Machine Learning *with* Python

Principles and Practical Techniques

Parteek Bhatia

Machine Learning with Python

Machine learning (ML) has become a dominant problem-solving technique in the modern world, with applications ranging from search engines and social media to self-driving cars and artificial intelligence (AI). This lucid textbook is an effort to present the contents in a simplified and practical manner without losing the details of the subject.

The book presents the theoretical foundations of ML algorithms, and then illustrates each concept with its detailed implementation in Python to allow beginners to effectively implement the principles in real-world applications. All major techniques, such as regression, classification, clustering, deep learning, and association mining, have been illustrated using step-by-step coding instructions to help inculcate a “learning by doing” approach. The book has no prerequisites, and covers the subject from the ground up, including a detailed introductory chapter on the Python language. As such, it is going to be a valuable resource not only for students of computer science, but also for anyone looking for a foundation in the subject, as well as professionals looking for a ready reckoner.

Parteek Bhatia is an Associate Professor in the School of Electrical Engineering & Computer Science, Washington State University, Pullman, WA, USA. He was previously a Professor in the Department of Computer Science and Engineering at Thapar Institute of Engineering and Technology, Patiala, India, and has also served as a Visiting Professor at Whitman College, Walla Walla, WA, USA, and at the LAMBDA Lab, Tel Aviv University, Israel. His research interests include machine learning, natural language processing, and explainable AI. He has authored several textbooks in the field of databases, including “Data Mining and Data Warehousing: Principles and Practical Techniques,” published by Cambridge University Press in 2019.

Machine Learning with Python

Principles and Practical Techniques

Parteek Bhatia

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

Shaftesbury Road, Cambridge CB2 8EA, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment,
a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of
education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781009170246

First published 2024

Printed in India

A catalogue record for this publication is available from the British Library

ISBN 978-1-009-17024-6 Paperback

To all my students from Thapar Institute of Engineering and Technology, Patiala; DAV College, Amritsar, India; Tel Aviv University, Israel; and Whitman College, Walla Walla, USA, as well as those I have had the privilege to interact with during workshops worldwide, you are the pillars of my life. Life without your presence and contributions would be unimaginably different.

Special thanks to my team led by Mr Aditya Ghodgaonkar, Ms Hemalatha Alapati, Mr Sukhwinder Singh, and Ms Saloni Saxena. Their invaluable contributions in refining the content, creating impactful images, and rigorously testing the code were instrumental in the development of this book.

Contents

List of Figures xxiii

List of Tables xlvii

Foreword by Irad Ben-Gal liii

Foreword by Ravi Shankar lv

Preface lvii

Acknowledgments lix

1. Overview of Machine Learning **1**

1.1 Introduction 1

1.2 Applications of Machine Learning 2

1.3 History of Machine Learning 7

1.3.1 Foundational Phase of Machine Learning (Before 1940) 8

1.3.2 Transitional Phase of Machine Learning (1940–2000) 9

1.3.3 Contemporary Phase of Machine Learning (Post 2000) 10

1.4 Definition of Machine Learning 11

1.5 Traditional Programming vs. Machine Learning 13

1.6 Classification of Machine Learning Algorithms 15

1.6.1 Supervised Learning 16

1.6.2 Unsupervised Learning 23

1.7 Supervised and Unsupervised Learning in a Real-life Scenario 26

1.8 Reinforcement Learning 27

1.8.1 Working Principle of Reinforcement Learning 28

1.8.2 Components of Reinforcement Learning 29

1.8.3 Reward Maximization and Discount 31

1.8.4 Exploration and Exploitation 32
1.8.5 Timing of Execution of Various Steps 33
1.8.6 Markov's Decision Process 34
1.8.7 Understanding Q-Learning 35
1.9 Data Science vs. Data Mining vs. Machine Learning vs. Deep Learning vs. Artificial Intelligence 44
REMIND ME 48
ADDITIONAL RESOURCES 49
BOOKS 49
WEBLINKS TO VIDEOS 50
WEBLINKS TO BLOGS 50
TEST ME (QUESTIONS) 51

2. Introduction to Python 53

2.1 Features of Python 54
2.2 History of Python 55
2.3 Installation of Python 56
2.4 My First Python Program 57
2.4.1 Interactive Mode Programming 58
2.4.2 Script Mode Programming 59
2.4.3 Running Python Script through IDE 60
2.5 Identifiers in Python 63
2.6 Keywords in Python 63
2.7 Indentation in Python 64
2.8 Comments in Python 65
2.9 Variables in Python 65
2.9.1 Declaring Variables and Assignment of Values to the Variables 66
2.10 Data Types in Python 67
2.11 Python Numbers 68
2.12 Python Strings 69
2.13 Python List 73
2.13.1 Main Features of the List Data Type 74
2.13.2 Basic Operations on List in Python 74
2.14 Python Tuples 81
2.14.1 Features of Tuples 82
2.14.2 Operations over the Tuple Data Type 83
2.14.3 Comparison between Lists and Tuples 87

2.15 Python Dictionary 88
2.15.1 Operations over the Dictionary Data Type 89
2.16 Python Set 96
2.16.1 Operations over the Set Data Type 96
2.17 Comparison between List, Tuple, Dictionary, and Set Data Types 101
2.18 If Statement 102
2.19 Looping Statements 105
2.19.1 `While` Loop 105
2.19.2 `For` Loop 107
2.20 Python Functions 110
REMIND ME 115
ADDITIONAL RESOURCES 116
BOOKS 116
WEBLINKS TO VIDEOS 116
WEBLINKS TO BLOGS 118
TEST ME (QUESTIONS) 118

3. Data Pre-processing 125
3.1 Need for Data Pre-processing 125
3.2 Data Pre-processing Techniques 129
3.2.1 Data Cleaning 130
3.2.2 Data Integration 137
3.2.3 Data Transformation 138
3.2.4 Data Reduction 142
3.3 Handling of Categorical Data 148
REMIND ME 151
ADDITIONAL RESOURCES 152
BOOKS 152
WEBLINKS TO BLOGS 152
TEST ME (QUESTIONS) 153

4. Implementing Data Pre-processing in Python 155
4.1 Step 1: Importing Libraries 156
4.2 Step 2: Loading the Dataset 159
4.3 Step 3: Handling of Missing Values 166
4.4 Step 4: Handling of Categorical Data 168
4.4.1 One-hot Encoding 170

4.5 Step 5: Splitting of Dataset into Training and Testing 175
4.6 Step 6: Feature Scaling 177
4.6.1 Normalization 179
4.6.2 Standardization 181
REMIND ME 182
ADDITIONAL RESOURCES 183
WEBLINKS TO VIDEOS 183
WEBLINKS TO BLOGS 183
TEST ME (QUESTIONS) 184
TEST ME (PRACTICAL SKILLS) 186

5. Simple Linear Regression 187
5.1 Introduction to Simple Linear Regression 187
5.2 Hypothesis or Function 194
5.3 Parameters of Simple Linear Regression 195
5.4 Cost Function and Its Minimization 198
5.4.1 Minimization of Cost Function with One Parameter: $\min_{\phi} J(\phi)$ 199
5.4.2 Minimization of the Cost Function with Both Parameters 203
5.5 Gradient Descent Algorithm 208
5.6 Mathematical Understanding of Gradient Descent Algorithm 213
5.6.1 Role of Learning Rate α 224
5.6.2 Calculating the Values of Parameters by Gradient Descent Algorithm 229
REMIND ME 235
ADDITIONAL RESOURCES 236
WEBLINKS TO VIDEOS 236
WEBLINKS TO BLOGS 238
TEST ME (QUESTIONS) 238

6. Implementing Simple Linear Regression 241
6.1 Step 1: Loading Dataset 243
6.2 Step 2: Splitting the Dataset into Training and Testing Datasets 246
6.3 Step 3: Building the Model and Making Predictions 249
6.4 Step 4: Visualizing the Training Set Results 251
6.5 Step 5: Visualizing the Testing Set Results 255
6.6 Step 6: Evaluating the Model Using Regression Metrics 257
REMIND ME 264
ADDITIONAL RESOURCES 264
WEBLINKS TO VIDEOS 264

