Thomas Rahlf

Data Visualisation with R

111 Examples

Second Edition

EXTRAS ONLINE

Springer

Data Visualisation with R

Thomas Rahlf

Data Visualisation with R

111 Examples

Second Edition

Springer

Thomas Rahlf
Rheinische Friedrich-Wilhelms-Universität
Bonn
Bonn, Germany

ISBN 978-3-030-28443-5 ISBN 978-3-030-28444-2 (electronic)
https://doi.org/10.1007/978-3-030-28444-2

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface to the Second English Edition

Due to the continued interest in the book, Springer Verlag has decided to publish an English translation of the second edition. Again, many thanks to Ralf Gerstner from Springer, and also to Tracey Duffy for translating all of the additions.

In 2017, Ulrike Grömpling reviewed the English translation of the first edition in the *Journal of Statistical Software*. Inspired by the book, and following the release of the second edition, she has published an R package "prepplot" which greatly simplifies the configuration of figure regions for base R graphics. I highly recommend trying this one out.

Meanwhile, the third edition of Paul Murrell's standard work *R Graphics* has been published. When I read in his foreword that my book—among others—was one inspiration for restructuring part of his book, I felt very honored.

Bonn, Germany Thomas Rahlf

Preface to the Second German Edition

The feedback on the first edition of this book (published by Open Source Press in 2014 and now out of print), and on the English edition that has been published in the meantime, was so pleasing that I was happy to accept the offer made by Springer Verlag to publish a new, updated edition of the book. The concept of explaining complete examples and the restriction to Base Graphics have been retained. Compared to the first edition of the book, this new, updated edition contains 11 additional, new examples, hence the new subtitle "111 Examples". There are two main additions: firstly, a section on visualising network relationships has been added to the chapter on categorical data. In addition to examples of classic network diagrams, an adapted heat map, and a multiple bar chart, this section also contains a chord diagram and a riverplot. Although the chord diagram and riverplot may appear unusual at first glance, they can now be found in publications of respected scientific journals such as Science, Nature or Cell. Examples on the use of georeferenced grid formats and on cartograms have been added to the chapter on maps. Three examples for the integration of data created with R in interactive JavaScript illustrations have also been added to the book. R now provides multiple concepts and packages that can be used to create JavaScript visualisations more or less directly. Ultimately, such packages form a type of container in R. In each case, a specific syntax developed by the authors of these packages translates the scripts written in this form into the notation required for the underlying JavaScript library. This means that we have to rely on the scope of the language of the R package and on the quality and flexibility of the translation routines. I am not sure if this is the right path. In this book, I have taken a different path. In three examples in Chap. 12, the data are prepared with R such that they are integrated in existing, only slightly adapted JavaScript code. This is done using Highcharts and Mapael, two JavaScript libraries that can be used to create very aesthetic illustrations "out of the box" with minimal change effort.

The primary objective of this book is still to explain how to create presentation graphics. For the exploratory visualisation within the scope of data analysis, I refer to the book Graphical Data Analysis with R (CRC Press, 2015) by Antony Unwin. The first people I would like to thank are Agnes Herrmann and Iris Ruhmann at

Springer, who have enabled me to create this updated edition of the book. For helpful comments, suggestions, and exchange of ideas for this edition, I would also like to thank Alberto Cairo, Martin S. Fischer, Sebastian Jeworutzki, Nikola Sander, Antony Unwin, January Weiner, and Stefan Fichtel. January Weiner has kindly included a comment in his riverplot package that is helpful for example 6.4.4.

Bonn, Germany Thomas Rahlf

Preface to the English Edition

This book is a translation of the German book "Datendesign mit R" that was published 2014 by Open Source Press. Due to the encouraging strong interest in the German edition Springer Verlag offered to publish an English translation. First of all I would like to thank Ralf Gerstner from Springer for this and for his helpful suggestions for improvement, as well as Annika Brun for translating most of the text, Colin Marsh for copy editing, and Katja Diederichs for converting all scripts from German to English. Last year I benefited a lot from a communication with Antony Unwin. His book "Graphical Data Analysis with R" can be seen as complementary to my own: while this one focusses on presentation of graphics, you will benefit from his book if you are interested in exploring data graphically.

Bonn, Germany Thomas Rahlf

Preface to the German Edition

Some 20 years ago, when I reviewed a score of books on statistical graphics and graphic-based data analysis, things were completely different: there were proprietary formats and operating systems, their character sets were incompatible, and graphic and statistical software was expensive. Since the turn of the century, the situation has changed fundamentally: the Internet has come of age, open-source projects have attracted more and more followers, and a handful of enthusiasts provided version 1.0 of the free statistical programming language R. Many developers were inspired to collaborate on this project. R reached version 3 in 2013, and in addition to the basic software, more than 7000 freely available extension packs are currently available. Companies and organisations such as Google, Facebook or the CIA are using R for their data analysis. Its graphic capabilities are again and again emphasised as its strong point. Pretty much all technologies relevant for data visualisation are quickly integrated into R. Through numerous functions, detailed designs of every imaginable figure, creation of maps and much more are made possible. All it takes is to know how—and that is where this book wants to contribute.

