mathbf{I}_T)\mathbf{X}_I\right]^{-1}\mathbf{X}_I'\mathbf{Z}\}^{-1}\mathbf{Z}'\mathbf{X}_I\left[\mathbf{X}_I'(\boldsymbol{\Sigma} \otimes \mathbf{I}_T)\mathbf{X}_I\right]^{-1}\mathbf{X}_I'\mathbf{Y}. \tag{22.29}$$

In this case, $\boldsymbol{\Sigma}$ is a diagonal matrix, the 3SLS estimator is equivalent to the 2SLS estimator. As discussed above, the GMM estimator with all exogenous variables $\mathbf{X} = [\mathbf{X}_1 \quad \mathbf{X}_2 \cdots \mathbf{X}_J]$ as instruments, the moment conditions of this system (22.25) are,

$$\begin{aligned} \mathrm{E}\left(\mathbf{X}_I'\boldsymbol{\varepsilon}\right) &= \mathrm{E}\left[\mathbf{X}_I'(\mathbf{Y} - \mathbf{Z}\beta)\right] \\ &= \left(\mathrm{E}\left[\mathbf{X}_I'(\mathbf{y}_1 - \mathbf{Z}_1\boldsymbol{\beta}_1)\right]\mathrm{E}[\mathbf{X}_I'(\mathbf{y}_2 - \mathbf{Z}_2\boldsymbol{\beta}_2)]\cdots\mathrm{E}[\mathbf{X}_I'(\mathbf{y}_J - \mathbf{Z}_J\boldsymbol{\beta}_J)]\right)' = 0 \end{aligned} \tag{22.30}$$

We can apply the 2SLS estimator with instruments $\mathbf{X}$ to estimate $\boldsymbol{\beta}_j$ and obtain the sample residuals $\hat{\varepsilon}_j = \mathbf{y}_j - \mathbf{Z}_j\hat{\beta}_{j,2SLS}$. Then, compute the weighting matrix $\widehat{\mathbf{W}}_{jl}$ for GMM estimator based on those residuals as follows:

$$\widehat{\mathbf{W}}_{jl} = \left[\frac{1}{\mathrm{T}^2}\left(\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'\hat{\varepsilon}_{jt}\hat{\varepsilon}_{lt}\mathbf{x}_t\right)\right]. \tag{22.31}$$

The system GMM estimator based on the moment conditions (22.30) minimizes the quadratic function:

$$\begin{bmatrix} \mathbf{X}'(\mathbf{y}_1 - \mathbf{Z}_1\boldsymbol{\beta}_1) \\ \mathbf{X}'(\mathbf{y}_2 - \mathbf{Z}_2\boldsymbol{\beta}_2) \\ \vdots \\ \mathbf{X}'(\mathbf{y}_\mathrm{J} - \mathbf{Z}_\mathrm{J}\boldsymbol{\beta}_\mathrm{J}) \end{bmatrix}' \begin{bmatrix} \widehat{\mathbf{W}}_{11} & \widehat{\mathbf{W}}_{12} & \cdots & \widehat{\mathbf{W}}_{1\mathrm{J}} \\ \widehat{\mathbf{W}}_{21} & \widehat{\mathbf{W}}_{22} & \cdots & \widehat{\mathbf{W}}_{2\mathrm{J}} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{\mathbf{W}}_{\mathrm{J}1} & \widehat{\mathbf{W}}_{\mathrm{J}2} & \cdots & \widehat{\mathbf{W}}_{\mathrm{JJ}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X}'(\mathbf{y}_1 - \mathbf{Z}_1\boldsymbol{\beta}_1) \\ \mathbf{X}'(\mathbf{y}_2 - \mathbf{Z}_2\boldsymbol{\beta}_2) \\ \vdots \\ \mathbf{X}'(\mathbf{y}_\mathrm{J} - \mathbf{Z}_\mathrm{J}\boldsymbol{\beta}_\mathrm{J}) \end{bmatrix}. \tag{22.32}$$

The GMM estimator that minimizes this quadratic function (22.32) is obtained as

$$\begin{bmatrix} \hat{\boldsymbol{\beta}}_{1,GMM} \\ \hat{\boldsymbol{\beta}}_{2,GMM} \\ \vdots \\ \hat{\boldsymbol{\beta}}_{J,GMM} \end{bmatrix} == \begin{bmatrix} \mathbf{Z}_1\mathbf{X}\widehat{\mathbf{W}}_{11}^{-1}\mathbf{X}\mathbf{Z}_1 & \cdots & \mathbf{Z}_l'\mathbf{X}\widehat{\mathbf{W}}_{1J}^{-1}\mathbf{X}'\mathbf{Z}_J \\ \mathbf{Z}_2'\mathbf{X}\widehat{\mathbf{W}}_{21}^{-1}\mathbf{X}\mathbf{Z}_1 & \cdots & \mathbf{Z}_l'\mathbf{X}\widehat{\mathbf{W}}_{2J}^{-1}\mathbf{X}'\mathbf{Z}_J \\ \vdots & \cdots & \vdots \\ \mathbf{Z}_J'\mathbf{X}\widehat{\mathbf{W}}_{J1}^{-1}\mathbf{X}'\mathbf{Z}_l & \cdots & \mathbf{Z}_J'\mathbf{X}\widehat{\mathbf{W}}_{JJ}^{-1}\mathbf{X}'\mathbf{Z}_J \end{bmatrix}^{-1} \begin{bmatrix} \sum_{l=1}^{J}\mathbf{Z}_l'\mathbf{X}\widehat{\mathbf{W}}_{1l}^{-1}\mathbf{y}_l \\ \sum_{l=1}^{J}\mathbf{Z}_2'\mathbf{X}\widehat{\mathbf{W}}_{2l}^{-1}\mathbf{y}_l \\ \vdots \\ \sum_{l=1}^{J}\mathbf{Z}_J'\mathbf{X}\widehat{\mathbf{W}}_{Jl}^{-1}\mathbf{y}_l \end{bmatrix}. \tag{22.33}$$

The 2SLS and 3SLS estimators are the special cases of system GMM estimators. If $\widehat{\mathbf{W}}_{jj} = \left[\frac{\hat{\sigma}_{jj}}{\mathrm{T}}\left(\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'\mathbf{x}_t\right)\right]$ and $\widehat{\mathbf{W}}_{jl} = 0$ for $j \neq l$, then the system GMM estimator is equivalent to the 2SLS estimator. In the case that $\widehat{\mathbf{W}}_{jl} = \left[\frac{\hat{\sigma}_{jl}}{\mathrm{T}}\left(\sum_{t=1}^{\mathrm{T}}\mathbf{x}_t'\mathbf{x}_t\right)\right]$, the system GMM estimator is equivalent to the 3SLS estimator.

22.3.3 Weak Instruments

As mentioned above, we introduce three alternative approaches, 2SLS, 3SLS, and GMM estimations to estimate a simultaneous equations system. Regardless of whether 2SLS, 3SLS, or GMM estimation is used to estimate in the second stage, the first-stage regression instrumenting for endogenous regressors is estimated via OLS. The choice of instruments is critical to the consistent estimation of the IV methods. Previous works have demonstrated that if the instruments are weak, the IV estimator will not possess its ideal properties and will be misleading (e.g., Bound et al.1995; Staiger and Stock, 1997; Stock and Yogo, 2005).

A simple way to detect the presence of weak instruments is to look at the R^2 or F-statistic of first-stage regression testing the hypothesis that the coefficients on the instruments are jointly equal to zero (Wang 2015). Institutively, the first-stage F-statistic must be large, typically exceeding 10, for inference of 2SLS estimation to be reliable (Staiger and Stock 1997; Stock et al. 2002). In addition, Hahn and

Hausman(2005) show that the relative bias of 2SLS estimation declines as the strength of the correlation between the instruments and the endogenous regressor increases, but grows with the number of instruments. Stock and Yogo (2005) tabulate critical values for the first-stage F-statistic to test whether instruments are weak. They report, for instance, that when there is one endogenous regressor, the first-stage F-statistic of the 2SLS regression should have a value higher than 9.08 with three instruments and 10.83 with five instruments.

To sum up, the choice of instruments is critical to the consistent estimation of the instrumental variable methods. As the weakness of instruments in explaining the endogenous regressor can be measured by F-statistic from first-stage regression and compared to the critical value in Stock and Yogo (2005). In addition, the traditional IV models such as 2SLS and 3SLS overcome the endogeneity problem by instrumenting for variables that are endogenous.

22.4 Applications in Investment, Financing, and Dividend Policy

22.4.1 Model and Data

The investment, dividend, and debt financing are major decisions of a firm. Past studies argue some relations among investment, dividend, and debt financing. To control for the possible endogenous problems among these three decisions, we apply 2SLS, 3SLS, and GMM methods to estimate the simultaneous-equations model that considers the interaction of the three policies.

There are three equations in our simultaneous-equations system; each equation contains the remaining two endogenous variables as explanatory variables along with other exogenous variables. The three endogenous variables are investment (Inv_{it}), dividend (Div_{it}), and debt financing ($Leverage_{it}$) of firm i in year t. Investment is measured by the net property, plant, and equipment. Following Fama (1974), both investment and dividend are measured on a per-share basis. We follow Fama and French (2002) to use book leverage as the proxy for debt financing. Book leverage is measured as the ratio of total liabilities to total assets.

We also use the following exogenous variables in the model. In addition to lag-terms of the three policies, we follow Fama (1974) to respectively incorporate sales plus the change in inventories (Q_{it}) and net income minus preferred dividends (P_{it}) into investment and dividend decisions. Moreover, we follow Fama and French (2002) to add natural logarithm of lagged total assets ($\ln A_{i,t-1}$) and the lag of earnings before interest and taxes divided by total assets ($E_{i,t-1}/A_{i,t-1}$) as the determinants of leverage.

The structural equations are estimated as follows:

$$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}, \tag{22.34}$$

$$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}, \tag{22.35}$$

$$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}. \tag{22.36}$$

Our sample consists of Johnson & Johnson, and IBM companies' annual data from 1966 to 2019. Table 22.1 presents summary statistics on the investment, dividend, and debt financing for two companies, namely IBM and Johnson & Johnson.

22.4.2 Results of Weak Instruments

We perform the first-stage F-statistic to test whether instruments are weak. Table 22.2 shows the results of testing the

Table 22.1 Summary statistics

	Mean	Median	Q1	Q3	Standard deviation
Panel A. Johnson & Johnson case					
Inv	7.7107	6.3985	5.0117	9.4102	3.8303
Div	1.4714	1.2752	0.8478	1.9341	0.8242
Leverage	0.3996	0.4419	0.2844	0.4815	0.1176
Panel B. IBM case					
Inv	27.5106	27.5306	11.4498	39.7225	16.1379
Div	3.7218	3.7672	1.5499	4.8527	2.5784
Leverage	0.5821	0.6842	0.3766	0.7666	0.2231

This table presents the summary statistics where we show the mean, median, first quartile, third quartile, and the standard deviation of each variable from 1966 to 2019 consists of total 54 observations. *Inv* denotes net property, plant, and equipment. *Div* denotes dividends. Both *Inv* and *Div* are measured on a per share basis. *Leverage* refers to book leverage, defined as the ratio of total liabilities to total assets

Table 22.2 Results of testing the relevance of instruments and heteroskedasticity

Instruments	*Inv*	*Div*	*Leverage*
Panel A. Johnson & Johnson case			
First-stage R^2	0.9798	0.9847	0.8966
F-statistic	319.3	423.7	56.9
Panel B. IBM case			
First-stage R^2	0.9448	0.8688	0.9807
F-statistic	112.5	43.53	334.6

We regress each endogenous variable on all exogenous variables in the system to receive the prediction of endogenous variable and obtain R^2 as well as F-statistics for each firm. The null hypothesis of F test is that the instruments are jointly equal to zero. The three endogenous variables are Inv_{it}, Div_{it} and $Leverage_{it}$, which are net plant and equipment, dividends, and book leverage ratio, respectively

relevance of instruments. We regress each endogenous variable on all exogenous variables in the system to receive the prediction of endogenous variable and obtain as well as F-statistics for each firm. In Johnson & Johnson's case, the values of R^2 for investment, dividend, and book leverage equations are 0.9798, 0.9847, and 0.8966 respectively that show the strength of the instrument. Likewise, in the IBM case, the values of R^2 for the investment, dividend, and financing decision equations are 0.9448, 0.8688, and 0.9807, respectively. Moreover, the ratios of F-statistics over 10 for three endogenous variables both in Johnson & Johnson, and IBM cases. All results support that instruments are sufficiently strong.

