

# Contents

xvii

## Preface

xxxiii

## Before You Begin

<b>1</b>	<b>Introduction to Computers and Python</b>	<b>1</b>
1.1	Introduction	2
1.2	A Quick Review of Object Technology Basics	3
1.3	Python	5
1.4	It's the Libraries!	7
	1.4.1 Python Standard Library	7
	1.4.2 Data-Science Libraries	8
1.5	Test-Drives: Using IPython and Jupyter Notebooks	9
	1.5.1 Using IPython Interactive Mode as a Calculator	9
	1.5.2 Executing a Python Program Using the IPython Interpreter	10
	1.5.3 Writing and Executing Code in a Jupyter Notebook	12
1.6	The Cloud and the Internet of Things	16
	1.6.1 The Cloud	16
	1.6.2 Internet of Things	17
1.7	How Big Is Big Data?	17
	1.7.1 Big Data Analytics	22
	1.7.2 Data Science and Big Data Are Making a Difference: Use Cases	23
1.8	Case Study—A Big-Data Mobile Application	24
1.9	Intro to Data Science: Artificial Intelligence—at the Intersection of CS and Data Science	26
1.10	Wrap-Up	29
<b>2</b>	<b>Introduction to Python Programming</b>	<b>31</b>
2.1	Introduction	32
2.2	Variables and Assignment Statements	32
2.3	Arithmetic	33
2.4	Function print and an Intro to Single- and Double-Quoted Strings	36
2.5	Triple-Quoted Strings	38
2.6	Getting Input from the User	39
2.7	Decision Making: The if Statement and Comparison Operators	41
2.8	Objects and Dynamic Typing	45
2.9	Intro to Data Science: Basic Descriptive Statistics	46
2.10	Wrap-Up	48

<b>3</b>	<b>Control Statements</b>	<b>49</b>
3.1	Introduction	50
3.2	Control Statements	50
3.3	if Statement	51
3.4	if...else and if...elif...else Statements	52
3.5	while Statement	55
3.6	for Statement	55
	3.6.1 Iterables, Lists and Iterators	56
	3.6.2 Built-In range Function	57
3.7	Augmented Assignments	57
3.8	Sequence-Controlled Iteration; Formatted Strings	58
3.9	Sentinel-Controlled Iteration	59
3.10	Built-In Function range: A Deeper Look	60
3.11	Using Type Decimal for Monetary Amounts	61
3.12	break and continue Statements	64
3.13	Boolean Operators and, or and not	65
3.14	Intro to Data Science: Measures of Central Tendency— Mean, Median and Mode	67
3.15	Wrap-Up	69
<b>4</b>	<b>Functions</b>	<b>71</b>
4.1	Introduction	72
4.2	Defining Functions	72
4.3	Functions with Multiple Parameters	75
4.4	Random-Number Generation	76
4.5	Case Study: A Game of Chance	78
4.6	Python Standard Library	81
4.7	math Module Functions	82
4.8	Using IPython Tab Completion for Discovery	83
4.9	Default Parameter Values	85
4.10	Keyword Arguments	85
4.11	Arbitrary Argument Lists	86
4.12	Methods: Functions That Belong to Objects	87
4.13	Scope Rules	87
4.14	import: A Deeper Look	89
4.15	Passing Arguments to Functions: A Deeper Look	90
4.16	Recursion	93
4.17	Functional-Style Programming	95
4.18	Intro to Data Science: Measures of Dispersion	97
4.19	Wrap-Up	98
<b>5</b>	<b>Sequences: Lists and Tuples</b>	<b>101</b>
5.1	Introduction	102
5.2	Lists	102