WEBLINKS TO BLOGS 265

TEST ME (PRACTICAL SKILLS) 265

7. Multiple Linear Regression and Polynomial Linear Regression 267

7.1 Introduction to Multiple Linear Regression 267

7.2 Handling of Nominal Variable and Concept of Dummy Variables 270

7.3 Building the Multiple Regression Model 275

7.4 Concept and Need of P-value 278

7.5 Calculating P-value 280

7.6 Using P-value and Defining Null Hypothesis for Building Multiple Linear Regression Model 292

7.7 Variable Selection Methods 293

7.7.1 All-in (Knowledge-based Method) 293

7.7.2 Backward Elimination Method 293

7.7.3 Forward Selection Method 295

7.7.4 Bi-directional Elimination 296

7.7.5 All Possible Models 297

7.7.6 Choosing the Most Preferred Variable Selection Method 297

7.8 Importance of Adjusted R-squared Metric in Multiple Linear Regression 298

7.9 Polynomial Linear Regression 299

7.9.1 Case Study of Rank–Salary 301

7.9.2 Case Study of Chemistry Students 306

7.9.3 Polynomial Linear Regression in Case of Multiple Input Attributes 308

7.9.4 Exploring the Versatility of Polynomial Regression in Modeling Various Data Shapes 308

REMIND ME 310

ADDITIONAL RESOURCES 311

WEBLINKS TO VIDEOS 311

WEBLINKS TO BLOGS 311

TEST ME (QUESTIONS) 311

8. Implementation of Multiple Linear Regression and Polynomial Linear Regression 313

8.1 Introduction to Multiple Linear Regression 313

8.1.1 Understanding Dataset and Problem Statement 314

8.2 Polynomial Linear Regression 327

8.2.1 Case Study of Rank Salary 327

8.3 What Will Happen if You Increase the Degree Value Too Much? 342

8.4 Comparison of Simple Linear Regression and Polynomial Regression Results 344

8.5 Case Study of Chemistry Lab Experiment 348

REMIND ME 355

ADDITIONAL RESOURCES 356

WEBLINKS TO VIDEOS 356

WEBLINKS TO BLOGS 356

TEST ME (PRACTICAL SKILLS) 357

9. Classification 359

9.1 Introduction to Classification 359

9.2 Types of Classification 361

9.2.1 Posteriori Classification 362

9.2.2 Priori Classification 362

9.3 Input and Output Attributes 362

9.4 Working of Classification 363

9.5 Guidelines for Size and Quality of the Training Dataset 367

9.6 Lazy and Eager Learner Classifiers 367

9.7 Introduction to Decision Tree Classifier 367

9.7.1 Building Decision Tree 370

9.7.2 Concept of Information Theory 370

9.7.3 Defining Information in Terms of Probability 371

9.7.4 Rule for Selection of Root Node and Other Nodes 373

9.7.5 Information Gain 374

9.7.6 Building Decision Tree Using Information Gain 375

9.7.7 Drawbacks of Information Gain Theory 408

9.7.8 Split Algorithm Based on Gini Index 408

9.7.9 Building Decision Tree Using Gini Index 412

9.7.10 Advantages of Decision Tree 445

9.7.11 Disadvantages of Decision Tree 445

9.8 Random Forest Classification 445

9.9 Naive Bayes Method 448

9.9.1 Bayes' Theorem in Action 449

9.9.2 Using Bayes' Theorem for Classification 451

9.9.3 Working of Naive Bayes Classifier Using Laplace Estimator 466

9.10 k-Nearest Neighbor Algorithm 467

9.11 Logistic Regression 470

9.12 Understanding Metrics to Know the Quality of Classifier 478

9.12.1 True Positive 479

9.12.2 True Negative 479
9.12.3 False Positive 480
9.12.4 False Negative 480
9.12.5 Confusion Matrix 480
9.12.6 Precision 481
9.12.7 Recall 482
9.12.8 F-Measure 483
9.12.9 Evaluating a Confusion Matrix Have More Than Two Classes 484
REMIND ME 486
ADDITIONAL RESOURCES 487
BOOKS 487
WEBLINKS TO VIDEOS 487
WEBLINKS TO BLOGS 488
TEST ME (QUESTIONS) 488

10. Support Vector Machine Classifier 495

10.1 Support Vector Machines 495
10.1.1 SVM Working Principle 495
10.2 Rules for Identification of Correct Hyperplane 499
10.3 Some Important Concepts and Terminologies Used in SVM 503
10.3.1 Support Vectors 504
10.3.2 Maximized Margin 504
10.3.3 Positive and Negative Hyperplane 505
10.4 Types of SVM Classifiers 506
10.4.1 Linear SVM Classifier 506
10.4.2 Non-linear SVM Classifier 507
10.5 Deep Dive into Non-linear SVM Classifier 507
10.5.1 Case 1: When Non-linear Data Is in One-Dimension 509
10.5.2 Case 2: When Non-linear Data Is Two-Dimensional 513
10.6 Defining Hyperplane 514
10.7 Limitation of Mapping Data Points to Higher Dimensional Space and Projecting Them Back to Lower Dimensional Space 515
10.8 SVM Kernel 515
10.8.1 Radial Basis Function (RBF) or Gaussian Kernel Function 516
10.8.2 Selecting the Appropriate Value of the Hyperparameter (σ) 519
10.8.3 Understanding RBF Kernel Function from a 3D Perspective 521
10.8.4 Applying Multiple Kernels to a Dataset 525

10.9 Tuning Parameters of SVM 526
10.9.1 Gamma 526
10.9.2 Regularization 527
REMIND ME 528
ADDITIONAL RESOURCES 530
WEBLINKS TO VIDEOS 530
WEBLINKS TO BLOGS 530
TEST ME (QUESTIONS) 531

11. Implementation of Classification Algorithms 533

11.1 Introduction to Classification Algorithms and Steps for Its Implementation 533
11.2 Preparing the Pre-processing Data Template (Step 1 to Step 4)—Common to All the Classifier Models 535
11.3 Implementation of Decision Tree Classifier 544
11.4 Implementation of Random Forest Classifier 554
11.5 Implementation of Naive Bayes Classifier 561
11.6 Implementation of k-NN 567
11.7 Implementation of Logistic Regression Classifier 574
11.8 Implementation of Support Vector Machine Linear Model 581
11.9 Implementation of Non-linear Support Vector Machine Model 587
11.10 Comparative Analysis of Different Classifiers 592
11.11 Text Classification 594
REMIND ME 604
ADDITIONAL RESOURCES 606
WEBLINKS TO VIDEOS 606
WEBLINKS TO BLOGS 607
TEST ME (QUESTIONS) 607
TEST ME (PRACTICAL SKILLS) 609

12. Clustering 611

12.1 Introduction to Clustering 611
12.2 Applications of Clustering 613
12.3 Features of an Ideal Clustering Algorithm 614
12.4 Distance Metrics 615
12.4.1 Euclidean Distance 615
12.4.2 Manhattan Distance 618
12.4.3 Minkowski Distance 619
12.4.4 Chebyshev Distance 620

12.5 Major Clustering Methods/Algorithms 621
12.6 Partitioning Clustering 622
12.6.1 k-means Clustering 622
12.6.2 Finding the Optimal Number of Clusters 646
12.6.3 Issues of k-means Algorithm 647
12.7 Hierarchical Clustering Algorithms (HCA) 648
12.7.1 Agglomerative Clustering 649
12.7.2 Divisive Clustering 675
12.7.3 Density-based Clustering 681
REMIND ME 692
ADDITIONAL RESOURCES 693
BOOKS 693
WEBLINKS TO VIDEOS 693
WEBLINKS TO BLOGS 694
TEST ME (QUESTIONS) 695

13. Implementation of Clustering 699

13.1 Implementation of k-means Clustering and Hierarchical Clustering 699
13.2 Case Study I: Clustering the Customers of a Mall 700
13.2.1 Step 1: Importing Libraries and the Dataset 702
13.2.2 Step 2: Find the Optimal Number of Clusters 704
13.2.3 Step 3: Fitting k-means to the Dataset 707
13.2.4 Step 4: Visualizing the Clusters 709
13.2.5 Step 5: Creating a Dendrogram and Finding the Optimal Number of Clusters 712
13.2.6 Step 6: Fitting the Hierarchical Clustering Model over the Dataset 714
13.2.7 Step 7: Visualizing the Different Clusters 716
13.3 Case Study II: U.S. Arrests Dataset 719
13.3.1 Step 1: Importing the Libraries and Dataset 720
13.3.2 Step 2: Find the Optimal Number of Clusters 721
13.3.3 Step 3: Fitting k-means to the Dataset 722
13.3.4 Step 4: Visualizing the Clusters 724
13.3.5 Step 5: Creating a Dendrogram and Finding the Optimal Number of Clusters 726
13.3.6 Step 6: Fitting the Hierarchical Clustering Model over the Dataset 727
13.3.7 Step 7: Visualizing the Different Clusters 728
13.4 Case Study III: Iris Dataset 729
13.4.1 Step 1: Importing the Libraries and Dataset 730
13.4.2 Step 2: Find the Optimal Number of Clusters 731

13.4.3 Step 3: Fitting k-means to the Dataset 732
13.4.4 Step 4: Visualizing the Clusters 734
REMIND ME 736
ADDITIONAL RESOURCES 737
WEBLINKS TO VIDEOS 737
WEBLINKS TO BLOGS 737
TEST ME (QUESTIONS) 737

14. Association Mining **739**
14.1 Introduction to Association Rule Mining 739
14.2 Defining Association Rule Mining 743
14.3 Representations of Items for Association Mining 744
14.4 Metrics for Evaluating the Strength of Association Rules 745
14.4.1 Support 745
14.4.2 Confidence 747
14.4.3 Lift 749
14.5 Naive Algorithm for Finding Association Rules 753
14.5.1 Working of Naive Algorithm 753
14.5.2 Limitations of Naive Algorithm 755
14.5.3 Improved Naive Algorithm to Deal with a Larger Dataset 756
14.6 Approaches for Transaction Database Storage 757
14.6.1 Simple Transaction Storage 758
14.6.2 Horizontal Storage 758
14.6.3 Vertical Representation 759
14.7 Apriori Algorithm 760
14.7.1 A Brief Discussion about the Inventors 760
14.7.2 Working of the Apriori Algorithm 762
14.7.3 Apriori Property 781
14.8 Closed and Maximal Itemsets 791
14.9 Limitations of Apriori Algorithm 795
14.10 Apriori–TID Algorithm for Generating Association Mining Rules 795
14.10.1 Advantages of the Apriori–TID Algorithm 799
REMIND ME 800
ADDITIONAL RESOURCES 801
BOOKS 801
WEBLINKS TO VIDEOS 801