What This Book Wants to Be—and What It Doesn't Want to Be

This book is not an introduction that systematically explains all the graphic tools R has to offer. Rather, its aim is to use 100 complete script examples to introduce the reader to the basics of designing presentation graphics, and to show how bar and column charts, population pyramids, Lorenz curves, box plots, scatter plots, time series, radial polygons, Gantt charts, heat maps, bump charts, mosaic and balloon charts, and a series of different thematic map types can be created using

R's Base Graphics System. Every example uses real data and includes step-by-step explanations of the figures and their programming. The selection is based on my personal experiences—it is likely that readers will find one or another illustration lacking, and consider some too detailed. However, a large scope should be covered. This book is aimed at:

- R experts: You can most likely skip Part I. For you, the examples are particularly useful, especially the code.
- Readers that have heard of R and maybe even tried R before and are not daunted by programming; you will profit from both parts.
- Beginners: for you, the finished graphics pictured here will be most helpful. You will see what R can do. Or, in other words: you will realise that there is such a tool as R, and that it can be used to create graphics you have wanted to create for a long time, but merely never knew how. The code will be too complicated for you, but you may be able to commission others to do your graphics programming in R.

Windows, Mac, and Linux

All of the scripts and working steps will yield identical results when executed in Windows, Mac OS X or Linux. All of the examples were created in Mac OS X and then tested in Ubuntu 12.04 and an evaluation copy of Windows 8.1.

Acknowledgements

The following people deserve my thanks for hints, comments, feedback, data, discussions or help: Gregor Aisch, Insa Bechert, Evelyn Brislinger, Giuseppe Casalicchio, Arnulf Christl, Katja Diederichs, Günter Faes, Mira Hassan, Mark Heckmann, Daniel Hienert, Bruno Hopp, Uwe Ligges, Lorenz Matzat, Meinhard Moschner, Stefan Müller, Paul Murrell, David Phillips, Duncan Temple Lang, Martijn Tennekes, Patrick R. Schmid, Thomas Schraitle, Valentin Schröder, Torsten Steiner, Michael Terwey, Katrin Weller, Bernd Weiss, Nils Windisch, Benjamin Zapilko, and Lisa Zhang. This manuscript particularly benefited from discussions with an infographic designer and a data journalist. Stefan Fichtel looked over every figure and provided critical feedback. For selected figures, he designed his own suggestions; this has been an invaluable help. We did not always agree, and I have disregarded his advice here or there. Therefore, any remaining errors and shortcomings are mine. Björn Schwentker went to the trouble of proof reading large parts of the manuscript. I am very grateful for his valuable notes which have surely

made some parts of the text clearer and more readable. Finally, I want to thank Markus Wirtz for tackling the experiment of ultimately printing everything into a book.

On the Internet

The figures are conceived for different final output options. The format of the book implies that some details have become very small, e.g. in maps and radial column charts. Particularly for such cases, please refer to the book's website, on which all figures are available in high resolution or as vector graphics in PDF format:
http://www.datavisualisation-r.com

Bonn, Germany Thomas Rahlf

Contents

Chapter 1
Data for Everybody

The type and scope of data, our attitudes towards them, and their availability have fundamentally changed in recent years. Never before has there been more data than today. Never have they been so readily available. And never have there been greater opportunities for analysis, preparation, and presentation. So-called infographics, frequently animated and interactive, are spreading through the Internet. Genuine, standard-setting offerings are based on the work of extensive teams of experts and become research subjects themselves. In this book, all data are visualised using the free statistical software R. Getting started is easier than for many other programming languages, since R is specifically designed for data and statistics, and for their visualisation.

1.1 Data Visualisation Between Science and Journalism

Some scientists, like mathematician Stephen Wolfram, believe that the process of data analysis can largely be automated and, in that context, even speak about a democratisation of science. Others, like Google chief economist Hal Varian on the other hand, believe that this would mean several skills had to be acquired and that these would become future central key qualifications. Over the last few years, a plethora of new websites, books, and other publications have emerged devoted to visualisation of data. Here, the priority is their narrative rather than exploratory visualisation. One of the most well known examples is the mission of Gapminder author and inventor Hans Rosling to find catchy ways to illustrate statistics on global societal developments for a wide audience. In 2012, Time Magazine counted Hans

Electronic Supplementary Material The online version of this chapter (https://link.springer.com/chapter/10.1007/978-3-030-28444-2_1) contains supplementary material, which is available to authorized users.

