Module 1 Introduction to data exploration and visualisation

Kimbal Marriott

Contents

Chapter 1

Introduction to data exploration and visualisation: Overview

By Kimbal Marriott

Updated 28 February 2019

This is the first module of six in FIT5147 Data Exploration and Visualisation. FIT5147 introduces statistical and visualisation techniques for the exploratory analysis of data.

1.1 Aims of this module

After completing this module you will:

- Understand the important roles that visualisation plays in data science: data checking and cleaning, exploration and discovery, and the presentation and communication of results;
- Introduce visual analytics and the complementary roles of visualisation and data analysis in data science;
- Appreciate that data visualisation is not (just) about making cool looking InfoGraphics, it is about really showing the data in all of its messy detail;
- Know a little about the history of data visualisation and how it has been shaped by factors such as: presentation technology, human cognitive abilities and limits, societal needs, data availability and the invention of different kinds of information graphics;
- Have first-hand experience in data exploration and visualisation with easy-to-use visual analytics tools like Tableau and with programming languages like R.
- Know about the **What-Why-How** framework for understanding data visualisation design
- Have first-hand experience using the **five design sheet methodology** for designing data visualisations

1.2 How to study for this module

In this module we draw on material in the public domain, including journal articles and quite a few videos.

In this module there is one assessment activity: creating an interactive data visualisation with Tableau Public.

You should also start your data exploration project. By the end of the module you should have identified what you want to investigate and where you are going to get your data from.

Good general introductions to data visualisation and visual analytics are:

- Munzner, Tamara. *Visualization Analysis and Design.* CRC Press, 2014.

- Ward, Matthew O., Georges Grinstein, and Daniel Keim. *Interactive data visualization: foundations, techniques, and applications (2nd Ed).* CRC Press, 2015.

Other more specific references will be mentioned throughout the course.

Chapter 2

Data Visualisation

By Kimbal Marriott, Richard Cox

Updated 28 February 2019

The human visual system is amazing, almost half the human brain is devoted to vision. It is our primary sense for understanding the world. To "see" something is to understand it. In this module you will learn that interactive visualisation is one of the most important tools in the data scientists' workbench, helping them to both understand their data and then to communicate what they have discovered to other people. Data visualisation isn't about creating cool InfoGraphics, it's about creating visualisations that really allow you to understand the data. If you have a strong emotional reaction to the data visualisation it should be because of the underlying data, not the visual representation. To get a feel for the kind of visualisations we will be interested in take a look at the following two TED talks:

- The first is by the late Hans Rosling The best stats you've ever seen (20 mins) and has become a classic in the data visualisation field. Bill Gates credits one of Rosling's talks for convincing him to give billions of dollars to healthcare projects in developing countries (see Hans Rosling: the man who makes statistics sing).
- Another great TED talk The beauty of data visualization (20 minutes) is by David McCandless

And spend some time looking at data visualisations developed by Tableau and Spotfire users

- https://public.tableau.com/s/gallery
- http://spotfire.tibco.com/demos/

Think about what you like in these visualisation and start to think about whether they are effective in allowing you to understand the underlying data. If they are then try and understand why are they effective. If not, what could be done to improve them.

Chapter 3

Visual analytics and the role of data visualisation

By Kimbal Marriott, Richard Cox

Updated 28 February 2019

Data visualisation is an extremely effective way of understanding your data. The combination of visualisation with statistics, data mining and other kinds of computational analytics is often called visual analytics. This term was coined by James Thomas and Kristin Cook just after the terrorist attack on the US in September 11, 2001. The original focus of visual analytics was on helping security analysts and emergency response services but it is now widely used in business intelligence and sciences for data analysis. In A Visual Analytics Agenda, visual analytics is defined by Thomas and Cook "as the science of analytical reasoning facilitated by interactive visual interfaces." It is designed to "detect the expected and discover the unexpected."

Visual analytics is all about putting the human-in-the-loop when doing data analytics. If you know exactly what you are testing for then the human is not needed and you can just run the analysis. However if you are not sure what you are looking for-which is mostly the case in data science-then visual analytics supports interactive exploration of the data:

- Visualise the data,
- Make some tentative hypothesis,
- Run appropriate analytics and visualise the results.
- Repeat this until you have found what you need.

One reason that visualisations are effective is that they can contains a huge amount of information. Standard statistical measures such as the mean, median, standard deviation or correlation summarise the data using just a few numbers. An information graphic potentially provides much more information about the data as it can show thousands (even million) of graphic elements each of which can use position, colour, pattern or shape to encode information about the data.

In 1973 the statistician Francis Anscombe constructed four data sets to show the importance of graphing data and also the effect of outliers on commonly used summary statistics. These sets are called the Anscombe Quartet. The x and y values in the four sets are carefully chosen so that the x-values have almost identical mean and variance, y-values have almost identical mean and variance, and they have the same linear regression with the same correlation. If you didn't look at the data more closely you would believe that they were very similar. However the moment the data is graphed it is very clear that the data sets are different and that they have very different characteristics.

Anscombe's Quartet of Data (Generated with R):

The other reason that data visualisation is effective is the human visual system. Much of our visual processing

