LEARNING MADE EASY

2nd Edition

Python® for Data Science

for dummies®
A Wiley Brand

Learn Python data analysis programming and statistics

Write code in the cloud with Google Colab™

Wrangle data and visualize information

John Paul Mueller
Luca Massaron

Authors of *Machine Learning for Dummies* and *Artificial Intelligence For Dummies*

Python® for Data Science

for dummies®
A Wiley Brand

Python® for Data Science

2nd Edition

by John Paul Mueller and Luca Massaron

for dummies®
A Wiley Brand

Python® for Data Science For Dummies®, 2nd Edition

Published by: **John Wiley & Sons, Inc.,** 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Published simultaneously in Canada

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit https://hub.wiley.com/community/support/dummies.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at http://booksupport.wiley.com. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2018967877

ISBN: 978-1-119-54762-4; ISBN: 978-1-119-54766-2 (ebk); ISBN: 978-1-119-54764-8 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Contents at a Glance

Table of Contents

Introduction

Data is increasingly used for every possible purpose, and many of those purposes elude attention, but every time you get on the Internet, you generate even more. It's not just you, either; the growth of the Internet has been phenomenal, according to Internet World Stats (`https://www.internetworldstats.com/emarketing.htm`). Data science turns this huge amount of data into something useful — something that you use absolutely every day to perform an amazing array of tasks or to obtain services from someone else.

In fact, you've probably used data science in ways that you never expected. For example, when you used your favorite search engine this morning to look for something, it made suggestions on alternative search terms. Those terms are supplied by data science. When you went to the doctor last week and discovered the lump you found wasn't cancer, the doctor likely made her prognosis with the help of data science. In fact, you might work with data science every day and not even know it. *Python for Data Science For Dummies,* 2nd Edition not only gets you started using data science to perform a wealth of practical tasks but also helps you realize just how many places data science is used. By knowing how to answer data science problems and where to employ data science, you gain a significant advantage over everyone else, increasing your chances at promotion or that new job you really want.

About This Book

The main purpose of *Python for Data Science For Dummies,* 2nd Edition is to take the scare factor out of data science by showing you that data science is not only really interesting but also quite doable using Python. You might assume that you need to be a computer science genius to perform the complex tasks normally associated with data science, but that's far from the truth. Python comes with a host of useful libraries that do all the heavy lifting for you in the background. You don't even realize how much is going on, and you don't need to care. All you really need to know is that you want to perform specific tasks, and Python makes these tasks quite accessible.

Part of the emphasis of this book is on using the right tools. You start with Anaconda, a product that includes IPython and Jupyter Notebook — two tools that take the sting out of working with Python. You experiment with IPython in a fully interactive environment. The code you place in Jupyter Notebook (also called just Notebook throughout the book) is presentation quality, and you can mix a number of presentation elements right there in your document. It's not really like using a development environment at all. To make this book easier to use on alternative platforms, you also discover an online Interactive Development Environment application (IDE) named Google Colab that allows you to interact with most, but not quite all, of the book examples using your favorite tablet or (assuming that you can squint well enough) your smart phone.

You also discover some interesting techniques in this book. For example, you can create plots of all your data science experiments using MatPlotLib, and this book gives you all the details for doing that. This book also spends considerable time showing you available resources (such as packages) and how you can use Scikit-learn to perform some really interesting calculations. Many people would like to know how to perform handwriting recognition, and if you're one of them, you can use this book to get a leg up on the process.

Of course, you might still be worried about the whole programming environment issue, and this book doesn't leave you in the dark there, either. At the beginning, you find complete installation instructions for Anaconda, which are followed by the methods you need to get started with data science using Jupyter Notebook or Google Colab. The emphasis is on getting you up and running as quickly as possible, and to make examples straightforward and simple so that the code doesn't become a stumbling block to learning.

This second edition of the book provides you with updated examples using Python 3.*x* so that you're using the most modern version of Python while reading. In addition, you find a stronger emphasis on making examples simpler, but also making the environment more inclusive by adding material on deep learning. Consequently, you get a lot more out of this edition of the book as a result of the input provided by hundreds of readers before you.

To make absorbing the concepts even easier, this book uses the following conventions:

- Text that you're meant to type just as it appears in the book is in **bold**. The exception is when you're working through a step list: Because each step is bold, the text to type is not bold.
- When you see words in *italics* as part of a typing sequence, you need to replace that value with something that works for you. For example, if you

see "Type ***Your Name*** and press Enter," you need to replace *Your Name* with your actual name.