T. Rahlf, *Data Visualisation with R*, https://doi.org/10.1007/978-3-030-28444-2_1

Rosling among the "100 most influential people in the world". Almost-forgotten social scientists who provided didactic visualisations, above all Otto Neurath, have been rediscovered. However, it is not as if the wheel were being reinvented. Data visualisations have always and continuously played an important role in science. Imaging processes are integral parts of many medical analyses, and almost all natural sciences use figurative representations of data to visually communicate their results. In the scope of statistical methodology, some scientists had already conducted basic research on statistical graphics many years ago. Aside from the works of William S. Cleveland, Edward Tufte's book *The Visual Display of Quantitative Information* was a revolutionising breakthrough. The book was published in 1983, and the first edition already saw 16 reprints. In combination with two subsequently published works, *Envisioning Information* and *Visual Explanations*, Edward Tufte found a genuine way to defining the standard of the topic. Also in economics, there is a long tradition of data presentation. Companies have not only collected and analysed data for internal purposes for years, but have also translated them into figures. Publications specifically prepared for external use showcase impressive displays of presentation graphics in annual reports. Finally, official statistics have been successfully dedicating their efforts towards presenting data not only in tabular but also graphic form. Here, a progressive trend towards better visualisation of official data material is evident virtually from year to year on both a national and international level. The massive influx of data pushing us to analyse them has one side-effect: their new, potential availability and openness brings with it a change in attitude towards usage rights and viewing rights. Increasing demands for transparency not only affect government but also company data. Environmental and weather records, expenditure data or those from the health or education sector, state parliament elections, legislative texts, traffic data or time tables for public transport should be free and publicly available. Big Data and Open Data have led to new methods and new approaches. An innovative variant that adopted the name "Data Science" takes this to mean a combination of programming skills, mathematical and statistical knowledge, and content-specific expertise. Drew Conway visualised this combination using a Venn diagram, which also clearly shows the overlaps.[1]

Generally, data science is highly mathematical and elaborate. However, even the journalistic sector shows a strong interest in data. Data journalism, research, and visualisations primarily offered by The New York Times and The Guardian are on the rise.

Additional popularity is enjoyed by individual offerings from "information designers" such as Catherine Mulbrandon, Stephen Few, Robert Kosara, Ben Fry or Nathan Yau, who develop their own data visualisation software, found consulting businesses, offer global workshops or set up blogs with thousands of registered users. If viewed from the viewpoint of "traditional" statistical graphics, some miss the point, and some are not only considered too colourful, too playful or too busy, but also confusing or even misleading. This is where a recent discussion arose that will eventually profit both sides.

[1] http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram.

1.2 Why R?

In scientific articles, R as programming language has become common and well liked. Outside of science, however, its potential for creation of purpose-adapted figures is not widely known. This is not surprising, since it is a known fact that graphic designers or journalists find programming difficult. Of course, it would be wrong to claim that mastering R was so easy that a first appealing graphic could be created within minutes. On the other hand, it offers several advantages that could be extremely valuable for journalists or data designers:

- All graphics can be saved in vector format (PDF, EPS or SVG) and processed with well-known vector graphic programs such as Adobe Illustrator or the free Inkscape, making each graphic element individually customisable.
- In R, the colour or shape of every graphic element can be changed whichever way you like. Text, symbols, arrows or entire drawings can be added, and different diagrams combined.
- The basic shapes of the most important diagram types, such as bar charts, line or pie charts, can frequently be created with a single command for a quick first impression.
- R can also handle maps, and can therefore be used for any geo-visualisations. The necessary maps can be loaded as, e.g., files in the common shape format.
- Since graphics are entirely programmed in a script, every step can be traced, every error found, and changes easily made. This also enables quality control by third parties and disclosure of the graphic source code in the scope of maximal open-data transparency.
- R is free.
- R is open.
- R can be extended with many software modules (packages) to visualise special graphic types or to perform advanced upstream data analyses. A growing international community continuously offers new extensions.

R graphics can also be used as a basis for interactive online graphics; a JavaScript package like, e.g., D3.js can be used to bring chart elements saved as SVG to life. An alternative is the complete JavaScript package Shiny which enables users to write interactive online data applications directly in R.