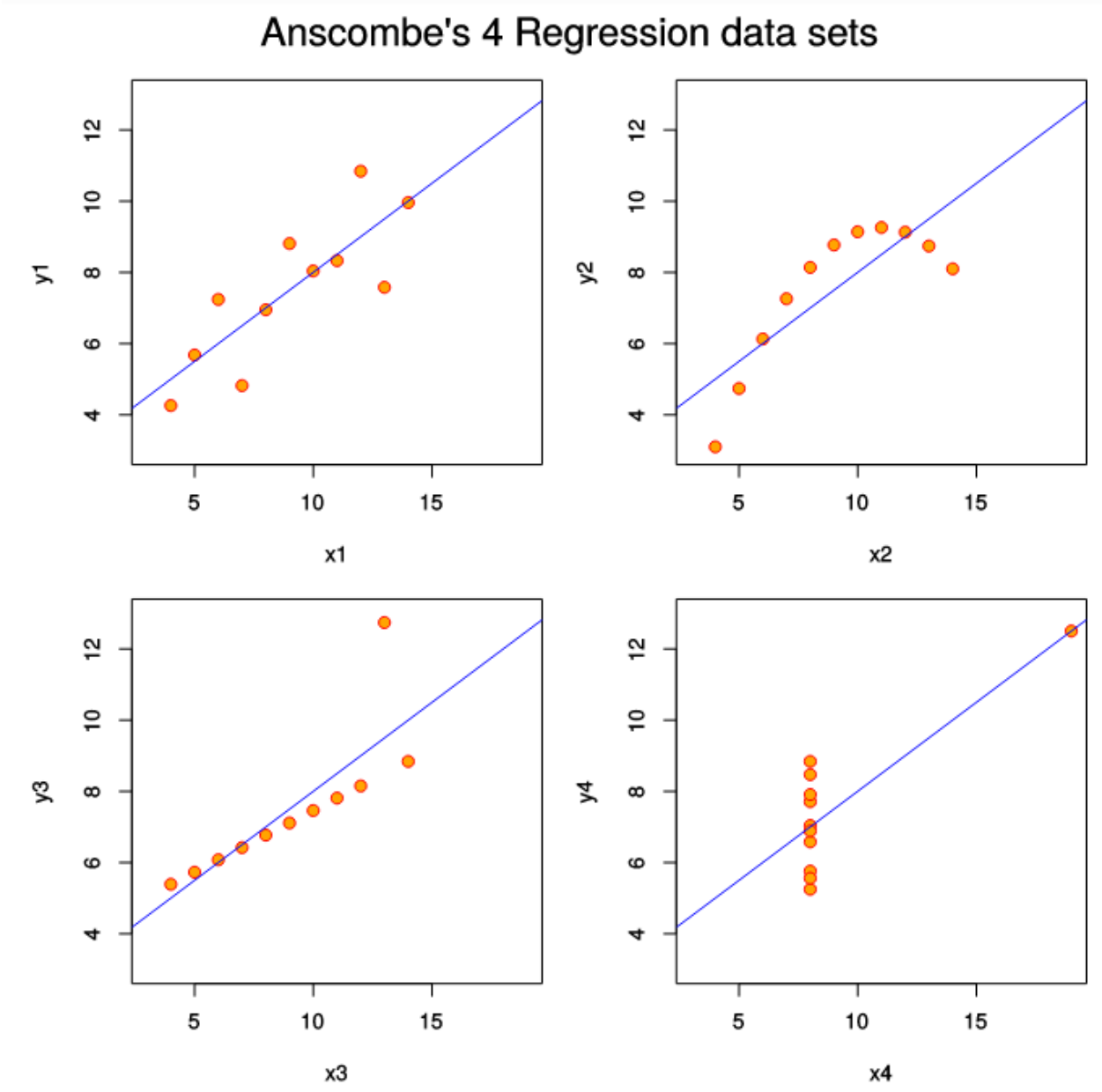

Anscombe's 4 Regression data sets
y1
y2
y3
y4
x1
x2
x3
x4
4
6
8
10
12
5
10
15

Figure 3.1:

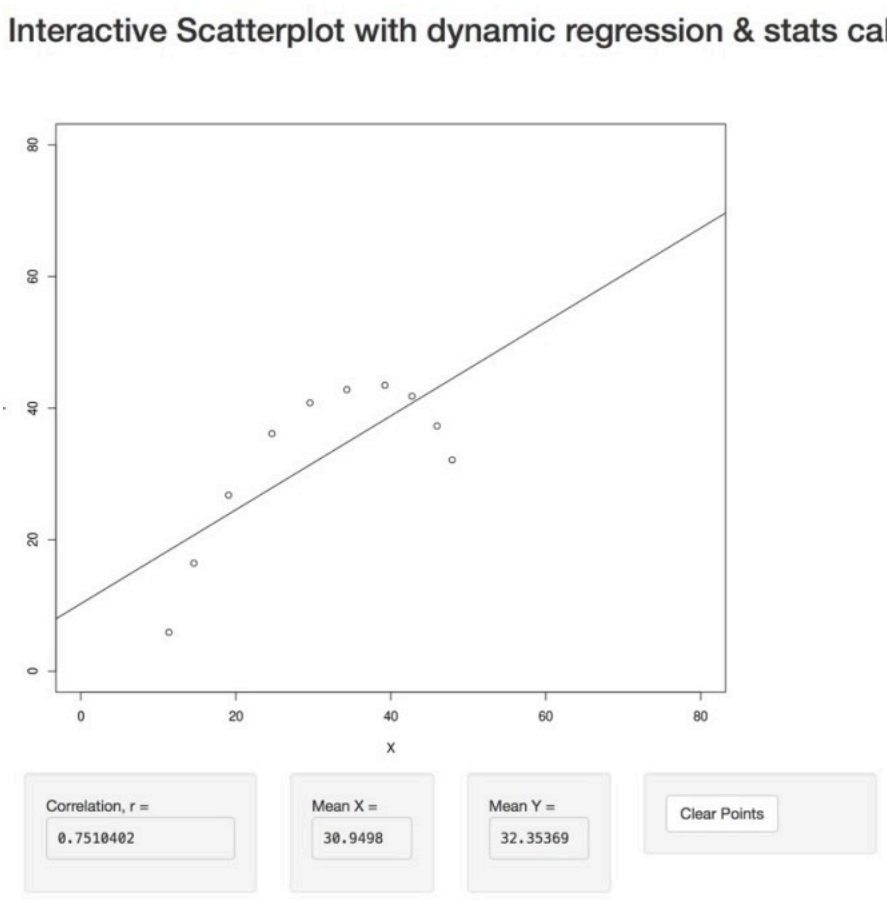
is pre-attentive and occurs in parallel. This means that with a well designed visualisation we can see patterns, anomalies, and trends very quickly-we don't have to read through the data line by line. We immediately see from the scatter plot of Anscombe's quartet how the different data sets are grouped in quite different patterns.

ACTIVITY

You can explore how data set groupings affect statistics such as the mean, standard deviation and correlation coefficient using this interactive scatterplot (made with R and Shiny).

This scatterplot allows you to add data points by clicking - you can create your own data set shapes. As you add points the regression line, correlation coefficient and means of X and Y are dynamically updated.

Add data points and observe the effect of various shapes of scatterplot on the mean of each variable and the correlation coefficient. Try and reproduce the four data set shapes of Anscombe's Quartet and also try constructing three differently shaped scatterplots that each yield a positive correlation with a coefficient value r=0.8. Explore the effect of outlying values by bunching most of the data points into a circular pattern (observe value of the correlation coefficient) - then put one point at a distance away (try various positions). Observe how the correlation coefficient is affected by just one such 'outlier'.



3.1 Role of data visualisation in data science

Data visualisation is used for three main purposes in Data Science.

1. *Data checking and cleaning.* When you first get your data you should do some quick plots of the

individual features to check that there are no obvious errors and to get a feel for the distribution of values.

2. *Exploration and discovery.* According to Mike Lourdes (What is Data Science?) Hilary Mason, one of the world's leading data scientists, says that when she gets a new data set, she starts by making a dozen or more scatter plots, trying to get a sense of what might be interesting. Visualisation reveals possible connections and patterns that can then be confirmed (or not) using other kinds of analysis. Visualisation also plays a key role in understanding any kind of spatial data.

3. *Presentation and communication of results.* The other important use of visualisation is to present the results of your analysis. This has two main purposes: (1) to help you and other modellers/analysts understand the results and (2) to communicate the results to other stakeholders.

3.2 Data checking and cleaning

Whenever you first obtain some data it is a really good idea to check it. This can reveal simple entry errors like an extra 0, missing values or strange patterns in the data like 20% of the product prices are $99.99. This might be correct or it might be an error in the data.

The following is a quick check list for each attribute:

- Look at some random records
- Compute the mean, median and quartiles for the data. Look at a box plot of these.
- Determine the number of missing values and invalid values (NaNs), number of special values like 0.
- Determine the number of distinct values and whether they really are distinct.
- Plot the frequency distribution of values. This might be with a histogram or density plot. You should play around with the choice of bin width as this smooths the data.
- Check for symmetry (skewness) and the flatness/spikiness of the distribution ('kurtosis', note platykurtic = flat distribution with low peak, leptokurtic = spikey peak around the mean).
- Look at the outliers and check whether they should be thrown away (trimmed) rounded up or down (Winsorised). This might be done, for instance, for data outside the 5th and 95th percentiles.
- Check formats for dates, that they are in comparable time zones.
- Plot latitude and longitude on a map to check they are sensible.
- Check text for strange characters or encoding

One quick way to look at your data to find missing values is using R with the 'mi' (multiple imputation) library. In the example below you can see (e.g.) income was not always recorded but age was (4th and 5th columns):

If you are using statistical tests that require a normal distribution then check that the data does appear to be normally distributed. There are statistical tests for doing this but they can be quite picky. A better approach is to actually plot the data and test for normality graphically.