WEBLINKS TO BLOGS 802

TEST ME (QUESTIONS) 802

15. Implementation of Association Mining **807**

15.1 Building Association Mining Model 807

15.2 Step 1: Importing Relevant Libraries and Loading the Dataset 810

15.3 Step 2: Making Transactions 812

15.4 Step 3: Building the Model 814

15.5 Step 4: Viewing and Interpreting the Mined Rules 816

REMIND ME 819

ADDITIONAL RESOURCES 819

WEBLINK TO VIDEO 819

WEBLINK TO BLOG 819

TEST ME (QUESTIONS) 820

16. Artificial Neural Network **821**

16.1 Introduction to Artificial Neural Network 821

16.2 Human Brain: An Inspiration for the Development of Neural Network 824

16.2.1 Neuron 825

16.3 Mapping of Human Brain Neurons to Artificial Neural Network 827

16.4 Working of Artificial Neural Network 829

16.5 How Does ANN Processes Multiple Input Values? 831

16.6 Role of Weights in Building ANN 833

16.7 Three-Step Learning Process of a Neuron 835

16.7.1 Commonly Used Activation Functions 837

16.8 Training of Neural Network 842

16.9 Finding Optimized Value of Weights for Minimizing the Cost Function 852

16.9.1 Brute-force Approach 852

16.9.2 Gradient Descent Approach 853

16.9.3 Stochastic Gradient Descent Algorithm to Find Global Minima 856

16.9.4 Mini-batch Method 858

16.10 Final Step-by-Step Procedure for Training the Neural Network 859

REMIND ME 860

ADDITIONAL RESOURCES 861

WEBLINKS TO VIDEOS 861

WEBLINKS TO BLOGS 862

TEST ME (QUESTIONS) 862

17. Implementation of the ANN **865**
17.1 Building Artificial Neural Network for Cancer Detection 865
17.2 Importing the Dataset and Pre-processing 866
17.2.1 Step 1: Importing the Libraries 866
17.2.2 Step 2: Creating Dependent and Independent Variables 867
17.2.3 Step 3: Encoding Categorical Data 869
17.2.4 Step 4: Splitting the Dataset into Training and Testing 870
17.2.5 Step 5: Feature Scaling 871
17.3 Building the Artificial Neural Network 871
17.3.1 Step 6: Importing the ANN Libraries 871
17.3.2 Step 7: Initializing the ANN 872
17.3.3 Step 8: Adding the Input Layer and the First Hidden Layer 872
17.3.4 Step 9: Adding Second and Third Hidden Layers 873
17.3.5 Step 10: Adding the Output Layer 874
17.3.6 Step 11: Compiling the ANN 875
17.3.7 Step 12: Fitting the ANN into the Training Set 876
17.4 Predicting the Results 878
17.4.1 Step 13: Predicting the Test Set Results 878
17.4.2 Step 14: Making the Confusion Matrix 880
REMIND ME 884
ADDITIONAL RESOURCES 885
WEBLINKS TO VIDEOS 885
WEBLINKS TO BLOGS 885
TEST ME (ASSIGNMENTS) 886
18. Deep Learning and Convolutional Neural Network **889**
18.1 Image Recognition 889
18.2 Identifying Digits with Machine Learning 892
18.2.1 Converting Image to Numbers 893
18.2.2 Issues with This Simple Image Classifier 896
18.2.3 Possible Solutions 897
18.3 How Does Human Brain Identify Images? 900
18.4 Working Principle of Convolutional Neural Network (CNN) 903
18.5 Architecture of a Basic CNN Model 905
18.5.1 Convolution Layer 906
18.5.2 Pooling Layer 915
18.5.3 Flattening 920

18.5.4 Full Connection 922
18.6 Training a CNN Model 923
18.7 Deciding the Optimal Number of Epochs to Train a Neural Network 925
Remind Me 930
Additional Resources 931
Weblinks to Videos 931
Weblinks to Blogs 931
Test Me (Questions) 932

19. Implementation of Convolutional Neural Network 937
19.1 Building Image Classifier with CNN 937
19.2 Dog–Cat Classifier 938
19.2.1 Step1: Importing the `keras` Library and Packages 940
19.2.2 Step 2: Initializing the CNN Model 940
19.2.3 Step 3: Add Convolution Layer 941
19.2.4 Step 4: Pooling 943
19.2.5 Step 5: Flattening 944
19.2.6 Step 6: Full Connection 945
19.2.7 Step 7: Compilation of CNN 946
19.3 Phase 2: Image Augmentation and Fitting the Model 947
19.4 Phase 3: Getting Predictions from Our Model and Model Improvement 952
19.5 Further Improving the Accuracy of the Model by Adding Another Convolutional and Max Pooling Layer 954
19.6 Case Study 2: Handwritten Digit Classifier Using MNIST Dataset 958
Remind Me 965
Additional Resources 966
Weblinks to Videos 966
Weblinks to Blogs and Datasets 967
Test Me (Practical Skills) 967

20. Recurrent Neural Network 969
20.1 Neural Networks and Human Brain 969
20.2 Extending Artificial Neural Network to Recurrent Neural Network 972
20.3 Limitations of Feed-forward Networks 975
20.4 Working Principle of Recurrent Neural Networks 976
20.5 Understanding the Recurrent Neural Networks by Using a Case Study of a Coaching Center 977
20.6 Mathematical Modeling of RNN 980

20.7 Limitations of Recurrent Neural Networks 983
20.7.1 Vanishing and Exploding Gradients 985
20.8 Long Short-Term Memory (LSTM) 989
20.9 Working Principle of LSTM 991
20.9.1 Internals of an RNN Cell 991
20.9.2 Internals of an LSTM Cell 995
20.10 Applications of RNN and LSTM 1001
REMIND ME 1003
ADDITIONAL RESOURCES 1004
WEBLINKS TO VIDEOS 1004
WEBLINKS TO BLOGS 1005
TEST ME (QUESTIONS) 1005

21. Implementation of Recurrent Neural Network 1007
21.1 Introduction 1007
21.2 Implementation of RNN in Python 1008
21.2.1 Data Pre-processing 1008
21.2.2 Building Architecture of RNN 1022
21.2.3 Making Predictions and Visualizing Results 1029
REMIND ME 1038
ADDITIONAL RESOURCES 1039
WEBLINKS TO VIDEOS 1039
WEBLINKS TO BLOGS 1040
TEST ME (PRACTICAL SKILLS) 1040

22. Genetic Algorithm 1041
22.1 Intuition of Genetic Algorithm 1041
22.2 The Inspiration behind Genetic Algorithm 1042
22.2.1 Natural Selection: Survival of the Fittest 1043
22.2.2 Recombination: Crossover 1044
22.2.3 Mutation: Changes in Genes 1046
22.3 Correlation between Nature and Genetic Algorithm 1049
22.4 Mathematical Representation of Genes and Fitness Theory 1050
22.5 Natural Selection Implementation through Roulette Wheel 1054
22.5.1 Working of Roulette Wheel Used in the Casino Game 1054
22.6 Implementation of Recombining or Crossover in Genetic Algorithm 1064
22.6.1 Single-point Crossover 1065
22.6.2 Double-point Crossover 1068

22.7 Implementation of Mutation in Genetic Algorithm 1072
22.7.1 Mathematical Modeling of Mutation 1072
22.7.2 How to Decide the Number of Genes to Be Mutated? 1074
22.8 Elitism 1075
22.8.1 Implementation of Elitism and Concept of Elitism Ratio 1077
22.9 Advantages and Disadvantages of Genetic Algorithm 1079
22.10 Applications of Genetic Algorithm 1080
REMIND ME 1080
ADDITIONAL RESOURCES 1081
WEBLINKS TO VIDEOS 1081
WEBLINKS TO BLOGS 1082
TEST ME (QUESTIONS) 1082
Index 1085

Figures

1.1	Applications of machine learning	2
1.2	Machine learning powered dynamic pricing	4
1.3	Applications of machine learning in healthcare	6
1.4	Phases of development in machine learning	8
1.5	A traditional programming model	14
1.6	A machine learning model	14
1.7	Types of machine learning	15
1.8	Price prediction dataset	16
1.9	Simplified version of price prediction dataset	17
1.10	Prediction of cost of the plot with a small dataset	17
1.11	Prediction of cost of the plot with a large dataset	18
1.12	Dataset of tumor and patient with its corresponding label	19
1.13	Simplified version of a dataset of tumor and patient with its corresponding label	19
1.14	Data plot for tumor size and malignant	20
1.15	Prediction of a tumor size A	20
1.16	Considering tumor size and age as features for classification	21
1.17	Prediction for the tumor size B	21
1.18	Prediction for tumor size B to benign	22
1.19	Working of unsupervised learning	23
1.20	Applications of unsupervised learning	24
1.21	Google News, an application of clustering	25
1.22	Learning of Adam and Eve vs. modern-day child	27
1.23	Comparison of supervised, unsupervised, and reinforcement learning	28
1.24	Sparrow, grain, and cat in the setting of reinforcement learning	32