1.3 The Concept of Data Design

This book follows a 100% approach: every example shows the entire design of a specific figure. The starting point is always the result, the initial questions were always: how does a specific graphic have to look, or how can existing data best be visualised? Irrespective of specific software, the first step was always a sketch

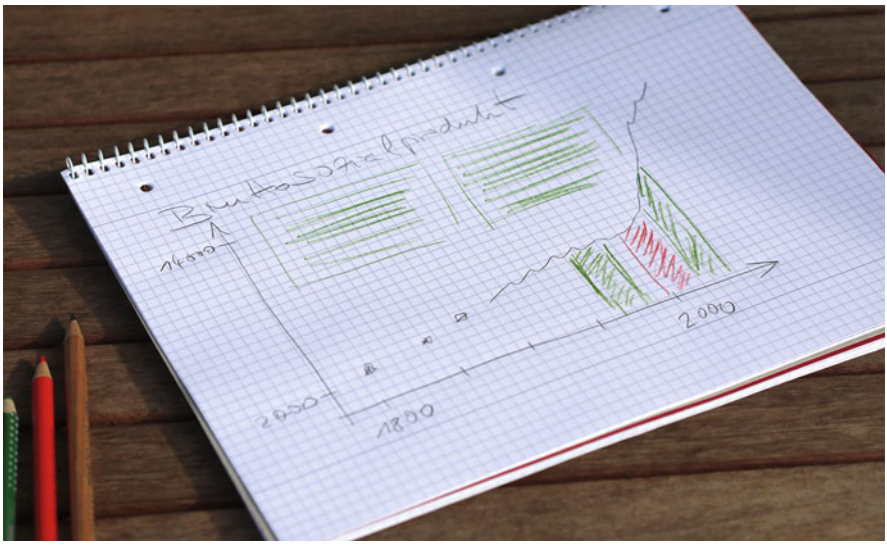

Fig. 1.1 Sketch of a picture

(Fig. 1.1). Only with the next step did the search for appropriate tools (packages and functions) and their use begin.

For the most part, the used data come from social science and official statistics, whereas some originate from business management, macroeconomics, politics, medicine, meteorology or social media. It was my goal to find suitable data for all of the selected presentation types. Obviously, this was sometimes more, sometimes less successful. However, data were not "tweaked", but used in the exact form in which they were provided. This means that scripts are sometimes a little bigger than in an ideal setting with data optimally prepared for the problem. On the other hand, this is much close to reality and may be useful when navigating some of your data pitfalls. All figures are designed as PDF files for preferably lossless and flexible further use. On average, 40 lines of code were required to create the result. It usually took a day to get from the first idea to the final product, but sometimes a week. In my opinion, the time investment is worth it, if you want to convey a message with your data.

Part I
Basics and Techniques

Chapter 2
Structure and Technical Requirements

Before we address the actual implementation of graphics in R, we will take a closer look at the structure of figures. Two examples for the different perception of graphics will be followed by a definition of graphic elements using schematic overviews that we call the "styling pattern", based on the term used in graphic design. Then follow explanations of important "ancillary elements" for figures, the used typefaces and symbols, as well as colour.

2.1 Terms and Elements

A figure can comprise one or more charts or graphs. For our purpose, the latter two terms will therefore be used synonymously. A chart consists of a data area (in R: plot region) and optional axes, axis labels, axis names, point names, legends, headings, and captions. A figure can contain several charts. In this case, every individual chart can have headings and captions, axes, legends, etc.; furthermore, there are titles and subtitles for the entire figure. If one figure contains several charts, then this is called a panel.

2.2 Illustration Grids

A figure principally comprises a title (1), a subtitle (2), a y-axis (3) including label (4) and name (5), the data area (6), a legend (7), an X-axis (8) including label (9) and name (10), and ultimately the sources (11). Figures can also contain further elements such as annotations, lines or symbols (Fig. 2.1).

T. Rahlf, *Data Visualisation with R*, https://doi.org/10.1007/978-3-030-28444-2_2

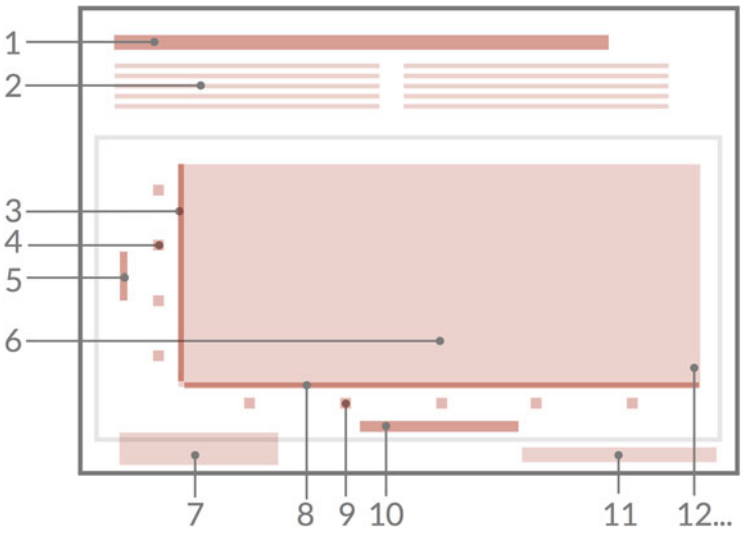

Fig. 2.1 Elements of a figure

First, consideration should be given to the aspect ratio of the figure. If, for example, both parameters of a scatter plot are percentages, and the range of values is to be represented from 0 to 100, then it would be logical to have axes of equal length, i.e. ,a square data area. In other cases, making a decision is not that simple. R gives you the opportunity to exactly specify these parameters during graphic creation (Sects. 3.3.3 and 3.3.7).

