1
Introduction

This book shares my experience of teaching undergraduate and graduate students in political science and public policy how to plot findings to guide decision-making in scientific and professional settings. At both levels of education, I found students with different backgrounds in quantitative tools. To my surprise, younger generations of undergrads, which master many technological gadgets, are having a hard time dealing with analytics, and have no basic understanding of how to encode information obtained into a visual. At graduate level, on the other hand, it seems easier to teach these contents as they relate analytics to problems they have been facing. That required that I developed a particular teaching approach for not leaving anybody behind, while keeping my class interesting, so the reader should expect detailed explanations in this book.

My courses emphasize three different skills: data pre-processing (collecting and organizing), data analysis (modeling and inferencing), and producing information (visualizing and publishing). This book focuses on the last stage, and all its related procedures. In a previous work, I covered the first skill (Magallanes Reyes, 2017), which is arguably the most time-consuming; but finding a way to produce insight once the second stage is done (analysis) is an art difficult to transfer, and even more so if the tools are difficult to learn (or to buy).

However, I think that it is not the lack of visualization tools that causes trouble; it is the opposite, there is an incredible amount of methods of producing graphics that might confuse social and policy science scholars during their first steps in data science. I plan to give a step-by-step approach to the basics of producing information visually, a crucial element in today's complex society.

To be honest, I thought that writing this book would not be a difficult task, but it has taken too long to produce as it was done during the COVID-19

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

pandemic and, like most you, my life habits changed in many ways. But even though my work took longer than expected, I also had the chance to confirm how valuable it is to have simple plans available to help decision-making, while explaining why a decision had to be made.

Keeping that in mind, I decided to prepare a book that will not contain beautiful visuals that are difficult to interpret, but a collection of plots familiar enough to most people, but with a lot of explanations on how to make plots most audiences can understand. I will do my best to avoid offering too many choices of plots in each case; but in general, I will produce a simple plot and expand on it as long as the extra detail will not distract the reader.

1.1 Some Assumptions

This book assumes you want to learn to produce the best possible visuals. So, I assume you are already producing plots, but you are not sure if your choices are the right ones. I also assume you do not know if Python is better than R or vice versa, so I offer you both versions; however, every visual you see in the book is made using R and I just offer a Python code that will produce the same result as R.

I also assume that you believe that using the *the grammar of graphics* that R uses via the package **ggplot** (Wickham, 2016) is a well-documented option; thus, every R plot will be made using that approach and I will avoid using a different one; this will put me in trouble with Python as many great libraries do not follow that same coding approach; this will not be solved easily, so several times you will not be able to see the same coding approach.

A strong assumption is that you know some basic R or Python. I will not use complex operations at all in R, but you might find some Python code that is not easy to get understand just by reading it (so pay attention to the explanations I will give). I also assume you know how to install libraries or packages in R and Python.

Finally, I have not produced interactive plots, because I assume that if you can prepare a static plot, you can take an extra step and make it interactive. After assuming so many things, I was surprised by the amount of pages this book contains.

Why R or Python? This book will use **R** and **Python** to teach you how to prepare an informative visualization. The selection does not mean that you should avoid visualization tools that require no coding, it just means that if you have done pre-processing and modeling in those languages, you can still use

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

them to communicate your results or findings. I am also using them because the market requests skills in both languages for the non computer scientist. If it were not for R or Python, data science would not have invaded the territory of social science. I can not imagine social science programs or government schools teaching JAVA or C++.

Python and R are also attractive as they are well documented, with lots of applications in different areas of knowledge, active communities in the web sharing code and examples; and, of course, you can use R and Python free of charge. Some other details follow:

- **R** is a high-level programming language. That is, its creators have tried to abstract it, so that some commands are in English. This means that there are low level languages that *talk* to the computer in a language closer to what a computer understands (far from direct human understanding). Because it is free-of-charge and open-source, it has allowed many scholars not only to carry out data analysis, but also to contribute to R itself with very specific functionalities, so that R now has support for almost any kind of quantitative technique. It has been said that R represents a slow learning curve at the beginning, but I found that working with R is just different, not harder. However, that depends on the coding style the user develops (messy codes are difficult to follow in any programming language). This book emphasizes a basic coding style and habits to make R instantaneously reachable.
- **Python**, is an all-purpose programming language with many more features for computational work than R, but, for the goals of this book, I will not make that difference clear for the reader. I can only say that you can not build sophisticated information systems with R, but for sure you can with Python. Python, as well as R, has a very active community of users and developers, and you can practically write any question about Python in your browser and find several answers. After reading this book you can try that, just keep in mind that those who reply are often advanced users who may use jargon that a novice user may not understand.

