Chapter 29. Telling a Story and Making a Point

Most data visualization is done for the purpose of communication. We have an insight about a dataset, and we have a potential audience, and we would like to convey our insight to our audience. To communicate our insight successfully, we will have to present the audience with a meaningful and exciting story. The need for a story may seem disturbing to scientists and engineers, who may equate it with making things up, putting a spin on things, or overselling results. However, this perspective misses the important role that stories play in reasoning and memory. We get excited when we hear a good story, and we get bored when the story is bad or when there is none. Moreover, any communication creates a story in the audience's minds. If we don't provide a clear story ourselves, then our audience will make one up. In the best-case scenario, the story they make up is reasonably close to our own view of the material presented. However, it can be and often is much worse. The made-up story could be "this is boring," "the author is wrong," or "the author is incompetent."

Your goal in telling a story should be to use facts and logical reasoning to get your audience interested and excited. Let me tell you a story about the theoretical physicist Stephen Hawking. He was diagnosed with motor neuron disease at age 21—one year into his PhD—and was given two years to live. Hawking did not accept this predicament and started pouring all his energy into doing science. He ended up living to be 76, became one of the most influential physicists of his time, and did all of his seminal work while being severely disabled. I'd argue that this is a compelling story. It's also entirely fact-based and true.

What Is a Story?

Before we can discuss strategies for turning visualizations into stories, we need to understand what a story actually is. A story is a set of observations, facts, or events, true or invented, that are presented in a specific order such that they create an emotional reaction in the audience. The emotional reaction is created through the buildup of tension at the beginning of the story followed by some type of resolution toward the end of the story. We refer to the flow from tension to resolution as the *story arc*, and every good story has a clear, identifiable arc.

Experienced writers know that there are standard patterns for storytelling that resonate with how humans think. For example, we can tell a story using the Opening–Challenge–Action–Resolution format. In fact, this is the format I used for the Hawking story. I opened the story by introducing the topic, the physicist Stephen Hawking. Next I presented the challenge, the diagnosis of motor neuron disease at age 21. Then came the action, his fierce dedication to science. Finally I presented the resolution, that Hawking led a long and successful life and ended up becoming one of the most influential physicists of his time. Other story formats are also commonly used. Newspaper articles frequently follow the Lead–Development–Resolution format, or, even shorter, just Lead–Development, where the lead gives away the main point up front and the subsequent material provides further details. If we wanted to tell the Hawking story in this format, we might start out with a sentence such as "The influential physicist Stephen Hawking, who revolutionized our understanding of black holes and of cosmology, outlived his doctors' prognosis by 53 years and did all of his most influential work while being severely disabled." This is the lead. In the development, we could follow up with a more in-depth description of Hawking's life, illness, and devotion to science. Yet another format is Action–Background–Development–Climax–Ending, which develops the story a little more rapidly than Opening–Challenge–Action–Resolution but not as rapidly as Lead–Development. In this format, we might open with a sentence such as "The young Stephen Hawking, facing a debilitating disability and the prospect of an early death, decided to pour all his efforts into his science, determined to make his mark while he still could." The purpose of this format is to draw in the audience and to create an emotional connection early on, but without immediately giving away the final resolution.

My goal in this chapter is not to describe these standard forms of storytelling in more detail. There are excellent resources that cover this material; for scientists and analysts, I particularly recommend Joshua Schimel's book *Writing Science* [Schimel 2011]. Instead, I want to discuss how we can bring data visualizations into the story arc. Most importantly, we need to realize that a single (static) visualization will rarely tell an entire story. A visualization may illustrate the opening, the challenge, the action, or the resolution, but it is unlikely to convey all these parts of the story at once. To tell a complete story, we will usually need multiple visualizations. For example, when giving a presentation, we may first show some background or motivational material, then a figure that creates a challenge, and eventually some other figure that provides the resolution. Likewise, in a research paper, we may present a sequence of figures that jointly create a convincing story arc. It is, however, also possible to

condense an entire story arc into a single figure. Such a figure must contain a challenge and a resolution at the same time, and it is comparable to a story arc that starts with a lead.

To provide a concrete example of incorporating figures into stories, I will now tell a story on the basis of two figures. The first creates the challenge and the second serves as the resolution. The context of my story is the growth of preprints in the biological sciences (see also Chapter 13). Preprints are manuscripts in draft form that scientists share with their colleagues before formal peer review and official publication. Scientists have been sharing manuscript drafts for as long as scientific manuscripts have existed. However, in the early 1990s, with the advent of the internet, physicists realized that it was much more efficient to store and distribute manuscript drafts in a central repository. They invented the preprint server, a web server where scientists can upload, download, and search for manuscript drafts.

To provide a concrete example of incorporating figures into stories, I will now tell a story on the basis of two figures. The first creates the challenge and the second serves as the resolution. The context of my story is the growth of preprints in the biological sciences (see also Chapter 13). Preprints are manuscripts in draft form that scientists share with their colleagues before formal peer review and official publication. Scientists have been sharing manuscript drafts for as long as scientific manuscripts have existed. However, in the early 1990s, with the advent of the internet, physicists realized that it was much more efficient to store and distribute manuscript drafts in a central repository. They invented the preprint server, a web server where scientists can upload, download, and search for manuscript drafts.

The preprint server physicists developed and still use today is called arXiv.org. Shortly after it was established, arXiv.org started to branch out and become popular in related quantitative fields, including mathematics, astronomy, computer science, statistics, quantitative finance, and quantitative biology. Here, I am interested in the preprint submissions to the quantitative biology (q-bio) section of arXiv.org. The number of submissions per month grew exponentially from 2007 to late 2013, but then the growth suddenly stopped (Figure 29-1). Something must have happened in late 2013 that radically changed the landscape in preprint submissions for quantitative biology. What caused this drastic change in submission growth?

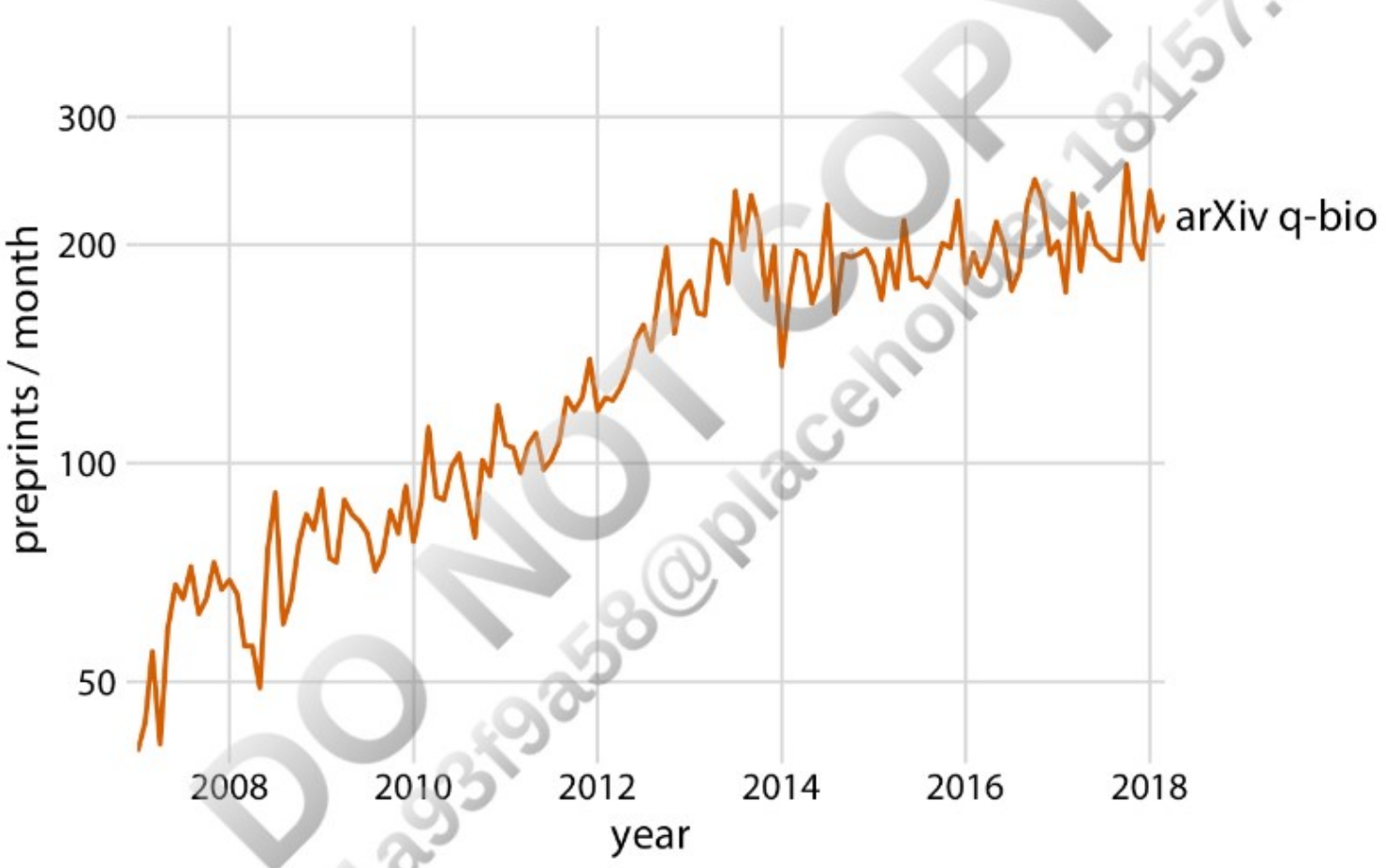


Figure 29-1. Growth in monthly submissions to the quantitative biology (q-bio) section of the preprint server arXiv.org. A sharp transition in the rate of growth can be seen around 2014. While growth was rapid up to 2014, almost no growth occurred from 2014 to 2018. Note that the y axis is logarithmic, so a linear increase in y corresponds to exponential growth in preprint submissions. Data source: Jordan Anaya, http://www.prepubmed.org/.

