An Introduction to Data Visualisation

Workshop Notes and References

7 July 2014

Grand Connaught Rooms

London

Dr Jonathan Minton

Applied Quantitative Methods Network

University of Glasgow

Please feel free to contact me:

[jonathan.minton@glasgow.ac.uk](mailto:jonathan.minton@glasgow.ac.uk)

For additional material and updates please visit:

<https://github.com/JonMinton/Data_Visualisation_Workshops>

Contents

[Take-Home Messages 2](#_Toc392247036)

[The Grammar of Graphics 4](#_Toc392247037)

[The Layers of Graphics 5](#_Toc392247038)

[The Three Hats of Data Visualisation 6](#_Toc392247039)

[Table Management 7](#_Toc392247040)

[References 8](#_Toc392247041)

[Books 8](#_Toc392247042)

[Software 10](#_Toc392247043)

|  |  |
| --- | --- |
| http://the-sra.org.uk/img/sralogo120x83.jpg | https://www.aqmen.ac.uk/sites/default/files/aqmen_logo.jpg |

# Take-Home Messages

The main take-home messages of the workshop are:

1. Data visualisation must involve *the consist mapping of variables within a dataset to aesthetic elements of an image*.
2. Data visualisations usually include a number of distinct layers, defined by the role they play in conveying information to the viewer. At its simplest, there are three layers: the **support layer**, the **data layer**, and the **annotation layer**. What makes a data visualisation a data visualisation, rather than some other kind of information visualisation, is the presence of the data layer.
3. There is a **grammar of graphics**, a way of describing the different elements that go into producing a data visualisation. Learning some of these terms, even if we don’t apply them consistently, can help us think and communicate more clearly about the elements, processes and aims of producing a visualisation.
4. You need three distinct mind-sets for data visualisation, three ‘hats’ that you need to swap between in order to complete different tasks involved in the process. These are the **Scientist**, the **Artist**, and the **Engineer**. On a day-to-day basis, you will spend most of your time wearing the Engineer’s hat. (Also, Scientists and Artists tend to squabble.)
5. *What counts as a good data visualisation is heavily context dependent*. A good data visualisation is one that is a good match for the audience. Audiences can be divided crudely into ‘internal’ and ‘external’ (we and they); and into ‘specialist’ and ‘generalist’. Often but not always, internal audiences are specialists, and external audiences are generalists. Data visualisations meant for internal audiences are visualisations we produce for ourselves. In these cases, visualisations are used as part of exploratory data analysis, to learn something new. By contrast, data visualisations meant for external audiences aim to communicate something we already know to other people.

# The Grammar of Graphics

Grammar is a set of tools for describing the hidden structure of sequences of words. Just as there is a grammar of language, so there is a grammar of graphics. The grammar of language distinguishes words and sequences as adjectives, nouns, pronouns, propositions, adverbs, and so on, based on the roles and functions words and sequences have in the context of sentences and paragraphs. Similarly, the grammar of graphics distinguishes different peices of a data visualisation according to their functional components.

A book, **The Grammar of Graphics**, was written by the statistician Leland Wilkinson in 2005. The book represents a highly formalized approach to understanding what goes into producing a data visualisation, providing terms and ideas for helping researchers distinguish between different component parts of a visualisation. The components of the grammar of graphics that Wilkinson identifies include:

* **Variables**: these are pieces of data, associated with observations, which contain values.
* **Scales**: these are choices of transformations that can be applied to the values associated with variables. For example, an identity scale produces as output whatever was given as its input, and a logarithmic transformation produces as output the logarithm of the input.
* **Geometries**: these are specific shapes and forms which you, as producer of the data visualiser, must choose between. Your choice will depend on your understanding of the types of variables being represented, and the types of relationship between them. For example, a line is a different geometry to a point, and a line which connects two points is different to a line which reaches upwards or sideways from an axis towards a point.
* **Coordinate systems**: these define how space is represented in an image. For example, a Cartesian coordinate system represents space in terms of latitude and longitude, whereas a Polar coordinate system represents space in terms of distance from origin and angle. Many visualisations that look very different are in fact the same, but for the coordinate system used.
* **Aesthetics**: these represent the rules by which different variables are mapped onto different geometric features placed in different positions within the space as defined by the coordinate system(!). This is an initially confusing idea, but in many ways the most important take-home message from the approach. The difference between data visualisation and other forms of information visualisation can mainly be understood in terms of this mapping process.
* **Facets**: these are tabular arrangements of multiple graphs. They are a vital tool for data visualisation meant for internal use (i.e. for exploratory data analysis), and are a simple but very important tool for high information density data visualisation more generally.
* **Guides**: these are additional pieces of information which help the viewer to understand the context of the data visualisation. Examples of guides include legends for distinguishing between different types of line and point, and scales for helping people work back from a position or other aesthetic feature of a graph and the underlying values.