As for speed, regular users may not find a difference between Python and R. If you find that computing results is taking too much time, writing a better code can improve the speed, but advanced programming techniques require more preparation, and the code may become difficult to read. This book will not turn you into an advanced programmer; it will turn you into an effective user of both languages in order to deal with situations similar to the examples. It may be that your code is very good, so take into consideration that other factors also affect speed, such as the size and contents of the file, your hardware (laptop,

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

tablet, etc.), your internet provider, your operating system version, the memory available, and so on. For the examples in this book, speed will not be an issue.

1.2 R and Python Environments

There are several ways you can use R or Python, but I have used these environments:

- **Anaconda** for **Python**.
- **RStudio** for **R**.

The code I use in the book is independent of what environment you use. Anaconda is a free **Python** distribution, which includes the most popular **Python** packages needed in this book for data analysis. That is, it is almost ready to be used without much downloading. Nevertheless, Anaconda has a simple way of adding packages that are not included. I will let you know when a package is needed. RStudio is also a coding environment that makes the R experience easier and more versatile. However, RStudio does not include R, so you will need to install **R** first. RStudio is also free-of-charge.

Both RStudio and Anaconda offer business options, which are not free. These options are needed to deploy large-scale applications. Not everything is free in the data analytics business, but the free versions are enough for all applications you need for academic and professional work.

However, they both have a way to help you install and use external programs called *packages/modules/libraries* (I will use them interchangeably). These are very useful when you need to carry out complicated tasks that need some large code, as it is likely that someone has created a package that does what you need. You do not need to know every package available, but progressively you will become familiar with packages that will save you lots of coding time.

I use RStudio because it gives me a nice interface to install any package I need. You can use the command `install.packages()` but I prefer using the RStudio window for that, as shown in Figure 1.1.

Anaconda has a window to help you install packages, as shown in Figure 1.2.

RStudio might not interact with you during installation, but Anaconda may request your permission after the four steps in Figure 1.2. In general, another window will pop up telling you what other changes will be done to proceed with the installation. While RStudio will show you "almost" every available package, Anaconda may miss several packages in the menu

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

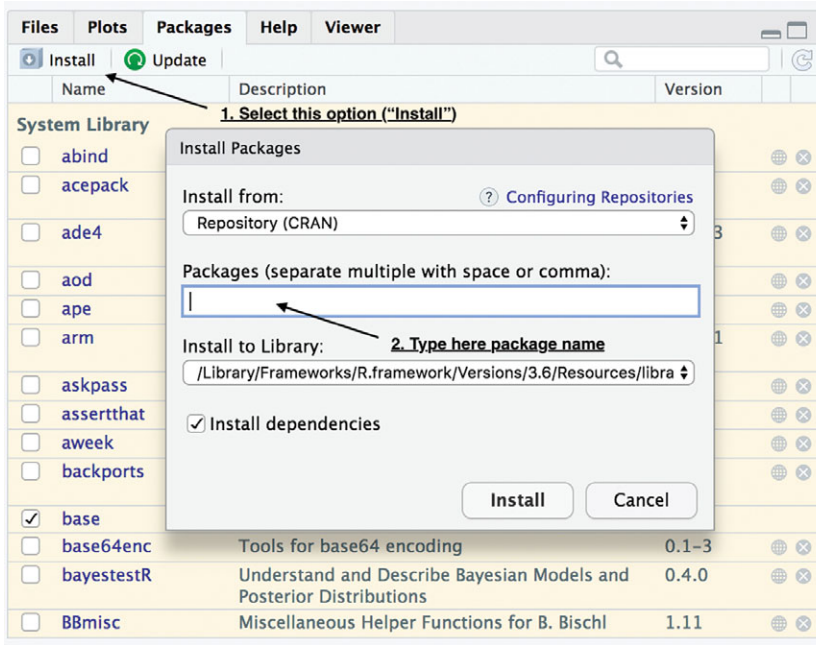

Figure 1.1 RStudio menu for installing packages

shown in Figure 1.2, in that situation you should first request *the terminal*, as shown in Figure 1.3.