22.4.3 Empirical Results

A. Johnson & Johnson case

Tables 22.3, 22.4, and 22.5 respectively show the 2SLS, 3SLS, and GMM estimation results for the simultaneous-equation model for Johnson & Johnson case. Overall, our findings of relations among these three financial decisions from 2SLS, 3SLS, and GMM methods are similar. The results of three financial decisions for Johnson & Johnson company are summarized as follows.

First, looking at the investment equation (e.g., Table 22.3), dividend (Div_{it}) has a negative impact on the level of investment expenditure (Inv_{it}). This negative relation between investment and dividend is consistent with McCabe (1979) and Peterson and Benesh (1983). They argue that dividend is a competing use of funds, the firm must choose whether to expend funds on investment or dividends. Moreover, financing decisions $((Leverage_{it}))$ has a positive impact on investment (Inv_{it}). Our finding that increases in debt financing enhance the funds available to outlays for investment is consistent with McDonald et al. (1975), McCabe (1979), Peterson and Benesh (1983), John and Nachman (1985), and Froot et al. (1993).

Second, as for dividend decision (e.g., Table 22.3), the impact of debt financing on the dividend is significantly positive, showing that an increase in external financing should exhibit a positive influence on the dividend. The positive relationship between leverage and dividend is consistent with McCabe (1979), Peterson and Benesh (1983), and Switzer (1984). Moreover, an increase in the level of investment expenditure has a negative influence on dividends since investment and dividends are competing uses for funds.

Third, turning to financing decision (e.g., Table 22.3), only lagged leverage has a significantly positive effect on the level of leverage. However, investment and dividend decisions do not have a significantly impact on the level of leverage. This finding supports that Johnson & Johnson company may have a desired optimal level of leverage.

In addition, the results of control variables for Johnson & Johnson company are shown as follows. First, the impact of output, Q_{it}, on the investment is significantly positive, which is consistent with Fama (1974). Second, the coefficient of P_{it} in the dividend model is significantly positive, implying that firms with high net income tend to increase to pay dividends. Third, in the debt financing equation, only the coefficient of $\ln A_{i,t-1}$ is significantly positive, indicating that large firms leverage more than smaller firms. This finding results from large firms that tend to have a greater reputation and less information asymmetry than small firms and thus large firms can finance at a lower cost. The positive relation between size and leverage is consistent with Fama and French (2002), Flannery and Rangan (2006), and Frank and Goyal (2009).

B. IBM case

Tables 22.6, 22.7, and 22.8 respectively show the 2SLS, 3SLS, and GMM estimation results for the simultaneous-equation model for the IBM case. Overall, our findings of relations among these three financial decisions from 2SLS, 3SLS, and GMM methods are similar. The results of three

Table 22.3 Results of 2SLS: Johnson & Johnson case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	-0.9507^{***}		0.0054
	(0.1664)		(0.0198)
$Leverage_{it}$	7.8215^{***}	1.0104^{**}	
	(1.2650)	(0.4489)	
Inv_{it}		-0.0276^{*}	0.0006
		(0.0148)	(0.0030)
$Inv_{i,t-1}$	0.0581^{*}		
	(0.0323)		
Q_{it}	0.2496^{***}		
	(0.0097)		
$Leverage_{i,t-1}$			0.7835^{***}
			(0.0989)
$lnA_{i,t-1}$			0.0097
			(0.0105)
$E_{i,t-1}/A_{i,t-1}$			−0.0653
			(0.3502)
$Div_{i,t-1}$		0.6196^{***}	
		(0.0766)	
P_{it}		0.2055^{***}	
		(0.0356)	
Constant	-2.3771^{***}	-0.5971^{***}	0.0029
	(0.5196)	(0.1893)	(0.0778)
Observations	54	54	54
Adjusted R^2	0.9701	0.9002	0.8549

This table presents the 2SLS regression results of a simultaneous equation system model for investment, dividend, and debt financing:
$Inv_{it} = \alpha_{1i} + \alpha_{2i}Div_{it} + \alpha_{3i}Leverage_{it} + \alpha_{4i}Inv_{i,t-1} + \alpha_{5i}Q_{it} + \epsilon_{it}$,
$Div_{it} = \beta_{1i} + \beta_{2i}Inv_{it} + \beta_{3i}Leverage_{it} + \beta_{4i}Div_{i,t-1} + \beta_{5i}P_{it} + \eta_{it}$,
$Leverage_{it} = \gamma_{1i} + \gamma_{2i}Inv_{it} + \gamma_{3i}Div_{it} + \gamma_{4i}Leverage_{i,t-1} + \gamma_{5i}lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}$,
where Inv_{it}, Div_{it}, and $Leverage_{it}$ are net plant and equipment, dividends, and book leverage ratio, respectively. The independent variables in investment regression are lagged investment ($Inv_{i,t-1}$), and sales plus change in inventories (Q_{it}). The independent variables in dividend regression are lagged dividends ($Div_{i,t-1}$), and net income minus preferred dividends (P_{it}). All variables in both of investment and dividend equations are measured on a per share basis. The independent variables in debt financing regression are lagged book leverage ($Leverage_{i,t-1}$), natural logarithm of lagged total assets ($lnA_{i,t-1}$), and the lag of earnings before interest and taxes divided by total assets ($E_{i,t-1}/A_{i,t-1}$). Numbers in parentheses are standard errors of coefficients. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

financial decisions for IBM company are summarized as follows.

First, as for investment decision, only financing decision has a significantly negative impact on the level of investment expenditure. Secondly, as for dividend decision, investment, and financing decisions both do not have a significant impact on the dividend payout. Thirdly, as for financing decision, only investment decision has a significantly positive impact on the level of leverage. Finally, the results of control variables for IBM company are similar to the findings in Johnson & Johnson company. Overall, our finding supports that the investment and financing decisions are made simultaneously for the IBM company. That is, the interaction between investment and financing decisions should be considered in a system of simultaneous equations framework.

Table 22.4 Results of 3SLS: Johnson & Johnson case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	-0.9827^{***}		-0.0035
	(0.0931)		(0.0103)
$Leverage_{it}$	8.1380^{***}	0.9683^{***}	
	(0.7077)	(0.2466)	
Inv_{it}		-0.0293^{***}	0.0010
		(0.0079)	(0.0016)
$Inv_{i,t-1}$	0.0953^{***}		
	(0.0168)		
Q_{it}	0.2436^{***}		
	(0.0053)		
$Leverage_{i,t-1}$			0.8220^{***}
			(0.0518)
$lnA_{i,t-1}$			0.0097^{*}
			(0.0054)
$E_{i,t-1}/A_{i,t-1}$			−0.2657
			(0.1790)
$Div_{i,t-1}$		0.6193^{***}	
		(0.0408)	
P_{it}		0.2080^{***}	
		(0.0183)	
Constant	-2.5608^{***}	-0.5792^{***}	0.0360
	(0.2906)	(0.1040)	(0.0406)
Observations	54	54	54
Adjusted R^2	0.9681	0.8980	0.8486

This table presents the 3SLS regression results of a simultaneous equation system model for investment, dividend, and debt financing:

$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}$,

$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}$,

$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}$,

The three endogenous variables are Inv_{it}, Div_{it} and $Leverage_{it}$, which are net plant and equipment, dividends, and book leverage ratio, respectively. The other variables are the same as in Table 22.4. Numbers in the parentheses are standard errors of coefficients. The sign in bracket is the expected sign of each variable of regressions. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Table 22.5 Results of GMM: Johnson & Johnson case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	-0.9014^{***}		-0.0045
	(0.0693)		(0.0070)
$Leverage_{it}$	7.8582^{***}	0.5768^{***}	
	(0.4749)	(0.1737)	
Inv_{it}		-0.0461^{***}	0.0016
		(0.0053)	(0.0011)
$Inv_{i,t-1}$	0.0790^{***}		
	(0.0084)		
Q_{it}	0.2456^{***}		
	(0.0041)		
$Leverage_{i,t-1}$			0.7838^{***}
			(0.0445)
$lnA_{i,t-1}$			0.0098^{*}
			(0.0043)
$E_{i,t-1}/A_{i,t-1}$			-0.1738
			(0.1219)
$Div_{i,t-1}$		0.5280^{***}	
		(0.0478)	
P_{it}		0.2585^{***}	
		(0.0210)	
Constant	-2.5196^{***}	-0.4035^{***}	0.0206
	(0.1751)	(0.0708)	(0.0295)
Observations	54	54	54
Adjusted R^2	0.9693	0.8871	0.8491

This table presents the GMM regression results of a simultaneous equation system model for investment, dividend, and debt financing:

$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}$,

$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}$,

$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}$,

The three endogenous variables are Inv_{it}, Div_{it}, and $Leverage_{it}$, which are net plant and equipment, dividends, and book leverage ratio, respectively. The other variables are the same as in Table 22.4. Numbers in the parentheses are standard errors of coefficients. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Table 22.6 Results of 2SLS: IBM case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	−0.2382		−0.0012
	(0.3815)		(0.0034)
$Leverage_{it}$	−45.4795***	1.0228	
	(5.6080)	(1.3274)	
Inv_{it}		0.0149	0.0012
		(0.0214)	(0.0009)
$Inv_{i,t-1}$			
	0.2404***		
	(0.0684)		
Q_{it}			
	0.2964***		
	(0.0366)		
$Leverage_{i,t-1}$			
			0.9306***
			(0.0640)
$lnA_{i,t-1}$			
			0.0284**
			(0.0112)
$E_{i,t-1}/A_{i,t-1}$			
			−0.0224
			(0.1599)
$Div_{i,t-1}$		0.5499***	
		(0.0835)	
P_{it}		0.1640***	
		(0.0309)	
Constant	23.2712***	−1.8168	−0.2790*
	(4.2838)	(1.2531)	(0.1502)
Observations	54	54	54
Adjusted R^2	0.9221	0.7734	0.9759

This table presents the 2SLS regression results of a simultaneous equation system model for investment, dividend, and debt financing:

$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}$,

$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}$,

$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}$,

The three endogenous variables are Inv_{it}, Div_{it}, and $Leverage_{it}$, which are net plant and equipment, dividends, and book leverage ratio, respectively. The independent variables in the investment regression are lagged investment ($Inv_{i,t-1}$), and sales plus change in inventories (Q_{it}). The independent variables in the dividend regression are lagged dividends ($Div_{i,t-1}$), and net income minus preferred dividends (P_{it}). All the variables in both of investment and dividend equations are measured on a per share basis. The independent variables in the debt financing regression are lagged book leverage ($Leverage_{i,t-1}$), natural logarithm of lagged total assets ($lnA_{i,t-1}$), and the lag of earnings before interest and taxes divided by total assets ($E_{i,t-1}/A_{i,t-1}$). Numbers in the parentheses are standard errors of coefficients. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Table 22.7 Results of 3SLS: IBM case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	−0.3028		−0.0015
	(0.2107)		(0.0018)
$Leverage_{it}$	−42.5825***	0.8453	
	(3.0531)	(0.7312)	
Inv_{it}		0.0112	0.0012**
		(0.0118)	(0.0005)
$Inv_{i,t-1}$	0.2958***		
	(0.0364)		
Q_{it}	0.2809***		
	(0.0200)		
$Leverage_{i,t-1}$			0.9285***
			(0.0349)
$lnA_{i,t-1}$			0.0304***
			(0.0061)
$E_{i,t-1}/A_{i,t-1}$			−0.0012
			(0.0872)
$Div_{i,t-1}$		0.5713***	
		(0.0454)	
P_{it}		0.1590***	
		(0.0163)	
Constant	21.5511***	−1.6140**	−0.3031***
	(2.3508)	(0.6898)	(0.0819)
Observations	54	54	54
Adjusted R^2	0.9190	0.7669	0.9753

This table presents the 3SLS regression results of a simultaneous equation system model for investment, dividend, and debt financing:
$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}$,
$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}$,
$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} \ln A_{i,t-1} + \gamma_{6i}(E_{i,t-1}/A_{i,t-1}) + \xi_{it}$,
The three endogenous variables are Inv_{it} , Div_{it} , and $Leverage_{it}$, which are net plant and equipment, dividends, and book leverage ratio, respectively. The other variables are the same as in Table 22.4. Numbers in the parentheses are standard errors of coefficients. The sign in bracket is the expected sign of each variable of regressions. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Table 22.8 Results of GMM: IBM case