5.3	Tuples	106
5.4	Unpacking Sequences	108
5.5	Sequence Slicing	110
5.6	del Statement	112
5.7	Passing Lists to Functions	113
5.8	Sorting Lists	115
5.9	Searching Sequences	116
5.10	Other List Methods	117
5.11	Simulating Stacks with Lists	119
5.12	List Comprehensions	120
5.13	Generator Expressions	121
5.14	Filter, Map and Reduce	122
5.15	Other Sequence Processing Functions	124
5.16	Two-Dimensional Lists	126
5.17	Intro to Data Science: Simulation and Static Visualizations	128
	5.17.1 Sample Graphs for 600, 60,000 and 6,000,000 Die Rolls	128
	5.17.2 Visualizing Die-Roll Frequencies and Percentages	129
5.18	Wrap-Up	135
<b>6</b>	<b>Dictionaries and Sets</b>	<b>137</b>
6.1	Introduction	138
6.2	Dictionaries	138
	6.2.1 Creating a Dictionary	138
	6.2.2 Iterating through a Dictionary	139
	6.2.3 Basic Dictionary Operations	140
	6.2.4 Dictionary Methods keys and values	141
	6.2.5 Dictionary Comparisons	143
	6.2.6 Example: Dictionary of Student Grades	143
	6.2.7 Example: Word Counts	144
	6.2.8 Dictionary Method update	146
	6.2.9 Dictionary Comprehensions	146
6.3	Sets	147
	6.3.1 Comparing Sets	148
	6.3.2 Mathematical Set Operations	150
	6.3.3 Mutable Set Operators and Methods	151
	6.3.4 Set Comprehensions	152
6.4	Intro to Data Science: Dynamic Visualizations	152
	6.4.1 How Dynamic Visualization Works	153
	6.4.2 Implementing a Dynamic Visualization	155
6.5	Wrap-Up	158
<b>7</b>	<b>Array-Oriented Programming with NumPy</b>	<b>159</b>
7.1	Introduction	160
7.2	Creating arrays from Existing Data	160
7.3	array Attributes	161

7.4	Filling arrays with Specific Values	163
7.5	Creating arrays from Ranges	164
7.6	List vs. array Performance: Introducing %timeit	165
7.7	array Operators	167
7.8	NumPy Calculation Methods	169
7.9	Universal Functions	170
7.10	Indexing and Slicing	171
7.11	Views: Shallow Copies	173
7.12	Deep Copies	174
7.13	Reshaping and Transposing	175
7.14	Intro to Data Science: pandas Series and DataFrames	177
	7.14.1 pandas Series	178
	7.14.2 DataFrames	182
7.15	Wrap-Up	189

## **8 Strings: A Deeper Look** **191**

8.1	Introduction	192
8.2	Formatting Strings	193
	8.2.1 Presentation Types	193
	8.2.2 Field Widths and Alignment	194
	8.2.3 Numeric Formatting	195
	8.2.4 String's format Method	195
8.3	Concatenating and Repeating Strings	196
8.4	Stripping Whitespace from Strings	197
8.5	Changing Character Case	197
8.6	Comparison Operators for Strings	198
8.7	Searching for Substrings	198
8.8	Replacing Substrings	199
8.9	Splitting and Joining Strings	200
8.10	Characters and Character-Testing Methods	202
8.11	Raw Strings	203
8.12	Introduction to Regular Expressions	203
	8.12.1 re Module and Function fullmatch	204
	8.12.2 Replacing Substrings and Splitting Strings	207
	8.12.3 Other Search Functions; Accessing Matches	208
8.13	Intro to Data Science: Pandas, Regular Expressions and Data Munging	210
8.14	Wrap-Up	214

## **9 Files and Exceptions** **217**

9.1	Introduction	218
9.2	Files	219
9.3	Text-File Processing	219
	9.3.1 Writing to a Text File: Introducing the with Statement	220
	9.3.2 Reading Data from a Text File	221