There are two ways of doing this. The first is to look at a histogram or density plot of the data distribution and see if it looks like a normal distribution entered around the mean. Another way of testing for normality is to use a Q-Q plot. This is discussed more fully in Analysis of trends and patterns in tabular data

3.3 Exploration and discovery

The heart of data science is exploring the data and discovering patterns and trends. Visualisation plays a core role in this process. Scatter plots, time series, data maps, tag clouds and many other kinds of graphics that you will see later in this unit allow the data scientist to get to know their data and the connections between it. As part of this process visualisation is used to understand the results of analysis such as clustering or curve fitting. Visualisation plays a complementary role to computational analysis: it is how the results of the analysis are presented to the data scientist so that they see all of the details and "confirm the expected" or see the "unexpected."

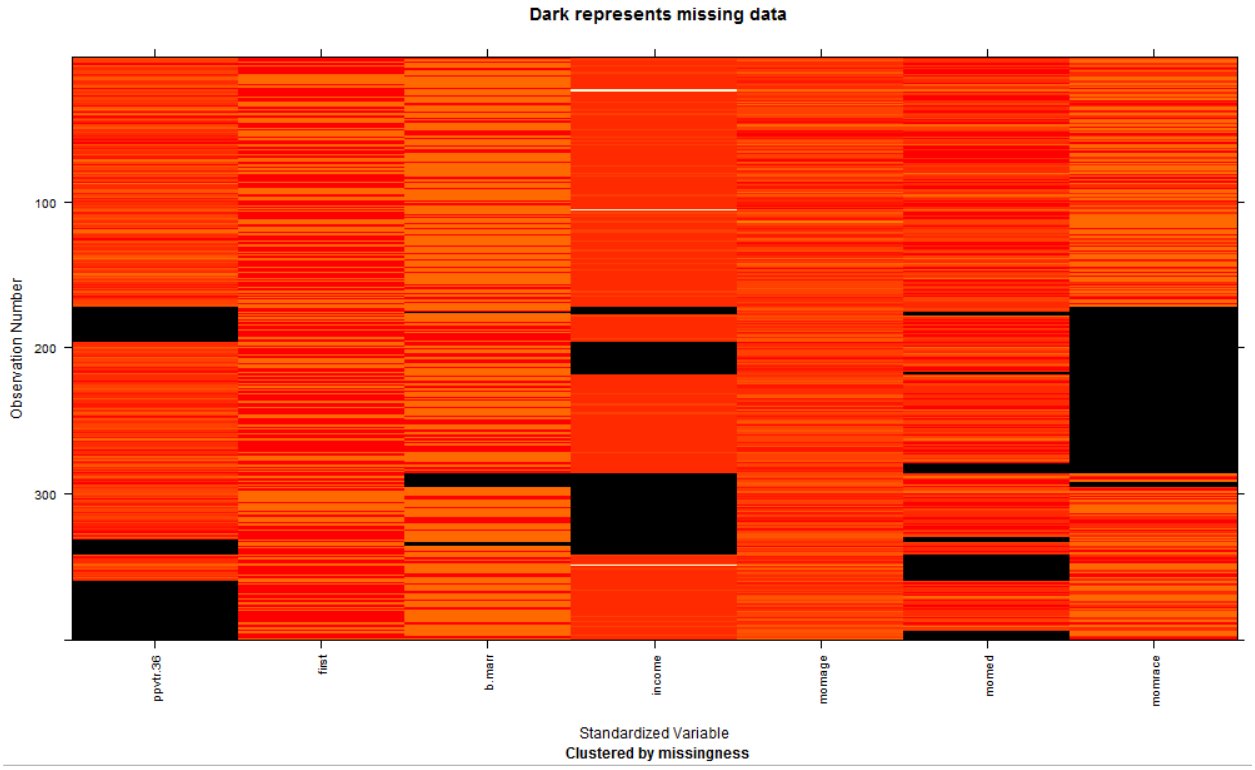


Figure 3.2:

Exploration is inherently incremental. A visualisation leads to a hypothesis which leads to another visualisation to see if this is supported by the data and perhaps some confirmatory statistical testing of the hypothesis. This in turn might lead to another hypothesis. As part of this process data is fused from different sources and it may be that the process identifies that new data must be collected or found. Typical task during exploration are:

- Search for elements that satisfy certain properties, if they exist. This might be locating a known data point, filtering the data, or finding outliers.
- Identify the properties of a single data item
- Compare or rank elements
- Visually identify patterns in some subset of elements. Examples include trends, correlations, clusters or categories.
- Calculate derived properties not originally in the data. These may be data transformations, data aggregations or may be statistical properties such as regression lines or clusters

In many of the activities for this unit you will explore data and discover patterns and trends. And I leave the last word to the great mathematician John W. Tukey:

The greatest value of a picture is when it forces us to notice what we never expected to see. J.W. Tukey. Exploratory Data Analysis, 1977.

Tukey invented the box plot and according to Wikipedia he also invented the words “bit” and “software.”

3.4 Presentation of results

The other common use of visualisation in data science is for communicating the results of the analysis.

This communication might be to other analysts working on the same problem. In this case the visualisations

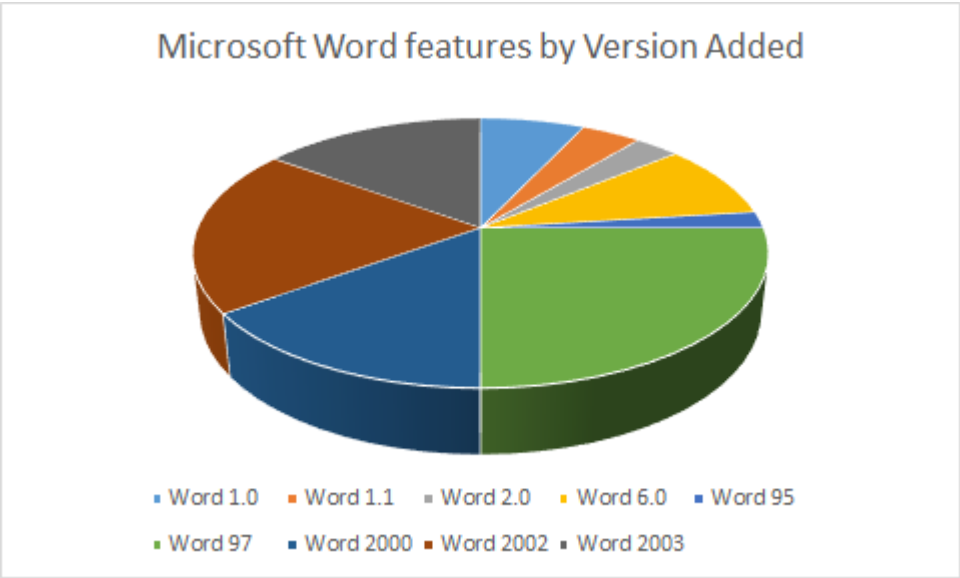


Figure 3.3:

will typically be those used in exploration and discovery, perhaps slightly cleaned up.

Much more effort needs to be put into preparing graphics and visualisations for communicating results to stakeholders who are not data scientists. These might be managers, policy makers or students in the case of educational projects or the general public in the case of journalists.

These kinds of visualisations require considerable time to prepare. They typically are used to communicate a particular message or narrative. The graphic needs to be designed to clearly communicate that message to the reader. The production values are also very high: plots and graphs produced by standard graphing packages are often touched up using graphics editing tools like Adobe Illustrator or Inkscape.

Once these presentation graphics were static, typically printed on glossy paper or shown in PowerPoint presentations. Nowadays they are often interactive and published on the Web.