	Dependent variable		
	Inv_{it}	Div_{it}	$Leverage_{it}$
Div_{it}	−0.0309		−0.0017*
	(0.1634)		(0.0010)
$Leverage_{it}$	−35.8505***	−0.9110**	
	(2.1937)	(0.3846)	
Inv_{it}		−0.0120	0.0016***
		(0.0088)	(0.0003)
$Inv_{i,t-1}$	0.4382***		
	(0.0300)		
Q_{it}	0.2016***		
	(0.0160)		
$Leverage_{i,t-1}$			0.9434***
			(0.0299)
$\ln A_{i,t-1}$			0.0437***
			(0.0043)
$E_{i,t-1}/A_{i,t-1}$			0.1056
			(0.0710)
$Div_{i,t-1}$		0.8039***	
		(0.0520)	
P_{it}		0.0921***	
		(0.0156)	
Constant	19.9544***	0.3580	−0.4845***
	(1.4990)	(0.4224)	(0.0534)
Observations	54	54	54
Adjusted R^2	0.9063	0.7013	0.9190

This table presents the GMM regression results of a simultaneous equation system model for investment, dividend, and debt financing:

$Inv_{it} = \alpha_{1i} + \alpha_{2i} Div_{it} + \alpha_{3i} Leverage_{it} + \alpha_{4i} Inv_{i,t-1} + \alpha_{5i} Q_{it} + \epsilon_{it}$,

$Div_{it} = \beta_{1i} + \beta_{2i} Inv_{it} + \beta_{3i} Leverage_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it}$,

$Leverage_{it} = \gamma_{1i} + \gamma_{2i} Inv_{it} + \gamma_{3i} Div_{it} + \gamma_{4i} Leverage_{i,t-1} + \gamma_{5i} lnA_{i,t-1} + \gamma_{6i}\left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}$,

The three endogenous variables are Inv_{it}, Div_{it} , and Leverage_{it} , which are net plant and equipment, dividends, and book leverage ratio, respectively. The other variables are the same as in Table 22.4. Numbers in the parentheses are standard errors of coefficients. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

22.5 Conclusion

In this chapter, we investigate the endogeneity problems related to the simultaneous equations system and introduce how 2SLS, 3SLS, and GMM estimation methods deal with endogeneity problems. In addition to reviewing applications of simultaneous equations in capital structure decisions, we also use Johnson & Johnson, and IBM companies' annual data from 1966 to 2019 to examine the interrelationship among corporate investment, leverage, and dividend payout policies in a simultaneous-equation system by employing 2SLS, 3SLS, and GMM methods. Our findings of relations among these three financial decisions from 2SLS, 3SLS, and GMM methods are similar. Overall, our study suggests that these three corporate decisions are jointly determined and the interaction among them should be taken into account in a simultaneous equations framework.

Appendix 22.1: Data for Johnson & Johnson and IBM

1.1 Johnson & Johnson data

fyear	pstar	div	inv	q	debtratio	et	lna	debtratio_peer	invlag_1	divlag_l	ebtratiolag.	debtratiolag_peer	etlag	lnalag
1966	8.6129	1.6441	18.7782	83.7590	0.2160	0.1696	5.7948	0.3105	16.3320	1.4456	0.2152	0.3146	0.1538	5.7206
1967	3 2342	0.6139	7.0284	29.4397	0.2084	0.1613	5.9186	0.3180	18.7782	1.6441	0.2160	0.3105	0.1696	5.7948
1968	3 7749	0.6481	7.5312	33.0988	0.2073	0.1841	6.0540	0.3494	7.0284	0.6139	0.2084	0.3180	0.1613	5.9186
1969	4 3568	0.8413	8.4597	36.8895	0.1868	0.1904	6.1772	0.3560	7.5312	0.6481	0.2073	0.3494	0.1841	6.0540
1970	2 0867	0.3387	4.2893	18.9954	0.2422	0.2109	6.5604	0.3635	8.4597	0.8413	0.1868	0.3560	0.1904	6.1772
1971	2.4711	0.4285	4.8064	20.7572	0.2477	0.2102	6.7215	0.3929	4.2893	0.3387	0.2422	0.3635	0.2109	6.5604
1972	2 8925	0.4455	5.3408	23.6784	0.2506	0.2131	6.8890	0.3952	4.8064	0.4285	0.2477	0.3929	0.2102	6.7215
1973	3 4686	0.5194	6.3477	29.0629	0.2663	0.2168	7.0809	0.4191	5.3408	0.4455	0.2506	0.3952	0.2131	6.8890
1974	3 8532	0.7233	8.0820	36.3433	0.2844	0.1850	7.2483	0.4631	6.3477	0.5194	0.2663	0.4191	0.2168	7.0809
1975	4 3495	0.8475	9.1023	37.7274	0.2551	0.1934	7.3470	0.4280	8.0820	0.7233	0.2844	0.4631	0.1850	7.2483
1976	4 8568	1.0473	9.7586	44.3202	0.2442	0.2033	7.4563	0.4482	9.1023	0.8475	0.2551	0.4280	0.1934	7.3470
1977	5 7056	1.3976	11.1505	50.6008	0.2644	0.2040	7.6108	0.4793	9.7586	1.0473	0.2442	0.4482	0.2033	7.4563
1978	6 7198	1.6827	13.1696	60.0825	0.2825	0.2076	7.7759	0.4906	11.1505	1.3976	0.2644	0.4793	0.2040	7.6108
1979	7 7318	1.9964	15.4836	71.1573	0.3059	0.1960	7.9634	0.4719	13.1696	1.6827	0.2825	0.4906	0.2076	7.7759
1980	8 7276	2.2151	18.7998	79.9422	0.3183	0.1843	8.1145	0.5060	15.4836	1.9964	0.3059	0.4719	0.1960	7.9634
1981	3 3144	0.8478	7.1399	29.1363	0.3353	0.1877	8.2481	0.4093	18.7998	2.2151	0.3183	0.5060	0.1843	8.1145
1982	3 6991	0.9644	8.3431	30.7638	0.3327	0.1658	8.3451	1.6483	7.1399	0.8478	0.3353	0.4093	0.1877	8.2481
1983	3 6524	1.0694	8.7191	31.3993	0.3205	0.1666	8.4032	0.3971	8.3431	0.9644	0.3327	1.6483	0.1658	8.3451
1984	4 0515	1.2027	9.4102	33.2396	0.3544	0.1642	8.4210	0.4385	8.7191	1.0694	0.3205	0.3971	0.1666	8.4032
1985	4 7262	1.2753	10.0622	35.1705	0.3423	0.1649	8.5360	0.5177	9.4102	1.2027	0.3544	0.4385	0.1642	8.4210
1986	3 4984	1.4157	11.0865	40.8348	0.5194	0.1674	8.6787	0.4621	10.0622	1.2753	0.3423	0.5177	0.1649	8.5360
1987	6 6591	1.6154	13.0741	47.4532	0.4676	0.1847	8.7866	0.4540	11.0865	1.4157	0.5194	0.4621	0.1674	8.6787
1988	7 8722	1.9636	14.9698	54.6912	0.5079	0.1972	8.8705	0.5207	13.0741	1.6154	0.4676	0.4540	0.1847	8.7866
1989	4 3236	1.1199	8.5452	29.5358	0.4762	0.2097	8.9770	0.5143	14.9698	1.9636	0.5079	0.5207	0.1972	8.8705
1990	4 6536	1.3090	9.7485	34.2925	0.4845	0.2096	9.1597	0.5147	8.5452	1.1199	0.4762	0.5143	0.2097	8.9770
1991	5 6999	1.5398	11.0065	37.8370	0.4649	0.2058	9.2604	0.5172	9.7485	1.3090	0.4845	0.5147	0.2096	9.1597
1992	3.2408	0.8956	6.2786	21.0453	0.5649	0.1916	9.3829	0.3546	11.0065	1.5398	0.4649	0.5172	0.2058	9.2604
1993	3 6393	1.0249	6.8525	21.9493	0.5452	0.1956	9.4126	0.3472	6.2786	0.8956	0.5649	0.3546	0.1916	9.3829
1994	4.2457	1.1306	7.6360	25.1598	0.5454	0.1792	9.6594	0.8785	6.8525	1.0249	0.5452	0.3472	0.1956	9.4126
1995	5.0334	1.2769	8.0225	29.2691	0.4939	0.1964	9.7910	1.5659	7.6360	1.1306	0.5454	0.8785	0.1792	9.6594
1996	2.9239	0.7310	4.2410	16.3919	0.4585	0.2150	9.9040	0.5254	8.0225	1.2769	0.4939	1.5659	0.1964	9.7910

(continued)

fyear	pstar	div	inv	q	debtratio	et	lna	debtratio_peer	invlag_1	divlag_1	ebtratiolag.	debtratiolag_peer	etlag	lnalag
1997	3.2487	0.8453	4.3193	16.8362	0.4239	0.2154	9.9736	0.5929	4.2410	0.7310	0.4585	0.5254	0.2150	9.9040
1998	3.2030	0.9709	4.6427	17.8520	0.4815	0.1925	10.1739	0.5440	4.3193	0.8453	0.4239	0.5929	0.2154	9.9736
1999	4.0376	1.0643	4.8349	19.9420	0.4441	0.2032	10.2807	4.0815	4.6427	0.9709	0.4815	0.5440	0.1925	10.1739
2000	4.5401	1.2395	5.0117	20.7673	0.3995	0.2068	10.3520	4.3154	4.8349	1.0643	0.4441	4.0815	0.2032	10.2807
2001	2.3868	0.6718	2.5331	10.8801	0.3704	0.2049	10.5581	3.5906	5.0117	1.2395	0.3995	4.3154	0.2068	10.3520
2002	2.7824	0.8021	2.9343	12.3333	0.4404	0.2386	10.6104	3.7052	2.5331	0.6718	0.3704	3.5906	0.2049	10.5581
2003	3.0546	0.9252	3.3174	14.2006	0.4433	0.2252	10.7844	2.0155	2.9343	0.8021	0.4404	3.7052	0.2386	10.6104
2004	3.5789	1.0942	3.5126	15.9891	0.4033	0.2413	10.8840	2.2194	3.3174	0.9252	0.4433	2.0155	0.2252	10.7844
2005	4.2038	1.2752	3.6410	17.0279	0.3473	0.2291	10.9686	2.4548	3.5126	1.0942	0.4033	2.2194	0.2413	10.8840
2006	4.5727	1.4748	4.5085	18.7071	0.4427	0.1925	11.1642	0.5554	3.6410	1.2752	0.3473	2.4548	0.2291	10.9686
2007	4.7014	1.6442	4.9943	21.5673	0.4649	0.1872	11.3016	0.5565	4.5085	1.4748	0.4427	0.5554	0.1925	11.1642
2008	5.6988	1.8143	5.1875	22.9992	0.4994	0.1904	11.3494	1.2258	4.9943	1.6442	0.4649	0.5565	0.1872	11.3016
2009	5.4605	1.9341	5.3585	22.5192	0.4657	0.1772	11.4583	1.5623	5.1875	1.8143	0.4994	1.2258	0.1904	11.3494
2010	5.9432	2.1197	5.3150	22.5649	0.4502	0.1606	11.5416	1.2891	5.3585	1.9341	0.4657	1.5623	0.1772	11.4583
2011	4.7094	2.2596	5.4101	24.2027	0.4977	0.1430	11.6408	1.3479	5.3150	2.1197	0.4502	1.2891	0.1606	11.5416
2012	5.2255	2.3804	5.7934	24.6299	0.4658	0.1413	11.7064	2.5761	5.4101	2.2596	0.4977	1.3479	0.1430	11.6408
2013	6.3585	2.5831	5.9242	25.4181	0.4419	0.1429	11.7957	1.9465	5.7934	2.3804	0.4658	2.5761	0.1413	11.7064
2014	7.2642	2.7910	5.7940	26.8168	0.4680	0.1629	11.7839	0.8565	5.9242	2.5831	0.4419	1.9465	0.1429	11.7957
2015	6.9524	2.9664	5.7728	25.3862	0.4667	0.1377	11.8012	1.1927	5.7940	2.7910	0.4680	0.8565	0.1629	11.7839
2016	7.4982	3.1853	5.8792	26.5955	0.5013	0.1502	11.8580	0.6180	5.7728	2.9664	0.4667	1.1927	0.1377	11.8012
2017	2.5879	3.3338	6.3392	28.7308	0.6176	0.1262	11.9659	0.9751	5.8792	3.1853	0.5013	0.6180	0.1502	11.8580
2018	8.3483	3.5661	6.3985	30.5804	0.6093	0.1398	11.9379	0.5902	6.3392	3.3338	0.6176	0.9751	0.1262	11.9659
2019	8.4057	3.7671	7.0712	31.3314	0.6230	0.1339	11.9686	0.5648	6.3985	3.5661	0.6093	0.5902	0.1398	11.9379