Do we need a legend? When? Where? The best-case scenario is one that can work without a legend. Generally speaking, this is the case with time series, because the labels can be written directly on the data: they are connected with lines and therefore clearly defined. However, this is not the case with scatter plots, where the meaning attributed to the colours has to be explained in a legend. In R, the legend() function lets you choose almost any setting for shape and placement of the legend.

If we include several charts in one figure, then we are creating a panel. In this case, certain elements can appear repeatedly (Fig. 2.2).

The arrangement of individual elements can vary, as can the number of charts included in one figure (Fig. 2.3).

In this book, we present examples for figures containing more than 40 graphics. R offers different ways for definition of such panels (Sect. 3.3.4 and 3.3.5).

Clearly, there cannot be universally applicable rules for the creation of an styling pattern. However, the following notes should be kept in mind:

1. It does make a difference whether graphics are free standing or embedded in body text. In the latter case, the heading is different, font sizes have to be adjusted for each element, and an explanatory subheading and explanatory labels and arrows are omitted or used more sparingly.

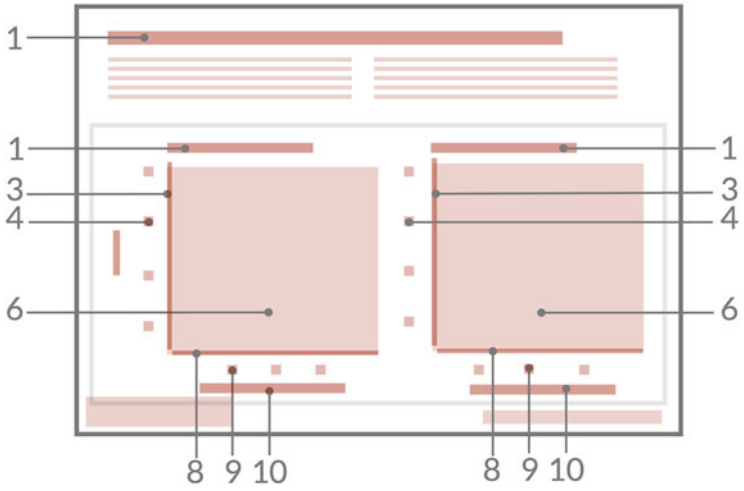

Fig. 2.2 Elements of a figure with two diagrams

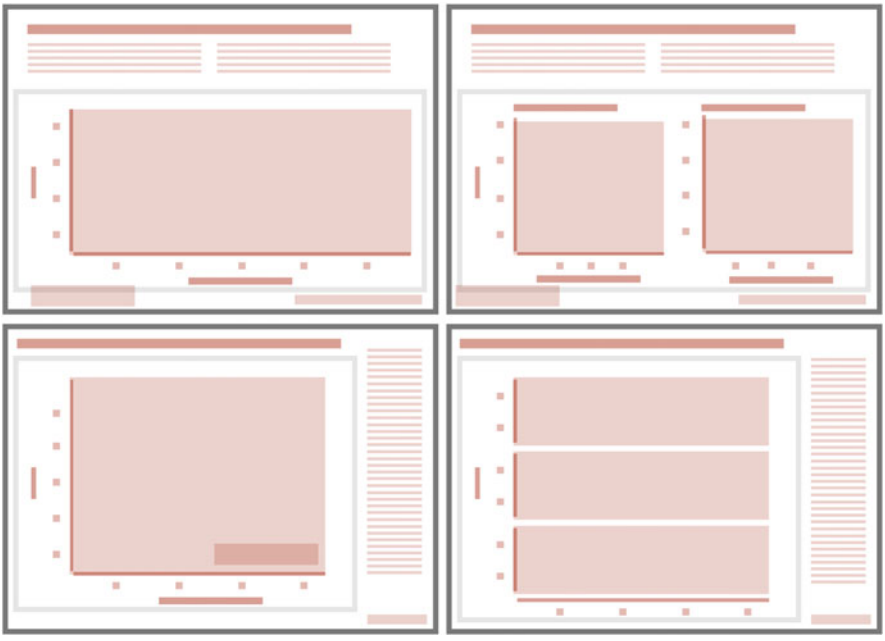

Fig. 2.3 Exemplary layout of individual elements

2. Generally, there is not just one adequate presentation of the data, but several. Whether to use stacked columns, or several column charts in one panel has to be decided for each specific case, depending on the specific data.

3. The source and title of a figure within an essay, book or website can be omitted, if they can be included where the figure is embedded.