9.4	Updating Text Files	222
9.5	Serialization with JSON	223
9.6	Focus on Security: pickle Serialization and Deserialization	226
9.7	Additional Notes Regarding Files	226
9.8	Handling Exceptions	227
9.8.1	Division by Zero and Invalid Input	227
9.8.2	try Statements	228
9.8.3	Catching Multiple Exceptions in One except Clause	230
9.8.4	What Exceptions Does a Function or Method Raise?	230
9.8.5	What Code Should Be Placed in a try Suite?	230
9.9	finally Clause	231
9.10	Explicitly Raising an Exception	233
9.11	(Optional) Stack Unwinding and Tracebacks	233
9.12	Intro to Data Science: Working with CSV Files	235
9.12.1	Python Standard Library Module csv	235
9.12.2	Reading CSV Files into Pandas DataFrames	237
9.12.3	Reading the Titanic Disaster Dataset	238
9.12.4	Simple Data Analysis with the Titanic Disaster Dataset	239
9.12.5	Passenger Age Histogram	240
9.13	Wrap-Up	241
<b>10</b>	<b>Object-Oriented Programming</b>	<b>243</b>
10.1	Introduction	244
10.2	Custom Class Account	246
10.2.1	Test-Driving Class Account	246
10.2.2	Account Class Definition	248
10.2.3	Composition: Object References as Members of Classes	249
10.3	Controlling Access to Attributes	249
10.4	Properties for Data Access	250
10.4.1	Test-Driving Class Time	250
10.4.2	Class Time Definition	252
10.4.3	Class Time Definition Design Notes	255
10.5	Simulating “Private” Attributes	256
10.6	Case Study: Card Shuffling and Dealing Simulation	258
10.6.1	Test-Driving Classes Card and DeckOfCards	258
10.6.2	Class Card—Introducing Class Attributes	259
10.6.3	Class DeckOfCards	261
10.6.4	Displaying Card Images with Matplotlib	263
10.7	Inheritance: Base Classes and Subclasses	266
10.8	Building an Inheritance Hierarchy; Introducing Polymorphism	267
10.8.1	Base Class CommissionEmployee	268
10.8.2	Subclass SalariedCommissionEmployee	270
10.8.3	Processing CommissionEmployees and SalariedCommissionEmployees Polymorphically	274

10.8.4	A Note About Object-Based and Object-Oriented Programming	274
10.9	Duck Typing and Polymorphism	275
10.10	Operator Overloading	276
10.10.1	Test-Driving Class Complex	277
10.10.2	Class Complex Definition	278
10.11	Exception Class Hierarchy and Custom Exceptions	279
10.12	Named Tuples	280
10.13	A Brief Intro to Python 3.7's New Data Classes	281
10.13.1	Creating a Card Data Class	282
10.13.2	Using the Card Data Class	284
10.13.3	Data Class Advantages over Named Tuples	286
10.13.4	Data Class Advantages over Traditional Classes	286
10.14	Unit Testing with Docstrings and doctest	287
10.15	Namespaces and Scopes	290
10.16	Intro to Data Science: Time Series and Simple Linear Regression	293
10.17	Wrap-Up	301

## **11 Natural Language Processing (NLP) 303**

11.1	Introduction	304
11.2	TextBlob	305
11.2.1	Create a TextBlob	307
11.2.2	Tokenizing Text into Sentences and Words	307
11.2.3	Parts-of-Speech Tagging	307
11.2.4	Extracting Noun Phrases	308
11.2.5	Sentiment Analysis with TextBlob's Default Sentiment Analyzer	309
11.2.6	Sentiment Analysis with the NaiveBayesAnalyzer	310
11.2.7	Language Detection and Translation	311
11.2.8	Inflection: Pluralization and Singularization	312
11.2.9	Spell Checking and Correction	313
11.2.10	Normalization: Stemming and Lemmatization	314
11.2.11	Word Frequencies	314
11.2.12	Getting Definitions, Synonyms and Antonyms from WordNet	315
11.2.13	Deleting Stop Words	317
11.2.14	n-grams	318
11.3	Visualizing Word Frequencies with Bar Charts and Word Clouds	319
11.3.1	Visualizing Word Frequencies with Pandas	319
11.3.2	Visualizing Word Frequencies with Word Clouds	321
11.4	Readability Assessment with Textastic	324
11.5	Named Entity Recognition with spaCy	326
11.6	Similarity Detection with spaCy	327
11.7	Other NLP Libraries and Tools	328
11.8	Machine Learning and Deep Learning Natural Language Applications	328
11.9	Natural Language Datasets	329
11.10	Wrap-Up	330