When developing data visualisation for presentation don't get sucked in by trying to create a cool looking InfoGraphic. You need to carefully think about the best way to present your data and story and make sure that it communicates it effectively and does not simplify the story too much by hiding complexity.

ACTIVITY

Take a look at the following graphics showing feature additions to Microsoft Word. Below is a recreation of a 3D pie chart (using MS Excel, 2013), that seems to have first appeared on this Microsoft blog (the original is even worse, note the conflicting colours). Don't worry about the accuracy of the data, this is all about the graphic design.

Consider what is bad (or good) about this chart. Now, same data, different chart (also MS Excel, 2013):

What other types of chart would be appropriate for this data? There's a critique of the original pie chart here

Which criticisms, if any, apply to the bar chart?

There is increasing sophistication in the data visualisations produced by magazines and newspapers. According to Nathan Yau the New York Times employs data visualisation experts in both its graphics de-

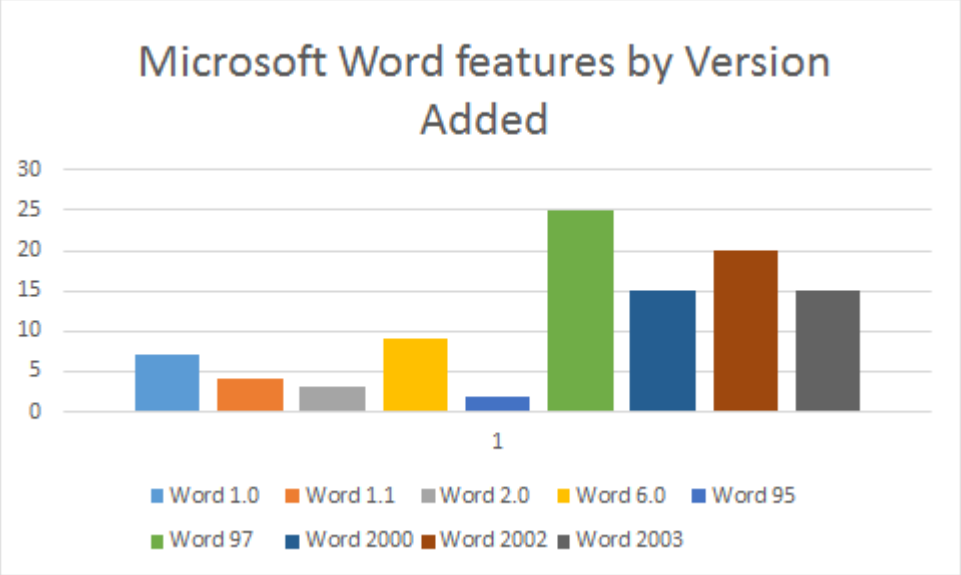


Figure 3.4:

partment and in a group dedicated to online interactive graphics. (http://flowingdata.com/2008/12/10/what-jobs-are-there-in-data-visualization/)

3.5 Summary

Visualisation is used for three important purposes in data science:

- initial data checking and cleaning,
- exploration and discovery,
- and presentation of results.

Visualisation is effective because the human visual system allows parallel perception of large amounts of information. Visual analytics is the name given to the combination of interactive visualisation with statistics, data mining and other kinds of analytics.

FURTHER READING

Take a look at Daniel Keim's 1-hour seminar on visual analytics. He is one of the pioneers in this field and he provides a great introduction to why visualisation is indispensable in data exploration.

Nathan Yau's website and books are also well worth checking out. He is an expert in making easy to understand and interesting data visualisations. Take a look at some of his project on his Flowing Data website. Also look at

Visualize This: The FlowingData Guide to Design, Visualization, and Statistics. Nathan Yau. 2011

Also recommended reading is:

The chapter on "Exploring Data with Graphs" in "Discovering Statistics Using R." Andy Field, Jeremy Miles, Zoe Field. 2012

Chapter 4

Brief history of data visualisation

By Kimbal Marriott

Updated 28 february 2019

Our modern world is awash with information graphics; maps, plans, timetables, charts and diagrams. We take them for granted, tools to help us communicate and to understand the world. It has not always been like this: Very, very few information graphics are more than one thousand years old, suggesting that they were relatively rare until this time.

By understanding why information graphics and then data visualisations became commonplace we can better appreciate their role in data science and also how they may change in the future. In this module we will present a brief history of data visualisation and the main reasons for its rise. I believe there are four kinds of reasons:

1. *Improved technology for producing and presenting graphics.* Widespread use of information graphics could not have happened without paper and printing and interactive data visualisation was impossible without the computer.
2. *Changes in societal needs and attitudes to graphics.* The Scientific Revolution required scientific and medical illustrations, the Age of Exploration needed maps, the Industrial Revolution required engineering drawings and the 21st century requires interactive data visualisations.
3. *Availability of data.* Detailed maps require accurate surveys while graphs showing population or wealth distribution require census data.
4. *Invention of graphical notations and interaction techniques.* These allow the data to be shown in ways that are useful, that answer society's needs. They form a visual language.

We can break the history of data visualisation into four main periods. We look at these in turn.

4.1 Prehistory

It is widely believed that information graphics such as schematic maps and cosmological drawings have been used in traditional hunter-gathering and subsistence farming societies for thousands of years. However the evidence for this is somewhat inconclusive as most of these were probably produced on ephemeral materials such as sand, bark or even drawn on the human body.

One hunter-gathering culture with a long tradition of using schematic maps and cosmological diagrams are the Australian aborigines. They are believed to be one of the oldest surviving cultures in the world and have inhabited Australia for more than 60,000 years. Much of their art shows connection to place and travels by dream time beings across the land. Rock paintings and engravings from the Western Desert may be the oldest known maps. These show circles representing water holes with connecting lines representing journeys between the water holes.

4.2 Early civilisations

The first civilisations arose in Mesopotamia and the Nile Valley in Egypt around 3500BCE. These were followed by civilisations in the Indus Valley in India at about 2500BCE and in China around the Hwang-Ho (Yellow River) in about 2000BCE. Civilisations also independently emerged in the Americas, first in Mesoamerica around 500BCE and then in South America. All are believed to have arisen when population growth or a changing climate put pressure on subsistence level farmers living in river valleys or flood plains surrounded by arid land. Because the surrounding land was not suitable for farming, the only way to increase food production was by building large scale irrigation and drainage works which could dramatically improve the productivity of the arable land allowing it to support a much larger population.

In most of these early civilisations we find evidence of information graphics. While there was contact between ancient Mesopotamia and Egypt, the other civilisations had little or not contact with each other so it is likely that their use of information graphics evolved independently.

This is perhaps not very surprising; early civilisations had similar problems and so came up with similar solutions. Centralized political authority required ways to record ownership of land and its boundaries-this led to detailed maps recording property boundaries (these are called cadastral maps). The need to compute taxes and land area led to geometry and geometric diagrams; the need to plan military campaigns, towns and buildings led to plans; while the need to predict the seasons and other cyclic events led to star maps. And the need to understand one's place in the world gave rise to symbolic cosmological maps.

4.3 Printing and paper

While some information graphics survive from prehistoric times or early civilisations they are rare. One of the main reasons for this was that it was very difficult to accurately reproduce graphics. Before the invention of printing all copies had to be done by hand. This was time consuming and extremely error prone.

During the European Renaissance this all changes. Paper and wood-block printing, which were invented in China, made their way to Europe via the Islamic world. Paper arrived in about the twelfth century soon followed by printing. At first woodblock printing was used for producing items such as tarot and playing cards. It was also used to print books though it was not well suited to this because the fine detail required for text is difficulty and time consuming to carve. Around 1440, Johannes Gensfleisch zur Laden zum Gutenberg invented the printing press and moveable metal type. This revolutionized the production of books and soon woodblock illustrations were combined with printing, allowing the production of illustrated books.

