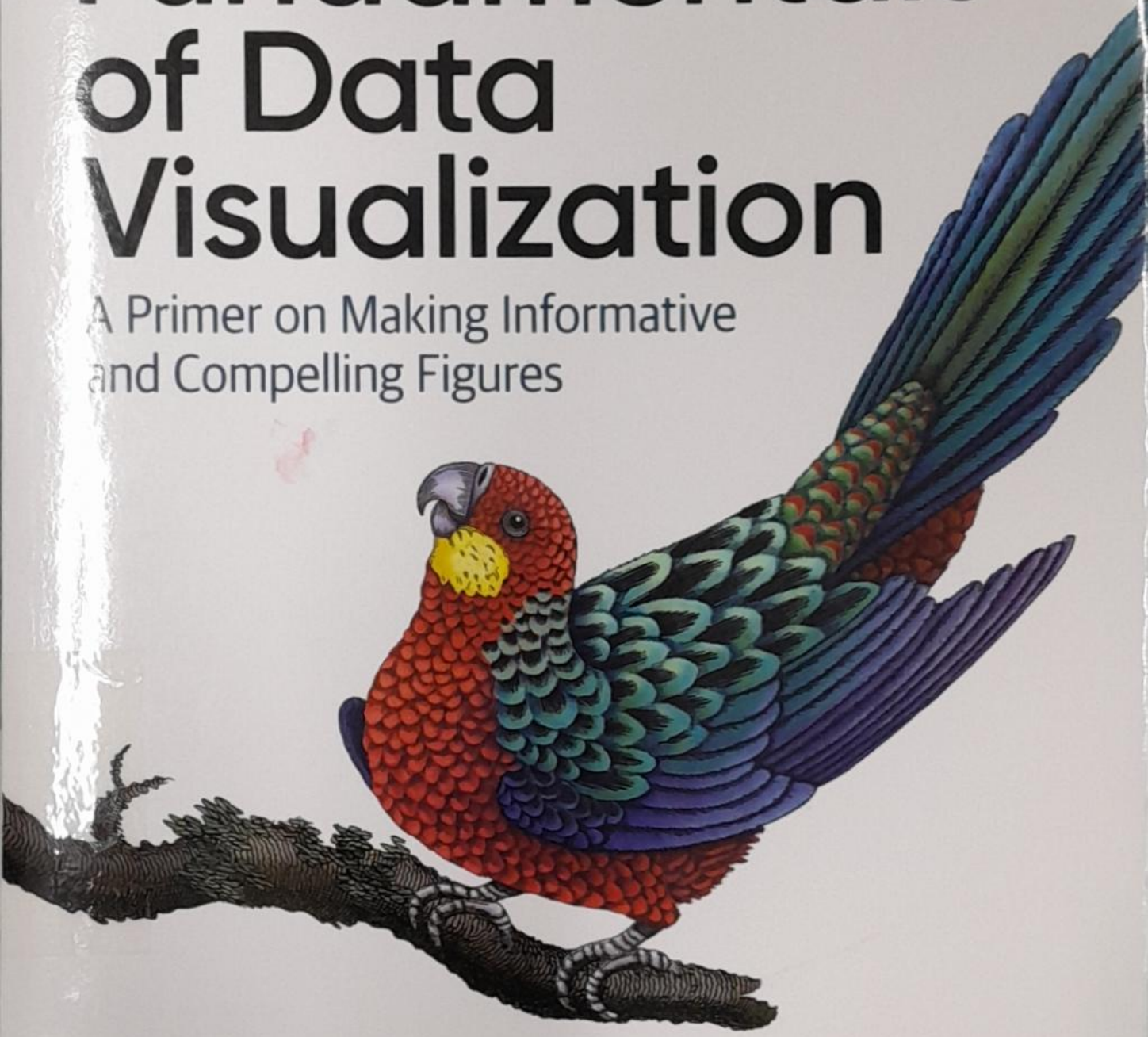
O'REILLY®

Fundamentals of Data Visualization

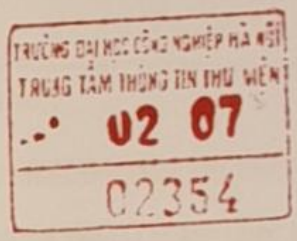
A Primer on Making Informative and Compelling Figures



Claus O. Wilke

Fundamentals of Data Visualization

A Primer on Making Informative and Compelling Figures



Claus O. Wilke

Beijing · Boston · Farnham · Sebastopol · Tokyo

O'REILLY®

Fundamentals of Data Visualization
by Claus O. Wilke

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editors: Mike Loukides and Melissa Potter
Production Editor: Kristen Brown
Copyeditor: Rachel Head
Proofreader: James Fraleigh

Indexer: Ellen Troutman-Zaig
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Claus Wilke

March 2019: First Edition

Revision History for the First Edition

2019-03-15: First Release
2019-05-03: Second Release
2019-06-07: Third Release
2020-03-27: Fourth Release

See *http://oreilly.com/catalog/errata.csp?isbn=9781492031086* for release details.

978-1-492-03108-6

[LSI]

Table of Contents

Preface

If you are a scientist, an analyst, a consultant, or anybody else who has to prepare technical documents or reports, one of the most important skills you need to have is the ability to make compelling data visualizations, generally in the form of figures. Figures will typically carry the weight of your arguments. They need to be clear, attractive, and convincing. The difference between good and bad figures can be the difference between a highly influential or an obscure paper, a grant or contract won or lost, a job interview gone well or poorly. And yet, there are surprisingly few resources to teach you how to make compelling data visualizations. Few colleges offer courses on this topic, and there are not that many books on this topic either. (Some exist, of course.) Tutorials for plotting software typically focus on how to achieve specific visual effects rather than explaining why certain choices are preferred and others not. In your day-to-day work, you are simply expected to know how to make good figures, and if you're lucky you have a patient adviser who teaches you a few tricks as you're writing your first scientific papers.