1.25	SAR triples	33
1.26	Markov's decision process to find the shortest path	34
1.27	Representation of a house	35
1.28	Representation house in the form of a graph with reward points	36
1.29	Reward matrix (R)	36
1.30	Graph for calculation of $Q(1, 5)$	38
1.31	Graph for calculation of $Q(2, 3)$	40
1.32	Graph for calculation of $Q(3, 1)$	41
1.33	Graph for calculation of $Q(1, 5)$	42
1.34	Domains of all fields	44
1.35	Traditional machine learning vs. deep learning	45
1.36	Artificial intelligence is a superset of machine learning	46
1.37	Types of artificial intelligence	46
1.38	Relationship between artificial intelligence, machine learning, and deep learning	47
2.1	Ranking of computer programming languages	54
2.2	Features of Python	55
2.3	History of Python	56
2.4	Installation of Python	57
2.5	Saving the *hello.py*	59
2.6	Spyder in Anaconda Navigator	60
2.7	Spyder IDE	61
2.8	Jupyter in Anaconda Navigator	61
2.9	Homepage of Jupyter Notebook	62
2.10	Jupyter IDE	62
2.11	List of keywords in Python	64
2.12	Data types in Python	67
2.13	Dictionary as a *key:value* pair	88
3.1	Idea dataset for house price prediction	126
3.2	Messy data collected for house price prediction	126
3.3	Precision agriculture data with a lot of missing values and noise	127
3.4	Garbage data input to ML models results in garbage output	128
3.5	Data pre-processing to process the raw data into quality data for training ML models	128
3.6	Various stages of pre-processing	129
3.7	Use of mean or median over numeric data	131
3.8	Clustering of data to identify the outliers and noise	136
3.9	Regression for identification of noise	137

3.10 Effect of an outlier on normalization 140
3.11 Customer dataset 141
3.12 Dropping columns for feature selection 141
3.13 Feature selection vs. feature extraction 143
3.14 Matrix and its eigenvector and eigenvalue 145
3.15 Concept of principal components 147
3.16 Labeling of the categorical variable 148
3.17 Label-encoded data 149
3.18 Creating three separate attributes for each of the nominal value 149
3.19 Concept of dummy variables 150
3.20 Processed data 151
4.1 File Explorer in Spyder 159
4.2 Save and run file options in Spyder 160
4.3 Importing the dataset 161
4.4 Content of *dataset* data frame 162
4.5 Various ways to perform row and column selection 162
4.6 Content of X and Y matrices 165
4.7 (a) Missing values of X replaced with corresponding columns' mean 168
4.7 (b) Mean values of City and Age columns calculated in Excel 168
4.8 Output of code snippet 4 169
4.9 Encoded City column of matrix X 170
4.10 Results of OneHotEncoding on console 171
4.11 Results of OneHotEncoding 172
4.12 Encoded Purchase column 174
4.13 *X_train*, *X_test*, *Y_train*, and *Y_test* dataset after the split 177
4.14 Euclidean distance 178
4.15 Results of normalization: (a) original *X_train*, (b) normalized *X_train* 180
4.16 Effect of an outlier on normalization 180
4.17 *X_train* and *X_test* after scaling 182
5.1 Research experience versus stipend plot 189
5.2 Stipend versus research experience plot based on dataset in Table 5.1 189
5.3 Equation of the straight line 190
5.4 Stipend value at research experience of 0 year 191
5.5 Plot for finding an increase in stipend 191
5.6 Fitting a straight line to model 192
5.7 Estimating stipend for five years of research experience 192

5.8	Mathematical equation of simple linear regression	193
5.9	Role of hypothesis in supervised learning	194
5.10	(a) ϕ_0 is 1.5 and ϕ_1 is 0	195
5.10	(b) ϕ_0 is 0 and ϕ_1 is 0.5	195
5.10	(c) ϕ_0 is 1 and ϕ_1 is 0.5	196
5.11	Error between predicted and actual values	197
5.12	Scattering of data points for dataset	199
5.13	Tracing cost function for different values of the parameter	202
5.14	A 3D graph of cost function $J(\phi_0, \phi_1)$ versus ϕ_0 and ϕ_1	203
5.15	Contour plot of cost function and its parameters	204
5.16	Points on the same color ring have the same cost	205
5.17	Hypothesis plot for the cost function given in Figure 5.16	205
5.18	Minimum value of cost function dependent on two parameters	206
5.19	Gradient descent's basic idea	209
5.20	3D plot of the cost function for which minimization has to be done	209
5.21	Initialization of parameters at point A	210
5.22	Cost function after taking a tiny step	210
5.23	Starting from A and moving toward a lower altitude	211
5.24	Starting from A and moving toward a lower altitude	211
5.25	Reaching minima from initial point A	211
5.26	Moving downhill from initial point B	212
5.27	Different local minima for the same plot	212
5.28	Cost function for simple linear regression	213
5.29	Plot of a cost function with one parameter	213
5.30	Initial point for gradient descent (Case 1)	214
5.31	The given line whose slope is to be found	214
5.32	Drawing a triangle out of the given line	215
5.33	(a) When the slope of the line is larger	215
5.33	(b) When the slope of the line is smaller	215
5.34	Computing the slope of line *AB*	216
5.35	Positive slope of the line	217
5.36	Negative slope of a line	218
5.37	Zero slope of a line	218
5.38	Various cases of the slope of a line	219
5.39	Slope of a curve given by $f(x)$ at point T	219
5.40	Points T and T′ on the curve $y = f(x)$	219

5.41 Construction of line TT′ on the curve $y = f(x)$ 220
5.42 Finding the slope of a curve 221
5.43 Slope of curve at an initial point in Case 1 222
5.44 Converging to the point of minima 222
5.45 Chosen initial point for gradient descent (Case 2) 223
5.46 Slope of curve at an initial point in Case 2 223
5.47 Converging to the point of minima 224
5.48 When the learning rate is low 224
5.49 When the learning rate is large 225
5.50 When the learning rate is too large 225
5.51 Initialization at point A with a very large value of learning rate α 226
5.52 A large step size from point A to point B 226
5.53 A large step size from point B to point C 226
5.54 A large step size from point C to point D 227
5.55 Reducing the step size when approaching the point of minima 227
5.56 Initialization at point A 228
5.57 Step from point B to point C 228
5.58 Step from point C to point D 229
5.59 Notations used in gradient descent algorithm 230
5.60 The cost function for the linear regression model 235
6.1 Problem representation 242
6.2 Best-fitting line for Stipend versus ResearchExperience plot 242
6.3 File explorer in Spyder 243
6.4 (a) Variable explorer in Spyder 244
6.4 (b) Contents of dataframe *dataset* 245
6.5 Snapshot of variable explorer 248
6.6 Values of *X_train*, *X_test*, *y_train*, and *y_test* 248
6.7 Snapshot of variable explorer 250
6.8 Values of *y_pred* and *y_test* 251
6.9 Scatter plot for a given dataset 252
6.10 Scatter plot for a given dataset with labels 253
6.11 The resulting regression line 254
6.12 Regression line for the scatter plot of the training dataset 254
6.13 Regression line for the scatter plot of the testing dataset 256
6.14 Regression line for the scatter plot on the execution of code snippet 13 257
6.15 The value of MAE 258

6.16	The value of MSE	259
6.17	The value of RMSE	259
6.18	Simple Linear Regression Line	261
6.19	Mean (average) Line	262
7.1	Simple and multiple linear regression mathematical realization	268
7.2	Handling nominal attributes	271
7.3	Creating two separate attributes for each of the nominal values	271
7.4	Dummy variables	271
7.5	Replacing nominal attribute with a dummy variable	272
7.6	Mathematical representation of multiple linear regression for a given dataset	272
7.7	Nominal attribute with three values	273
7.8	Replacing nominal attribute with dummy variables	273
7.9	Mathematical representation for multiple linear regression with two dummy variables	274
7.10	Dataset with more than one categorical attribute—State and Company Type	274
7.11	Handling the models with two or more categorical attributes	275
7.12	(a) Multiple regression model, dependent variable y is dependent on more than one variable, and in (b) multiple regression model for predicting the final grades of students by selecting all the attributes	276
7.13	Selection of attributes on which y is dependent	277
7.14	Garbage in–garbage out	277
7.15	Too complex model	277
7.16	Selection of relevant attributes on which final grades of students depend	278
7.17	Expected results of the experiment	281
7.18	Observed results of the experiment	281
7.19	Chi-square distribution table	284
7.20	Finding P-value with chi-square distribution table	284
7.21	Finding P-value with chi-square value 3	285
7.22	Finding P-value with chi-square value of 0.75	287
7.23	Finding P-value with chi-square value 0.12	288
7.24	Finding P-value with chi-square value 12	290
7.25	Finding P-value with chi-square value 27	291
7.26	Backward elimination method	294
7.27	Forward selection method	295
7.28	Bi-directional elimination	296
7.29	Polynomial behavior of COVID confirmed cases	300
7.30	Plot of salary versus rank	302