<b>12</b>	<b>Data Mining Twitter</b>	<b>331</b>
12.1	Introduction	332
12.2	Overview of the Twitter APIs	334
12.3	Creating a Twitter Account	335
12.4	Getting Twitter Credentials—Creating an App	335
12.5	What's in a Tweet?	337
12.6	Tweepy	340
12.7	Authenticating with Twitter Via Tweepy	341
12.8	Getting Information About a Twitter Account	342
12.9	Introduction to Tweepy Cursors: Getting an Account's Followers and Friends	344
12.9.1	Determining an Account's Followers	344
12.9.2	Determining Whom an Account Follows	346
12.9.3	Getting a User's Recent Tweets	346
12.10	Searching Recent Tweets	347
12.11	Spotting Trends: Twitter Trends API	349
12.11.1	Places with Trending Topics	350
12.11.2	Getting a List of Trending Topics	351
12.11.3	Create a Word Cloud from Trending Topics	352
12.12	Cleaning/Preprocessing Tweets for Analysis	353
12.13	Twitter Streaming API	354
12.13.1	Creating a Subclass of StreamListener	355
12.13.2	Initiating Stream Processing	357
12.14	Tweet Sentiment Analysis	359
12.15	Geocoding and Mapping	362
12.15.1	Getting and Mapping the Tweets	364
12.15.2	Utility Functions in tweetutilities.py	367
12.15.3	Class LocationListener	369
12.16	Ways to Store Tweets	370
12.17	Twitter and Time Series	370
12.18	Wrap-Up	371
<b>13</b>	<b>IBM Watson and Cognitive Computing</b>	<b>373</b>
13.1	Introduction: IBM Watson and Cognitive Computing	374
13.2	IBM Cloud Account and Cloud Console	375
13.3	Watson Services	376
13.4	Additional Services and Tools	379
13.5	Watson Developer Cloud Python SDK	381
13.6	Case Study: Traveler's Companion Translation App	381
13.6.1	Before You Run the App	382
13.6.2	Test-Driving the App	383
13.6.3	SimpleLanguageTranslator.py Script Walkthrough	384
13.7	Watson Resources	394
13.8	Wrap-Up	395

<b>14</b>	<b>Machine Learning: Classification, Regression and Clustering</b>	<b>397</b>
14.1	Introduction to Machine Learning	398
14.1.1	Scikit-Learn	399
14.1.2	Types of Machine Learning	400
14.1.3	Datasets Bundled with Scikit-Learn	402
14.1.4	Steps in a Typical Data Science Study	403
14.2	Case Study: Classification with k-Nearest Neighbors and the Digits Dataset, Part 1	403
14.2.1	k-Nearest Neighbors Algorithm	404
14.2.2	Loading the Dataset	406
14.2.3	Visualizing the Data	409
14.2.4	Splitting the Data for Training and Testing	411
14.2.5	Creating the Model	412
14.2.6	Training the Model	412
14.2.7	Predicting Digit Classes	413
14.3	Case Study: Classification with k-Nearest Neighbors and the Digits Dataset, Part 2	413
14.3.1	Metrics for Model Accuracy	414
14.3.2	K-Fold Cross-Validation	417
14.3.3	Running Multiple Models to Find the Best One	418
14.3.4	Hyperparameter Tuning	420
14.4	Case Study: Time Series and Simple Linear Regression	420
14.5	Case Study: Multiple Linear Regression with the California Housing Dataset	425
14.5.1	Loading the Dataset	426
14.5.2	Exploring the Data with Pandas	428
14.5.3	Visualizing the Features	430
14.5.4	Splitting the Data for Training and Testing	434
14.5.5	Training the Model	434
14.5.6	Testing the Model	435
14.5.7	Visualizing the Expected vs. Predicted Prices	436
14.5.8	Regression Model Metrics	437
14.5.9	Choosing the Best Model	438
14.6	Case Study: Unsupervised Machine Learning, Part 1—Dimensionality Reduction	438
14.7	Case Study: Unsupervised Machine Learning, Part 2—k-Means Clustering	442
14.7.1	Loading the Iris Dataset	444
14.7.2	Exploring the Iris Dataset: Descriptive Statistics with Pandas	446
14.7.3	Visualizing the Dataset with a Seaborn pairplot	447
14.7.4	Using a KMeans Estimator	450
14.7.5	Dimensionality Reduction with Principal Component Analysis	452
14.7.6	Choosing the Best Clustering Estimator	453
14.8	Wrap-Up	455