In the context of writing, experienced editors talk about "ear," the ability to hear (internally, as you read a piece of prose) whether the writing is any good. I think that when it comes to figures and other visualizations, we similarly need "eye," the ability to look at a figure and see whether it is balanced, clear, and compelling. And just as is the case with writing, the ability to see whether a figure works or not can be learned. Having eye means primarily that you are aware of a larger collection of simple rules and principles of good visualization, and that you pay attention to little details that other people might not.

In my experience, again just as in writing, you don't develop eye by reading a book over the weekend. It is a lifelong process, and concepts that are too complex or too subtle for you today may make much more sense five years from now. I can say for myself that I continue to evolve in my understanding of figure preparation. I routinely try to expose myself to new approaches, and I pay attention to the visual and design choices others make in their figures. I'm also open to changing my mind. I might today consider a given figure great, but next month I might find a reason to

criticize it. So with this in mind, please don't take anything I say as gospel. Think critically about my reasoning for certain choices and decide whether you want to adopt them or not.

While the materials in this book are presented in a logical progression, most chapters can stand on their own, and there is no need to read the book cover to cover. Feel free to skip around, to pick out a specific section that you're interested in at the moment, or one that covers a particular design choice you're pondering. In fact, I think you will get the most out of this book if you don't read it all at once, but rather read it piecemeal over longer stretches of time, try to apply just a few concepts from the book in your figuremaking, and come back to read about other concepts or reread sections on concepts you learned about a while back. You may find that the same chapter tells you different things if you reread it after a few months have passed.

Even though nearly all of the figures in this book were made with R and ggplot2, I do not see this as an R book. I am talking about general principles of figure preparation. The software used to make the figures is incidental. You can use any plotting software you want to generate the kinds of figures I'm showing here. However, ggplot2 and similar packages make many of the techniques I'm using much simpler than other plotting libraries. Importantly, because this is not an R book, I do not discuss code or programming techniques anywhere in this book. I want you to focus on the concepts and the figures, not on the code. If you are curious about how any of the figures were made, you can check out the book's source code at its GitHub repository (*https://github.com/clauswilke/dataviz*).

Thoughts on Graphing Software and Figure-Preparation Pipelines

I have over two decades of experience preparing figures for scientific publications and have made thousands of figures. If there has been one constant over these two decades, it's been the change in figure preparation pipelines. Every few years, a new plotting library is developed or a new paradigm arises, and large groups of scientists switch over to the hot new toolkit. I have made figures using gnuplot, Xfig, Mathematica, Matlab, matplotlib in Python, base R, ggplot2 in R, and possibly others I can't currently remember. My current preferred approach is ggplot2 in R, but I don't expect that I'll continue using it until I retire.

This constant change in software platforms is one of the key reasons why this book is not a programming book and why I have left out all code examples. I want this book to be useful to you regardless of which software you use, and I want it to remain valuable even once everybody has moved on from ggplot2 and is using the next new thing. I realize that this choice may be frustrating to some ggplot2 users who would like to know how I made a given figure. However, anybody who is curious about my

coding techniques can read the source code of the book. It is available. Also, in the future I may release a supplementary document focused just on the code.

One thing I have learned over the years is that automation is your friend. I think figures should be autogenerated as part of the data analysis pipeline (which should also be automated), and they should come out of the pipeline ready to be sent to the printer, with no manual post-processing needed. I see a lot of trainees autogenerate rough drafts of their figures, which they then import into Illustrator for sprucing up. There are several reasons why this is a bad idea. First, the moment you manually edit a figure, your final figure becomes irreproducible. A third party cannot generate the exact same figure you did. While this may not matter much if all you did was change the font of the axis labels, the lines are blurry, and it's easy to cross over into territory where things are less clear-cut. As an example, let's say you want to manually replace cryptic labels with more readable ones. A third party may not be able to verify that the label replacement was appropriate. Second, if you add a lot of manual post-processing to your figure-preparation pipeline, then you will be more reluctant to make any changes or redo your work. Thus, you may ignore reasonable requests for change made by collaborators or colleagues, or you may be tempted to reuse an old figure even though you've actually regenerated all the data. Third, you may yourself forget what exactly you did to prepare a given figure, or you may not be able to generate a future figure on new data that exactly visually matches your earlier figure. These are not made-up examples. I've seen all of them play out with real people and real publications.

For all these reasons, interactive plot programs are a bad idea. They inherently force you to manually prepare your figures. In fact, it's probably better to autogenerate a figure draft and spruce it up in Illustrator than to make the entire figure by hand in some interactive plot program. Please be aware that Excel is an interactive plot program as well and is not recommended for figure preparation (or data analysis).

One critical component in a book on data visualization is the feasibility of the proposed visualizations. It's nice to invent some elegant new type of visualization, but if nobody can easily generate figures using this visualization then there isn't much use to it. For example, when Tufte first proposed sparklines nobody had an easy way of making them. While we need visionaries who move the world forward by pushing the envelope of what's possible, I envision this book to be practical and directly applicable to working data scientists preparing figures for their publications. Therefore, the visualizations I propose in the subsequent chapters can be generated with a few lines of R code via ggplot2 and readily available extension packages. In fact, nearly every figure in this book, with the exception of a few figures in Chapters 26, 27, and 28, was autogenerated exactly as shown.