- Web addresses and programming code appear in `monofont`. If you're reading a digital version of this book on a device connected to the Internet, note that you can click the web address to visit that website, like this: `http://www.dummies.com`.
- When you need to type command sequences, you see them separated by a special arrow, like this: File ➪ New File. In this example, you go to the File menu first and then select the New File entry on that menu.

Foolish Assumptions

You might find it difficult to believe that we've assumed anything about you — after all, we haven't even met you yet! Although most assumptions are indeed foolish, we made these assumptions to provide a starting point for the book.

You need to e familiar with the platform you want to use because the book doesn't offer any guidance in this regard. (Chapter 3 does, however, provide Anaconda installation instructions, and Chapter 4 gets you started with Google Colab.) To provide you with maximum information about Python concerning how it applies to data science, this book doesn't discuss any platform-specific issues. You really do need to know how to install applications, use applications, and generally work with your chosen platform before you begin working with this book.

You must know how to work with Python. This edition of the book no longer contains a Python primer because you can find such as wealth of tutorials online (see `https://www.w3schools.com/python/` and `https://www.tutorialspoint.com/python/` as examples).

This book isn't a math primer. Yes, you see lots of examples of complex math, but the emphasis is on helping you use Python and data science to perform analysis tasks rather than teaching math theory. Chapters 1 and 2 give you a better understanding of precisely what you need to know to use this book successfully.

This book also assumes that you can access items on the Internet. Sprinkled throughout are numerous references to online material that will enhance your learning experience. However, these added sources are useful only if you actually find and use them.

Icons Used in This Book

As you read this book, you see icons in the margins that indicate material of interest (or not, as the case may be). This section briefly describes each icon in this book.


TIP

Tips are nice because they help you save time or perform some task without a lot of extra work. The tips in this book are time-saving techniques or pointers to resources that you should try in order to get the maximum benefit from Python or in performing data science–related tasks.


WARNING

We don't want to sound like angry parents or some kind of maniacs, but you should avoid doing anything that's marked with a Warning icon. Otherwise, you might find that your application fails to work as expected, you get incorrect answers from seemingly bulletproof equations, or (in the worst-case scenario) you lose data.


TECHNICAL STUFF

Whenever you see this icon, think advanced tip or technique. You might find these tidbits of useful information just too boring for words, or they could contain the solution you need to get a program running. Skip these bits of information whenever you like.


REMEMBER

If you don't get anything else out of a particular chapter or section, remember the material marked by this icon. This text usually contains an essential process or a bit of information that you must know to work with Python or to perform data science–related tasks successfully.

Beyond the Book

This book isn't the end of your Python or data science experience — it's really just the beginning. We provide online content to make this book more flexible and better able to meet your needs. That way, as we receive e-mail from you, we can address questions and tell you how updates to either Python or its associated add-ons affect book content. In fact, you gain access to all these cool additions:

- **Cheat sheet:** You remember using crib notes in school to make a better mark on a test, don't you? You do? Well, a cheat sheet is sort of like that. It provides you with some special notes about tasks that you can do with Python, IPython, IPython Notebook, and data science that not every other person knows. You can find the cheat sheet by going to `www.dummies.com`, searching this book's title, and scrolling down the page that appears. The cheat sheet contains really

neat information such as the most common programming mistakes that cause people woe when using Python.

- **Updates:** Sometimes changes happen. For example, we might not have seen an upcoming change when we looked into our crystal ball during the writing of this book. In the past, this possibility simply meant that the book became outdated and less useful, but you can now find updates to the book by searching this book's title at `www.dummies.com`.

  In addition to these updates, check out the blog posts with answers to reader questions and demonstrations of useful book-related techniques at `http://blog.johnmuellerbooks.com/`.

- **Companion files:** Hey! Who really wants to type all the code in the book and reconstruct all those plots manually? Most readers would prefer to spend their time actually working with Python, performing data science tasks, and seeing the interesting things they can do, rather than typing. Fortunately for you, the examples used in the book are available for download, so all you need to do is read the book to learn Python for data science usage techniques. You can find these files at `www.dummies.com`. Search this book's title, and on the page that appears, scroll down to the image of the book cover and click it. Then click the More about This Book button and on the page that opens, go to the Downloads tab.