2.3 Perception

The most important aspect when it comes to designing figures is the accurate perception of the data. This can be severely impaired by an unfortunate choice of the presentation format. Two examples: In the first example, body heights of selected celebrities are presented (Fig. 2.4)

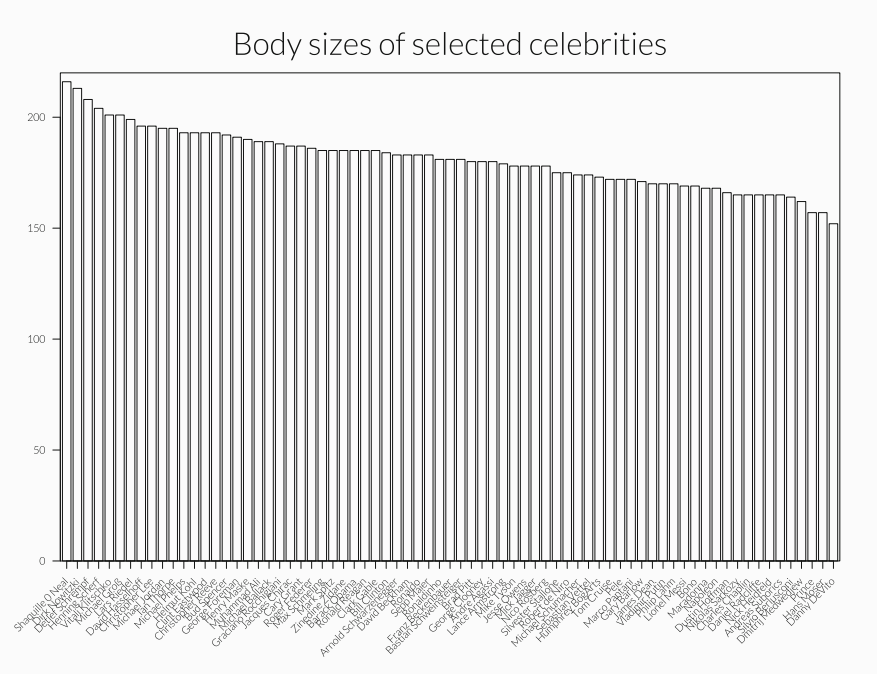

Fig. 2.4 Heights of selected celebrities

y-Axis scaling starts, as frequently requested, at zero. This gives the impression that these persons' heights are quite close to each other. This effect is even further enhanced by the use of (unfilled) bars, whose total volume takes up a large part of the entire area of the figure.

This contradicts our everyday experiences of considerable differences in body height. Searching the Internet reveals a picture showing Danny de Vito next to Christopher Reeve. Most people looking at this picture will likely think that the bar chart does not adequately reflect the differences in height. For these data, the following dot chart, repeatedly recommended by William Cleveland, is the more sensible option (Fig. 2.5).