7.31	Modeling the rank–salary problem using polynomial linear regression	302
7.32	Modeling the rank–salary problem using linear regression	303
7.33	Resulting polygon when the degree is 2	304
7.34	Resulting polygon when the degree is 3	304
7.35	Resulting polygon when the degree is 4	305
7.36	Predicting the salary of a person working at rank 7.5	305
7.37	Temperature versus pressure plot for case study	306
7.38	Linear regression when applied to chemistry students' problem	307
7.39	Resulting polynomial trend when scatter points are joined	307
7.40	Complex curve shapes	309
8.1	Data analysis of the *50_AdAgency* dataset	315
8.2	Importing the *50_AdAgency.csv* dataset to the current working directory of Spyder	316
8.3	Variable explorer tab after executing code snippets 1 to 4	317
8.4	(a) Data frame *dataset*	317
8.4	(b) Values loaded into *X*	317
8.4	(c) Values loaded into *Y*	318
8.5	Creating three separate attributes for each of the nominal values	318
8.6	Independent variable *X* after *onehot encoding* over the City attribute	320
8.7	Explanation of dummy variables	320
8.8	Updated *X*- after freeing it from *dummy trap*	321
8.9	*X_train, X_test, Y_train, Y_test* in the variable explorer tab	322
8.10	Output message on building the model successfully	323
8.11	*Y_pred* in variable explorer tab	324
8.12	Actual and predicted Profit values	325
8.13	Results of the model evaluation	326
8.14	Importing the dataset	329
8.15	Data frame dataset as shown by variable explorer	329
8.16	Contents of the *dataset* data frame	330
8.17	Independent Rank and dependent Salary variable	331
8.18	Two-step process of fitting a polynomial regression model	331
8.19	Contents of *X_poly*	332
8.20	Scatter plot for a given dataset	334
8.21	Scatter plot for a given dataset with labels	334
8.22	The resulting regression line	335
8.23	Regression line for the scatter plot of the training dataset	336
8.24	Performance evaluation at degree 2	337

8.25	Regression plot for 3-degree polynomial	338
8.26	Performance evaluation at degree 3	339
8.27	Regression plot for 4-degree polynomial	340
8.28	(a) Performance evaluation at degree 4	341
8.28	(b) Performance evaluation at degrees 2, 3, and 4	341
8.29	Predicting *Rahat's* salary, an employee at level 7.5	342
8.30	Regression plot for 50-degree polynomial	342
8.31	*X_poly* up to degree 50	343
8.32	Comparison of regression plot for 50-degree and 100-degree	343
8.33	Visualizing the results of linear regression	345
8.34	Performance evaluation for simple linear regression	346
8.35	Predicting salary using simple linear regression	347
8.36	The content of the *dataset*	349
8.37	Contents of *X* and *Y*	349
8.38	*X_poly* when the degree is 2	350
8.39	The resulting regression line fitting the data points	351
8.40	Performance evaluation at degree 2	352
8.41	*X_poly* when the degree is 3	353
8.42	The resulting regression line for a degree of 3	353
8.43	Performance evaluation at degree 3	353
8.44	Predictions made by polynomial regression	354
8.45	Results of linear regression	355
9.1	Input and output attributes of glass identification dataset	363
9.2	Process of learning in the case of a decision tree classifier	364
9.3	Building classifier to predict the chance of placements	365
9.4	Predicting the chance of placement for a student based on the input attributes	366
9.5	Training and testing of the classifier	366
9.6	Structure of a decision tree	368
9.7	Decision tree to predict whether the customer will buy the phone or not	368
9.8	Decision tree to predict the species of an animal	369
9.9	Data splitting based on Age attribute	379
9.10	Decision tree after splitting of attribute Age being "Old"	379
9.11	Decision tree after splitting of attribute Age being "New"	380
9.12	Decision tree after splitting based on attribute "Competition"	382
9.13	Data split based on Student attribute as the root node	387
9.14	Data split based on Age attribute	391

9.15 Decision tree after splitting of attribute Age "<= 30" 392
9.16 Decision tree after selecting Income as best split when Age is "31–40" 393
9.17 Decision tree after splitting of attribute Income 394
9.18 Decision tree after splitting of attribute Age "> 40" 395
9.19 Data split based on Age attribute when the customer is not a student 399
9.20 Decision tree after selecting Income as best split when the customer is not a student and Age <= 30 400
9.21 Decision tree after splitting of attribute Income when the customer is not a student and Age <= 30 402
9.22 Decision tree after selecting Credit rating as the next split 403
9.23 Decision tree after splitting of attribute Credit rating when the customer is non-student and age is 31–40 404
9.24 Decision tree after selecting Credit rating as the next split 406
9.25 Decision tree after splitting Credit rating attribute 407
9.26 Prediction of buying a computer for an unknown instance 408
9.27 Gini index representing perfect equality 409
9.28 Lorenz curve 410
9.29 Lorentz curves with varying income distributions 411
9.30 Data splitting based on Age attribute 416
9.31 Decision tree after splitting of attribute Age being "Old" 417
9.32 Decision tree after splitting of attribute Age being "New" 417
9.33 Decision tree after splitting based on attribute "Competition" 419
9.34 Data split based on Student attribute as the root node 424
9.35 Data split based on Age attribute 428
9.36 Decision tree after splitting of attribute Age "<= 30" 429
9.37 Decision tree after selecting Income as the next split 430
9.38 Decision tree after splitting of attribute Age "31–40" 431
9.39 Decision tree after splitting of attribute Age "> 40" 432
9.40 Data split based on Age attribute when the customer is not a student 436
9.41 Decision tree after selecting Income as best split when Age <= 30 and customer is not a student 438
9.42 Decision tree after splitting of attribute Income when Age <= 30 and customer is not a student 439
9.43 Decision tree after selecting Credit rating as the next split 441
9.44 Decision tree after splitting of attribute Credit rating when Age is 31–40, the customer is not a student 442
9.45 Decision tree after analysis of Age attribute 443

9.46 Prediction of buying a computer for an unknown instance 444
9.47 The working of the random forest algorithm 447
9.48 Kinect, a gaming device using random forest for the classification of body parts of a gamer 447
9.49 Manufactured screws 449
9.50 Graph depicting information of employees whether they are commuting on two-wheeler or four-wheeler based on their Salary and Age 452
9.51 New data point indicating unknown instance 452
9.52 Area of likelihood around unknown instance X 454
9.53 Calculation of probability of likelihood 455
9.54 Calculation of probability of likelihood for four-wheeler 457
9.55 Summarization of count calculations of all input attributes 463
9.56 Probability of play held or not for each value of the attribute 464
9.57 Probability for play yes for an unknown instance 464
9.58 Probability for a play taking a class no for an unknown instance 465
9.59 Probability of play not being performed when the outlook is "overcast" 466
9.60 Values of attributes after adding Laplace estimator 466
9.61 Probability of play held or not for each modified value of the attribute 467
9.62 Attribute values for given example instance 467
9.63 Sample plot of our classification problem 468
9.64 The new data point whose category needs to be predicted 468
9.65 Five closest neighbors of X (new data point) 469
9.66 Category-wise neighbor count 470
9.67 Customers' age and response (whether they bought insurance or not?) 471
9.68 Linear regression model of researchers' stipend and respective experience 471
9.69 Modeling the buying insurance problem using linear regression 472
9.70 Likelihood of customers buying insurance for a given age 472
9.71 Studying the relevant part of the linear regression model 473
9.72 Studying the non-relevant part of the linear regression model 473
9.73 Modified plot modeling the buying insurance problem 474
9.74 Effect of Sigmoid function on linear regression model 474
9.75 Plot for Equation (9.6) in XY plane 475
9.76 Predicting the chances of buying insurance 476
9.77 Computing probabilities of buying insurance 477
9.78 Mapping probability values to binary outcome (0 or 1) 477
9.79 Building binary classifier using a logistic regression model 478
9.80 Confusion matrix for bird classifier 479