Notice that in Figure 1.2 you can have the option to have "Environments". There, I have created an environment where I will install all the libraries needed for this book. In Figure 1.3 I am using my "bookVisualDS" environment, and I decided to "Open Terminal" because I need to install a library not available in Anaconda. In Figure 1.4 you can see the terminal, where you can type the commands needed to install other libraries (pip install, in this case). Notice that the installation will be done in the "bookVisualDS" environment. I highly recommend you create an environment for this book, and, if you do so, request Python 3.7 or above. After the terminal window appears, you can install a package using the command `pip install` plus the name of the package to install. I show you that in Figure 1.4.

The command *pip* is a familiar alternative, but there are more options for installing, like *conda install*. Every package recommends a particular procedure for this, and you should follow that advice.

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

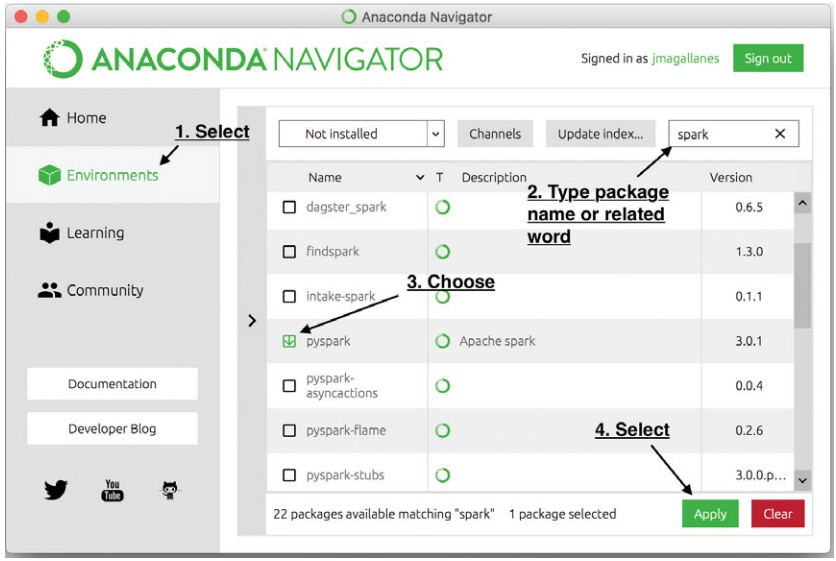

Figure 1.2 Anaconda menu for installing packages

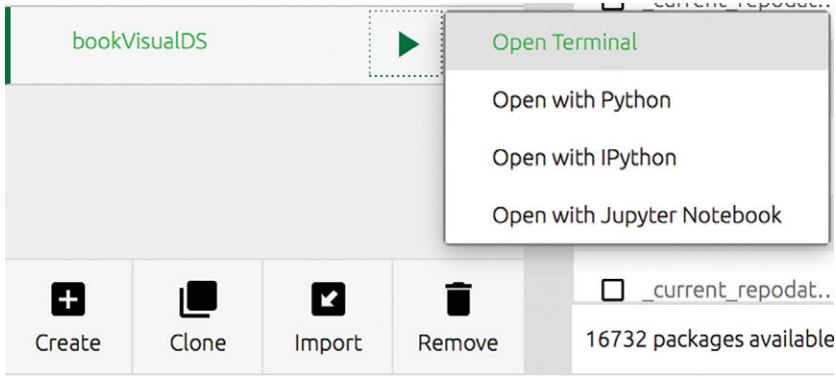

Figure 1.3 Calling the terminal from Anaconda
The call is done from an environment named `bookVisualDS`.