# The Layers of Graphics

There are essentially three layers of a data visualisation:

* The **support layer**, containing background features such as gridlines, tickmarks, axis labels, legends and so on. In terms of the grammatical elements listed earlier, the support layer tends to contain **scales** and **guides**.
* The **data layer**, containing the results of consistently applying a series of rules for mapping from data variables to aesthetic forms. In order to be a data visualisation, a graphic must contain at least one data layer, and at its most minimal a data visualisation can contain nothing but the data layer. The grammatical elements listed earlier that data layers are made up of tends to include **geometries**, **aesthetics** and **facets**.
* The **annotation layer**, containing, well, everything else. The annotation layer can be what makes an otherwise generic graph stand out, for better or for worse. The annotation layer actively discusses and describes the data within the same frame within which it is being presented. Think of the annotation layer as providing, in print, the added value (and value judgements) that you would provide an audience if presenting the visualisation in person. Just as you might want to direct the audience’s attention to a particular point or line with a laser pointer, so you might want to add an arrow and some additional text to discuss the data within the visualisation itself.

# The Three Hats of Data Visualisation

To be effective and efficient at data visualisation, we need to learn to swap between three very different frames of mind, three ‘hats’. We need to swap between the **Artist**’s hat, the **Scientist’**s hat, and the **Engineer**’s hat.

* As an **Artist**, you will be concerned with beauty, with proportion, with form, with shape, with engagement. You will approach the task of producing a data visualisation as an opportunity to engage and inspire audiences, to bring feeling to numbers, to express yourself, to make art.
* As a **Scientist**, you will be concerned with accuracy, with precision, with efficiency, with testing and exploring theories, with disentangling and teasing apart complex causal relationships between variables, with learning something new about the world, and explaining what you have found to others clearly and honestly.
* As an **Engineer**, you will have to implement the plans agreed upon by the **Artist** and the **Scientist**, turning sketches and aspirations into reality. You will be faced with overcoming constant, annoying, niggling technical challenges and obstacles to realizing the blueprints. You will spend your time answering the question how. How do I access that data? How to I derive this variable? How do I change the line width? How do I label that point? How do I change the tick-mark on the axis? How do I do this? How can I do that?

As an **Engineer**, you will have a raw deal. If you fail to figure out how to do something, you will be blamed for your lack of knowledge and relevant skills. If you succeed, however, you will not be rewarded. Instead, the **Artist** and the **Scientist** will be thanked for the inspiration, boldness and insight of their plan. You merely carried out their instructions correctly, as you are expected to.

However – and this point is critical for understanding what data visualisation is in practice – *you will spend much more time wearing an* ***Engineer****’s hat than an* ***Artist****’s hat or a* ***Scientist****’s hat*. As a rule of thumb: data visualisation is about 80% engineering, 10% art, and 10% science.

# Table Management

As most of the job of data visualisation involves wearing the engineer’s hat, moving from one technical challenge to the next, you can usually improve the effectiveness and efficiency as a data visualiser by learning to be just a little bit more efficient and effective in data engineering issues. An important source of ideas and advice on effective data management comes from database theories, and thinking carefully about the distinction between **keys**, **variables**, and **values**. Crudely, if **keys** show *something that was* measured, **variables** record *what* was measured (of the ‘*something that was measured’)*, and **values** report *the result* of measuring (what was measured of the something that was measured).