I will argue that late 2013 marks the point in time when preprints took off in biology, and ironically this caused the q-bio archive to slow its growth. In November 2013, the biology-specific preprint server bioRxiv was launched by Cold Spring Harbor Laboratory (CSHL) Press. CSHL Press is a publisher that is highly respected among biologists. The backing of CSHL Press helped tremendously with the acceptance of preprints in general and bioRxiv in particular among biologists. The same biologists that would have been quite suspicious of arXiv.org were much more comfortable with bioRxiv. As a result bioRxiv quickly gained acceptance among biologists, to a degree that arXiv had never managed. In fact, soon after its launch, bioRxiv started experiencing rapid, exponential growth in monthly submissions, and the slowdown in q-bio submissions exactly coincides with the start of this exponential growth of bioRxiv (Figure 29-2). It appears to be the case that many quantitative biologists who otherwise might have deposited a preprint with q-bio decided to deposit it with bioRxiv instead.

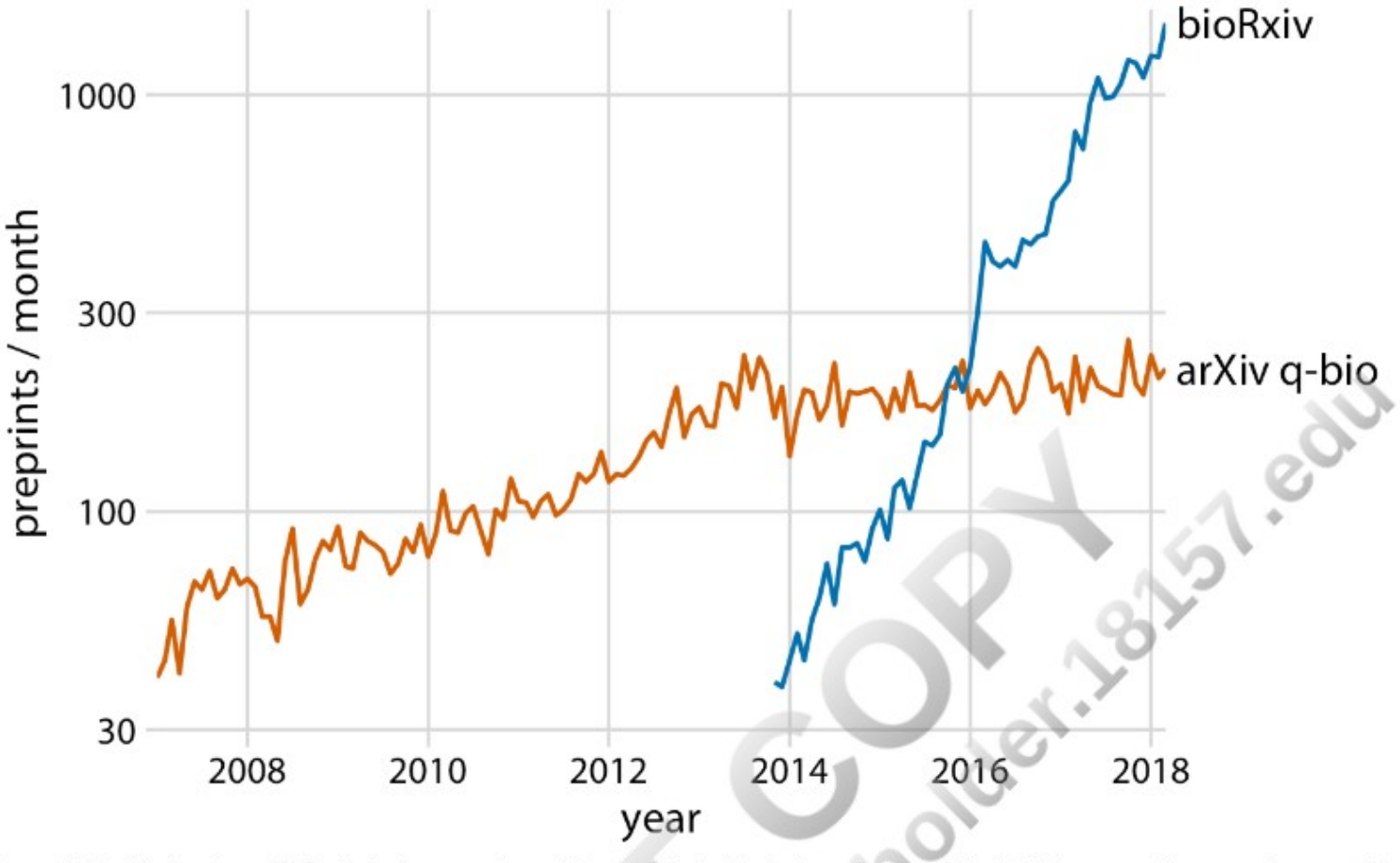


Figure 29-2. The leveling off of submission growth to q-bio coincided with the introduction of the bioRxiv server. Shown are the growth in monthly submissions to the q-bio section of the general-purpose preprint server arXiv.org and to the dedicated biology preprint server bioRxiv. The bioRxiv server went live in November 2013, and its submission rate has grown exponentially since. It seems likely that many scientists who otherwise would have submitted preprints to q-bio chose to submit to bioRxiv instead. Data source: Jordan Anaya, http://www.prepubmed.org/.

This is my story about preprints in biology. I purposefully told it with two figures, even though the first (Figure 29-1) is fully contained within the second (Figure 29-2). I think this story has the strongest impact when broken into two pieces, and this is how I would present it in a talk. However, Figure 29-2 alone can be used to tell the entire story, and the single-figure version might be more suitable to a medium where the audience can be expected to have short attention span, such as in a social media post.

Make a Figure for the Generals

For the remainder of this chapter, I will discuss strategies for making individual figures and sets of figures that help your audience to connect with your story and remain engaged throughout your entire story arc. First, and most importantly, you need to show your audience figures they can actually understand. It is entirely possible to follow all the recommendations I have provided throughout this book and still prepare figures that confuse. When this happens, you may have fallen victim to two common misconceptions: first, that the audience can see your figures and immediately infer the points you are trying to make, and second, that the audience can rapidly process complex visualizations and understand the key trends and relationships that are shown. Neither of these assumptions is true. We need to do everything we can to help our readers understand the meaning of our visualizations and see the same patterns in the data that we see. This usually means less is more. Simplify your figures as much as possible. Remove all features that are tangential to your story. Only the important points should remain. I refer to this concept as “making a figure for the generals.”

For several years, I was in charge of a large research project funded by the US Army. For our annual progress reports, I was instructed by the program managers to not include a lot of figures, and that any figure I did include should show very clearly how our project was succeeding. A general, the program managers told me, should be able to look at each figure and immediately see how what we were doing was improving upon or exceeding prior capabilities. Yet when my colleagues who were part of this project sent me figures for the annual progress report, many of the figures did not meet this criterion. The

figures usually were overly complex, were labeled in confusing, technical terms, or did not make any obvious point at all. Most scientists are not trained to make figures for the generals.

> **NOTE**
>
> Never assume your audience can rapidly process complex visual displays.

Some might hear this story and conclude that the generals are not very smart or just not that into science. I think that's exactly the wrong take-home message. The generals are simply very busy. They can't spend 30 minutes trying to decipher a cryptic figure. When they give millions of dollars of taxpayer funds to scientists to do basic research, the least they can expect in return is a handful of clear demonstrations that something worthwhile and interesting was accomplished. This story should also not be misconstrued as being about military funding in particular. The generals are a metaphor for anybody you may want to reach with your visualization: a scientific reviewer for your paper or grant proposal, a newspaper editor, or your supervisor or your supervisor's boss at the company where you're working. If you want your story to come across, you need to make figures that are appropriate for your generals.

The first thing that will get in the way of making a figure for the generals is, ironically, the ease with which modern visualization software allows us to make sophisticated data visualizations. With nearly limitless power of visualization, it becomes tempting to keep piling on more dimensions of data. And in fact, I see a trend in the world of data visualization to make the most complex, multifaceted visualizations possible. These visualizations may look very impressive, but they are unlikely to convey a meaningful story. Consider Figure 29-3, which shows the arrival delays for all flights departing out of the New York City area in 2013. I suspect it will take you a while to process this figure.