1.3 The Rest of the Book

This book is organized into three parts. The first part includes two chapters. The first, chapter one, is a review on data nature, and how the identification

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

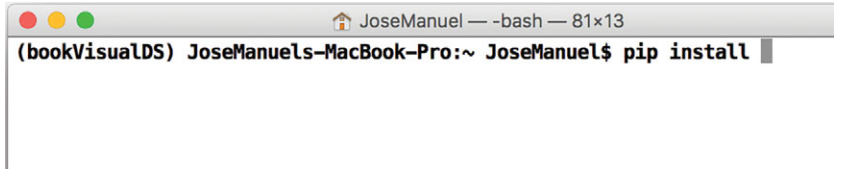

Figure 1.4 Using pip for installing a package
The installation will be done only at the environment that made the call.

of data type and structure is an important previous step for producing visuals; the second chapter deals with basic concepts you should master when making plots. I recommend you pay close attention to this chapter as it mentions elements you should include in your plots, elements I may not include in every plot in the book. The second part of the book deals with data organized in tables, the most well-known structure for people familiar with spreadsheets. I have organized this part into three chapters, each one dealing with plots for one, two, or three or more variables. In each case, you will see options for numerical and categorical data. The last part will offer two chapters to briefly cover data that are not exactly tables, such as maps, and networks; the chapter on networks, the last one, will include some basic plots for text data coming from Twitter.

1.4 How to Read This Book

In this book, each section depends on something previously said; so, I recommend you read it from beginning to end because in each chapter I am assuming you read the previous chapter. Therefore, I may not explain why I do something if I have explained it before. Of course, if you know most of the material covered, you are most welcome to visit the chapter or section you believe you need, while skipping previous material. I tried to include references to previous sections as needed.

This book has been conceived for students, professors and professionals of social and policy sciences at all levels. It can be used for self-learning, and to complement any quantitative analysis course.

All the book codes and data are stored in repositories in GitHub, which I will make available upon request to my email address: `jmagallanes@pucp.edu.pe` or `magajm@uw.edu`.

https://doi.org/10.1017/9781108625425.001 Published online by Cambridge University Press

1.5 Acknowledgements

I am very grateful to Cambridge University Press for giving me the opportunity to share this new work, specially to my editor Lauren Cowles. I am also very grateful to the reviewers that had the time to comment and suggest improvements on my initial drafts.

This book has been possible due to the continuous support from the eScience Institute at the University of Washington (UW) and the Evans School of Public Policy from UW; and the support from my home institution in Peru, the Pontificia Universidad Catolica del Peru, particularly from the support given by the Department of Social Sciences, and the grant (Concurso Anual de Proyectos PUCP - ID 713/Codigo 2019-3-0026) which required me to share these techniques with the data analytics team. I have to express my particular thankfulness to Ed Lazowska, Bill Howe, Tyler McCormick, Bernease Herman, Micaela Parker, and Sarah Stone from eScience; and to Sandra Archibald, Allison Cullen, Craig Thomas, David Layton, Ann Bostrom, and Leigh Anderson from the Evans School. They have been very supportive during my stay at UW and motivated me finish this book in different ways. I am also thankful to the graduate students at the Evans School, who help me to better organize these contents with their feedback from my teaching. I also would like to thank my colleagues at George Mason University, Robert Axtell, William Kennedy, Annetta Burger, and especially Claudio Cioffi-Revilla, my PhD advisor who recently retired as an Emeritus Professor.

I also express my deepest thanks to my home institution, the Department of Social Sciences of the Pontificia Universidad Catolica del Peru, which has made great efforts to support my work in Peru and in the United States. I am particularly very thankful for that to Alejandro Diez, David Sulmont, Aldo Panfichi, Eduardo Dargent, Sinesio Lopez, and Catalina Romero. I would also like to send special thanks to the crew of my 'Grupo Interdisciplinario de Prospectiva para Políticas Públicas' – GI3P (Interdisciplinary Group on Foresigth for Public Policy) Chiara Zamora, Gabriela Rengifo, Airam Bello, Diana Heredia, Claudia Linares, Manuel Sigueñas, Nicolas Jacobs, Fresia Gómez, Sofia Ticliahuanca, Luis Torres, Yurfa Toralva, Alejandro Boyco, and Pavel Coronado.

This book was finished while I suffered the lost of my "viejitos" Alfredo, and Jorge. However, it was not all bad news: Diana, my wife, was finishing her masters dissertation; and Rafael, my son, was starting high school in Lima and improving his tennis skills. Seeing them flourish keeps me moving forward.