Where to Go from Here

It's time to start your Python for data science adventure! If you're completely new to Python and its use for data science tasks, you should start with Chapter 1 and progress through the book at a pace that allows you to absorb as much of the material as possible.

If you're a novice who's in an absolute rush to get going with Python for data science as quickly as possible, you can skip to Chapter 3 with the understanding that you may find some topics a bit confusing later. Skipping to Chapter 5 is okay if you already have Anaconda (the programming product used in the book) installed, but be sure to at least skim Chapter 3 so that you know what assumptions we made when writing this book. If you plan to use your tablet to work with this book, be certain to review Chapter 4 so that you understand the limitations presented by Google Colab in running the example code; not all of the examples work in this IDE. Make sure to install Anaconda with Python version 3.6.5 installed to obtain the best results from the book's source code.

Readers who have some exposure to Python and have Anaconda installed can save reading time by moving directly to Chapter 5. You can always go back to earlier chapters as necessary when you have questions. However, you should understand how each technique works before moving to the next one. Every technique, coding example, and procedure has important lessons for you, and you could miss vital content if you start skipping too much information.

1

Getting Started with Data Science and Python

IN THIS PART . . .

Understanding how Python can make data science easier.

Defining the Python features commonly used for data science.

Creating a Python setup of your own.

Working with Google Colab on alternative devices.

IN THIS CHAPTER

- Discovering the wonders for data science
- Exploring how data science works
- Creating the connection between Python and data science
- Getting started with Python

Chapter 1

Discovering the Match between Data Science and Python

Data science may seem like one of those technologies that you'd never use, but you'd be wrong. Yes, data science involves the use of advanced math techniques, statistics, and big data. However, data science also involves helping you make smart decisions, creating suggestions for options based on previous choices, and making robots see objects. In fact, people use data science in so many different ways that you literally can't look anywhere or do anything without feeling the effects of data science on your life. In short, data science is the person behind the partition in the experience of the wonderment of technology. Without data science, much of what you accept as typical and expected today wouldn't even be possible. This is the reason that being a data scientist is the sexiest job of the twenty-first century.

REMEMBER

To make data science doable by someone who's less than a math genius, you need tools. You could use any of a number of tools to perform data science tasks, but Python is uniquely suited to making it easier to work with data science. For one thing, Python provides an incredible number of math-related libraries that help you perform tasks with a less-than-perfect understanding of precisely what is

going on. However, Python goes further by supporting multiple coding styles (programming paradigms) and doing other things to make your job easier. Therefore, yes, you could use other languages to write data science applications, but Python reduces your workload, so it's a natural choice for those who really don't want to work hard, but rather to work smart.

This chapter gets you started with Python. Even though this book isn't designed to provide you with a complete Python tutorial, exploring some basic Python issues will reduce the time needed for you to get up to speed. (If you do need a good starting tutorial, please get *Beginning Programming with Python For Dummies*, 2nd Edition, by John Mueller (Wiley). You'll find that the book provides pointers to tutorials and other aids as needed to fill in any gaps that you may have in your Python education.

CHOOSING A DATA SCIENCE LANGUAGE

There are many different programming languages in the world — and most were designed to perform tasks in a certain way or even make it easier for a particular profession's work to be done with greater ease. Choosing the correct tool makes your life easier. It's akin to using a hammer to drive a screw rather than a screwdriver. Yes, the hammer works, but the screwdriver is much easier to use and definitely does a better job. Data scientists usually use only a few languages because they make working with data easier. With this in mind, here are the top languages for data science work in order of preference:

- **Python (general purpose):** Many data scientists prefer to use Python because it provides a wealth of libraries, such as NumPy, SciPy, MatPlotLib, pandas, and Scikit-learn, to make data science tasks significantly easier. Python is also a precise language that makes it easy to use multi-processing on large datasets — reducing the time required to analyze them. The data science community has also stepped up with specialized IDEs, such as Anaconda, that implement the Jupyter Notebook concept, which makes working with data science calculations significantly easier (Chapter 3 demonstrates how to use Jupyter Notebook, so don't worry about it in this chapter). Besides all of these things in Python's favor, it's also an excellent language for creating glue code with languages such as C/C++ and Fortran. The Python documentation actually shows how to create the required extensions. Most Python users rely on the language to see patterns, such as allowing a robot to see a group of pixels as an object. It also sees use for all sorts of scientific tasks.
- **R (special purpose statistical):** In many respects, Python and R share the same sorts of functionality but implement it in different ways. Depending on which source you view, Python and R have about the same number of proponents, and some people use Python and R interchangeably (or sometimes in tandem). Unlike

Python, R provides its own environment, so you don't need a third-party product such as Anaconda. However, R doesn't appear to mix with other languages with the ease that Python provides.