As a simple example, the following table can be decomposed into a format where these three fundamental components are made explicit rather than implicit.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Surname** | **First name** | **Month of trial** | **Height** | **Condition** | **Weight** |
| Smith | Mary | 1 | 78 | Healthy | 45 |
| Smith | John | 2 | 62 | Sick | 52 |
| Jones | William | 1 |  | Healthy |  |
| Jones | William | 2 | 86 |  | 84 |
| Smith | Mary | 3 | 78 | Sick | 43 |

Table 1 Example 'Messy' wide format data

|  |  |  |
| --- | --- | --- |
| **Key** | **Variable** | **Value** |
| Smith\_Mary\_1 | Height | 78 |
| Smith\_Mary\_1 | Condition | Healthy |
| Smith\_Mary\_1 | Weight | 45 |
| Smith\_John\_2 | Height | 62 |
| Smith\_John\_2 | Condition | Sick |
| Smith\_John\_2 | Weight | 52 |
| Jones\_William\_1 | Condition | Healthy |
| Jones\_William\_2 | Height | 86 |
| Jones\_William\_2 | Weight | 84 |
| Smith\_Mary\_3 | Height | 78 |
| Smith\_Mary\_3 | Condition | Sick |
| Smith\_Mary\_3 | Weight | 43 |

Table 2 Long format data created by 'melting' Table 1 into its component parts.

Although the long format data is not as human readable, it is often easier to work with and reshape to produce other summary tables and graphs. Time invested in learning more about the fundamentals of data management, including using filters and pivot-tables, merging and summarising data, and reading and saving data in a variety of formats (including plain text formats such as .csv), will pay dividends in your work as a data visualiser.

# References

## Books

If you only want to read one book on data visualisation, and want to focus on using Excel, and using it to produce something professional and consistent now, then I would recommend the following book by Steven Few. One of the few ‘how-to’ book that would also look good on a coffee table:

* Few, Steven (2012) **Show Me the Numbers: Designing Tables and Graphs to Enlighten.** Second Edition, Analytics Press: Burlington

Other books by Steven Few include:

* Few, Steven (2006) **Information Dashboard Design: The Effective Visual Communication of Data.** O’Reilly: Sebastopol, California
* Few, Steven (2009) **Now you see it: Simple Visualisation Techniques for Quantitative Analysis.** Analytics Press: Burlington

If you would like to learn more about Edward Tufte’s data visualisation design principles and philosophy, then his books are a pleasure to read and view. The first and most famous of which is:

* Tufte, Edward (2001) **The Visual Display of Quantitative Information** Second Edition, Graphics Press: Cheshire, Conneticut

Two books from the ‘infographics’ end of the data visualisation spectrum, which have arguably done much to bring the idea of data visualisation to public consciousness, while also often misrepresenting the field in ways that cause more technically minded data visualisers to wince, are:

* McCandless, David (2012) **Information is Beautiful.** New Edition, HarperCollins: London (More so)
* Rogers, Simon (2013) **Facts are Sacred.** Faber & Faber: London (Less so)

(Declaration: I own both books.)

A more nuanced and balanced book from the world of data journalism is the following:

* Cairo, Alberto (2013) **The Functional Art: An introduction to information graphics and visualization**. New Riders: Berkeley, California

If you only want to buy two books, I would recommend buying Cairo’s after buying Few’s! Whereas Few’s book will help produce sturdy, clean, simple, honest graphics for helping organisations understand themselves better, Cairo’s book is an engaging and wide ranging tour popular data visualisation and infographics. A take-home message from the book is that the general public can probably cope with more information rich visualisations than they’re usually provided with.

If you are feeling brave, and want to learn about how to use statistical programming languages such as Javascript, Python and R to produce data visualisations, then I would recommend the following books by Nathan Yau, a US-based statistician who runs the website flowingdata.com

* Yau, N (2011) **Visualize This: The FlowingData Guide to Design, Visualization and Statistics.** Wiley: Indianapolis, Indiana
* Yau, N (2013) **Data Points: Visualization That Means Something.** John Wiley & Sons: Indianapolis, Indiana

Although Yau’s books are fairly technical, and producing visualisations as he does requires a lot of specialist knowledge in programming, his graphics also demonstrate a high level of care and attention to issues of aesthetics and graphic design. As an effective, all-round data visualiser, there are very few competitors to Yau.