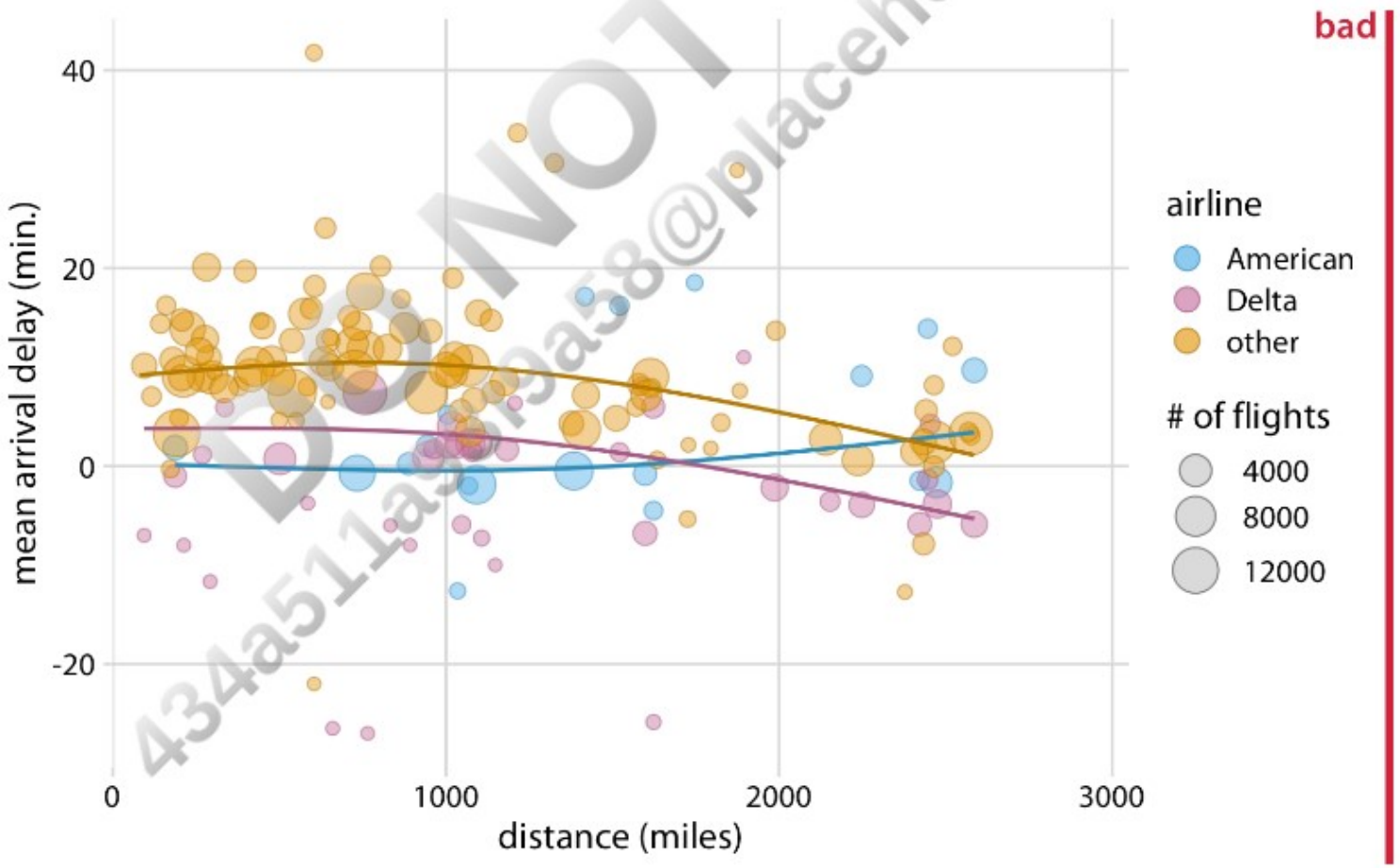


Figure 29-3. Mean arrival delay versus distance from New York City. Each point represents one destination, and the size of each point represents the number of flights from one of the three major New York City airports (Newark, JFK, or LaGuardia) to that destination in 2013. Negative delays imply that the flight arrived early. Solid lines represent the mean trends between arrival delay and distance. Delta has consistently lower arrival delays than other airlines, regardless of distance traveled. American has among the lowest delays, on average, for short distances, but has among the highest delays for longer distances traveled. This figure is labeled as "bad" because it is overly complex. Most readers will find it confusing and will not intuitively grasp what it is the figure is showing. Data source: US Dept. of Transportation.

Most readers will find it confusing and will not intuitively grasp what it is the figure is showing. Data source: US Dept. of Transportation, Bureau of Transportation Statistics.

I think the most important feature of Figure 29-3 is that American and Delta have the shortest arrival delays. This insight is much better conveyed in a simple bar graph (Figure 29-4). Therefore, Figure 29-4 is the correct figure to show if the story is about arrival delays of airlines, even if making that graph doesn't challenge your data visualization skills. And if you're then wondering whether these airlines have small delays because they don't fly that much out of the New York City area, you could present a second bar graph highlighting that both American and Delta are major carriers in this area (Figure 29-5). Both of these bar graphs discard the distance variable shown in Figure 29-3. This is OK. We don't need to visualize data dimensions that are tangential to our story, even if we have them and even if we could make a figure that showed them. Simple and clear is better than complex and confusing. When you're trying to show too much data at once, you may end up not showing anything.

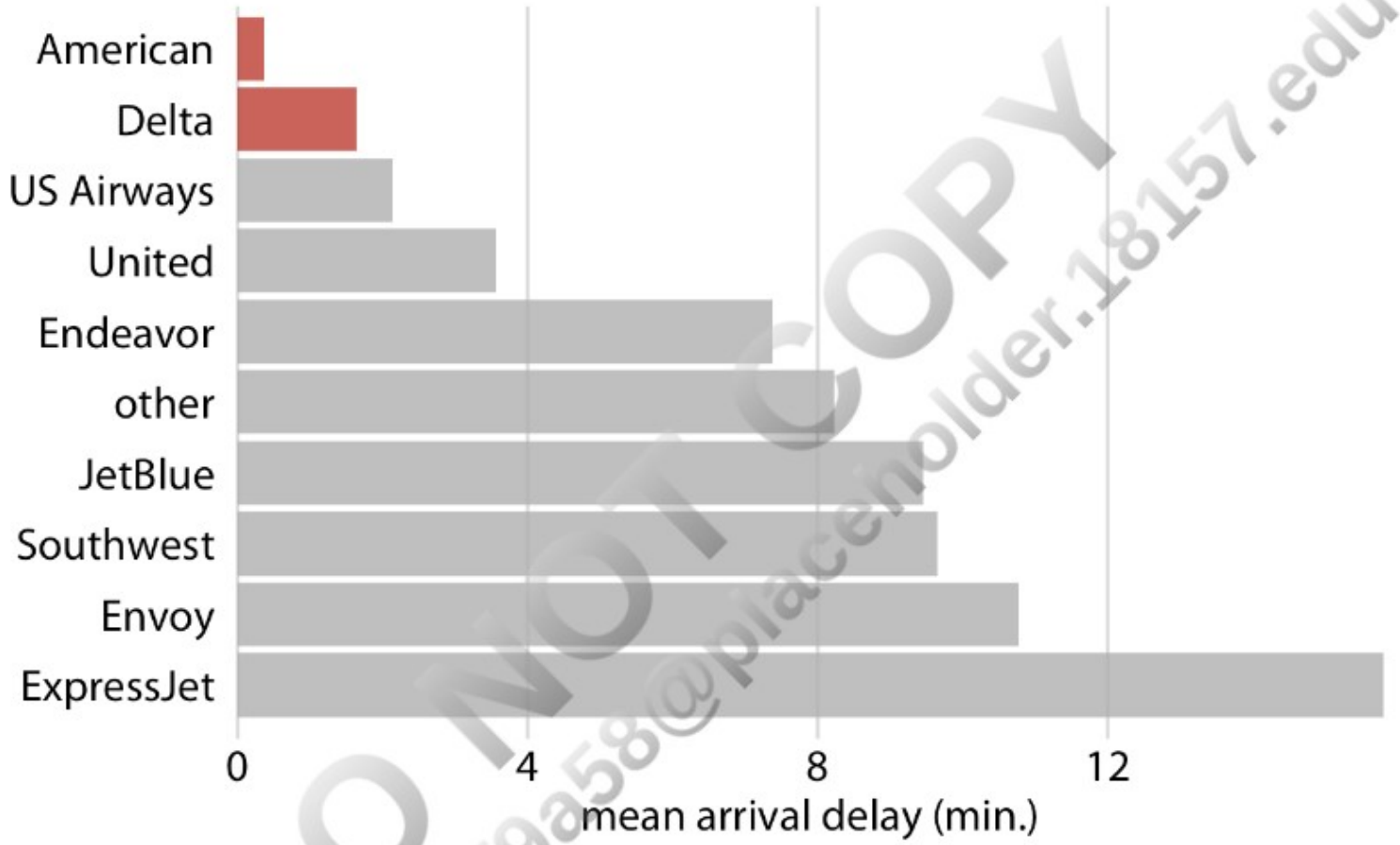
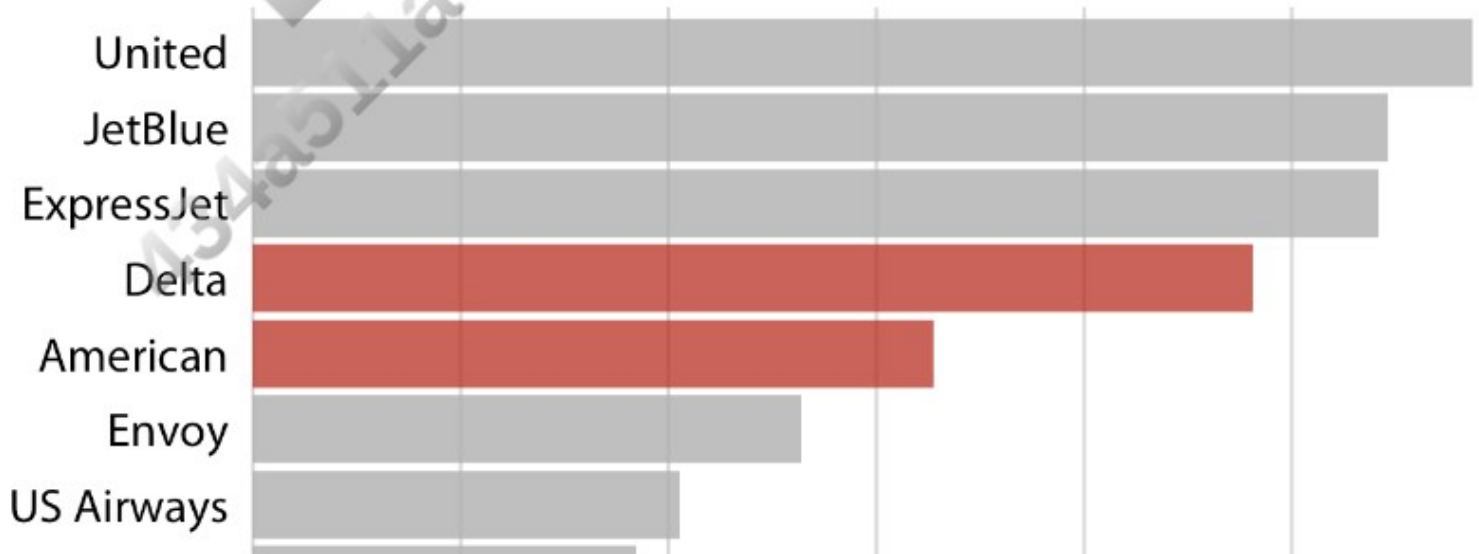


Figure 29-4. Mean arrival delay for flights out of the New York City area in 2013, by airline. American and Delta have the lowest mean arrival delays of all airlines flying out of the New York City area. Data source: US Dept. of Transportation, Bureau of Transportation Statistics.