<b>15</b>	<b>Deep Learning</b>	<b>457</b>
15.1	Introduction	458
15.1.1	Deep Learning Applications	460
15.1.2	Deep Learning Demos	461
15.1.3	Keras Resources	461
15.2	Keras Built-In Datasets	461
15.3	Custom Anaconda Environments	462
15.4	Neural Networks	463
15.5	Tensors	465
15.6	Convolutional Neural Networks for Vision; Multi-Classification with the MNIST Dataset	467
15.6.1	Loading the MNIST Dataset	468
15.6.2	Data Exploration	469
15.6.3	Data Preparation	471
15.6.4	Creating the Neural Network	473
15.6.5	Training and Evaluating the Model	480
15.6.6	Saving and Loading a Model	485
15.7	Visualizing Neural Network Training with TensorBoard	486
15.8	ConvnetJS: Browser-Based Deep-Learning Training and Visualization	489
15.9	Recurrent Neural Networks for Sequences; Sentiment Analysis with the IMDb Dataset	489
15.9.1	Loading the IMDb Movie Reviews Dataset	490
15.9.2	Data Exploration	491
15.9.3	Data Preparation	493
15.9.4	Creating the Neural Network	494
15.9.5	Training and Evaluating the Model	496
15.10	Tuning Deep Learning Models	497
15.11	Convnet Models Pretrained on ImageNet	498
15.12	Wrap-Up	499
<b>16</b>	<b>Big Data: Hadoop, Spark, NoSQL and IoT</b>	<b>501</b>
16.1	Introduction	502
16.2	Relational Databases and Structured Query Language (SQL)	506
16.2.1	A books Database	507
16.2.2	SELECT Queries	511
16.2.3	WHERE Clause	511
16.2.4	ORDER BY Clause	512
16.2.5	Merging Data from Multiple Tables: INNER JOIN	514
16.2.6	INSERT INTO Statement	514
16.2.7	UPDATE Statement	515
16.2.8	DELETE FROM Statement	516
16.3	NoSQL and NewSQL Big-Data Databases: A Brief Tour	517
16.3.1	NoSQL Key-Value Databases	517
16.3.2	NoSQL Document Databases	518

16.3.3	NoSQL Columnar Databases	518
16.3.4	NoSQL Graph Databases	519
16.3.5	NewSQL Databases	519
16.4	Case Study: A MongoDB JSON Document Database	520
16.4.1	Creating the MongoDB Atlas Cluster	521
16.4.2	Streaming Tweets into MongoDB	522
16.5	Hadoop	530
16.5.1	Hadoop Overview	531
16.5.2	Summarizing Word Lengths in <i>Romeo and Juliet</i> via MapReduce	533
16.5.3	Creating an Apache Hadoop Cluster in Microsoft Azure HDInsight	533
16.5.4	Hadoop Streaming	535
16.5.5	Implementing the Mapper	536
16.5.6	Implementing the Reducer	537
16.5.7	Preparing to Run the MapReduce Example	537
16.5.8	Running the MapReduce Job	538
16.6	Spark	541
16.6.1	Spark Overview	541
16.6.2	Docker and the Jupyter Docker Stacks	542
16.6.3	Word Count with Spark	545
16.6.4	Spark Word Count on Microsoft Azure	548
16.7	Spark Streaming: Counting Twitter Hashtags Using the pyspark-notebook Docker Stack	551
16.7.1	Streaming Tweets to a Socket	551
16.7.2	Summarizing Tweet Hashtags; Introducing Spark SQL	555
16.8	Internet of Things and Dashboards	560
16.8.1	Publish and Subscribe	561
16.8.2	Visualizing a PubNub Sample Live Stream with a Freeboard Dashboard	562
16.8.3	Simulating an Internet-Connected Thermostat in Python	564
16.8.4	Creating the Dashboard with Freeboard.io	566
16.8.5	Creating a Python PubNub Subscriber	567
16.9	Wrap-Up	571

**Index**