9.81	Confusion matrix for tumor prediction	480
10.1	Distribution of dataset of apples and peaches based on their size and color	496
10.2	Standard apples and peaches at the heart of the dataset	496
10.3	Apples that appear like peaches are chosen by SVM during training	497
10.4	Peaches that appear like apples are chosen by SVM during training	498
10.5	Hyperplane separating the two classes—apples and peaches	498
10.6	One of the possible hyperplanes (separating lines)	499
10.7	Some other possible hyperplanes (separating lines)	499
10.8	Taking three sample hyperplanes, A, B, and C	500
10.9	Selection of B as best hyperplane based on Rule 2	501
10.10	Choosing the hyperplane with maximum margin	501
10.11	Spotting an outlier	502
10.12	Resulting hyperplane in the presence of an outlier	502
10.13	Given hyperplane for a scatter plot	503
10.14	Some important terminologies related to SVM	503
10.15	Maximized margin	504
10.16	Locating positive and negative hyperplane in different cases	505
10.17	Understanding positive, negative, and optimal hyperplanes	505
10.18	Linearly separable data with linear SVM classifier	506
10.19	Non-linearly separable data	507
10.20	Non-linear data separated with different possibilities of linear hyperplane	508
10.21	Non-linearly separable dataset	509
10.22	Sample data points in 1D	509
10.23	Assuming *A* as the instance of hyperplane	510
10.24	Assuming *A'* as the instance of hyperplane	510
10.25	Non-linear dataset with 9 different items	510
10.26	Resulting plot on applying function *f* over data points	511
10.27	Change after applying the *f* function	511
10.28	Identifying a hyperplane after the transformation of data points	512
10.29	Hyperplane segregating the data points into two classes	512
10.30	A non-linearly separable dataset in 2D space	513
10.31	Mapping data points from 2D data space to the 3D data space	513
10.32	Mapping back the data points from the 3D data space to the 2D data space	514
10.33	Distance between two nodes	516
10.34	Relationship between kernel and distance between points ($\sigma = 4$)	518
10.35	Choosing the appropriate value of σ	519

10.36 Relationship between kernel and distance (width of the bump from -3 to 3) 520

10.37 Plot of kernel value versus the respective distance when sigma is large ($\sigma = 10$) 521

10.38 Gaussian RBF kernel in 3-dimension 522

10.39 Selection of point of the landmark 522

10.40 2D plot of a dataset 523

10.41 Relationship between dataset and the *kernel* function 523

10.42 Relationship between dataset and the *kernel* function 524

10.43 Scatter plot of a complex dataset 525

10.44 Using two landmarks $\overline{x}_1$ and $\overline{x}_2$ 525

10.45 Decision boundary for the complex dataset 526

10.46 Effects of the low and high value of gamma parameter 527

10.47 Effect of regularization parameter 528

11.1 *Alexa_dataset* 536

11.2 Importing the libraries 538

11.3 Importing the *Alexa_dataset.csv* to the current working directory of `Spyder` 538

11.4 Importing the dataset 539

11.5 Contents of the *dataset* variable 539

11.6 Variable explorer depicting dataset, *X,* and *y* 540

11.7 Contents of *X* and *y* 541

11.8 Variable explorer 542

11.9 *X_train* and *X_test* after feature scaling 543

11.10 Variable explorer showing *y_pred* 545

11.11 (a) *y_test* 546

11.11 (b) *y_pred* 546

11.12 Confusion matrix for the decision tree classifier 547

11.13 Analyzing the confusion matrix for the decision tree classifier 548

11.14 Resulting performance metrics for our decision tree model 550

11.15 Visualization of the decision tree (training dataset) 553

11.16 Visualization of the decision tree (testing dataset) 554

11.17 Parameters of the `RandomForestClassifier` 555

11.18 First 10 values of *y_test* and *y_pred* 557

11.19 Confusion matrix (*cm* variable) for random forest 558

11.20 Resulting performance metrics for our random forest model 559

11.21 Visualization of the prediction results (training dataset) 560

11.22 Visualization of the prediction results (testing dataset) 561

11.23 First 20 observations of naive Bayes *y_test*, *y_pred* 563

11.24 Confusion matrix (*cm* variable) for naive Bayes 564
11.25 Resulting performance metrics for our naive Bayes 565
11.26 Visualization of the prediction results (training dataset) 566
11.27 Visualization of the prediction results (testing dataset) 567
11.28 Properties of the k-NN model 569
11.29 Comparing the values of *y_test* (actual values) with *y_pred* (predicted values) 570
11.30 Confusion matrix of the *k-NN* model 571
11.31 Resulting performance metrics for our *k-NN* classifier 572
11.32 Visualization of the prediction results (training dataset) 573
11.33 Visualization of the prediction results (testing dataset) 574
11.34 Comparing the values of *y_test* (actual values) with *y_pred* (predicted values) 577
11.35 Confusion matrix of the logistic regression classifier 577
11.36 Resulting performance metrics for our logistic regression classifier 579
11.37 Visualization of the prediction results (training dataset) 579
11.38 Visualization of the prediction results (testing dataset) 580
11.39 Parameters for the *SVM* model 582
11.40 Comparing the values of *y_test* (actual values) with *y_pred* (predicted values) 583
11.41 Confusion matrix of the *SVM* model 584
11.42 Resulting performance metrics for our *SVM* classifier 585
11.43 Visualization of the prediction results (training dataset) 586
11.44 Visualization of the prediction results (testing dataset) 587
11.45 Comparing the values of *y_test* (actual values) with *y_pred* (predicted values) 589
11.46 Confusion matrix for the *non-linear SVM* model 590
11.47 Resulting performance metrics for our non-linear *SVM* classifier 591
11.48 Visualization of the training results 591
11.49 Visualizing the test set results 592
11.50 *X* as an array of int64 with size (100, 1500) in variable explorer tab after BoW operation 599
11.51 Contents of *X* after applying `CountVectorizer` 599
11.52 *X* as an array of float64 with size (100, 3622) in variable explorer tab 601
11.53 Contents of *X* after applying TF–IDF 601
11.54 Comparing predicted values with actual values 603
12.1 Characteristics of cluster 612
12.2 Clustering model 612
12.3 Euclidean distance between (6, –2) and (9, 4) 616
12.4 Manhattan distance 618
12.5 Major clustering methods/algorithms 622

12.6 Flowchart for the k-means algorithm 624
12.7 Database after initialization 625
12.8 Plot of data for $k = 2$ 630
12.9 Plot of data for $k = 3$ 633
12.10 Plot of data in Example 12.2 for $k = 3$ 640
12.11 Use the elbow method to determine the optimal number of clusters 647
12.12 Illustration of agglomerative and divisive clustering 649
12.13 Hierarchical clustering tree diagram for agglomerative clustering 651
12.14 Single linkage 652
12.15 Complete linkage 652
12.16 Average linkage 653
12.17 Graphical representation of given data points 654
12.18 Clustering of data points using single linkage 654
12.19 Clustering of data points using complete linkage 655
12.20 Clustering using complete linkage 658
12.21 Clustering using single linkage 659
12.22 Dendrogram of single-linkage clustering of customer segmentation 663
12.23 Hierarchical tree of single-linkage clustering of customer segmentation 663
12.24 Dendrogram obtained by agglomerative clustering 674
12.25 Hierarchical tree of clustering of students based on their marks in tests 674
12.26 Hierarchical clustering tree diagram for divisive algorithm 675
12.27 Dendrogram of divisive clustering 680
12.28 Hierarchical tree of clustering of students based on their marks in tests 681
12.29 Clustering using k-means 681
12.30 Data points clustered in arbitrary shapes and having noise 682
12.31 Concept of neighborhood and MinPts 683
12.32 Concept of core, border, outlier, MinPts, and ε (epsilon) 684
12.33 Representation of types of data points 684
12.34 Concept of directly density-reachability 685
12.35 Concept of density-reachability 686
12.36 Illustration of density-connectivity 687
12.37 Example of density-connectivity 688
12.38 Some more examples of DBSCAN 689
12.39 Advantages of DBSCAN over k-means with some examples 690
12.40 Clusters with unequal densities 691
12.41 Clusters with unequal densities 692

13.1 Dataset of the customers 702
13.2 *X* `NumPy` array 703
13.3 Use of the elbow method to determine the optimal number of clusters 704
13.4 Parameters of the `KMeans()` constructor 705
13.5 *Elbow* curve to find the number of clusters 706
13.6 Optimal number of clusters in elbow graph 707
13.7 Contents of *y_kmeans* 708
13.8 Mapping customer information with their cluster number 708
13.9 Details of scatter method 710
13.10 Five distinct clusters with their labels 710
13.11 Five distinct clusters 711
13.12 Five clusters with their respective class labels 712
13.13 Dendrogram for mall customers 713
13.14 Possible boxes for dendrogram of mall customer clustering problem 713
13.15 Total number of vertical lines within the box with a maximum height difference 714
13.16 Contents of *y_hc* 715
13.17 Interpreting the contents of *y_hc* 716
13.18 Results of agglomerative clustering 717
13.19 Clusters with their respective centroid 718
13.20 Contents of *X* 721
13.21 Elbow plot for the U.S. arrests dataset 722
13.22 First and last 10 rows of *y_means* variable 723
13.23 Mapping data of the U.S. states with their respective cluster number 724
13.24 The U.S. states are clustered into three distinct groups (or clusters) 725
13.25 Labeling clusters of the *U.S. states* 726
13.26 Dendrogram for the U.S. arrests dataset 727
13.27 Clusters formed by the agglomerative clustering algorithm 728
13.28 Description of *dataset* dictionary 730
13.29 Contents of *X* 731
13.30 Elbow graph for the *Iris* dataset 732
13.31 First and last 10 rows of *y_means* variable (*Iris* dataset) 733
13.32 Mapping data of flowers with their respective cluster number 733
13.33 Three clusters of the *Iris* dataset 734
13.34 Labeled clusters of the *Iris* dataset 735
14.1 Need for association mining 740
14.2 Association of sale of beer and diapers 741