In the next five centuries there was continuous improvement in printing technologies. Copper-plate printing allowed much finer details while lithography in the late 19th century provided cheap high-quality prints with colour for the first time.

By the beginning of the twentieth century information graphics were everywhere. Educational and reference books as well popular media like newspapers and magazines contain a wide variety of charts and maps, street maps and atlases are in common use, and graphics are widely used in science, medicine, architecture and engineering as well as business and government. We now look at when these various kinds of graphics were developed.

4.4 The rise of information graphics

Modern scientific and technical drawing techniques originated in the European Renaissance. They underpinned the emergence of modern science, medicine, engineering and architecture. Sketching is the primary tool that architects and engineers use for design, they learn their trade by looking at drawings and plans of existing buildings and machines, and use drawings and plans to communicate with the client and to specify the final design to the builders.

One of the great discoveries of the Renaissance were rules for drawing buildings and machines in linear and parallel perspective and in using measured plans and multi-view orthographic projections to precisely

Figure 4.1: Clay tablet showing a Babylonian world map from 500BCE. Babylon and the Euphrates river are shown by rectangles at the center of the map. Other towns are shown by circles and they are surrounded by the Bitter or Salty Sea. Copyright © The Trustees of the British Museum

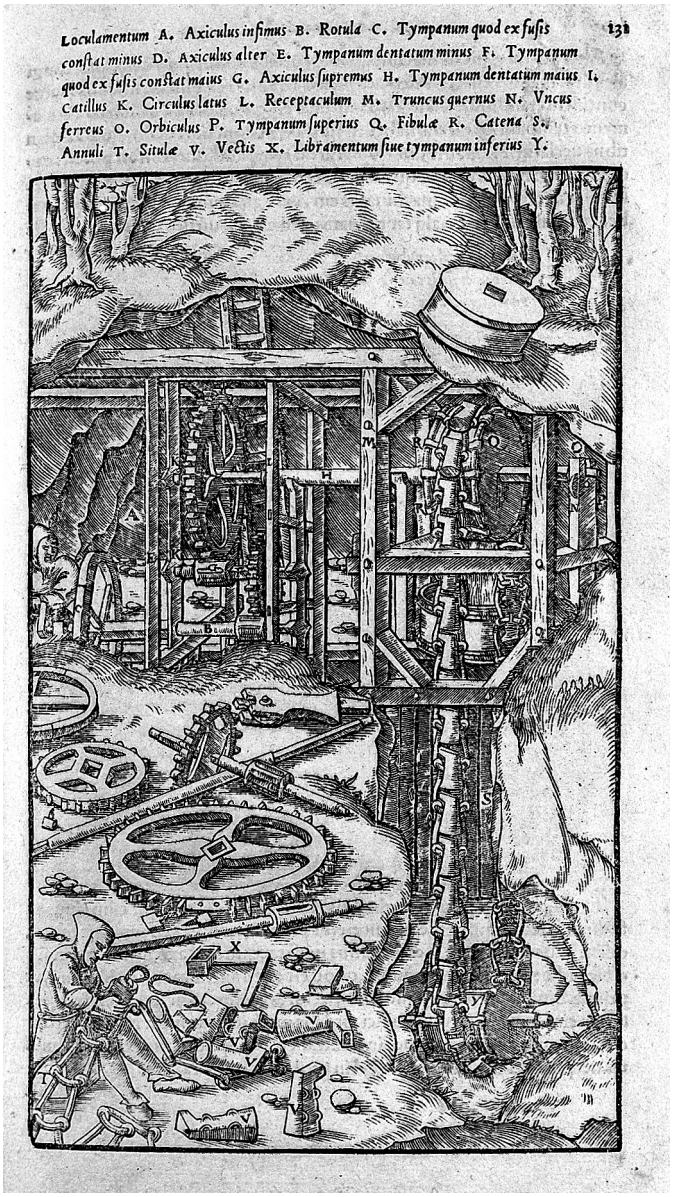


Figure 4.2: Machine for Drawing Water from Agricola, De re metallica libri XII. This was one of the first printed technical books. Credit: Wellcome Library, London.License: CC Attribution 4.0 International (CC BY 4.0)

Figure 4.3: Martin Waldseemüller World Map from 1508 showing European discoveries in Africa and of the Americas. Placed in the public domain by the Regents of the University of Minnesota, Twin Cities. License: Public Domain

specify their dimensions. These were needed as architects and engineers became separate professions and craft methods of construction were replaced by a separation between design and construction.

Drawings and diagrams are equally important to modern science and medicine. These rely on scientists sharing their observations and theories. Without illustrations it would have been impossible for early scientists to share their observations of nature, of human anatomy, of the new kinds of animals and plants encountered by early European explorers, and of the new worlds visible through the telescope and microscope.

Illustrations were not only important for recording observations, they allowed scientists to communicate new experimental apparatus and designs. And for many scientists, diagrams and mental imagery were tools that allowed them to understand nature and to powerfully convey the resulting theories to other scientists.

Maps also became more important during the Renaissance. Portuguese explorers made their way down the coast of Africa, eventually finding their way around the bottom of Africa to the "Spice Islands" or Moluccas in Indonesia and Christopher Columbus made his way to America, though he persisted in thinking he had actually reached Asia.

Sailors needed accurate maps and map making techniques improved rapidly. Latitude and longitude along with accurate scales became standard. Printing allowed the general public to see maps of these new discoveries and the world's first atlas was printed in 1570.

It wasn't until the 17th and 18th centuries that charts and graphs of more abstract material became common. Before that time, tables were the usual way of showing abstract information. For instance, many of the clay tablets surviving from Ancient Mesopotamia contain tables of mathematical exercises completed by novice scribes. One can almost hear the young boys (no girls of course) groaning as they are forced to memorize the very large multiplication tables required for a numbers system built on base 20 and base 60.

One key idea in information graphics is to use location on a plane to represent 2D numerical data-the *Cartesian plane*. This generalises latitude and longitude used in maps. Rene Descarte and Pierre de Fermat independently invented the basic idea a bit after 1600. In 1686 Edmund Halley was the first to use Cartesian plots for data analysis by fitting a hyperbolic curve to air pressure vs elevation but scatter plots with curve fitting were not commonly used until the 19th century.

Often we want to overlay maps with other kinds of information. Edmund Halley also pioneered this kind of information graphic. These are called *thematic maps* by cartographers but I prefer to use Edward Tufte's name for them, *data maps*. Halley created one of the first data maps in 1686 showing wind direction and another in 1701 using isograms showing magnetic variation. Also noteworthy is a map by Jodocus Hondius from 1607 showing the distribution of religions

Cognitive studies suggest we use spatial reasoning for thinking about time. This is probably why English (and many other languages) implicitly use a spatial metaphor for time: an event occurs before or after another event. One event takes longer than another event. It is surprising therefore, how long it took before time-lines were invented. Joseph Priestly introduced them in 1769: see his Chart of Biography. Before this

tables were the standard way to organize temporal data: lists of kings and queens etc. William Playfair built on Priestly's ideas in 1786 and invented the line graph, bar chart and pie chart.

From this time there was a fast and furious invention of different kinds of charts, diagrams and graphs for showing all kinds of data. And this hasn't stopped, there are many hundreds of different kinds of graphics. We will look at the most useful of these in the next three modules.

4.5 Interactive data visualisation

The widespread use of information graphics in the first half of the twentieth century was made possible by the invention of paper and printing. In the middle of the twentieth century there was another invention, one that influenced data visualisation as much as paper and printing. This was, of course, the computer.