1.2 IBM Data

fyear	pstar	div	inv	q	debtratio	et	lna	debtratio_peer	invlag_1	divlag_1	ebtratiolag_	debtratiolag peer	etlag	lnalag
1966	8.5221	4.5439	16.2576	71.1472	0.3244	0.2413	9.4662	0.4644	14.5714	5.2414	0.3455	0.4282	0.3119	9.4404
1967	8.1448	3.7954	16.8217	70.4698	0.3022	0.2185	9.4935	0.4608	16.2576	4.5439	0.3244	0.4644	0.2413	9.4662
1968	8.5660	4.2949	17.1392	80.3665	0.3036	0.2423	9.5475	0.4619	16.8217	3.7954	0.3022	0.4608	0.2185	9.4935
1969	11.7416	4.2953	19.7534	86.1986	0.3099	0.2227	9.6037	0.4726	17.1392	4.2949	0.3036	0.4619	0.2423	9.5475
1970	7.3457	3.3945	22.3586	66.7942	0.3048	0.0459	9.5592	0.4816	19.7534	4.2953	0.3099	0.4726	0.2227	9.6037
1971	12.9961	3.3975	21.6745	98.3157	0.4077	0.2004	9.8115	0.4876	22.3586	3.3945	0.3048	0.4816	0.0459	9.5592
1972	13.7900	4.4525	21.6790	107.1750	0.3607	0.2215	9.8132	0.4931	21.6745	3.3975	0.4077	0.4876	0.2004	9.8115
1973	15.3130	5.2543	21.8781	128.7056	0.3809	0.2099	9.9182	0.5305	21.6790	4.4525	0.3607	0.4931	0.2215	9.8132
1974	9.2478	3.3985	24.5583	114.4483	0.3878	0.0760	9.9266	0.5541	21.8781	5.2543	0.3809	0.5305	0.2099	9.9182
1975	11.6256	2.4013	24.7175	122.1282	0.3961	0.1100	9.9834	0.5242	24.5583	3.3985	0.3878	0.5541	0.0760	9.9266
1976	17.9336	5.5570	24.3210	167.0687	0.4115	0.2051	10.1041	0.5306	24.7175	2.4013	0.3961	0.5242	0.1100	9.9834
1977	20.0019	6.8103	28.7248	195.4323	0.4086	0.2244	10.1909	0.5301	24.3210	5.5570	0.4115	0.5306	0.2051	10.1041
1978	22.9187	6.0032	33.6705	223.0148	0.4258	0.2119	10.3287	0.5379	28.7248	6.8103	0.4086	0.5301	0.2244	10.1909
1979	21.0012	5.2539	40.2199	230.8880	0.4047	0.1448	10.3802	0.5441	33.6705	6.0032	0.4258	0.5379	0.2119	10.3287
1980	11.4936	2.9093	50.6284	192.1633	0.4848	-0.0343	10.4511	0.5594	40.2199	5.2539	0.4047	0.5441	0.1448	10.3802
1981	15.5674	2.3634	66.0044	206.4705	0.5455	0.0101	10.5711	0.5103	50.6284	2.9093	0.4848	0.5594	-0.0343	10.4511
1982	17.6421	2.3649	69.0837	189.1997	0.5583	0.0232	10.6310	0.4923	66.0044	2.3634	0.5455	0.5103	0.0101	10.5711
1983	28.0638	2.7925	60.8639	238.2435	0.5455	0.1205	10.7297	0.5152	69.0837	2.3649	0.5583	0.4923	0.0232	10.6310

(continued)

fyear	pstar	div	inv	q	debtratio	et	lna	debtratio_peer	invlag_1	divlag_1	ebtratiolag_	debtratiolag peer	etlag	lnalag
1984	30.0178	4.7903	61.5003	268.2601	0.5356	0.0901	10.8618	0.5033	60.8639	2.7925	0.5455	0.5152	0.1205	10.7297
1985	32.2438	5.0766	77.9633	307.6458	0.5375	0.0660	11.0640	0.5969	61.5003	4.7903	0.5356	0.5033	0.0901	10.8618
1986	30.0407	5.2097	95.7772	320.9095	0.5774	0.0374	11.1926	0.6127	77.9633	5.0766	0.5375	0.5969	0.0660	11.0640
1987	31.0784	5.3279	103.1967	330.0916	0.6199	0.0294	11.3785	0.5799	95.7772	5.2097	0.5774	0.6127	0.0374	11.1926
1988	38.1362	5.3237	120.5252	404.2870	0.7826	0.0744	12.0080	0.6298	103.1967	5.3279	0.6199	0.5799	0.0294	11.3785
1989	18.7529	3.1863	64.5974	206.4430	0.7886	0.0766	12.0628	0.6199	120.5252	5.3237	0.7826	0.6298	0.0744	12.0080
1990	8.8140	3.1713	69.3982	206.2301	0.8216	0.0472	12.1020	0.6483	64.5974	3.1863	0.7886	0.6199	0.0766	12.0628
1991	4.5951	1.7718	72.4874	197.6762	0.8447	0.0236	12.1245	1.2313	69.3982	3.1713	0.8216	0.6483	0.0472	12.1020
1992	8.5340	1.5206	66.1788	183.9771	0.9634	0.0221	12.1601	2.2858	72.4874	1.7718	0.8447	1.2313	0.0236	12.1245
1993	16.0407	1.0097	65.7128	187.3181	0.9679	0.0405	12.1453	0.5685	66.1788	1.5206	0.9634	2.2858	0.0221	12.1601
1994	20.3661	1.0489	72.7017	203.5847	0.9332	0.0581	12.1990	0.5353	65.7128	1.0097	0.9679	0.5685	0.0405	12.1453
1995	24.5531	1.4840	86.9076	221.7449	0.8925	0.0612	12.2882	0.5368	72.7017	1.0489	0.9332	0.5353	0.0581	12.1990
1996	21.8776	2.0222	89.3659	208.8503	0.8946	0.0382	12.3111	0.5786	86.9076	1.4840	0.8925	0.5368	0.0612	12.2882
1997	27.2202	2.3361	97.8707	243.8843	0.9203	0.0519	12.3410	0.7819	89.3659	2.0222	0.8946	0.5786	0.0382	12.3111
1998	22.8623	2.1191	109.1803	242.1635	0.9392	0.0369	12.4583	1.1206	97.8707	2.3361	0.9203	0.7819	0.0519	12.3410
1999	28.7838	2.2069	122.8843	288.6657	0.9227	0.0562	12.5235	0.6431	109.1803	2.1191	0.9392	1.1206	0.0369	12.4583
2000	32.4783	2.3605	142.0021	330.0820	0.8981	0.0464	12.6218	0.6408	122.8843	2.2069	0.9227	0.6431	0.0562	12.5235
2001	24.1037	2.1483	131.9002	319.9569	0.9369	0.0253	12.6884	0.7494	142.0021	2.3605	0.8981	0.6408	0.0464	12.6218
2002	26.1559	2.0840	129.8675	336.3791	0.9794	0.0195	12.8234	0.7190	131.9002	2.1483	0.9369	0.7494	0.0253	12.6884
2003	29.9645	1.9947	129.1713	334.5991	0.9430	0.0233	13.0137	0.6427	129.8675	2.0840	0.9794	0.7190	0.0195	12.8234
2004	30.0036	1.9978	132.8610	340.4939	0.9422	0.0252	13.0814	0.6392	129.1713	1.9947	0.9430	0.6427	0.0233	13.0137
2005	9.3914	2.0052	138.6357	343.4957	0.9672	-0.0075	13.0733	0.6263	132.8610	1.9978	0.9422	0.6392	0.0252	13.0814
2006	15.8608	0.9953	105.8250	327.1360	1.0228	0.0726	12.1345	0.6057	138.6357	2.0052	0.9672	0.6263	-0.0075	13.0733
2007	-59.6828	1.0017	87.8513	321.7686	1.2383	-0.0023	11.9109	0.6066	105.8250	0.9953	1.0228	0.6057	0.0726	12.1345
2008	-34.2827	0.4636	68.5965	240.9273	1.9373	-0.1316	11.4191	0.7057	87.8513	1.0017	1.2383	0.6066	-0.0023	11.9109
2009	224.9040	0.0000	37.3800	203.3080	0.7876	-0.0895	11.8226	0.6845	68.5965	0.4636	1.9373	0.7057	-0.1316	11.4191
2010	7.5426	0.0000	12.8222	91.7316	0.7325	0.0429	11.8415	0.6697	37.3800	0.0000	0.7876	0.6845	-0.0895	11.8226
2011	9.0923	0.0000	15.1733	97.4451	0.7304	0.0492	11.8818	0.6128	12.8222	0.0000	0.7325	0.6697	0.0429	11.8415
2012	8.2093	0.0000	18.9150	111.7161	0.7524	0.0037	11.9145	0.6016	15.1733	0.0000	0.7304	0.6128	0.0492	11.8818
2013	6.4520	0.0000	19.5000	103.1680	0.7405	0.0403	12.0218	0.6003	18.9150	0.0000	0.7524	0.6016	0.0037	11.9145
2014	5.6513	1.2050	21.7519	97.2075	0.7973	0.0353	12.0877	0.6259	19.5000	0.0000	0.7405	0.6003	0.0403	12.0218
2015	11.2707	1.4493	34.2673	101.6520	0.7927	0.0504	12.1783	0.6661	21.7519	1.2050	0.7973	0.6259	0.0353	12.0877
2016	13.0247	1.5580	46.8973	110.9360	0.8012	0.0539	12.3090	0.6344	34.2673	1.4493	0.7927	0.6661	0.0504	12.1783
2017	8.7964	1.5821	56.5250	101.7593	0.8296	0.0619	12.2666	0.6303	46.8973	1.5580	0.8012	0.6344	0.0539	12.3090
2018	15.1614	1.5314	58.7979	104.4300	0.8118	0.0323	12.3342	0.6618	56.5250	1.5821	0.8296	0.6303	0.0619	12.2666
2019	13.9421	1.5464	58.5036	98.4422	0.7985	0.0260	12.3373	0.6853	58.7979	1.5314	0.8118	0.6618	0.0323	12.3342

Appendix 22.2: Applications of R Language in Estimating the Parameters of a System of Simultaneous Equations

In this appendix, we show the estimation procedure on how to apply 2SLS, 3SLS, and GMM techniques for estimating the parameters of a system of simultaneous equations through R language. The structural equations of this chapter are constructed as follows:

$$
\begin{aligned}
Inv_{it} &= \alpha_1 + \alpha_2 Div_{it} + \alpha_3 Debt_{it} + \alpha_4 Inv_{i,t-1} + \alpha_5 Q_{it} + \epsilon_{it} \\
Div_{it} &= \beta_1 + \beta_2 Inv_{it} + \beta_3 Debt_{it} + \beta_{4i} Div_{i,t-1} + \beta_{5i} P_{it} + \eta_{it} \\
Debt_{it} &= \gamma_1 + \gamma_2 Inv_{it} + \gamma_3 Div_{it} + \gamma_4 Debt_{i,t-1} + \gamma_5 ln\left(A_{i,t-1}\right) \\
&\quad + \gamma_6 \left(E_{i,t-1}/A_{i,t-1}\right) + \xi_{it}
\end{aligned}
$$

The three endogenous variables are investment (Inv_{it}), dividend (Div_{it}), and debt financing ($Leverage_{it}$) of firm i in year t. In addition to lag-terms of the three policies, we respectively incorporate sales plus the change in inventories (Q_{it}) and net income minus preferred dividends (P_{it}) into investment and dividend decisions. Moreover, we add the natural logarithm of lagged total assets ($\ln A_{i,t-1}$) and the lag of earnings before interest and taxes divided by total assets ($E_{i,t-1}/A_{i,t-1}$) as the determinants of leverage. Based upon Johnson & Johnson, and IBM companies' annual data from 1966 to 2019 as presented in Appendix A, the procedure to estimate the interrelationship among corporate investment, leverage, and dividend payout policies in a simultaneous-equation system is provided as follows.