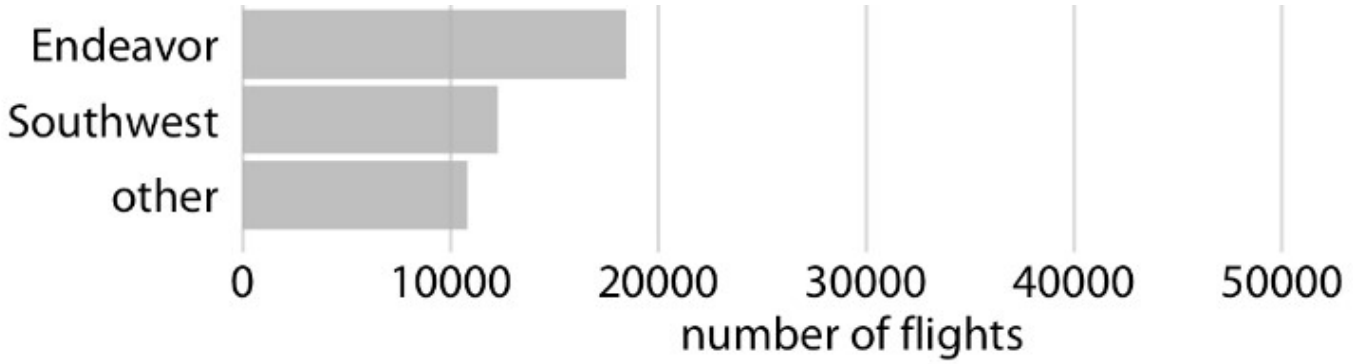
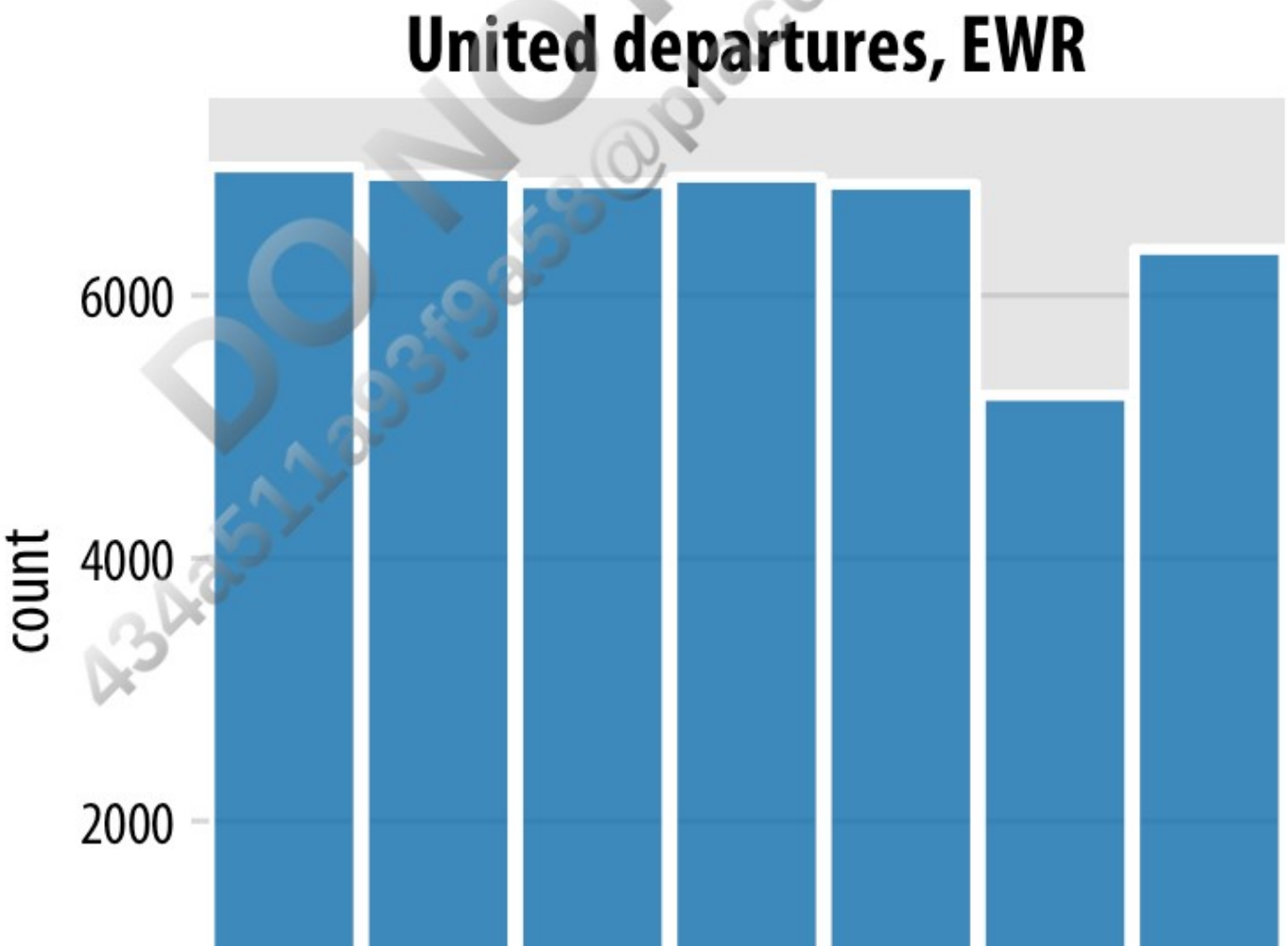


Figure 29-5. Number of flights out of the New York City area in 2013, by airline. Delta and American are the fourth and fifth largest carriers by flights out of the New York City area. Data source: US Dept. of Transportation, Bureau of Transportation Statistics.

Build Up Toward Complex Figures

Sometimes, however, we do want to show more complex figures that contain a large amount of information at once. In those cases, we can make things easier for our readers if we first show them a simplified version of the figure before we show the final one in its full complexity. The same approach is also strongly recommended for presentations. Never jump straight to a highly complex figure; first show an easily digestible subset of the information.

This recommendation is particularly relevant if the final figure is a small multiples plot (Chapter 21) showing a grid of subplots with similar structure. The full grid is much easier to digest if the audience has first seen a single subplot by itself. For example, Figure 29-6 shows the aggregate numbers of United Airlines departures out of Newark Airport (EWR) in 2013, broken down by weekday. Once we have seen and digested this figure, it's much easier to process the same information for 10 airlines and 3 airports at once (Figure 29-7).



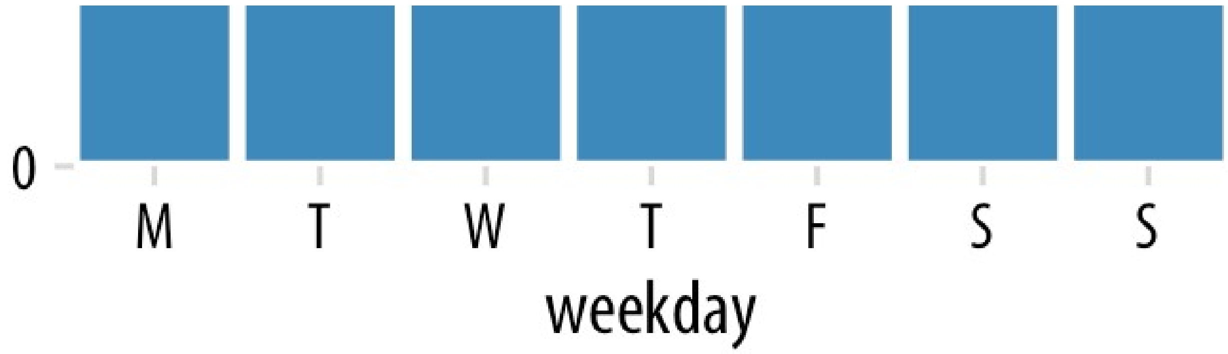


Figure 29-6. United Airlines departures out of Newark Airport (EWR) in 2013, by weekday. Most weekdays show approximately the same number of departures, but there are fewer departures on weekends. Data source: US Dept. of Transportation, Bureau of Transportation Statistics.