Visionaries like Ivan Sutherland saw that the computer could allow a completely different way of interacting with graphics. One in which the computer is used interactively to create new graphics. Sutherland wrote a revolutionary computer program, Sketchpad, in the early 1960s for his PhD at MIT. Sketchpad was the first interactive graphics editor. It introduced basic algorithms and techniques from computer graphics and is the ancestor of modern day CAD systems. It also introduced a simple kind of object-oriented programming. Sutherland received the Turing Award for this research in 1988. The following video shows a early film clip of Ivan Sutherland demonstrating Sketchpad in about 1962. The commentary is by Alan Kay, another pioneer in GUI research.

Also in the 1960s Roger Tomlinson created one of the first GIS systems, the Canada Geographic Information System (CGIS). And working at AT&T Bell labs Edward Fowlkes created an early interactive statistics package that used brushing to select outliers for removal and for linking points in multipart displays.

These systems were the first to support computer-mediated data visualisation in which the user can interactively explore and visualise data. They led to several distinct research fields: Geographic information systems (GIS) which capture, store, manipulate, analyze, manage, and present geographic data; scientific visualisation (SciVis) which is concerned with visualising scientific data that has an inherent spatial representation; information visualisation (InfoVis) for visualising more abstract data; and CAD/CAM systems for designing buildings and machines.

Changes in computer technology are continuing to drive innovations in data visualisation. The rise of the web has meant that presentation graphics are now published on the web rather than on paper, potentially allowing the reader to interact with the graphic. And new kinds of virtual reality (VR) and augmented reality (AR) devices such as the Oculus Rift, HTC Vive or Microsoft HoloLens look set to radically transform data visualisation yet again.

4.6 Summary

We have seen how data visualisation has been used for thousands of years. Its history has been shaped by society's needs, availability of data and by the underlying production and presentation technologies: paper and printing, then the computer. Most of the information graphics that we take for granted today, such as topographic maps, measured plans and even scatter plots and bar charts were painstakingly developed over the last six hundred years. Their invention is one of the great achievements of the modern world.

FURTHER READING

If you want to find out more about the history of data visualisation take a look at Michael Friendly's Milestones Project and his chapter on its history http://www.datavis.ca/papers/hbook.pdf

Also take a look at Edward Tufte's great books on data visualisation. These include:

- Edward Tufte. *The Visual Display of Quantitative Information* (2nd Edition) Graphics Press. 2001.

Figure 4.4: Using the Oculus Rift with Leap Motion tracking of hands to collaboratively explore air traffic patterns at the Monash University SensiLab. License: Copyright © Monash University, unless otherwise stated. All Rights Reserved.

- Edward Tufte. *Envisioning Information.* Graphics Press. 1992.

Chapter 5

Tools for Data Exploration and Visualisation

Data scientists use a wide variety of different tools for data exploration and visualisation. They fall into three main categories.

5.1 Introduction

By Kimbal Marriott

Updated 19 February 2018

5.1.1 General programming languages

There are two main programming languages used

- Probably the most widely used language for data visualisation and exploration is **R**. While I find R a difficult language to use, it has incredible libraries for data analysis and visualisation. One of the best of the graphics libraries is **ggplot2** by Hadley Wickham.
- The next most common choice is **Python**. This is a nicer language but does not have quite the choice of analysis packages. It is ideal for data wrangling and Python is often used for scrapping and fusing data, then R for exploration and visualisation.

An increasingly popular choice for creating interactive presentation graphics for the web is

- **JavaScript** with the **D3** data visualisation library. It is not commonly used for other purposes by data scientists.

The great advantage of a programming language approach is that you can, in principle, do anything.

5.1.2 Generic visual analytics tools

A number of easy to use tools for data analysis and visualisation are available. Examples include

- SAS Visual Analytics is a visual frontend to many of the SAS analytics tools.
- Tableau is a neasy to use visual analytics tool that is commonly used by business intelligence for interactive exploration of tabular data. It allows interactive graphics to be published on the web. It has a commercial version and a free limited offering called Tableau Public.
- Qlik is another visual analytics tool widely used in business intelligence.

	A	B	C	D	E	F
1	Year	Grantee	Discipline	Total Award Amount	City,Country	Country
2	2013	Complex	Visual Arts	8100	Tibet, China	China
3	2013	24th Street Theatre	Theater	8000	Mexico City, Mexico	Mexico
4	2013	Practice, LLC	Multi-discipline	12000	Rwanda; Uganda	Rwanda
5	2013	Ruben Martinez	Multi-discipline	18000	Mexico City, Mexico	Mexico
6	2013	Francois Eloi Perrin	Design (Architecture)	7500	Paris, France	France
7	2013	Community Partner	Music	5500	Lagos, Nigeria	Nigeria
8	2013	Lorena Ramos	Theater	11400	Salvador	El Salvador

Figure 5.1:

5.1.3 Application specific visual analytics tools

The third category of tools are systems designed for helping data analysis in a particular application area. There are many of these. As an example look at

- National Map is designed to show Australian government data on top of a map of Australia
- Scaffold Hunter is designed to analyse data in the life sciences with a focus on drug discovery.

FURTHER READING

Take a look at 385 Data Visualization Tools to find out about more tools

---

5.2 Activity: Exploring & Visualising Data with Tableau Public

By Kimbal Marriott, Yalong Yang

Updated 2 March 2018

5.2.1 Tableau Public

One of the best Data Analysis tools according to KDnuggets is:

No.1 Tableau Public: "Tableau democratizes visualization in an elegantly simple and intuitive tool." Top 10 Data Analysis Tools for Business

The 'Public' in the name refers to the promotion of online sharing of visualisations (in fact the only way you can save them, see: https://public.tableau.com/s/gallery).

Which may be why there's a free version of the software available, download and install: https://public.tableau.com/s/download

(Tableau Desktop is available for educational users if anyone is interested, and you can save your visulisations locally with it).

5.2.2 A. Worksheets and Dashboards

Step 1. Get data ready

Download and examine the data "Cultural-Exchange-International-Program-LA-Dept-of-Cultural-Affairs.xlsx" here:

The data is already in "**row-oriented tables**" format. Which means, in this data file:

- Each row is a entity (or an object, or a record)
- Each column is a property of this entity (or this object, or this record)