The UK-based data visualisation expert Andy Kirk has written the following book:

* Kirk, Andy (2012) **Data Visualization: a successful design process**. Pakt Publishing: Birmingham

It is a book of two halves: the first (shorter) half discusses the process of arriving at the correct design choices for a graphical visualisation; the second half is a reference guide describing and providing a wide range of example of different types of visualisation, and suggestions about the applications for which they would be best suited. I see it as a ‘conceptual’ practical guide: something to help the Scientist and Artist work well together in drafting something suitable for the Engineer to produce, rather than something that directly makes the Engineering involved any easier.

If you would like an even more comprehensive reference of data visualisations, then I would recommend the following:

* Harris, R (1999) **Information Graphics: A Comprehensive Illustrated Reference.** Oxford University Press: New York

Much more a catalogue than a coffee table book. Nonetheless, a good book to skim through if you don’t want to rely on Excel’s default options to spark your imagination.

Finally, if you would like to learn to use a statistical package, ggplot2, which explicitly uses the grammar of graphics paradigm described by Leland Wilkinson, I would recommend:

* Chang, Winston (2013) **R Graphics Cookbook**. O’Reilly: Sebastopol
* Wickham, Hadley (2009) **ggplot2** Springer: London
* Wilkinson, Leland (2005) **The Grammar of Graphics.** Second Edition, Springer, Chicago

However, anyone taking this dangerous path would have to learn R, the statistical programming language which ggplot2 is built on, as well. Two good introductions are:

* Kabacoff, Robert (2011) **R in Action.** Manning: New York
* Field, A., Miles J & Field, Z. (2012) **Discovering Statistics using R.** Sage: London

## Software

It’s frustrating, never quite right, and has bad taste. It’s also ubiquitous, strangely compelling and often surprisingly versatile. Excel seems here to stay. It’s the technical (as opposed to conceptual) focus of this course, and even for someone like me who tends not to finish data visualisations using this package, it’s often where I start.

If you want to stray away from Excel, it’s going to cost you. The question is: are you more willing to pay in time, or in money? The answer to this question will lead your data visualisation journey down one of two very different paths:

* ***“I want to pay in time”***: Then firstly consider learning many of the more technical features of Excel, such as VBA – the underlying programming language – pivot tables, connecting with databases, using macros and so on. This will allow you to increase the flexibility and functionality of Excel for data management and visualisation a lot. Unfortunately, there can be diminishing returns as you push further past the limits of Excel. If you have the time and inclination, and are willing to suffer the constant frustrations involved, then the dedicated statistical programming language R may allow you to travel further in the longer term. R is free in terms of money, but definitely not in time and effort. A range of books have been listed above, but if you want to dip your toes in the language for a few minutes, then the Codeschool website offers a free, gently paced, and surprisingly addictive, nautically themed introduction to the language, available from the following link: <http://tryr.codeschool.com/>
* ***“I want to pay in money”***: Of course, time is money, so even what on the face of it looks like a lot of money for a dedicated and user friendly data visualisation program might turn out cheaper than spending weeks, months, or years learning something like R. If you want something that’s less limited than Excel, but less scary than R, then Tableau is the main option on the table. Tableau is a commercial data visualisation program which makes it relatively easy to produce attractive publication ready visualisations, and to explore data visually using a largely intuitive interface. Conceptually, try to imagine the pivot table and pivot chart options in Excel, but *much, much better*, with better handling both of data management and visual design. Tableau comes in both a public version, and a commercial version. The difference between the versions is not in the functionality, but in the storage of the data. In the public version, data has to uploaded to Tableau’s website in order to be used. In the commercial version, you can keep the data to yourself. The cost of secrecy is about £1,000. The public version of Tableau can help you decide whether, for your organisation, this is a price worth paying: <http://www.tableausoftware.com/>