First, we load the data into the R environment and apply the 2SLS method to estimate the parameters of this simultaneous-equations system. Here, we use all the exogenous variables $(Inv_{i,t-1}, Q_{it}, Div_{i,t-1}, P_{it}, Debt_{i,t-1}, \ln(A_{i,t-1}), E_{i,t-1}/A_{i,t-1})$ in this system as instruments to obtain the prediction of each endogenous variable. By using `ivreg` package in R language, we obtain the following program code.

```
Data <- read.csv(file=``IBM.csv'')
library(ivreg)
```

Investment policy

```
INVeq <- ivreg(inv ~ div + debtratio + invlag_1 + q |
       invlag_1 + q+ divlag_1 + pstar+ debtratio-
lag_1 + lnalag + etlag, data=Data)
summary(INVeq)
```

Dividend policy

```
DIVeq <- ivreg(div ~ debtratio + inv + di-
vlag_1 + pstar |
       invlag_1 + q+ divlag_1 + pstar+ debtratio-
lag_1 + lnalag + etlag, data=Data)
```

summary(DIVeq)

Financing policy

```
FINeq <- ivreg(debtratio ~ inv + div + debtratio-
lag_1 + lnalag + etlag |
       invlag_1 + q+ divlag_1 + pstar+ debtratio-
lag_1 + lnalag + etlag, data=Data)
summary(FINeq)
```

Next, we introduce how to apply 3SLS and GMM methods to estimate the parameters of this simultaneous-equations system through the `gmm` package in R language. We first should specify a simultaneous equation system for investment policy, financing policy, and dividend policy and the instruments used in this system.

Specify the simultaneous-equation system

```
EqInv <- inv ~ div + debtratio + invlag_1 + q
EqDiv <- div ~ debtratio + inv + divlag_1 + pstar
EqFin <- debtratio ~ inv + div + debtratiolag_1 + lnalag
+ etlag
EqSystem <- list(Investment = EqInv, Dividend = EqDiv,
Financing = EqFin )
```

Specify the instruments

```
Insts <- list( ~ invlag_1 + q+ divlag_1 + pstar + deb-
tratiolag_1 + lnalag + etlag)
```

After specifying the simultaneous-equation system and his instruments, we then introduce how to apply 3SLS method to estimate the parameters of this simultaneous-equations system in R language. By using the `threeSLS` function in `gmm` package, we obtain the following program.

```
library(gmm)
```

3SLS method

```
threeSLS.fit <- threeSLS(EqSystem, Insts, data=Data)
summary( threeSLS.fit )
```

Finally, we apply the GMM method to estimate the parameters of this simultaneous-equations system. By using the `sysGmm` function in `gmm` package, we obtain the following program.

GMM method

```
GMM.fit <- sysGmm(EqSystem, Insts, data=Data)
summary(GMM.fit)
```

Notably, the default results list in `gmm` package does not provide the R-Squared and Adjusted R-Squared statistics. R language allows users to define their own functions and procedures to compute desired statistics. We here show how to define a `R_square` function in `gmm` package to calculate the R-squared and adjusted R-squared statistics as follows.

Define a function to calculate R-squared and adjusted R-squared statistics

```
R_square <- function(Model,i){
 TrueY = Model$fitted.values[[i]]+Model$residuals
[[i]]
 PredY = Model$fitted.values[[i]]
 R2 = cor(TrueY, PredY)^2
 Obs = length(Model$fitted.values[[i]])
 ParaNum = length(Model$coefficients[[i]])
 adj_R2 = 1-((1-R2)*( Obs -1)/( Obs - ParaNum -1))
 return(list(R2,adj_R2))
}
```

For example, after fitting a 3SLS model and saving results in `threeSLS.fit`, we can then call `R_square` function to calculate R-squared and adjusted R-squared statistics as follows.

```
R_square(threeSLS.fit,1) # 1st equation: Investment policy
R_square(threeSLS.fit,2) # 2nd equation: Dividend policy
R_square(threeSLS.fit,3) # 3rd equation: Financing policy
```

Similarly, by using the `R_square` function to GMM estimation results, we obtain the following program.

```
R_square(GMM.fit.fit,1) # Investment equation
R_square(GMM.fit.fit,2) # Dividend equation
R_square(GMM.fit.fit,3) # Financing equation
```

References

Aggarwal R, Kyaw NA (2010) Capital structure, dividend policy, and multinationality:

Berger AN, Bonaccorsi di Patti E (2006) Capital structure and firm performance: A new approach to testing agency theory and an application to the banking industry. J Bank Financ 30:1065–1102

Bhagat S, Black BS (2002) The non-correlation between board independence and long-term firm performance. J Corp Law 27:231–273

Billett MT, Xue H (2007) The takeover deterrent effect of open market share repurchases. J Financ 62:1827–1850

Billett MT, King THD, Mauer DC (2007) Growth opportunities and the choice of leverage, debt maturity, and covenants. J Financ 62:697–730

Boone AL, Field LC, Karpoff JM, Raheja CG (2007) The determinants of corporate board size and composition: An empirical analysis. J Financ Econ 85:66–101

Bound J, Jaeger D, Baker R (1995) Problems with instrumental variable estimation when the correlation between the instruments and the endogenous explanatory variables is weak. J Am Stat Assoc 90:443–450

Chen CR, Lee CF (2010) Application of simultaneous equation in finance research. In: Lee CF et al (eds) Handbook of quantitative finance and risk management. Springer, Berlin, pp 1301–1306.

Demsetz H, Villalonga B (2001) Ownership structure and corporate performance. J Corp Financ 7:209–233

Dhrymes PJ, Kurz M (1967) Investment, dividend, and external finance behavior of firms. In: Ferber R (ed) Determinants of investment behavior. NBER, pp 427–486

Fama EF (1974) The empirical relationships between the dividend and investment decisions of firms. Am Econ Rev 64:304–318

Fama EF, French KR (2002) Testing trade-off and pecking order predictions about dividends and debt. Rev Financ Stud 15:1–33

Ferreira MA, Matos P (2008) The colors of investors' money: The role of institutional investors around the world. J Financ Econ 88:499–533

Fich EM, Shivdasani A (2007) Financial fraud, director reputation, and shareholder wealth. J Financ Econ 86:306–336

Flannery MJ, Rangan KP (2006) Partial adjustment toward target capital structures. J Financ Econ 79:469–506

Frank MZ, Goyal VK (2009) Capital structure decisions: Which factors are reliably important? Financ Manag 38:1–37

Froot KA, Scharfstein DS, Stein JC (1993) Risk management: Coordinating corporate investment and financing policies. J Financ 48:1629–1658

Grabowski HG, Mueller DC (1972) Managerial and stockholder welfare models of firm expenditures. Rev Econ Stat 54:9–24

Greene WH (2011) Econometric analysis, 7th edn. Prentice Hall, New Jersey

Gugler K (2003) Corporate governance, dividend payout policy, and the interrelation between dividends, R&D, and capital investment. J Bank Financ 27:1297–1321

Hahn J, Hausman J (2005) Instrumental variable estimation with valid and invalid instruments. Working Paper.

Hansen LP (1982) Large sample properties of generalized method of moments estimators. Econometrica 50:1029–1054

Harford J, Klasa S, Maxwell WF (2014) Refinancing risk and cash holdings. J Financ 69:975–1012

Harvey CR, Lins KV, Roper AH (2004) The effect of capital structure when expected agency costs are extreme. J Financ Econ 74:3–30

Higgins RC (1972) The corporate dividend-saving decision. J Financ Quant Anal 7:1527–1541

Intriligator MD, Bodkin RG, Hsiao C (1996) Econometric models, techniques, and applications, 2nd edn. Prentice Hall, New Jersey

John K, Nachman DC (1985) Risky debt, investment incentives, and reputation in a sequential equilibrium. J Financ 40:863–878

Johnston J, DiNardo J (1997) Econometric methods. McGraw-Hill, New York

Lee CF, Chen HY, Lee J (2019) Financial econometrics, mathematics and statistics. Springer.

Lee CF, Liang Wl, Lin FL, Yang Y (2016) Applications of simultaneous equations in finance research: Methods and empirical results. Rev Quant Finan Acc, 47: 943–971

Lee, CF, Lee J (2020) Handbook of financial econometrics, mathematics, statistics, and machine learning. World Scientific, Singapore

Lee CF, Chen HY, Lee J (2019) Financial econometrics, mathematics and statistics. Springer.

Lee CF, Lin FL (2020) Impacts of measurement errors on simultaneous equation estimation of dividend and investment decisions. Handbook of Financial Econometrics, Mathematics, Statistics, and Machine Learning (Vol. IV), Chapter 116, 4001–4023. World Scientific, Singapore

Loderer C, Martin K (1997) Executive stock ownership and performance tracking faint traces. J Financ Econ 45:223–255

MacKay P, Phillips GM (2005) How does industry affect firm financial structure? Rev Financ Stud 18:1433–1466

McCabe GM (1979) The empirical relationship between investment and financing: A new look. J Financ Quant Anal 14:119–135

McDonald JG, Jacquillat B, Nussenbaum M (1975) Dividend, investment and financing decisions: Empirical evidence on French firms. J Financ Quant Anal 10:741–755

Morgan IG, Saint-Pierre J (1978) Dividend and investment decisions of Canadian firms. Can J Econ 11:20–37

Peterson PP, Benesh GA (1983) A reexamination of the empirical relationship between investment and financing decisions. J Financ Quant Anal 18:439–453

Prevost AK, Rao RP, Hossain M (2002) Determinants of board composition in New Zealand: A simultaneous equations approach. J Empir Financ 9:373–397

Ruland W, and Zhou P (2005) Debt, diversification, and valuation. Rev Quant Financ Acc 25:277–291

Sargan JD (1958) The estimation of economic relationships using instrumental variables. Econometrica 26:393–415

Sargan JD (1959) The estimation of relationships with autocorrelated residuals by the use of instrumental variables. J R Stat Soc B 21:91–105

Staiger D, Stock JH (1997) Instrumental variables regression with weak instruments. Econom 65:557–586

Stock JH, Yogo M (2005) Testing for weak instruments in linear IV regression. In: Andrews DWK (ed.) Identification and inference for econometric models. Cambridge Univ. Press, New York, pp 80–108

Stock JH, Wright JH, Yogo M (2002) A survey of weak instruments and weak identification in generalized method of moments. J Bus Econ Stat 20:518–529

Switzer L (1984) The determinants of industrial R&D: A funds flow simultaneous equation approach. Rev Econ Stat 66:163–168

Wang, CJ (2015) Instrumental variables approach to correct for endogeneity in finance. In: Lee CF and Lee J (eds.) Handbook of financial econometrics and statistics. Springer, New York, pp 2577–2600

Woidtke T (2002) Agents watching agents?: evidence from pension fund ownership and firm value. J Financ Econ 63:99–131

Ye P (2012) The value of active investing: Can active institutional investors remove excess comovement of stock returns? J Financ Quant Anal 47:667–688

23 Three Alternative Programs to Estimate Binomial Option Pricing Model and Black and Scholes Option Pricing Model

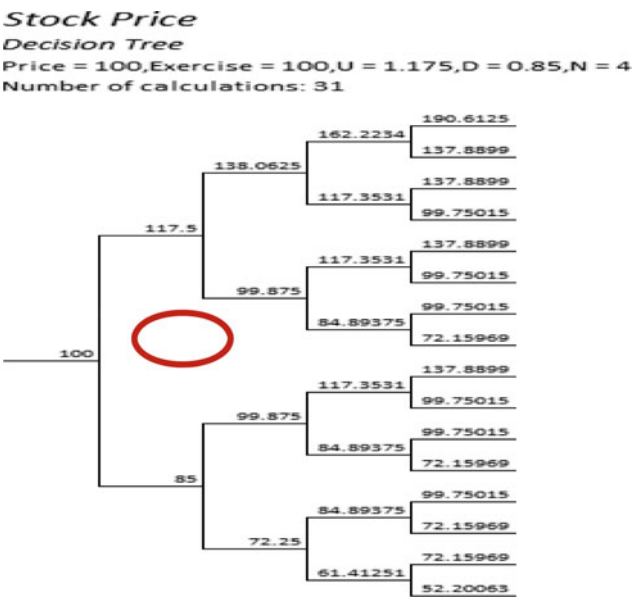
23.1 Introduction

In Chap. 5, we use Microsoft Excel programs to create large decision trees for the binomial pricing model to compute the prices of call and put options. In this chapter, we are going to present Microsoft Excel programs as well as R codes for call and put options prices in the following cases: (a) Black and Scholes model for individual stock, (b) Black and Scholes model for stock indices, and (c) Black and Scholes model for currencies. Section 23.2 presents an option pricing model for using Microsoft Excel program, Sects. 23.3–23.5 present Microsoft Excel programs for the computation of option prices for individual stocks, stock indices, and currencies, respectively. Section 23.6 presents the R codes to compute the option prices by the binomial tree model. Section 23.7 presents the R codes to compute the option prices by the Black and Scholes model. Section 23.8 summarizes this chapter. Appendix 23.1 presents the SAS program to calculate option prices using the binomial tree model. Appendix 23.2 presents the SAS program to calculate option prices using the Black and Scholes model.