14.3	Association of sale of beer and diapers	741
14.4	Association of sale of beer and diapers	742
14.5	Association of sale of beer and diapers	742
14.7	Representation of association rules	752
14.8	Process for identifying frequent itemsets	770
14.9	Generation of L1, frequent 1-itemset	776
14.10	Generating association rules from frequent itemsets	777
14.11	Lattice structure	782
14.12	Generation of C2 and L2	784
14.13	Generation of C3 and L3	785
14.14	Lattice structure	786
14.15	Relation between frequent, closed, and maximal itemsets	791
14.16	Lattice structure of itemsets	795
15.1	Snapshot of the dataset	810
15.2	Variable explorer	811
15.3	Contents of the *dataset* data frame with header	811
15.4	An updated snapshot of the *dataset* without a header	812
15.5	Preparing the *transactions* list	813
15.6	Contents of the *transactions* list	813
15.7	Installing the `apriori_python` library	814
15.8	Parameters of the *Apriori* method	814
15.9	The *rules* in the variable explorer tab	815
15.10	The *rules* variable contents	816
15.11	Analyzing the rules	818
15.12	Frequent items list	818
15.13	Comprehending the *Apriori rules*	818
16.1	A history of storage cost	822
16.2	Durability and access time of some storage techniques	823
16.3	Evolution of the processing capacity of computers	823
16.4	Processing of input signals to output action by the human brain	824
16.5	Network of neurons	825
16.6	Microscopic image of a neuron	825
16.7	MRI of a human brain showing billion of neurons	826
16.8	Anatomy of a neuron	826
16.9	Illustration of similarity between (a) biological neuron and (b) artificial neuron	827
16.10	Basic layer structure of ANN	828

16.11 Flow of information in a basic ANN 829
16.12 Deep learning neural network 829
16.13 Multiple output nodes in a case of multi-class classification 830
16.14 Single output node in case of regression or binary class classification 830
16.15 Layer structure of a typical ANN and deep learning neural network 831
16.16 Feature scaling independent variables 832
16.17 Input and output values for the same observation 832
16.18 Concept of weights in neural networks 833
16.19 Adjusting weights in ANN 834
16.20 Synapse is like a tightrope for a stuntman 835
16.21 Computing the weighted sum 835
16.22 Cooking of food is analogous to ANN's working 836
16.23 Application of activation function in Step 2 837
16.24 Placement of activation function in neural networks 837
16.25 A plot of the threshold function 838
16.26 Plot of the sigmoid function 839
16.27 Plot of ReLU 840
16.28 Plot of the hyperbolic tangent function 841
16.29 Passing signal to the next neuron 842
16.30 Basic neural network for predicting the price of real estate 843
16.31 Forward propagation in neural network 844
16.32 Comparing actual value (Y) and predicted value ($\hat{Y}$) 844
16.33 A cycle of forward and backward propagation making an epoch 845
16.34 Feeding input values and calculating predicted value through forwarding propagation 846
16.35 Adjusting weights to minimize cost function through backward propagation 847
16.36 Processing the complete dataset with six neurons working in parallel 848
16.37 Predicting estate prices of multiple flats (six flats) 849
16.38 Topmost neuron of hidden layer linked with two input parameters—*Covered Area* and *Distance from City* 850
16.39 Connections of input layer neurons with hidden layer neurons 851
16.40 Plot of a cost function with one parameter 853
16.41 Initial point for gradient descent 854
16.42 Finding the slope of a curve 854
16.43 Converging to the point of minima 855
16.44 Global and local minima of a cost function 855
16.45 Working of batch gradient descent algorithm 856

16.46	Working of stochastic gradient descent algorithm	857
16.47	Path taken by the gradient descent and the stochastic gradient descent to minimize the cost function	858
16.48	Batch sizes in stochastic gradient and gradient descent approach	859
17.1	The cancer dataset	866
17.2	Contents of variable explorer	868
17.3	(a) Snapshot of X (b) Snapshot of y	868
17.4	Encoding categorical data	869
17.5	Variable explorer after the training and testing of the dataset	870
17.6	*Softmax* activation function for multi-class classification	875
17.7	Fitting the training sets to the neural network	877
17.8	Fitting the training sets to the neural network	877
17.9	Fitting the training sets to the neural network	878
17.10	Predicted probability in the range of 0–1	879
17.11	Predicted probability after conversion to binary	879
17.12	Comparison of *y_pred* and *y_test*	880
17.12	Confusion matrix of the ANN classifier	881
17.13	Resulting performance metrics for our decision tree model	883
18.1	A scene in an IT company	890
18.2	Icon of Google Photos app	891
18.3	Google live albums in the Google Photos app	891
18.4	A snapshot from the MNIST dataset	893
18.5	A sample from the MNIST dataset for digit “5”	893
18.6	How computer perceives digit “5”	894
18.7	Converting image of digit “5” to the corresponding picture matrix	894
18.8	The pixel-vector for digit “5”	895
18.9	Using ANN for digit recognition	895
18.10	Digit “5” written in different handwriting	896
18.11	A subset of training data containing other handwritten digits	896
18.12	Variation in ANN results with the change of position of handwritten digits	897
18.13	Deep neural network for training on synthetic data	898
18.14	Sliding window approach	899
18.15	Ambiguous image	900
18.16	Ambiguous image	901
18.17	Optical illusion (Painting by Mexican artist Octavio Ocampo)	901
18.18	Optical illusion	902

18.19 Output of image classifier 903
18.20 A typical image classification task using CNN 903
18.21 How computer perceives an image 904
18.22 Processing of black and white image in a 2D array 904
18.23 Processing of a colored image in a 3D array 905
18.24 A black and white image of digit "5" and its pixel matrix representation 905
18.25 An overview of the CNN model 906
18.26 Input image pixel representation and feature detector or kernel 907
18.27 Dimension guidelines for convolution 907
18.28 Dimensions of feature map when a 7×7 image is convoluted with a 3×3 filter 908
18.29 Resulting feature map 913
18.30 Formation of a convolutional layer 914
18.31 Relevance of ReLU in CNN 915
18.32 Images of dogs with slight variations in them 916
18.33 Max pooling with a window of 2×2 and stride equal to 2 917
18.34 Max pooling with a window of 2×2 and stride equal to 2 918
18.35 Max pooling window of 2×2 providing positional independence of four locations 919
18.36 Max pooling versus average pooling 920
18.37 Convolution followed by pooling 920
18.38 Flattening operation 921
18.39 Feeding flattened feature map to subsequent ANN input layer 921
18.40 Overall architecture of CNN 922
18.41 Connecting the neural nets 922
18.42 Traditional artificial neural network 923
18.43 Full architecture of CNN 923
18.44 Fine-tuning weights and selection of feature descriptors 924
18.45 Example dataset 926
18.46 Model trained as a straight line (underfitted model) 926
18.47 Plot of an underfitting model 927
18.48 Model fitted as zig-zag curve (overfitted model) 927
18.49 Plot of an overfitting model 928
18.50 Right-fitted model as a parabola 928
18.51 Comparison of training and validation data trend concerning model 928
18.52 Training loss vs. validation loss plot based on some model's training 929
19.1 Directory structure of the downloaded dog–cat dataset 938
19.2 Creating training and testing set 939

19.3	Processing of input image	941
19.4	Convolution process	942
19.5	Max pooling with a window of 2×2 and stride equal to 2	943
19.6	Convolution followed by pooling	944
19.7	Flattening	944
19.8	Full CNN model	946
19.9	Code template for image augmentation	947
19.10	Accuracy of our trained CNN model	951
19.11	An unseen image	952
19.12	Input dimension of images fed to CNN	953
19.13	Improved accuracies of the model	956
19.14	Pop-up asking permission to access Google Drive	957
19.15	Directory structure of the *dog–cat* dataset	957
19.16	Some sample images of digits from the MNIST dataset	958
19.17	Our CNN model's performance	963
19.18	A handwritten sample digit	963
19.19	Uploading unseen image in sample_data folder	964
19.19	Prediction for unseen image	964
20.1	Structure of human brain	970
20.2	Architecture of ANN	972
20.3	Squashing ANN	972
20.4	Representation using a single arrow between layers	973
20.5	Vertical representation of layers	973
20.6	Network with a temporal loop	974
20.7	Network after unrolling the temporal loop	974
20.8	Classification of an image by feed-forward network	975
20.9	Relation between classification at time "t" and time "t+1"	975
20.10	Prediction of next word using feed-forward network	976
20.11	Relation between outputs of RNN	977
20.12	Schedule of classes	978
20.13	Predicting subject using feed-forward network	978
20.14	RNN to consider previous outputs	979
20.15	RNN to consider true and predicted outputs	979
20.16	Predicted vector and new information to find output vector	980
20.17	Mathematical representation of RNN	981
20.18	Processing at time interval t(0)	982