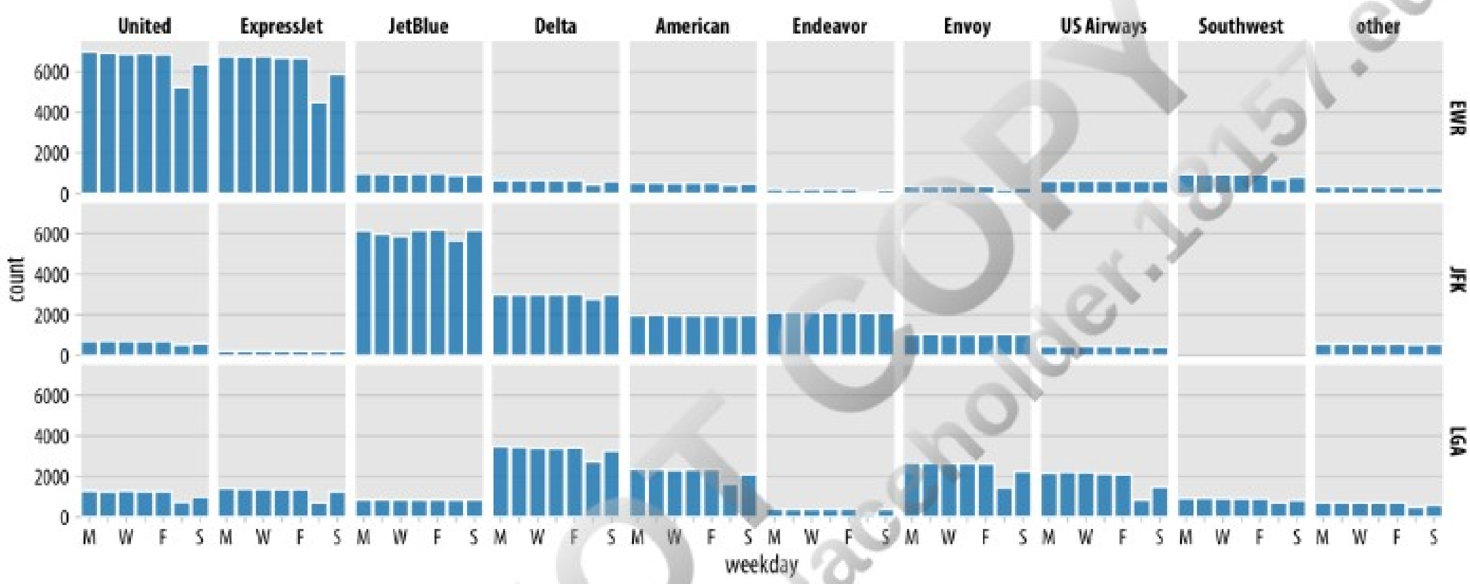
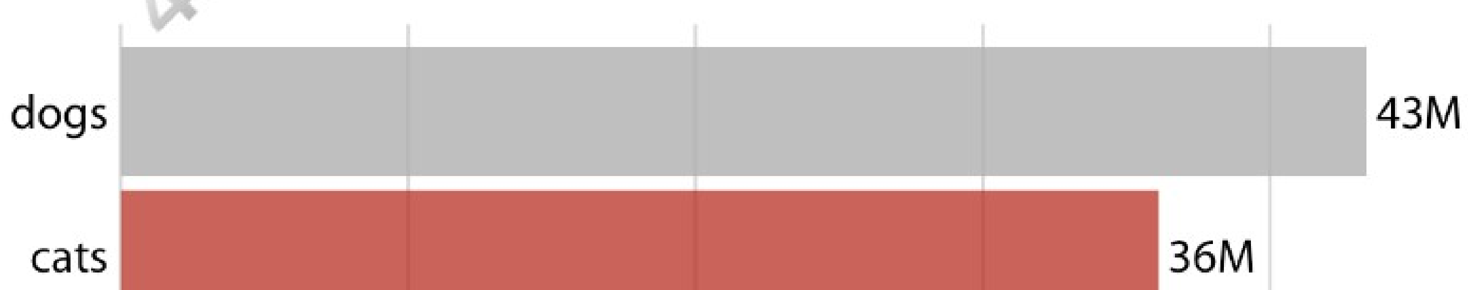


Figure 29-7. Departures out of airports in the New York City area in 2013, broken down by airline, airport, and weekday. United Airlines and ExpressJet make up most of the departures out of Newark Airport (EWR); JetBlue, Delta, American, and Endeavor make up most of the departures out of JFK; and Delta, American, Envoy, and US Airways make up most of the departures out of LaGuardia (LGA). Most but not all airlines have fewer departures on weekends than during the workweek. Data source: US Dept. of Transportation, Bureau of Transportation Statistics.

Make Your Figures Memorable

Simple and clean figures such as simple bar plots have the advantage that they avoid distractions, are easy to read, and let your audience focus on the most important points you want to bring across. However, the simplicity can come with a disadvantage: figures can end up looking generic. They don't have any features that stand out and make them memorable. If I showed you 10 bar graphs in quick succession you'd have a hard time keeping them apart and afterwards remembering what you saw. For example, if you take a quick look at Figure 29-8, you will notice the visual similarity to Figure 29-5 from earlier in this chapter. However, the two figures have nothing in common other than that they are bar graphs. Figure 29-5 showed the number of flights out of the New York City area by airline, whereas Figure 29-8 shows the most popular pets in US households. Neither figure has any element that helps you intuitively perceive what topic the figure covers, and therefore neither figure is particularly memorable.



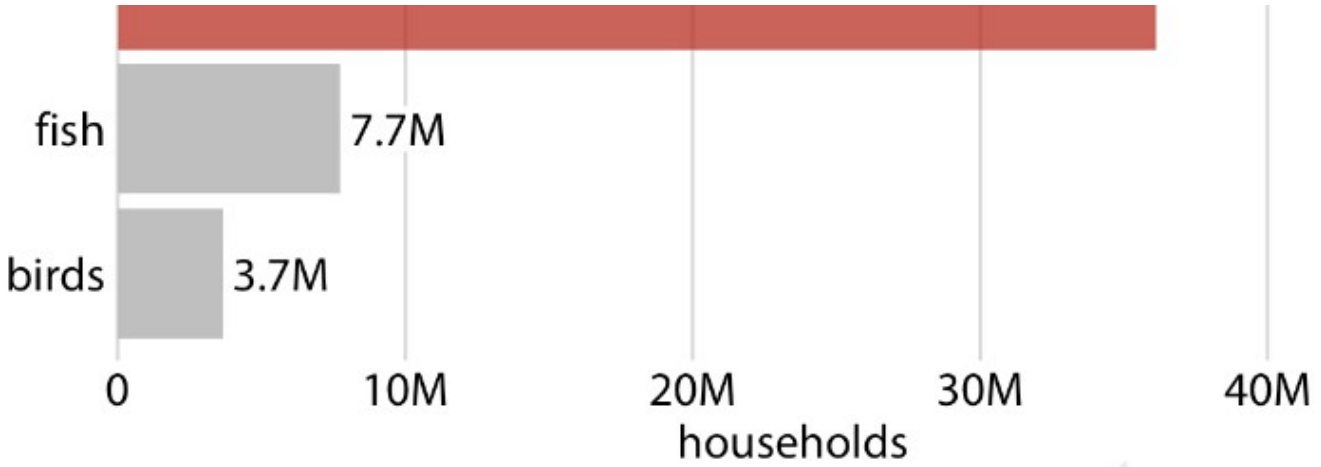


Figure 29-8. Number of households having one or more of the most popular pets: dogs, cats, fish, or birds. This bar graph is perfectly clear but not necessarily particularly memorable. The "cats" column has been highlighted solely to create visual similarity with Figure 29-5

Figure 29-5. Data source: 2012 US Pet Ownership & Demographics Sourcebook, American Veterinary Medical Association.

Research on human perception shows that more visually complex and unique figures are more memorable [Bateman et al. 2010]; [Borgo et al. 2012]. However, visual uniqueness and complexity do not just affect memorability, as they may hinder a person's ability to get a quick overview of the information or make it difficult to distinguish small differences in values. At the extreme, a figure could be very memorable but utterly confusing. Such a figure would not be a good data visualization, even if it works well as a stunning piece of art. At the other extreme, figures may be very clear but forgettable and boring, and those figures may not have the impact we might hope for either. In general, we want to strike a balance between the two extremes and make our figures both memorable and clear. (The intended audience matters as well, however. If a figure is intended for a technical scientific publication, we will generally worry less about memorability than if the figure is intended for a broadly read newspaper or blog.)

We can make a figure more memorable by adding visual elements that reflect features of the data, such as drawings or pictograms of the things or objects that the dataset is about. One approach that is commonly taken is to show the data values themselves in the form of repeated images, such that each copy of an image corresponds to a defined amount of the represented variable. For example, we can replace the bars in Figure 29-8 with repeated images of a dog, a cat, a fish, and a bird, drawn to a scale such that each complete animal corresponds to 5 million households (Figure 29-9). Thus, visually, Figure 29-9 still functions as a bar plot, but we now have added some visual complexity that makes the figure more memorable, and we have also shown the data using images that directly reflect what the data means. After only a quick glance at the figure, you may be able to remember that there were many more dogs and cats than fish or birds. Importantly, in such visualizations, we want to use the images to represent the data, rather than using images simply to adorn the visualization or to annotate the axes. In psychological experiments, the latter choices tend to be distracting rather than helpful [Haroz, Kosara, and Franconeri 2015].