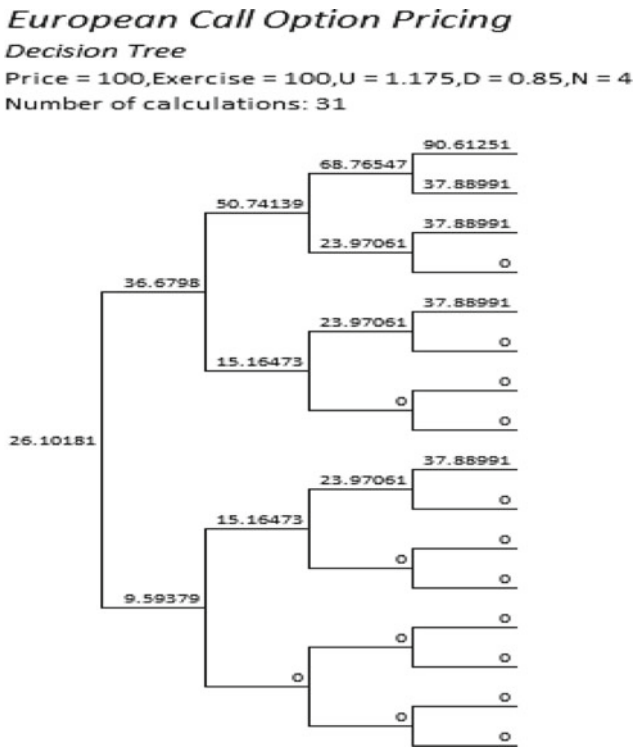
23.2 Microsoft Excel Program for the Binomial Tree Option Pricing Model

In Chap. 5, we priced the value of a call and put option by pricing backwards, from the last period to the first period. This method of pricing call and put options will work for any n period. To price the value of call options for two periods required seven sets of calculations. The number of calculations increases dramatically as n increases. Table 23.1 lists the number of calculations for a specific number of periods.

After two periods, it becomes very cumbersome to calculate and create the decision trees for a call and put option. To solve this problem, Microsoft Excel program *binomialoptionpricingmodel.xlsm* is developed to do the calculations and create the decision trees: (1) Stock Price; (2) Call Option Price; and (3) Put Option Price presented as follows:



© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
J. Lee et al., *Essentials of Excel VBA, Python, and R*,
https://doi.org/10.1007/978-3-031-14283-3_23



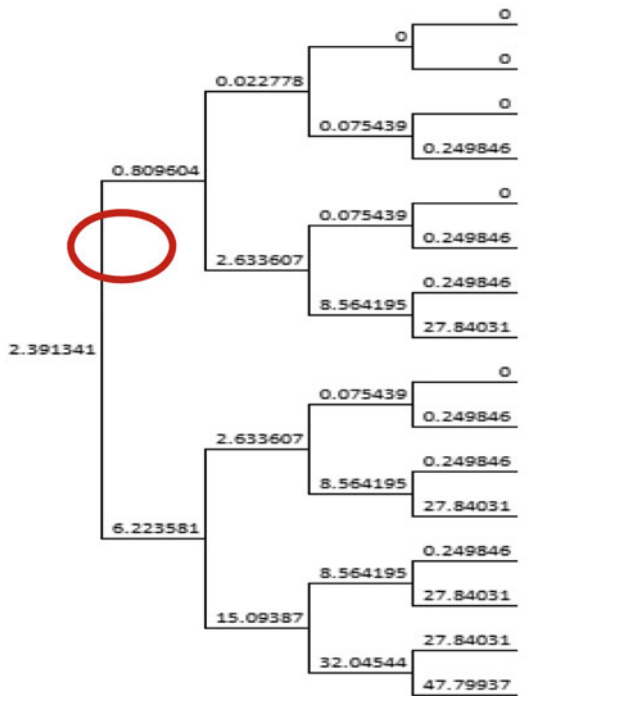


Table 23.1 Number of calculations for specific number of periods

Periods	Calculations
1	3
2	7
3	15
4	31
5	63
6	127
7	255
8	511
9	1023
10	2047
11	4065
12	8191

Benninga (2000, p 260) defined the price of a call option in a binomial option pricing model with n periods as

$$C = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[S(1+u)^i(1+d)^{n-i} - X, 0] \quad (23.1)$$

and the price of a put option in a binomial option pricing model with n periods as

$$P = \sum_{i=0}^{n} \binom{n}{i} q_u^i q_d^{n-i} \max[X - S(1+u)^i(1+d)^{n-i}, 0]. \quad (23.2)$$

Lee et al. (2000, p 237) defined the pricing of a call option in a binomial option pricing model with n period as

$$C = \frac{1}{R^n} \sum_{k=0}^{n} \frac{n!}{k!(n-k!)} p^k (1-p)^{n-k} \max[0, (1+u)^k (1+d)^{n-k} S - X]. \quad (23.3)$$

The definition of the pricing of a put option in a binomial option pricing model with n period would then be defined as

$$P = \frac{1}{R^n} \sum_{k=0}^{n} \frac{n!}{k!(n-k!)} p^k (1-p)^{n-k} \max[0, X - (1+u)^k (1+d)^{n-k} S]. \quad (23.4)$$

23.3 Black and Scholes Option Pricing Model for Individual Stock

The call option formula for an individual stock can be defined as

$$C = SN(d_1) - Xe^{-r(T)} N(d_2), \quad (23.5)$$

Table. 23.2 The inputs of European Call and Put options

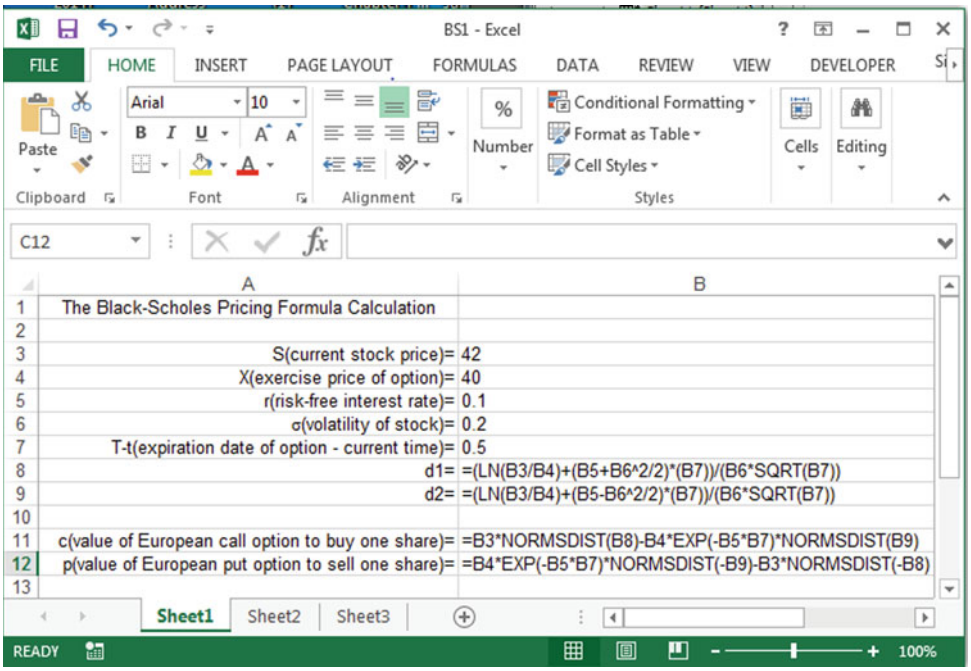
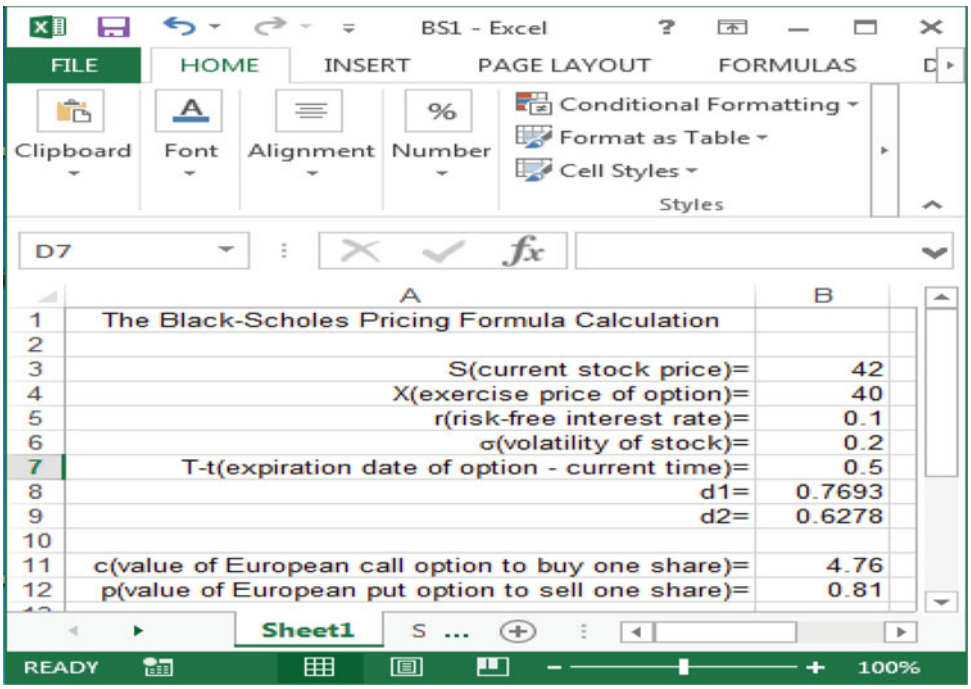


Table. 23.3 The outputs of European Call and Put options



where

$$d_1 = \frac{\ln\left(\frac{S}{X}\right) + \left(r + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}},$$

$$d_2 = \frac{\ln\left(\frac{S}{X}\right) + \left(r - \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T},$$

C = price of the call option,
S = current price of the stock,
X = exercise price of the option,
e = 2.71828,
r = short-term interest rate (T-Bill rate) = R_f,
T = time to expiration of the option, in years,
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1,2), and.
σ^2 = variance of the stock rate of return.

The put option formula can be defined as

$$P = Xe^{-r(T)}N(-d_2) - SN(-d_1), \tag{23.6}$$

where P = price of the put option. The following shows how to set up Microsoft Excel to solve the problem by assuming S = 42, X = 40, r = 0.1, σ = 0.2, and T = 0.5.

The following shows the outputs:

From the Excel output, we find that the price of a call option and a put option is \$4.76 and \$0.81, respectively.

23.4 Black and Scholes Option Pricing Model for Stock Indices

The call option formula for a stock index can be defined as

$$C = Se^{-q(T)}N(d_1) - Xe^{-r(T)}N(d_2), \tag{23.7}$$

where

$$d_1 = \frac{\ln(S/X) + (r - q + \frac{\sigma^2}{2})(T)}{\sigma\sqrt{T}},$$

$$d_2 = \frac{\ln(S/X) + (r - q - \frac{\sigma^2}{2})(T)}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T},$$

q = dividend yield,
S = value of index,
X = exercise price,
r = short-term interest rate (T-Bill rate) = R_f,
T = time to expiration of the option, in years,
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1, 2), and.
σ^2 = variance of the stock rate of return.

The put option formula for a stock index can be defined as

$$P = Xe^{-r(T)}N(-d_2) - Se^{-q(T)}N(-d_1), \tag{23.8}$$

where P = the price of the put option. The following shows how to set up Microsoft Excel to solve the problem by assuming that S = 950, X = 900, r = 0.06, σ = 0.15, q = 0.03, and T = 2/12.

The following shows the outputs:

From the Excel output, we find that the price of a call option and a put option is \$59.26 and \$5.01, respectively.