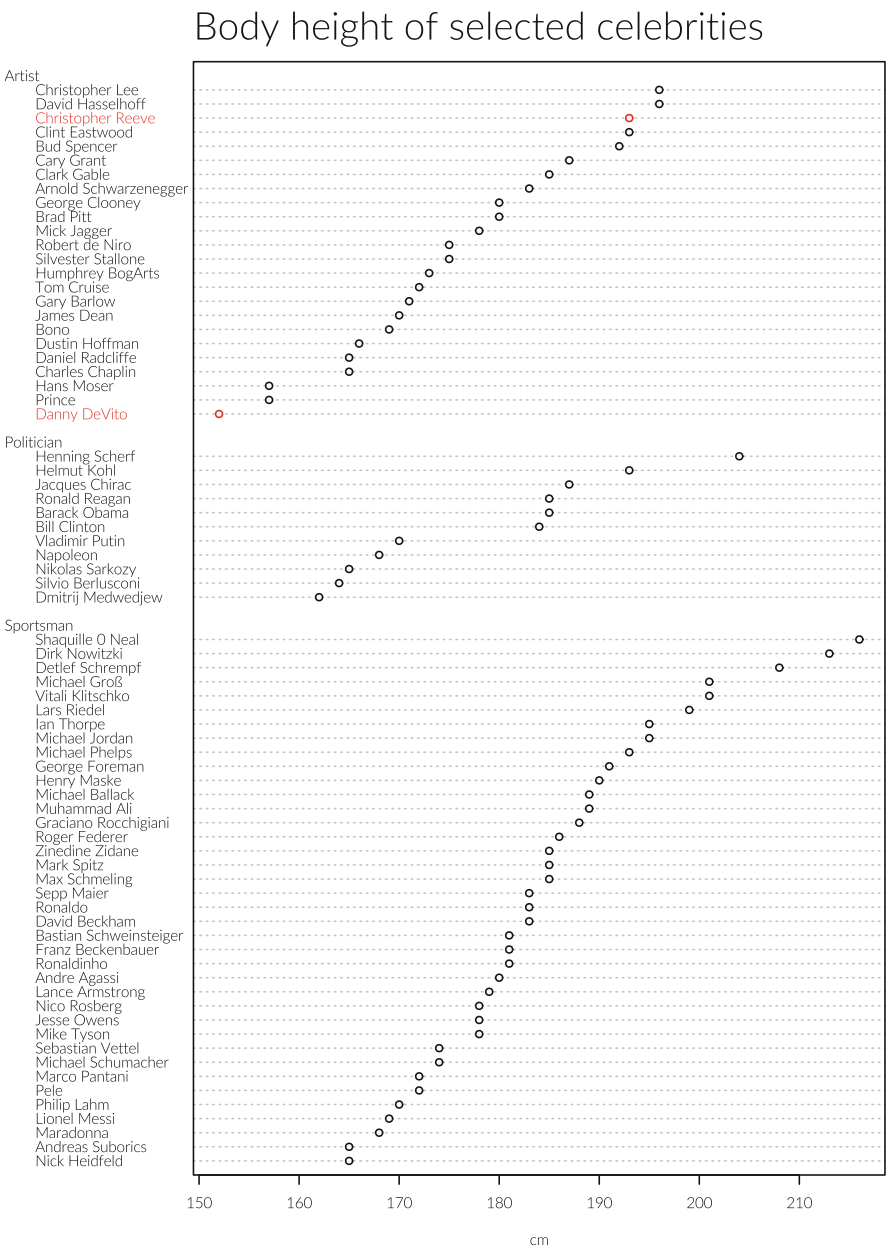

Fig. 2.5 Heights of selected celebrities as Dot chart

Four differences from the bar chart markedly improve perception:

1. Height information is depicted as dots rather than bars.
2. Grouping different "types of celebrities" offers an additional level of information and generally enhances clarity by grouping.
3. Scaling does not start at zero, but at the lowest data value.
4. Horizontal orientation makes the names easier to read.

A second example concerns time series. William S. Cleveland coined the term "banking" to describe an approach that ensures an appropriate presentation form for line charts. The main idea was that data characteristics are best perceived when data lines on average approach a 45° angle. We will illustrate this point using an example that depicts the monthly temperatures in New Jersey between 1895 and 2011 (Fig. 2.6).

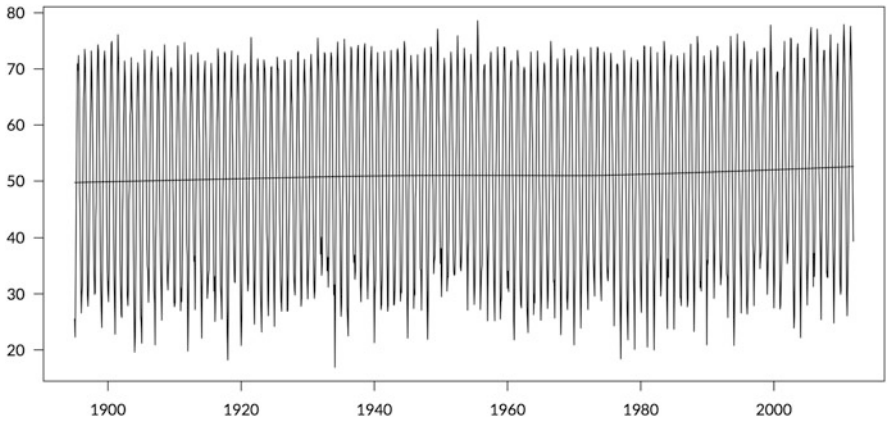

Fig. 2.6 Monthly temperatures in New Jersey between 1895 and 2011 with trend line

In this intentionally extreme example, lines are so compressed that the exact line of the actual data is practically invisible. However, the obvious if slight upward trend is easily recognisable.

If the illustration is "stretched" and made into a cut-and-stack plot, then the impression is markedly different (Fig. 2.7).

Here, the monthly temperatures' cyclical course is very easily perceived. On the other hand, no trend can be discerned from this figure. Therefore, the choice of presentation form is also dependent on the information one desires to convey.