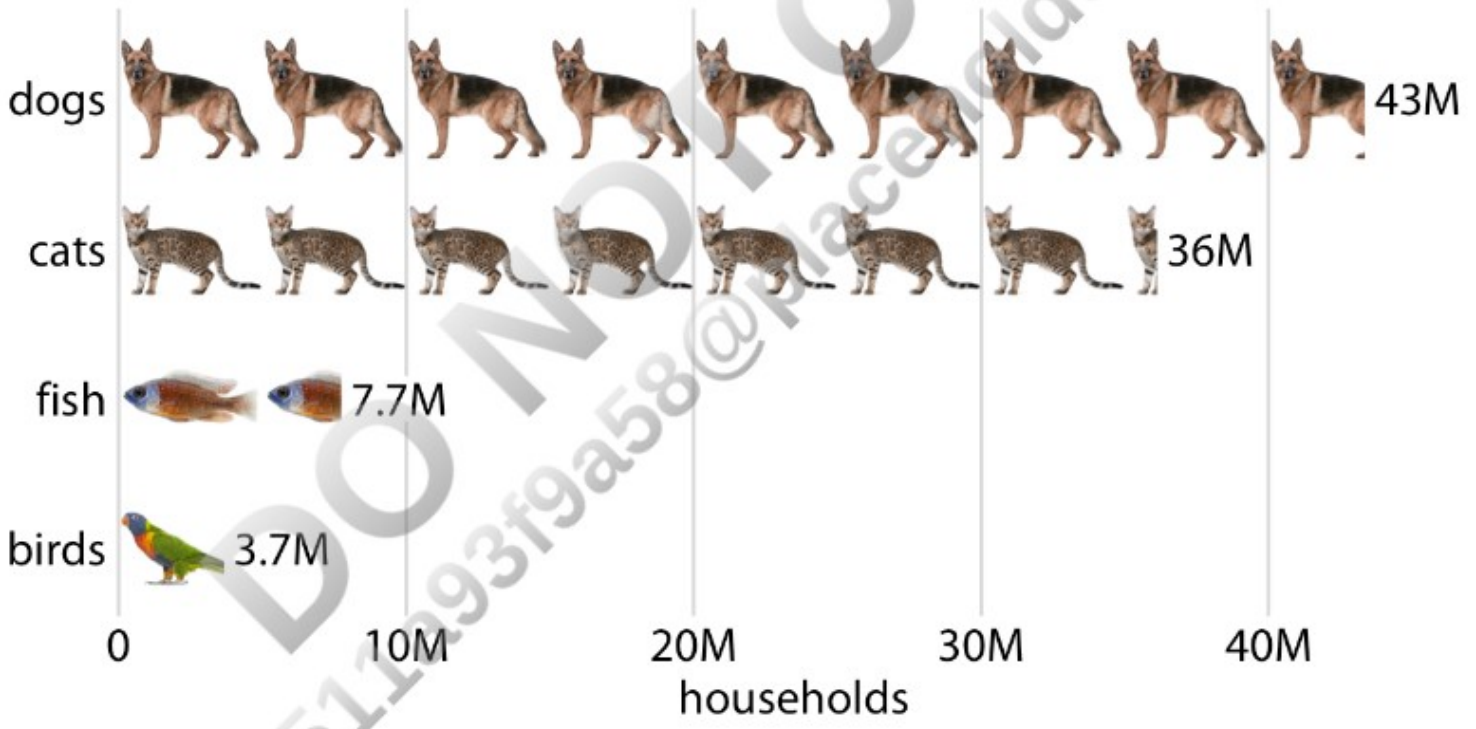


Figure 29-9. Number of households having one or more of the most popular pets, shown as an isotype plot. Each complete animal represents 5 million households that have that kind of pet. Data source: 2012 US Pet Ownership & Demographics Sourcebook, American Veterinary Medical Association.

Visualizations such as Figure 29-9 are often called *isotype plots*. The word "isotype" was introduced as an acronym of International System Of TYpographic Picture Education, and strictly speaking it refers to logo-like simplified pictograms that represent objects, animals, plants, or people [Haroz, Kosara, and Franconeri 2015]. However, I think it makes sense to use the term isotype plot more broadly to apply to any type of visualization where repeated copies of the same image are used to indicate the magnitude of a value. After all, the prefix "iso" means "the same" and "type" can mean a particular kind, class, or group.

Be Consistent but Don't Be Repetitive

Be Consistent but Don't Be Repetitive

When discussing compound figures in Chapter 21, I mentioned that it is important to use a consistent visual language for the different parts of a larger figure. The same is true across figures. If we make three figures that are all part of one larger story, then we need to design those figures so they look like they belong together. Using a consistent visual language does not mean, however, that everything should look exactly the same. On the contrary, it is important that figures describing different analyses look visually distinct, so that your audience can easily recognize where one analysis ends and another one starts. This is best achieved by using different visualization approaches for different parts of the overarching story. If you have used a bar plot already, next use a scatterplot, or a boxplot, or a line plot. Otherwise, the different analyses will blur together in your audience's mind, and your audience will have a hard time distinguishing one part of the story from another. For example, if we redesign Figure 21-8 from "Compound Figures" so it uses only bar plots, the result is noticeably less distinct and more confusing (Figure 29-10).

> **TIP**
>
> When preparing a presentation or report, aim to use a different type of visualization for each distinct analysis.

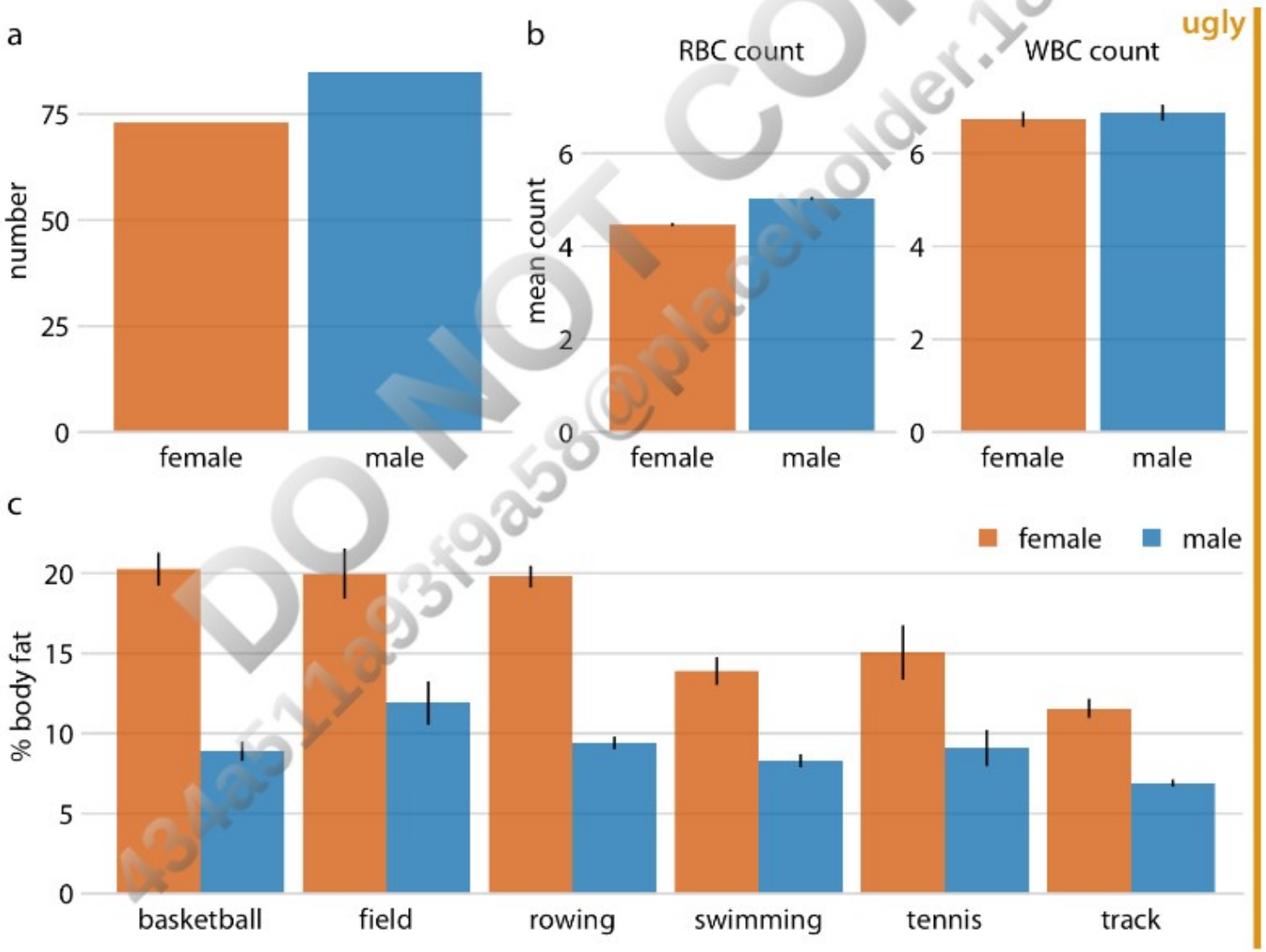


Figure 29-10. Physiology and body composition of male and female athletes. Error bars indicate the standard error of the mean. This figure is overly repetitive. It shows the same data as Figure 21-8 and it uses a consistent visual language, but all the subfigures use the same type of visualization (bar plots). This makes it difficult for the reader to process that parts (a), (b), and (c) show entirely different results. Data source: [Telford and Cunningham 1991].

Sets of repetitive figures are often a consequence of multipart stories where each part is based on the same type of raw data. In

those scenarios, it can be tempting to use the same type of visualization for each part. However, in aggregate, these figures will not hold the audience's attention. As an example, let's consider a story about the price of Facebook stock, in two parts: (i) the Facebook stock price has increased rapidly from 2012 to 2017, and (ii) the price increase has outpaced that of other large tech companies. You might want to visualize these two statements with two figures showing stock price over time, as demonstrated in Figure 29-11. However, while Figure 29-11a serves a purpose and should remain as is, Figure 29-11b is at the same time repetitive and obscures the main point. We don't particularly care about the exact temporal evolution of the stock price of Alphabet, Apple, or Microsoft; we just want to highlight that it grew less than the stock price of Facebook.