23.5 Black and Scholes Option Pricing Model for Currencies

The call option formula for a currency can be defined as

$$C = Se^{-r_f(T)}N(d_1) - Xe^{-r(T)}N(d_2),$$

where

$$d_1 = \frac{\ln(S/X) + (r - r_f + \frac{\sigma^2}{2})(T)}{\sigma\sqrt{T}},$$

$$d_2 = \frac{\ln(S/X) + (r - r_f - \frac{\sigma^2}{2})(T)}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T},$$

S = spot exchange rate,
r = risk-free rate for domestic country,
X = exercise price,
T = time to expiration of the option, in years,
$N(d_i)$ = value of the cumulative standard normal distribution (i = 1, 2), and.
σ = standard deviation of spot rate.

The put option formula for a currency can be defined as

$$P = Xe^{-r(T)}N(-d_2) - Se^{-r_f(T)}N(-d_1), \tag{23.9}$$

where P = the price of the put option. Assume that S = 130, X = 125, r = 0.06, r_f = 0.02, σ = 0.15, and T = 4/12. The following shows how to set up Microsoft Excel to solve the problem.

The following shows the outputs of Microsoft Excel program:

From the Excel output, we find that the price of a call option and a put option is \$8.43 and \$1.82, respectively.

23.6 R Codes to Implement the Binomial Trees Option Pricing Model

The current stock price is S = \$100 and the strike price is X = \$100. The increase and decrease factors are u = 1.175 and d = 0.85, respectively. The interest rate is r = 7%. The

Table. 23.4 The inputs of European call and put options

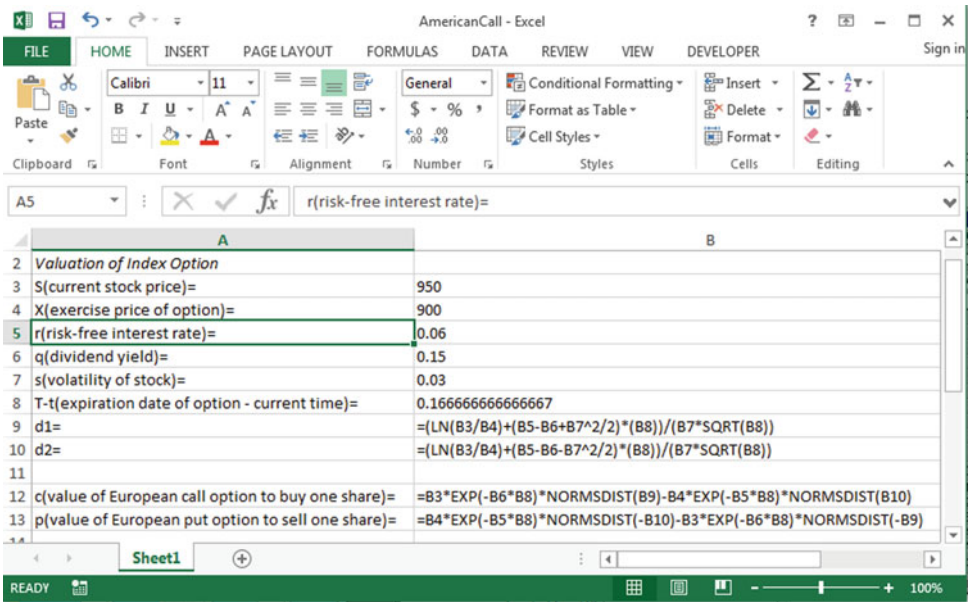


Table. 23.5 Results for functions contained in Table 23.5

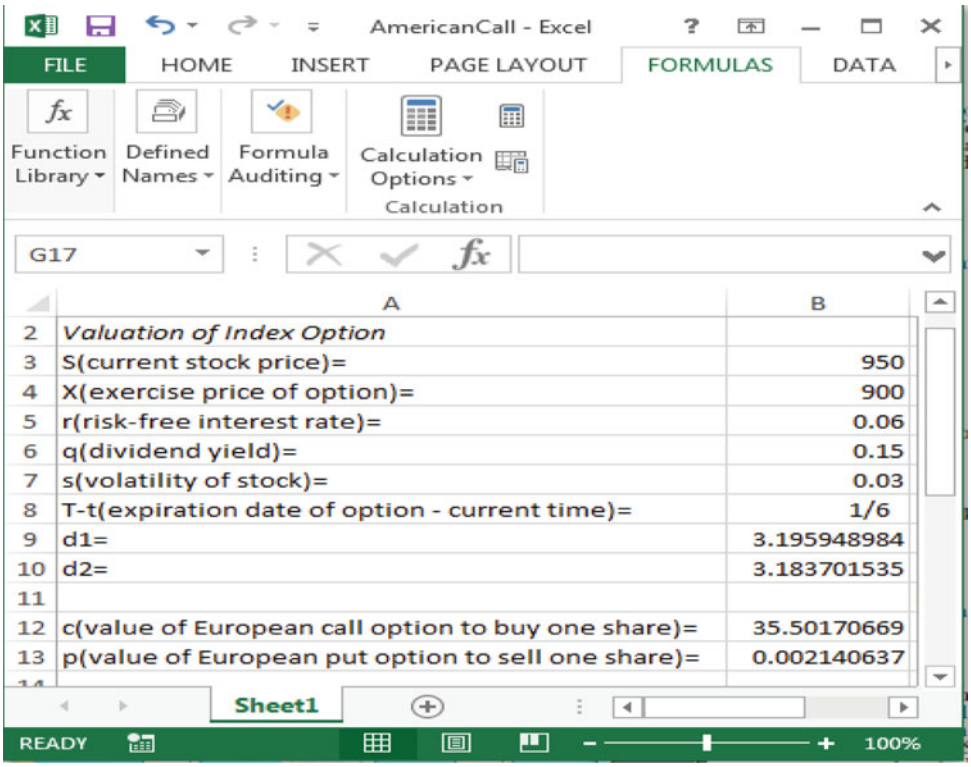


Table. 23.6 The inputs and excel functions of European call and put options

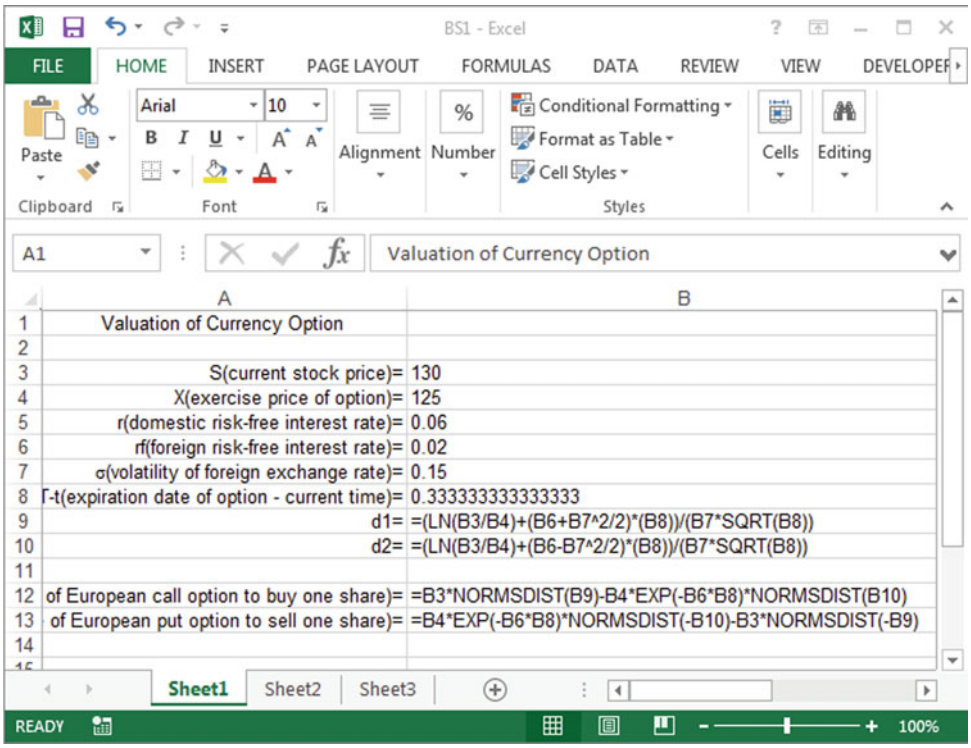
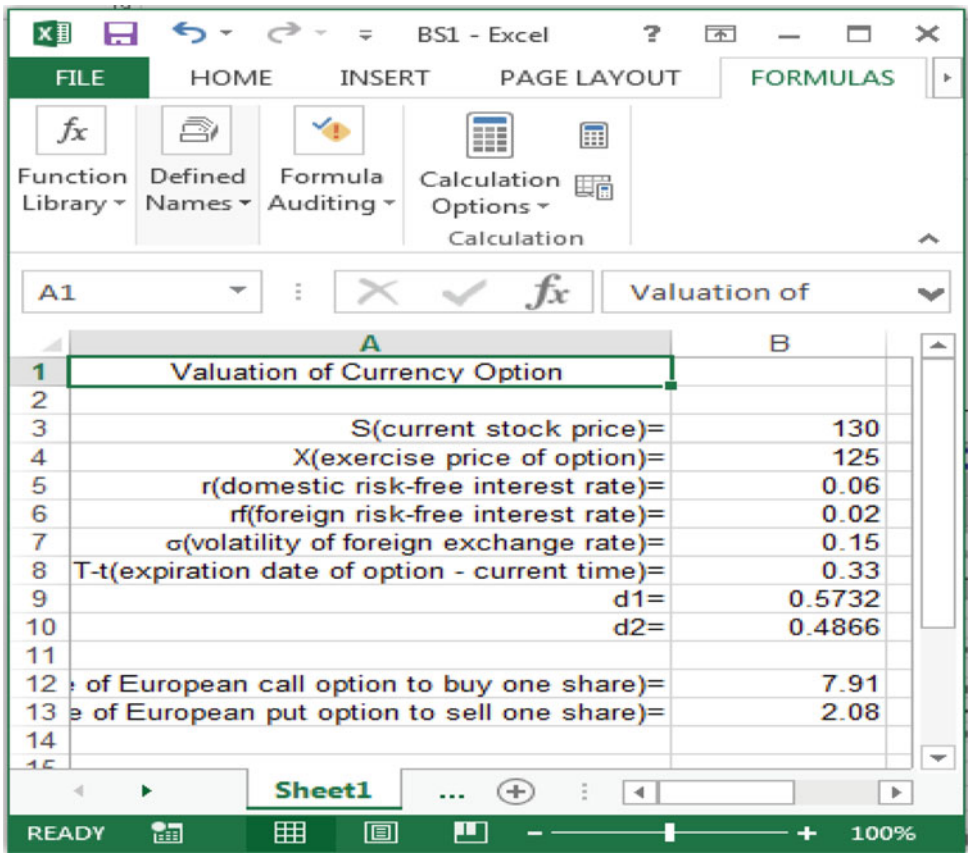


Table. 23.7 Results for functions contained in Table 23.6



$n = 5$ period binomial trees can be created by the following *R* code:

```
build_stock_tree <- function(S, u, d, n) {
        tree = matrix(0, nrow=n+1, ncol=n+1)
        for (i in 1: (n+1)) {
            for (j in 1: i) {
               tree[i, j] = S * u^(j-1) * d^((i-1)-(j-1))
        }   }
        return(tree)
}

q_prob <- function(r, u, d) {
          return((exp(r) - d)/(u-d))
}

value_binomial_option <- function (tree, r, u, d, X, type) {

       q = q_prob(r, u, d)
       option_tree = matrix(0, nrow=nrow(tree), ncol=ncol(tree))

        if (type == 'put') {
            option_tree[nrow(option_tree),] = pmax (X - tree[nrow(tree),], 0)
        } else { option_tree[nrow(option_tree),] = pmax(tree[nrow(tree),] - X, 0) }

       for (i in (nrow(tree)-1):1) {
            for (j in 1: i) {
              option_tree[i, j]=((1-q)*option_tree[i+1,j] +q*option_tree[i+1,
j+1])/exp(r)
                }
       }
       return (option_tree)
 }

binomial_option <- function (type, n, r, u, d, X, S) {

       q <- q_prob(r, u, d)
       tree <- build_stock_tree(S, u, d, n)

       option <- value_binomial_option(tree, r, u, d, X, type)
       return(list(stock_tree=tree, option_tree=option, option_price=option[1,1]))
}
```

The output is shown below:

(i) European call option:

```
> binomial_option <- function (type, n, r, u, d, X, S) {
+
+     q <- q_prob(r, u, d)
+     tree <- build_stock_tree(S, u, d, n)
+
+     option <- value_binomial_option(tree, r, u, d, X, type)
+     return(list(stock_tree=tree, option_tree=option, option_price=option[1,1]))
+ }
>
>
> c<-binomial_option(type, n, r, u, d, X, S)
> c
$stock_tree
          [,1]      [,2]      [,3]     [,4]     [,5]     [,6]
[1,] 100.00000   0.00000   0.00000   0.0000   0.0000   0.0000
[2,]  85.00000 117.50000   0.00000   0.0000   0.0000   0.0000
[3,]  72.25000  99.87500 138.06250   0.0000   0.0000   0.0000
[4,]  61.41250  84.89375 117.35313 162.2234   0.0000   0.0000
[5,]  52.20062  72.15969  99.75016 137.8899 190.6125   0.0000
[6,]  44.37053  61.33573  84.78763 117.2064 162.0207 223.9697

$option_tree
          [,1]      [,2]     [,3]     [,4]     [,5]     [,6]
[1,] 32.151885  0.000000  0.00000  0.00000  0.00000   0.0000
[2,] 15.563065 43.198142  0.00000  0.00000  0.00000   0.0000
[3,]  4.475864 22.318293 57.39081  0.00000  0.00000   0.0000
[4,]  0.000000  7.011564 31.73255 75.28761  0.00000   0.0000
[5,]  0.000000  0.000000 10.98381 44.65054 97.37316   0.0000
[6,]  0.000000  0.000000  0.00000 17.20643 62.02066 123.9697

$option_price
[1] 32.15188
```

(ii) European put option:

```
> c
$stock_tree
          [,1]      [,2]      [,3]     [,4]     [,5]     [,6]
[1,] 100.00000   0.00000   0.00000   0.0000   0.0000   0.0000
[2,]  85.00000 117.50000   0.00000   0.0000   0.0000   0.0000
[3,]  72.25000  99.87500 138.06250   0.0000   0.0000   0.0000
[4,]  61.41250  84.89375 117.35313 162.2234   0.0000   0.0000
[5,]  52.20062  72.15969  99.75016 137.8899 190.6125   0.0000
[6,]  44.37053  61.33573  84.78763 117.2064 162.0207 223.9697

$option_tree
          [,1]      [,2]       [,3] [,4] [,5] [,6]
[1,]  2.620694  0.000000  0.0000000    0    0    0
[2,]  6.141439  1.276516  0.0000000    0    0    0
[3,] 13.284289  3.501718  0.3867341    0    0    0
[4,] 25.523324  9.053637  1.3152468    0    0    0
[5,] 41.038757 21.079694  4.4730322    0    0    0
[6,] 55.629469 38.664266 15.2123672    0    0    0

$option_price
[1] 2.620694
```

23.7 R Codes to Compute Option Prices by Black and Scholes Model

In this section, we write R codes to price options for individual stocks, stock indices, and currencies. We use same examples in the previous sections to show the results in R.

```
BlackScholes <- function(S, K, r, T, sig, type){
    if(type=="C"){
        d1 <- (log(S/K) + (r + sig^2/2)*T) / (sig*sqrt(T))
        d2 <- d1 - sig*sqrt(T)
        value <- S*pnorm(d1) - K*exp(-r*T)*pnorm(d2)
        return(value)
        }
     if(type=="P"){
        d1 <- (log(S/K) + (r + sig^2/2)*T) / (sig*sqrt(T))
        d2 <- d1 - sig*sqrt(T)
        value <- (K*exp(-r*T)*pnorm(-d2) - S*pnorm(-d1))
        return(value)
        }
     }
```

(i) Option Model for Individual Stock

The current stock price is $S = \$42$ and the strike price is $X = \$40$. The interest rate is $r = 10\%$, volatility sig = 0.2, and time-to-maturity = 0.5 yr, the European call and put prices are.

Obs	d1	d2	C(the European call option price)	P(the European put option price)
1	0.7693	0.6278	4.7594	0.8086

(ii) Option Model for Stock Indices

The current stock index is $S = \$950$ and the strike price is $X = \$900$. The interest rate is $r = 6\%$, volatility sig = 0.15, dividend = 3%, and time-to-maturity = 1/6 yr, the European call and put prices are.

Obs	d1	d2	C(the European call option price)	P(the European put option price)
1	0.9952	0.9339	59.2225	5.0055

(iii) Option for Currencies

The current currency rate $S = \$130$ and the strike is $X = \$125$. The interest rate is $r = 6\%$, volatility sig = 0.15, foreign rate = 2%, and time-to-maturity = 1/3 yr, the European call and put prices are:

Obs	d1	d2	C(the European call option price)	P(the European put option price)
1	0.6501	0.5635	8.4275	1.8162

23.8 Summary

In this chapter, we presented the binomial option pricing model and Black and Scholes option pricing model, then we showed how Excel can be used to estimate binomial option pricing model and Black and Scholes model for individual stock options, index options, and currency options. We also showed how R language can be used to estimate binomial and Black and Scholes option pricing models. Finally, in the appendices, we showed how SAS language programming can be used to estimate binomial option pricing model and Black and Scholes option pricing model.

Appendix 23.1: SAS Programming to Implement the Binomial Option Trees

The following SAS macro is used to implement binomial trees and calculate the price of a stock, a call option, a put option, and a risk-free bond price. The parameters of this macro are.

S: Stock Price,
X: Strike Price,
U: Incease Factor,
D: Decrease Factor,
N: Periods, and.
r: Interest.

```
%macro test(S, X, U, D, N, r);
data a;
%do i=1 %to &N.;
        output;
%end;
run;

/*Stock Price*/
data b;
set a;
array S(&N.);
num=_N_;
if num=1 then do;
        S1=&S;
end;

%do j=1 %to (&N.-1);
        if num=1 then do;
                S(&j.+1)=&U*S(&j.);
        end;
        newS=lag(S(&j.));
        if num>1 then do;
                S(&j.+1)=&D*newS;
        end;

%end;

drop newS num;
run;

proc print data=b;
run;
```

```
/*Call Option Pricing*/
data cpre;
set b;
v=&N.+1-_N_;
run;

proc sort data=cpre;
by v;
run;

data c;
set cpre;
array C(&N.);
C&N.=max(S&N.-&X.,0);
prob=(1+&r.-&D.)/(&U.-&D.);

%do k=&N. %to 2 %by -1;
        newC=lag(C&k.);
        C(&k.-1)=((1-prob)*newC+prob*C(&k.))/(1+&r.);
%end;
drop newC prob S1-S&N.;
run;

proc sort data=c;
by descending v;
run;

data c;
set c;
drop v;
run;
proc print data=c;
run;
```

```
/*Put Option Pricing*/
data dpre;
set b;
u=&N.+1-_N_;
run;

proc sort data=dpre;
by u;
run;

data d;
set dpre;
array P(&N.);
P&N.=max(&X.-S&N.,0);
prob=(1+&r.-&D.)/(&U.-&D.);

%do k=&N. %to 2 %by -1;
        newP=lag(P&k.);
        P(&k.-1)=((1-prob)*newP+prob*P(&k.))/(1+&r.);
%end;
drop newP prob S1-S&N.;
run;

proc sort data=d;
by descending u;
run;

data d;
set d;
drop u;
run;

proc print data=d;
run;
```

```
/*Bond Pricing*/
data e;
set a;
array B(&N.);
num=_N_;
if num=1 then do;
B1=1;
end;

%do q=1 %to (&N.-1);
        if num=1 then do;
                B(&q.+1)=(1+&r.)*B(&q.);
        end;
        newB=lag(B(&q.));
        if num>1 then do;
                B(&q.+1)=(1+&r.)*newB;
end;
%end;
drop newB num;
run;

proc print data=e;
run;

%mend;
```

Appendix 23.2: SAS Programming to Compute Option Prices Using Black and Scholes Model

In this section, we write SAS macro function code to price options for individual stocks, stock indices, and currencies. We use same examples in previous sections to show the results in SAS.

1. **Option Model for Individual Stock**

```
%macro Individual_Stock(S, X, r, sigma, T);
data Individual;
      d1=(log(&S./&X.)+(&r.+1/2*&sigma.**2)*&T.)/(&sigma.*sqrt(&T.));
       d2=(log(&S./&X.)+(&r.-1/2*&sigma.**2)*&T.)/(&sigma.*sqrt(&T.));
       C=&S.*CDF('NORMAL',d1)-&X.*exp(-&r.*&T.)*CDF('NORMAL',d2);
       P=&X.*exp(-&r.*&T.)*CDF('NORMAL',-d2)-&S.*CDF('NORMAL',-d1);
run;

proc print data=Individual label;
     var d1 d2 C P;
     label C="C(the European call option price)"
              P="P(the European put option price)";
run;
%mend;
```

2. **Option Model for Stock Indices**

```
%macro Stock_Indices(S, X, r, sigma, q, T);
data Indices;
      d1=(log(&S./&X.)+(&r.-&q.+1/2*&sigma.**2)*&T.)/(&sigma.*sqrt(&T.));
       d2=d1-&sigma.*sqrt(&T.);
       C=&S.*exp(-&q.*&T.)*CDF('NORMAL',d1)-&X.*exp(-
&r.*&T.)*CDF('NORMAL',d2);
       P=&X.*exp(-&r.*&T.)*CDF('NORMAL',-d2)-&S.*exp(-
&q.*&T.)*CDF('NORMAL',-d1);
run;
proc print data=Indices label;
     var d1 d2 C P;
     label C="C(the European call option price)"
              P="P(the European put option price)";
run;
%mend;
```

3. **Option for Currencies**

```
%macro Currencies(S, X, r, rf, sigma, T);
data Individual;
      d1=(log(&S./&X.)+(&r.-&rf.+1/2*&sigma.**2)*&T.)/(&sigma.*sqrt(&T.));
      d2=d1-&sigma.*sqrt(&T.);
      C=&S.*exp(-&rf.*&T.)*CDF('NORMAL',d1)-&X.*exp(-
&r.*&T.)*CDF('NORMAL',d2);
      P=&X.*exp(-&r.*&T.)*CDF('NORMAL',-d2)-&S.*exp(-
&rf.*&T.)*CDF('NORMAL',-d1);
run;
proc print data=Individual label;
     var d1 d2 C P;
     label C="C(the European call option price)"
           P="P(the European put option price)";
run;
%mend;
```

References

Anderson, T. W. *An Introduction to Multivariate Statistical Analysis*, 3rd ed. New York: Wiley-Interscience, 2003.

Benninga, S. *Financial Modeling*. Cambridge, MA: MIT Press, 2000.

Benninga, S. *Financial Modeling*. Cambridge, MA: MIT Press, 2008.

Black, F. and M. Scholes. "The Pricing of Options and Corporate Liabilities." *Journal of Political Economy*, v. 31 (May–June 1973), pp. 637–659.

Black, F. "The Pricing of Commodity Contracts." *Journal of Financial Economics*, v. 3 (January-March 1976), pp.167–178.

Cox, J. C. and S. A. Ross. "The Valuation of Options for Alternative Stochastic Processes." *Journal of Financial Economics*, v. 3 (1976), 145–166.

Cox, J., S. Ross and M. Rubinstein. "Option Pricing: A Simplified Approach." *Journal of Financial Economics*, v. 7 (1979), 229–263.

Hull, J. *Options, Futures, and Other Derivatives* 10th ed. Pearson: 2017.

Johnson, N. L. and S. Kotz. *Distributions in Statistics: continuous Univariate Distributions 2*. New York: Wiley, 1970.

Johnson, N. L. and S. Kotz. *Distributions in Statistics: Continuous Multivariate Distributions*. New York: Wiley, 1972.

Lee, J. C., "Using Microsoft Excel and Decision trees to Demonstrate the Binomial Option Pricing Model." *Advances in Investment Analysis and Portfolio Management*, v. 8 (2001), pp. 303–329.

Lee, C. F. and Lee, A.C. *Encyclopedia of Finance* 3rd ed., Springer, forthcoming 2022.

Rubinstein, M. "The Valuation of Uncertain Income Streams and the Pricing of Options." *Bell Journal of Economics and Management Science*, v. 7 (1976), 407– 425.

Stoll, H. R. "The Relationship between Put and Call Option Prices." *Journal of Finance*, v. 24 (December 1969), pp. 801–824.

Whaley, R. E. "On the Valuation of American Call Options on Stocks with Known Dividends." *Journal of Financial Economics*, v. 9 (1981), 207–211.