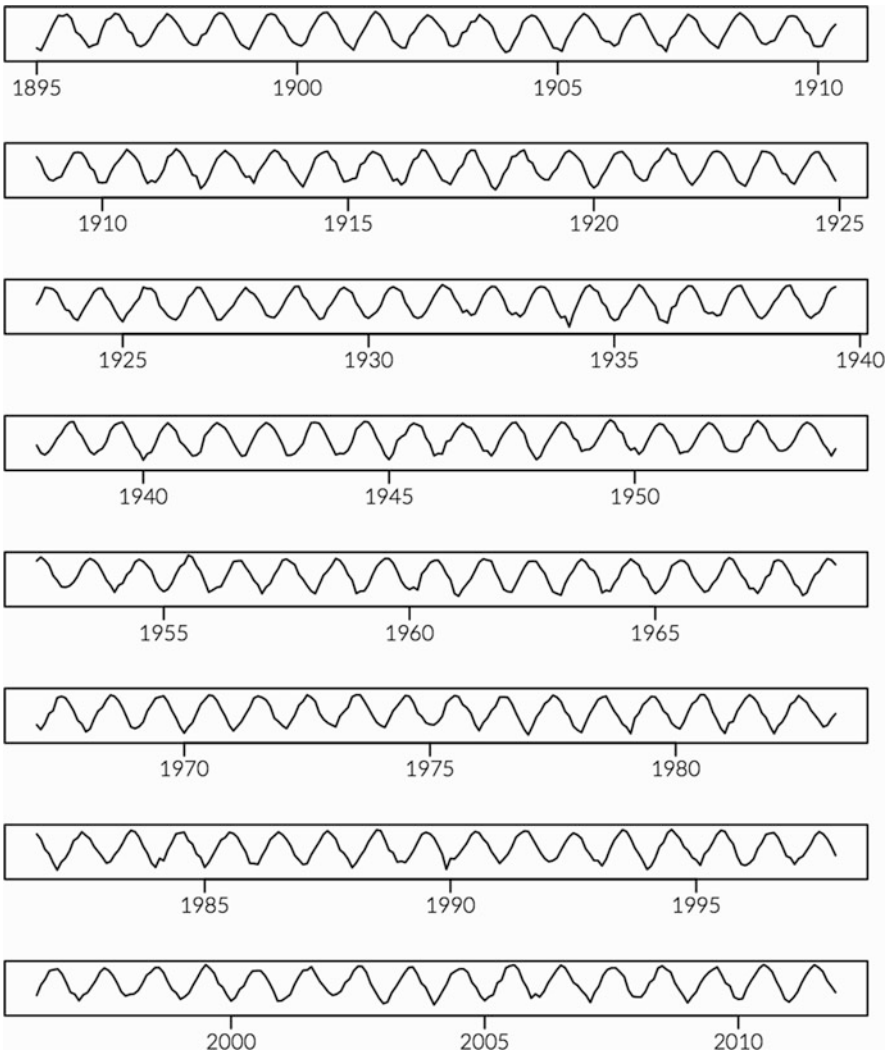

Fig. 2.7 Monthly temperatures in New Jersey between 1895 and 2011 as cut-and-stack plot

2.4 Typefaces

Typefaces make up a not insignificant part of figures. Unfortunately, they are usually neglected. However, use of the correct typeface can contribute much to clarity. An interesting study is available thanks to Sven Neumann from the design department of the HTW Berlin. He studied readability of typefaces in the public domain, and concluded from a survey of more than 100 people that the distance from which a

typeface is readable varies considerably from typeface to typeface. This is not only relevant for traffic signs as illustrations, too, profit from readable typefaces.

Many users restrict their choice of typefaces for texts and especially for illustrations to the requirements of their software or operating system. This is not only founded on pragmatism, but also has financial reasons: if you buy a high-quality typeface such as Frutiger, in regular, italics, and bold variants in three different widths, respectively, you will have to expend several hundred Euros—and still be unclear about the legal status governing its use. Fortunately, there are a fair number of free high-quality alternatives whose use makes sense even for illustrations. Before we shift our focus to these, we want to give you a quick overview over the most important properties of typefaces.

Fig. 2.8 Serif and non-serif typefaces

Currently, Germany categorises typefaces according to DIN 16518 into 11 groups. However, for everyday use, a much rougher classification suffices. Principally, proportional and non-proportional typefaces are distinguished. Especially the former are further differentiated into serif and non-serif types (Fig. 2.8). At first glance, serifs appear to be little letter ornaments: small, fine lines that are set perpendicular to the larger lines of a letter.

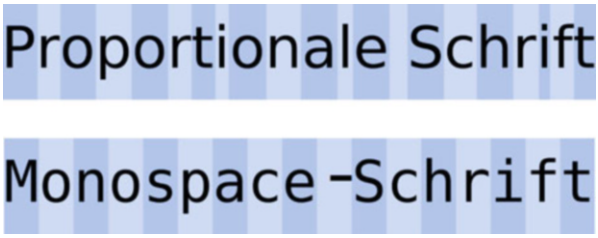

Fig. 2.9 Proportional and non-proportional typefaces. *Source*: de.wikipedia.org, Algos

Such typefaces are usually used for long texts, as serif-type typefaces are proven to be more pleasant to read. Non-serif typefaces on the other hand are used for headings or short texts. A proportional typeface (Fig. 2.9) is characterised by individual letters taking up different spaces in width. A lower ‘l’ or ‘i’ takes up less