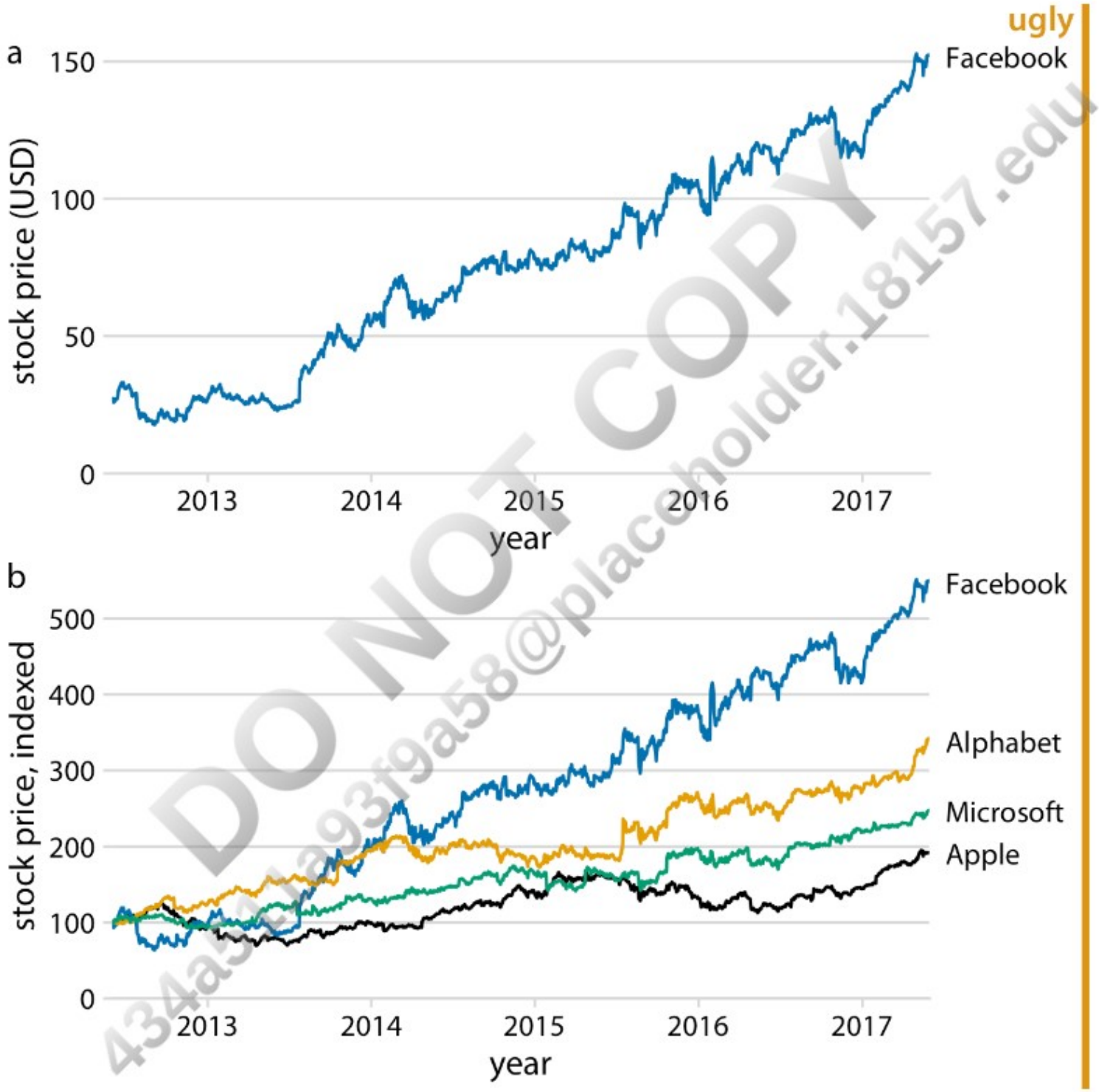


Figure 29-11. Growth of Facebook stock price over a five-year interval and comparison with other tech stocks. (a) The Facebook stock price rose from around $25/share in mid-2012 to $150/share in mid-2017. (b) The prices of other large tech companies did not rise comparably over the same time period. Prices have been indexed to 100 on June 1, 2012 to allow for easy comparison. This figure is labeled as "ugly" because parts (a) and (b) are repetitive. Data source: Yahoo! Finance.

I would recommend to leave part (a) as is but replace part (b) with a bar plot showing percent increase (Figure 29-12). Now we have two distinct figures that each make a unique point and that work well in combination. Part (a) allows the reader to get

familiar with the raw, underlying data and part (b) highlights the magnitude of the effect while removing any tangential information.

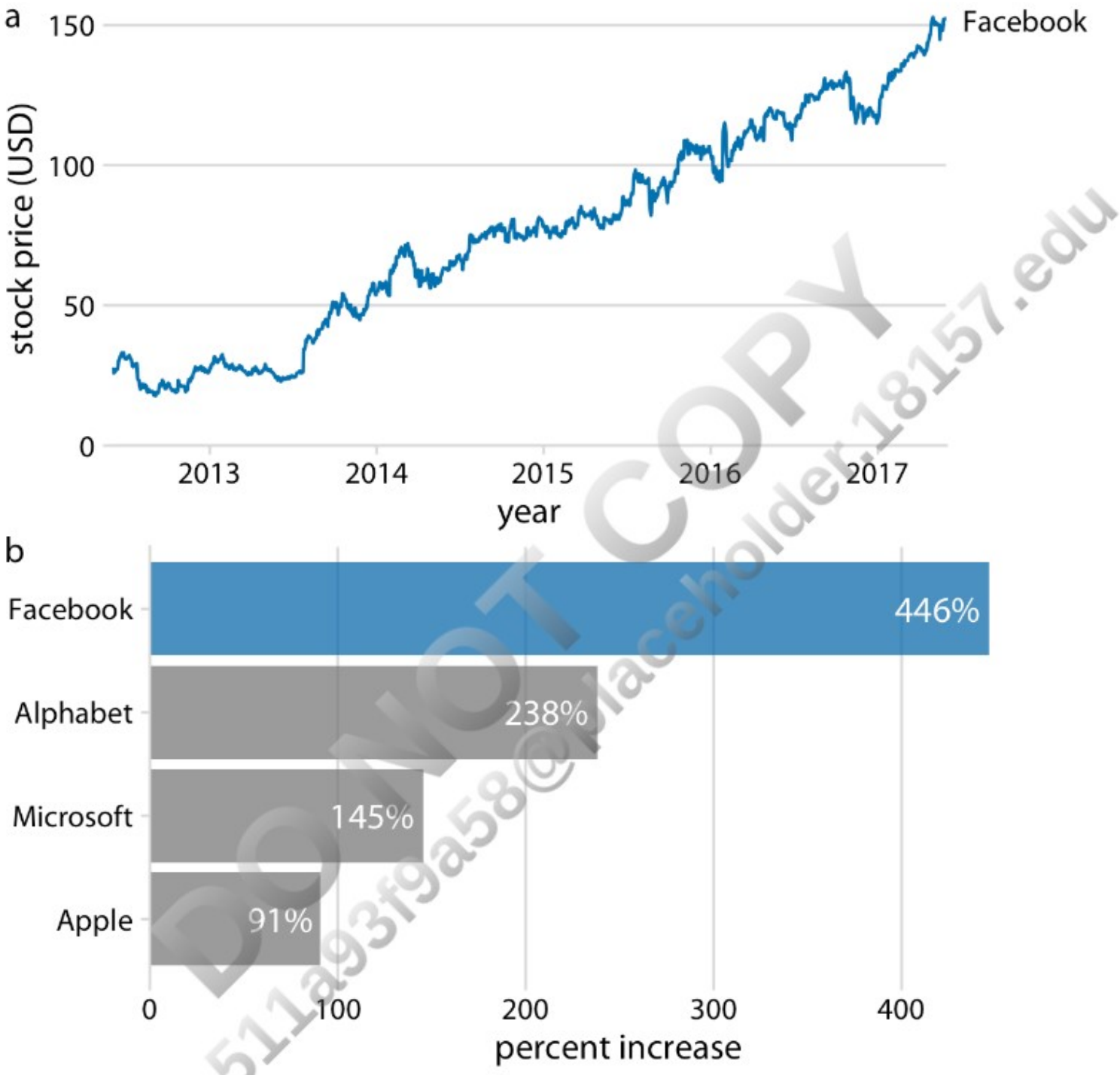


Figure 29-12. Growth of Facebook stock price over a five-year interval and comparison with other tech stocks. (a) The Facebook stock price rose from around $25/share in mid-2012 to $150/share in mid-2017, an increase of almost 450%. (b) The prices of other large tech companies did not rise comparably over the same time period. Price increases ranged from around 90% to almost 240%. Data source: Yahoo! Finance.

Figure 29-12 highlights a general principle that I follow when preparing sets of figures to tell a story: I start with a figure that is as close as possible to showing the raw data, and in subsequent figures I show increasingly more derived quantities. Derived quantities (such as percent increases, averages, coefficients of fitted models, and so on) are useful to summarize key trends in large and complex datasets. However, because they are derived they are less intuitive, and if we show a derived quantity before we have shown the raw data, our audience will find it difficult to follow. On the flip side, if we try to show all trends by showing raw data, we will end up needing too many figures and/or being repetitive.

How many figures should you use to tell your story? The answer depends on the publication venue. For a short blog post or

How many figures should you use to tell your story? The answer depends on the publication venue. For a short blog post or tweet, make one figure. For scientific papers, I recommend between three and six figures. If there are many more than six figures for a scientific paper, then some of them may need to be moved into an appendix or supplementary materials section. It is good to document all the evidence we have collected, but we must not wear out our audience by presenting excessive numbers of mostly similar-looking figures. In other contexts, a larger number of figures may be appropriate. However, in those contexts, we will usually be telling multiple stories, or an overarching story with subplots. For example, if I am asked to give an hour-long scientific presentation, I usually aim to tell three distinct stories. Similarly, a book or thesis will contain more than one story, and in fact may contain one story per chapter or section. In those scenarios, each distinct story line or subplot should be presented with no more than three to six figures. In this book, you will find that I follow this principle at the level of sections within chapters. Each section is approximately self-contained and will typically show no more than six figures.

Annotated Bibliography

No single book can cover everything there is to know about a topic. I encourage you to read other texts on data visualization to deepen your understanding and to develop your technical skills in making figures. Here, I provide a limited selection of books that I have personally found interesting, thought-provoking, or helpful. Books listed in the first section are the most similar in scope to the present book, and may provide complementary or alternative perspectives on the topics I have covered. Books listed in "Programming Books" address the important topic of how to make visualizations using programming approaches and available software libraries. The remaining sections list other books that will expand your knowledge of data visualization and help you communicate with visuals and data.