- **SQL (database management):** The most important thing to remember about Structured Query Language (SQL) is that it focuses on data rather than tasks. Businesses can't operate without good data management — the data is the business. Large organizations use some sort of relational database, which is normally accessible with SQL, to store their data. Most Database Management System (DBMS) products rely on SQL as their main language, and DBMS usually has a large number of data analysis and other data science features built in. Because you're accessing the data natively, there is often a significant speed gain in performing data science tasks this way. Database Administrators (DBAs) generally use SQL to manage or manipulate the data rather than necessarily perform detailed analysis of it. However, the data scientist can also use SQL for various data science tasks and make the resulting scripts available to the DBAs for their needs.
- **Java (general purpose):** Some data scientists perform other kinds of programming that require a general purpose, widely adapted and popular, language. In addition to providing access to a large number of libraries (most of which aren't actually all that useful for data science, but do work for other needs), Java supports object orientation better than any of the other languages in this list. In addition, it's strongly typed and tends to run quite quickly. Consequently, some people prefer it for finalized code. Java isn't a good choice for experimentation or ad hoc queries.
- **Scala (general purpose):** Because Scala uses the Java Virtual Machine (JVM) it does have some of the advantages and disadvantages of Java. However, like Python, Scala provides strong support for the functional programming paradigm, which uses lambda calculus as its basis (see *Functional Programmming For Dummies,* by John Mueller [Wiley] for details). In addition, Apache Spark is written in Scala, which means that you have good support for cluster computing when using this language; — think huge dataset support. Some of the pitfalls of using Scala are that it's hard to set up correctly, it has a steep learning curve, and it lacks a comprehensive set of data science specific libraries.

Defining the Sexiest Job of the 21st Century

At one point, the world viewed anyone working with statistics as a sort of accountant or perhaps a mad scientist. Many people consider statistics and analysis of data boring. However, data science is one of those occupations in which the more you learn, the more you want to learn. Answering one question often spawns more questions that are even more interesting than the one you just answered.

However, the thing that makes data science so sexy is that you see it everywhere and used in an almost infinite number of ways. The following sections provide you with more details on why data science is such an amazing field of study.

Considering the emergence of data science

Data science is a relatively new term. William S. Cleveland coined the term in 2001 as part of a paper entitled "Data Science: An Action Plan for Expanding the Technical Areas of the Field of Statistics." It wasn't until a year later that the International Council for Science actually recognized data science and created a committee for it. Columbia University got into the act in 2003 by beginning publication of the *Journal of Data Science.*


REMEMBER

However, the mathematical basis behind data science is centuries old because data science is essentially a method of viewing and analyzing statistics and probability. The first essential use of statistics as a term comes in 1749, but statistics are certainly much older than that. People have used statistics to recognize patterns for thousands of years. For example, the historian Thucydides (in his History of the Peloponnesian War) describes how the Athenians calculated the height of the wall of Platea in fifth century BC by counting bricks in an unplastered section of the wall. Because the count needed to be accurate, the Athenians took the average of the count by several solders.

The process of quantifying and understanding statistics is relatively new, but the science itself is quite old. An early attempt to begin documenting the importance of statistics appears in the ninth century when Al-Kindi wrote *Manuscript on Deciphering Cryptographic Messages.* In this paper, Al-Kindi describes how to use a combination of statistics and frequency analysis to decipher encrypted messages. Even in the beginning, statistics saw use in practical application of science to tasks that seemed virtually impossible to complete. Data science continues this process, and to some people it might actually seem like magic.

Outlining the core competencies of a data scientist

As is true of anyone performing most complex trades today, the data scientist requires knowledge of a broad range of skills to perform the required tasks. In fact, so many different skills are required that data scientists often work in teams. Someone who is good at gathering data might team up with an analyst and someone gifted in presenting information. It would be hard to find a single person with all the required skills. With this in mind, the following list describes areas in which a data scientist could excel (with more competencies being better):