20.19 Processing at time interval t(1) 983

20.20 RNN in case of short-term dependencies 984

20.21 RNN in case of long-term dependencies 985

20.22 Backpropagation in RNN 985

20.23 Vanishing gradient 987

20.24 Derivative in case of sigmoid activation function 987

20.25 Exploding gradient 988

20.26 Gated units in LSTM 990

20.27 Online review of laptop 990

20.28 Processing the information one by one 991

20.29 Machine-readable vectors 991

20.30 Hidden state flow between cells 992

20.31 Working inside a cell of an RNN 992

20.32 *tanh* function regulating input values between –1 and 1 993

20.33 Transformation of a vector in the absence of *tanh* function 994

20.34 Transformation of a vector with *tanh* function 994

20.35 LSTM cell 995

20.36 Sigmoid function regulating input values between 0 and 1 997

20.37 Operations of a forget gate 998

20.38 Operations of an input gate 999

20.39 Cell state calculation 1000

20.40 Operations of an output gate 1001

21.1 *DelhiDieselPrice* dataset 1009

21.2 Importing training data and testing data 1010

21.3 Inserting *Stacked Line* charts 1010

21.4 Plotting trend of diesel prices in Delhi based on training data 1011

21.5 Plotting trend of diesel prices in Delhi based on testing data 1011

21.6 Contents of *dataset_train* and *training_set* 1013

21.7 Contents of *training_set_scaled* (*scaled training_set*) 1014

21.8 Different instances of independent variables and their corresponding dependent variables for the training dataset (5299 records) 1016

21.9 Preparing *x_train* and *y_train* from *training_set_scaled* 1017

21.10 Contents of *x_train* 1019

21.11 Contents of *y_train* 1019

21.12 Identical values in the rows of *x_train* 1020

21.13 Variable explorer tab after *reshaping x_train* 1021

21.14 Visualizing *x_train* on the different axis after reshaping it 1022
21.15 Fitting the RNN model to the training data 1029
21.16 Contents of *diesel_price* and *dataset_test* 1029
21.17 Contents of *dataset_total* 1031
21.18 Preparing the data structure for *test data* 1032
21.19 Sub-setting *datset_total* for preparing the required data structure 1032
21.20 *Inputs* data frame after reshaping and transformation 1033
21.21 Process of making predictions (right from generation of *inputs* to making predictions) 1035
21.22 Contents of *predicted_diesel_price* and *diesel_price* 1036
21.23 Actual diesel prices versus predicted diesel prices in Delhi 1037
21.24 Actual diesel prices versus predicted diesel prices in Delhi (timesteps = 100) 1038
22.1 Evolution of humanity 1042
22.2 Food chain of birds 1043
22.3 Most fit sparrows get a chance to survive long 1044
22.4 Process of recombination or crossover 1044
22.5 Process of recombination or crossover 1045
22.6 Process of recombination or crossover 1045
22.7 Process of recombination or crossover 1046
22.8 Process of mutation 1047
22.9 Effect of mutation on human beings 1047
22.10 Mutation followed by natural selection and recombination 1047
22.11 Mutation in apples with a section of different colors over them 1048
22.12 Introduction of abnormalities due to mutation 1048
22.13 Population, chromosomes, and genes in GA 1049
22.14 Variables or parameters of helicopter rotor 1050
22.15 A single-objective optimization problem 1051
22.16 Search space when designing a helicopter rotary 1051
22.17 Role of GA in getting the optimal solution for an optimization problem 1052
22.18 Genetic representation (bitmap) 1053
22.19 Some of the commonly used genetic representations 1053
22.20 Role of a fitness function 1054
22.21 A typical setup of roulette wheel in a casino 1055
22.22 Species playing roulette wheel with a hope to survive to the next generation 1056
22.23 Plotting cumulative frequencies on a number line 1059
22.24 Plotting cumulative frequencies on a number line 1060
22.25 Plotting cumulative frequencies on a number line 1061

22.26	Plotting cumulative frequencies on a number line	1061
22.27	Stochastic universal sampling	1062
22.28	Tournament selection	1063
22.29	Roulette wheel selection where fitness values are very close	1063
22.30	Role of recombination	1065
22.31	Single-point crossover	1065
22.32	Stochastic single-point crossover	1066
22.33	Double-point crossover	1068
22.34	Double-point crossover with overlapping crossover points	1068
22.35	Possible cases of mating	1069
22.36	The process of mutation	1072
22.37	Single-point crossover of a given parent	1073
22.38	Applying mutation over the results of a crossover	1073
22.39	Using the probability of mutation to decide the mutation of a gene	1075
22.40	Working of the main operators in genetic algorithm	1076
22.41	Some populations subjected to selection, crossover, mutation, and elitism	1077
22.42	Elites getting replaced with fitter individuals	1078

Tables

1.1	Key milestones of the foundational phase of machine learning	8
1.2	Key milestones of the transitional phase of machine learning	9
1.3	Key milestones of the contemporary phase of machine learning	10
1.4	Comparison of supervised and unsupervised machine learning	27
1.5	Intial state of Q-Matrix	38
1.6	Updated Q-Matrix after $Q(1,5)$	39
1.7	Updated Q-Matrix after $Q(2, 3)$	40
1.8	Updated Q-Matrix after $Q(3, 1)$	42
1.9	Updated Q-Matrix after $Q(1,5)$	43
1.10	Tabular comparison of data science, data mining, artificial intelligence, machine learning, and deep learning	48
2.1	Differences between lists and tuples	87
2.2	Tabular comparison of list, tuple, dictionary, and set	101
5.1	Instances of the dataset to be used for simple linear regression study	188
5.2	Notations used for simple linear regression	193
5.3	Example dataset	199
5.4	Calculation of cost function	200
5.5	Analysis of cost function with different parameters	207
5.6	Scenario for updating the values of parameters	230
6.1	Computing MSE, MAE, and RMSE manually	260
7.1	Dataset of companies	269
7.2	Multiple linear regression dataset	269
7.3	Populated table with dummy variables	273
7.4	Mathematical representation of different regression models	300
7.5	The institutional position dataset	301

7.6	Effect of degree on polynomial equation	303
7.7	Temperature and vapor pressure readings	306
8.1	Instance of *50_AdAgency*	314
8.2	The institutional position dataset	328
8.3	Predicted salary of the employee at level 7.5 with different models	347
8.4	Temperature and vapor pressure readings	348
9.1	Example to check trend of Profit as Up or Down	375
9.2	Dataset of customer to indicate whether they bought computer or not	382
9.3	Information and Gini index for several events	412
9.4	Dataset for Profit prediction based on the input attributes	412
9.5	Dataset of customers to indicate whether they bought computer or not	420
9.6	Database of customers	459
9.7	Unknown sample X	459
9.8	Calculating the class of unknown instances in the database	460
9.9	Dataset for play-prediction based on given day weather conditions	462
9.10	Precision and recall for bird classifier	482
9.11	Confusion matrix of a three-class classifier	484
10.1	Summary of different Euclidean spaces and their corresponding hyperplanes	514
10.2	Properties of exponent function (e^x)	517
11.1	Some instances of *Alexa_dataset*	536
11.2	Confusion matrix of the decision tree model	548
11.3	Confusion matrix for random forest model	558
11.4	Confusion matrix of the naive Bayes	564
11.5	Confusion matrix for the k-NN	571
11.6	Confusion matrix of the logistic regression	578
11.7	Confusion matrix of the SVM model	584
11.8	Confusion matrix of the SVM model	590
11.9	Comparing the performance of implemented classification models on given dataset	593
11.11	BoW over the documents	598
12.1	Data to calculate Euclidean distances among three students	616
12.2	Data to calculate Chebyshev distances between object A and object B	620
12.3	Example database for k-means algorithm	624
12.4	Database after the first iteration	626
12.5	Database after the second iteration	628
12.6	Database after the second iteration	629
12.7	Initial dataset for $k = 3$	631

12.8 Final assigned cluster for $k = 3$ 632
12.9 Sample instances of *Mall–Customer* dataset 633
12.10 Seed records 634
12.11 Euclidean distance of each record from seed records 635
12.12 Euclidean distance of each record from modified centroids 637
12.13 Euclidean distance of each record from centroids after the second iteration 639
12.14 Record of students' performance 641
12.15 Seed records 641
12.16 First iteration–allocation of each object to its nearest cluster 642
12.17 Updated centroids after the first iteration 643
12.18 Second iteration–allocation of each object to its nearest cluster 643
12.19 Final allocation 644
12.20 Within (intra) cluster distance of records in cluster 1 with its centroid C1(12.67, 22, 17, 38.33) 645
12.21 Within (intra) cluster and between (inter) cluster distance 646
12.22 Distance matrix for a dataset 657
12.23 Distance matrix after clustering 1, 5 657
12.24 Distance matrix after clustering 3, 4 658
12.25 Distance matrix after clustering 2, 1/5 658
12.26 Customers and items brought by them per month 659
12.27 Distance matrix of given data 661
12.28 Updated distance matrix 661
12.29 Updated distance matrix 662
12.30 Updated distance matrix 662
12.31 Record of students' performance 664
12.32 Distance matrix at m:1 665
12.33 Cells involved in C1 666
12.34 Updated distance matrix at m:2 666
12.35 Cells involved in cluster C2 (100/106) 667
12.36 Updated distance matrix at m:3 668
12.37 Cells involved in C3 668
12.38 Updated distance matrix at m:4 669
12.39 Cells involved in C4 670
12.40 Updated distance matrix at m:5 670
12.41 Cells involved in C5 671
12.42 Updated distance matrix at m:6 671