Thinking About Data and Visualization

The following books discuss the thought processes and decision making required for turning data into visualizations. They serve as introductory texts on how to choose what visualizations to make and what pitfalls to look out for:

Alberto Cairo. The Truthful Art. New Riders, 2016.

An excellent all-around introduction to data visualization, in particular for journalists. The book covers many important concepts of data visualization, such as how to visualize distributions, trends, uncertainty, and maps. In many chapters, it also serves as an introduction to basic statistical principles, explaining concepts such as populations, samples, and confidence levels.

Stephen Few. Show Me the Numbers. Analytics Press, 2012.

A book about data visualization for the business professional. It is similar in scope and target audience to the following reference but contains more material and covers many topics in more depth. However, it is not as well written or carefully produced as the following book.

Cole Nussbaumer Knaflic. Storytelling with Data. John Wiley & Sons, 2015.

A well-written and carefully produced book on how to turn data into visuals. The book's primary audience is people making business graphics, and it's an excellent reference for the topics it covers. However, it does not cover many topics of importance to scientists, such as the visualization of distributions, trends, or uncertainty.

Programming Books

The following references are all how-to books that teach programming approaches to data visualization:

Kieran Healy. Data Visualization: A Practical Introduction. Princeton University Press, 2018.

An introduction to using ggplot2 for data visualization. Recommended as follow-up after Wickham and Grolemund's *R for Data Science* (mentioned later in this list).

Scott Murray. Interactive Data Visualization for the Web: An Introduction to Designing with D3. 2nd ed. O'Reilly Media, 2017.

An introduction to making interactive online visualizations with D3, using HTML, CSS, JavaScript, and SVG.

Jake VanderPlas. Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media, 2016.

An introduction to using the programming language Python for data science. Has extensive material on data visualization using Python's Matplotlib and Seaborn.

Hadley Wickham, Garrett Grolemund. R for Data Science. O'Reilly Media, 2017.

An all-around introduction to using the programming language R for data science. Contains several chapters on using ggplot2 for data visualization.

Statistics Texts

Introductory texts in statistics will generally contain material on data visualization, covering topics such as scatterplots, histograms, boxplots, and line graphs. There are many such texts that could be listed. Here, I mention just a few recent additions that are worth a look:

David M. Diez, Christopher D. Barr, Mine Çetinkaya-Rundel. OpenIntro Statistics. 3rd ed. OpenIntro, Inc., 2015.

An open source introductory statistics text book. The entire book is freely available, as are the LaTeX files and R code used to compile the book and make the figures.

Susan Holmes, Wolfgang Huber. Modern Statistics for Modern Biology. Cambridge University Press, 2018.

A statistics text that emphasizes computational tools needed for modern biology. The entire book is freely available, and R code for all examples is provided.

Chester Ismay, Albert Y. Kim. Modern Dive—An Introduction to Statistical and Data Sciences via R. https://moderndive.com.

An online-only introductory textbook that teaches basic statistics and data science. The book covers both theoretical concepts and practical approaches using R.

Historical Texts

The books in this section are of interest primarily for historical reasons. They were influential at the time of their publication, but similar material can now be found elsewhere or in more modern form:

William S. Cleveland. The Elements of Graphing Data. 2nd ed. Hobart Press, 1994.

One of the first books about information design written for statisticians. The book contains many examples of scatterplots, line graphs, histograms, and boxplots, and it discusses them in the context of data analysis and statistical modeling. It also popularized the Cleveland dot plot.

William S. Cleveland. Visualizing Data. Hobart Press, 1993.

Companion book to *The Elements of Graphing Data* by the same author. This one is more mathematical and doesn't talk about human perception.

Edward R. Tufte. Envisioning Information. Graphics Press, 1990.

This book popularized the concept of the small multiple.

Edward R. Tufte. The Visual Display of Quantitative Information. 2nd ed. Graphics Press, 2001.

First published in 1983, this book has been highly influential in the field of data visualization. It introduced concepts such as chart junk, data–ink ratio, and sparklines. The book also showed the first slopegraph (but didn't name it). However, it does contain several recommendations that have not stood the test of time. In particular, it recommends an excessively minimalistic plot design.

Books on Broadly Related Topics

The following books are all broadly related to the topics of data visualization and effective communication:

Joshua Schimel. Writing Science. Oxford University Press, 2011.

Teaches how to write about scientific and other technical topics in an engaging way, by telling a story. While not primarily a book about data visualization, this is an indispensable text for anybody who needs to write technical articles and/or

proposals.

Jonathan Schwabish. Better Presentations. Columbia University Press, 2016.

A short and informative guide for making presentations. A must-read for anybody who routinely uses slides to give talks or presentations.

Maureen C. Stone. A Field Guide to Digital Color. A K Peters, 2003.

A comprehensive guide to how colors are captured, processed, and reproduced by computers.

Colin Ware. Information Visualization. 3rd ed. Morgan Kaufmann, 2012.

A book about principles of visualization, specifically addressing topics such as how the human visual system works and how different graphical patterns are perceived. The book covers many different visualization scenarios, including user interfaces and virtual worlds, but it puts comparatively less emphasis on visualizing data in the form of 2D figures.

Technical Notes

The entire book was written in R Markdown, using the `bookdown`, `rmarkdown`, and `knitr` packages. All figures were made with ggplot2, with the help of several add-on packages including `cowplot`, `geofacet`, `ggforce`, `ggmap`, `ggrepel`, `ggridges`, `hexbin`, `patchwork`, `sf`, `statebins`, `tidybayes`, and `treemapify`. Color manipulations were done with the `colorspace` and `colorblindr` packages. For many of these packages, the current development version is required to compile all parts of the book.

The source code for the book is available at *https://github.com/clauswilke/dataviz*. The book also requires a supporting R package, `dviz.supp`, whose code is available at *https://github.com/clauswilke/dviz.supp*.

The book was last compiled using the following environment:

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Sierra 10.12.6
##
## Matrix products: default
## BLAS: /Library/Frameworks/ ... /libRblas.0.dylib
## LAPACK: /Library/Frameworks/ ... /libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/ ... /C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats    graphics  grDevices utils   datasets  methods   base
##
## other attached packages:
##  [1] nycflights13_1.0.0   gapminder_0.3.0  RColorBrewer_1.1-2
##  [4] gganimate_1.0.0.9000 ungeviz_0.1.0    emmeans_1.3.1
##  [7] mgcv_1.8-24          nlme_3.1-137     broom_0.5.1
## [10] tidybayes_1.0.3      maps_3.3.0       statebins_2.0.0
## [13] sf_0.7-1             maptools_0.9-4   sp_1.3-1
## [16] rgeos_0.3-28         ggspatial_1.0.3  geofacet_0.1.9
## [19] plot3D_1.1.1         magick_1.9       hexbin_1.27.2
## [22] treemapify_2.5.0     gridExtra_2.3    ggmap_2.7.904
## [25] ggthemes_4.0.1       ggridges_0.5.1   ggrepel_0.8.0
## [28] ggforce_0.1.1        patchwork_0.0.1  lubridate_1.7.4
## [31] forcats_0.3.0        stringr_1.3.1    purrr_0.2.5
## [34] readr_1.1.1          tidyr_0.8.2      tibble_1.4.2
## [37] tidyverse_1.2.1      dviz.supp_0.1.0  dplyr_0.8.0.9000
## [40] colorblindr_0.1.0    ggplot2_3.1.0    colorspace_1.4-0
## [43] cowplot_0.9.99
##
## loaded via a namespace (and not attached):
##  [1] rjson_0.2.20            deldir_0.1-15
##  [3] class_7.3-14            rprojroot_1.3-2
##  [5] estimability_1.3        ggstance_0.3.1
##  [7] rstudioapi_0.7          farver_1.0.0.9999
##  [9] ggfittext_0.6.0         svUnit_0.7-12
## [11] mvtnorm_1.0-8           xml2_1.2.0
## [13] knitr_1.20              polyclip_1.9-1
## [15] jsonlite_1.5            png_0.1-7
## [17] compiler_3.5.0          httr_1.3.1
## [19] backports_1.1.2         assertthat_0.2.0
## [21] Matrix_1.2-14           lazyeval_0.2.1
## [23] cli_1.0.1.9000          tweenr_1.0.1
## [25] prettyunits_1.0.2       htmltools_0.3.6
## [27] tools_3.5.0             misc3d_0.8-4
## [29] coda_0.19-2             gtable_0.2.0
## [31] glue_1.3.0              Rcpp_1.0.0
## [33] cellranger_1.1.0        imguR_1.0.3
## [35] xfun_0.3                strapgod_0.0.0.9000
## [37] rvest_0.3.2             MASS_7.3-50
## [39] scales_1.0.0            hms_0.4.2
## [41] yaml_2.2.0              stringi_1.2.4
## [43] e1071_1.7-0             spData_0.2.9.4
## [45] RgoogleMaps_1.4.3       rlang_0.3.0.1
## [47] pkgconfig_2.0.2         bitops_1.0-6
## [49] geogrid_0.1.1           evaluate_0.11
## [51] lattice_0.20-35         tidyselect_0.2.5
## [53] plyr_1.8.4              magrittr_1.5
## [55] bookdown_0.7            R6_2.3.0
```

```
## [55] bookdown_0.7              Rd_2.3.0
## [57] generics_0.0.2            DBI_1.0.0
## [59] pillar_1.3.0              haven_1.1.2
## [61] foreign_0.8-71            withr_2.1.2.9000
## [63] units_0.6-1               modelr_0.1.2
## [65] crayon_1.3.4              arrayhelpers_1.0-20160527
## [67] rmarkdown_1.10            progress_1.2.0.9000
## [69] jpeg_0.1-8                rnaturalearth_0.1.0
## [71] grid_3.5.0                readxl_1.1.0
## [73] digest_0.6.18             classInt_0.2-3
## [75] xtable_1.8-3              munsell_0.5.0
## [77] concaveman_1.0.0
```