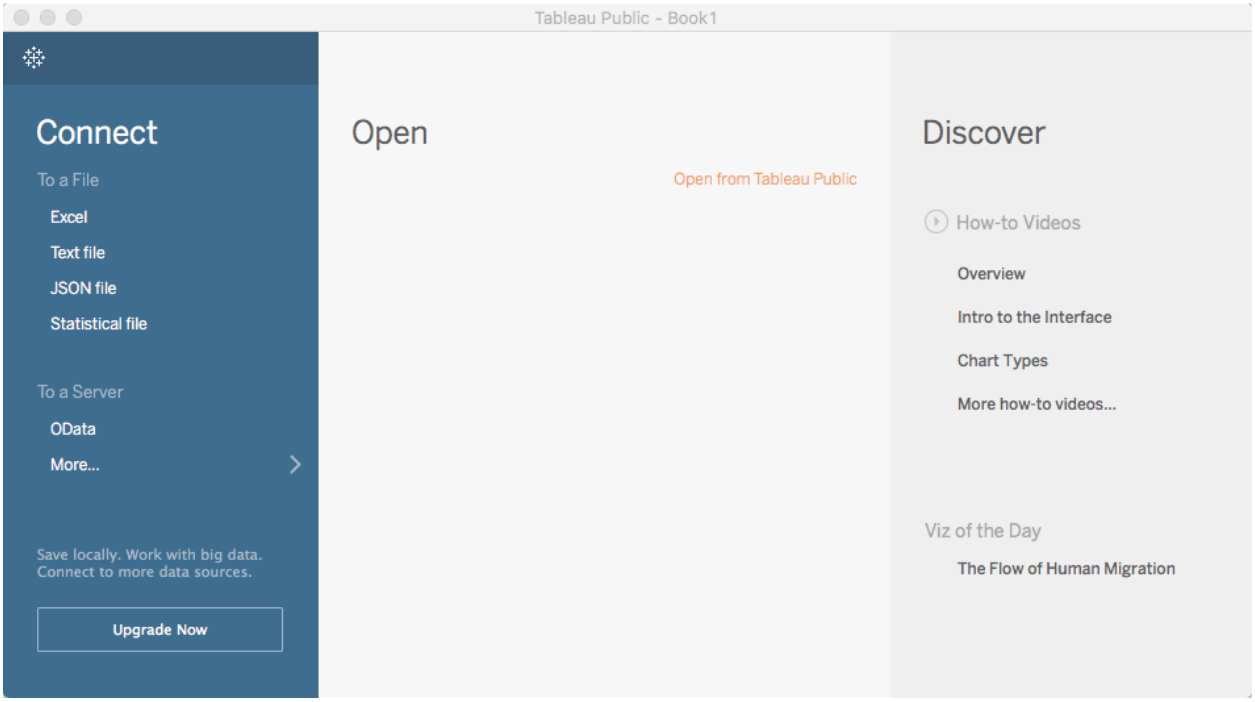


Figure 5.2:

Usually, this is the preferred format for most tools and data processing languages (like R).

However, you cannot expect all data are in this perfect format. If it is not the case, do Data wrangling!

More information about file format can be found at: Tableau Help: Tips for Working with Your Data.

Step 2. Load data into Tableau

Open Tableau Public (TP), there are two ways to load the data:

- at the left top click "**Excel**" and navigate to the file you just downloaded.
- Just drag your file and drop it into Tableau Public window.

Step 3. Data Source View

TP will try to determine your **data types** automatically, the common types are:

- # => numbers
- Abc => text (strings)
- An earth symbol => geographic/locations

Usually this automatic process works fine, but please remember to check that the data type inferred by TP is exactly the data type you want, sometime TP will make mistakes.

The most common mistake is that TP will sometimes treat longitude and latitude numbers as #, but we usually want them be treated as geographic type.

To change the type, just click the **symbol (#, Abc, Earth symbol and etc.)**.

Step 4. Create worksheet

Go to worksheet by clicking on Sheet 1 (at bottom of display, as shown below)

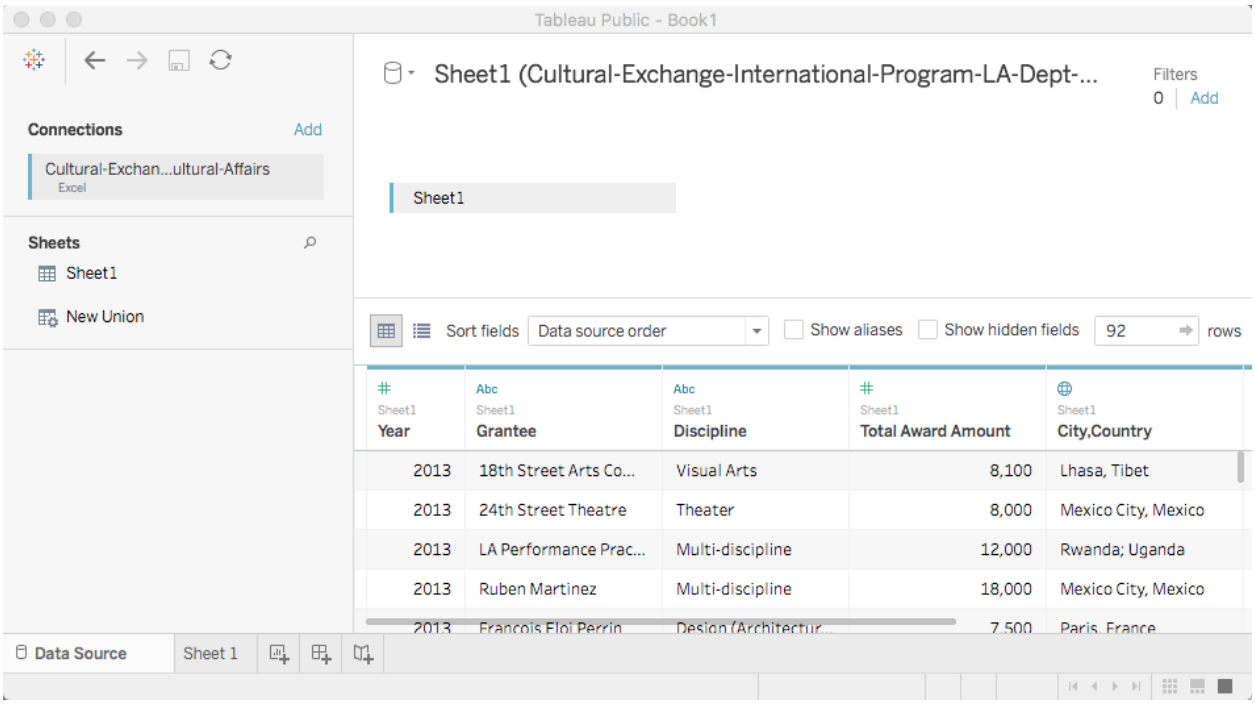


Figure 5.3:

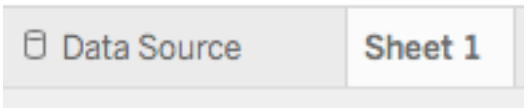


Figure 5.4:

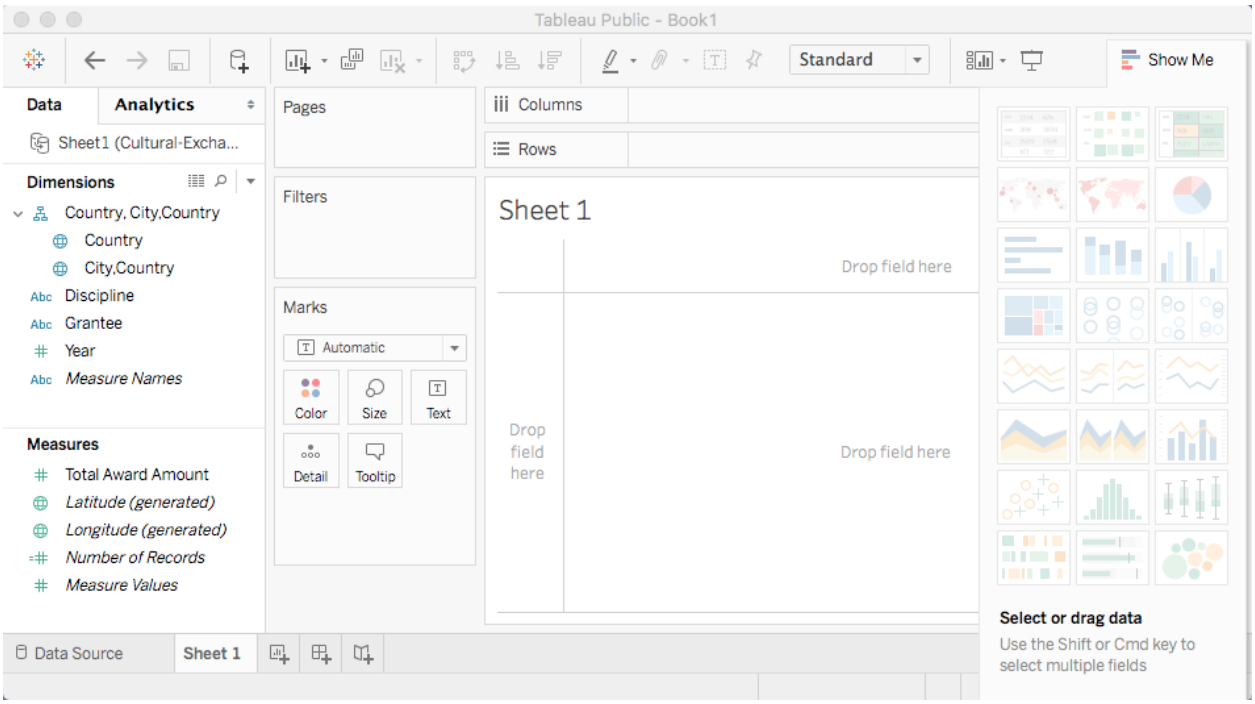


Figure 5.5:

Figure 5.6:

to see the TP interface:

Shown on the left, data has been split into

- **Dimensions**; by default, Tableau treats any field containing qualitative, categorical information as a dimension;
- **Measures**; and any field containing numeric (quantitative) information as a measure.

This modular treatment of information – that is to say, the treatment of individual data fields as independent components instead of an interdependent table – enables us to pick and choose what specific pieces of data we want to visualise against one another.

Shown in the middle, top down, are '**Pages**' '**Filters**' and '**Marks**' and then '**Columns**' & '**Rowsm**' then a '**Drop field here**' table (you can drop fields on all of these middle controls):

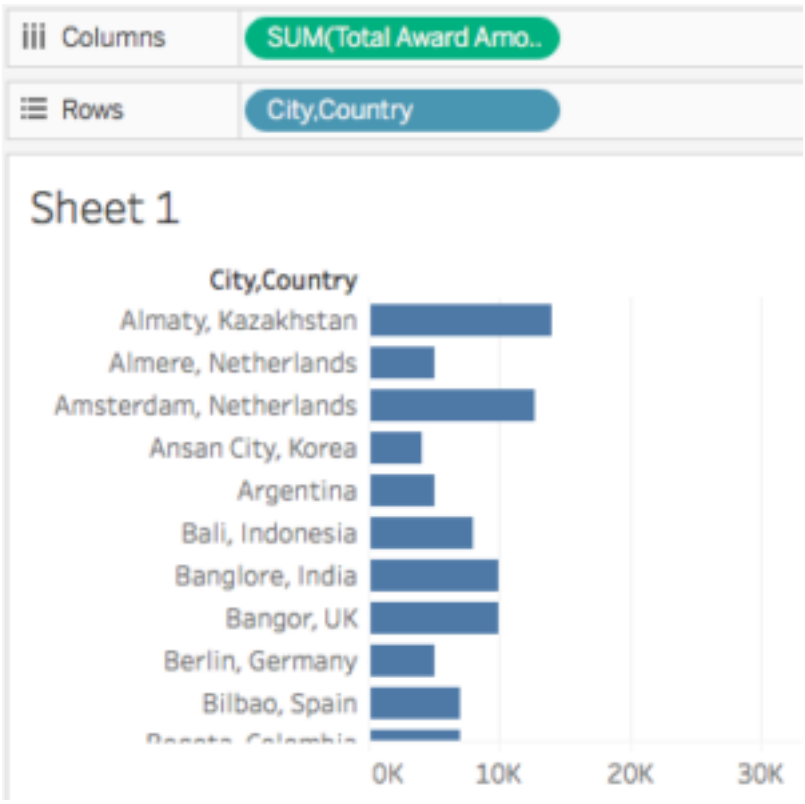


Figure 5.7:

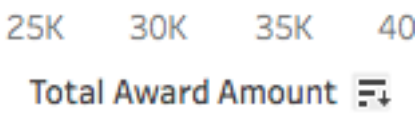


Figure 5.8:

Step 5. Create Visualisations

For our first visualisation, we will create a simple horizontal bar graph measuring City/Country against the Total Award Amounts granted to the artists from said locations.

- Drag 'City,Country' from **Dimensions** and drop it to **Rows**
- Drag 'Total Award Amount' from **Measures** and drop it to **Columns**
- Maybe also try to switch Rows and Columns

Step 6

Try sorting by using the control attached to "Total Award Amount' on the axis.

Step 7

Imagine we wanted to see what individual grant amounts compose the Total Award Amount for each Country/Region. To achieve this, we would want to differentiate between "Grantees".

The "Marks" functions will allow you to insert further detail into your visualization. Click and drag your "Grantee" dimension into the "Marks" table beneath the icons (as below). Your visualization should now

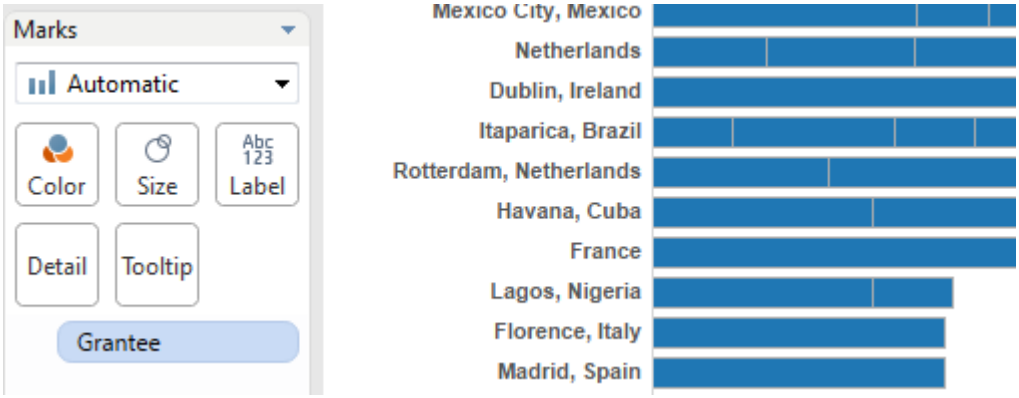


Figure 5.9:

feature individual segments, which you can click for details about all the dimensions and measures you have worked into your visualization (you can see for example that Cuba has two segments).

Who were the two recipients in Havana, Cuba and how much?

(Notice the SUM(Total Award Amount) above, how about other measures, try max, min, total, mean..?)

Step 8. One more dimension

We can add another dimension, e.g. 'Discipline' but let's distinguish it somehow (colour!). Drag 'Discipline' onto 'Colour':

Which Discipline was granted the most, the least? Which countries got funding for film?

Step 9. Change colors

Don't like those colours? By clicking the "Color" in the "Marks" window, you can customize the color palette, adjusting to the distribution of information present in your visualization.

Then 'Edit colors'

Step 10. Filtering

Filter by year. Drag 'Year' onto 'Filters' to see:

And 'OK' to select all (2009 to 2013). Now all this data is wrapped up and ready to go, let's try a geographical view.

Step 11. Map

Start a new sheet, the icon to the right of Sheet1 below

Drag and drop 'Country' onto the table below 'Rows' (use the larger of the 'Drop field here' cells):

=>

This launches a map, change it from a symbol map to a **filled map** using the 'Show me' dialog at right, drag 'Country' onto 'Label' also:

Now look for the measures labeled 'Latitude' and 'Longitude' **What does 'generated' mean?**

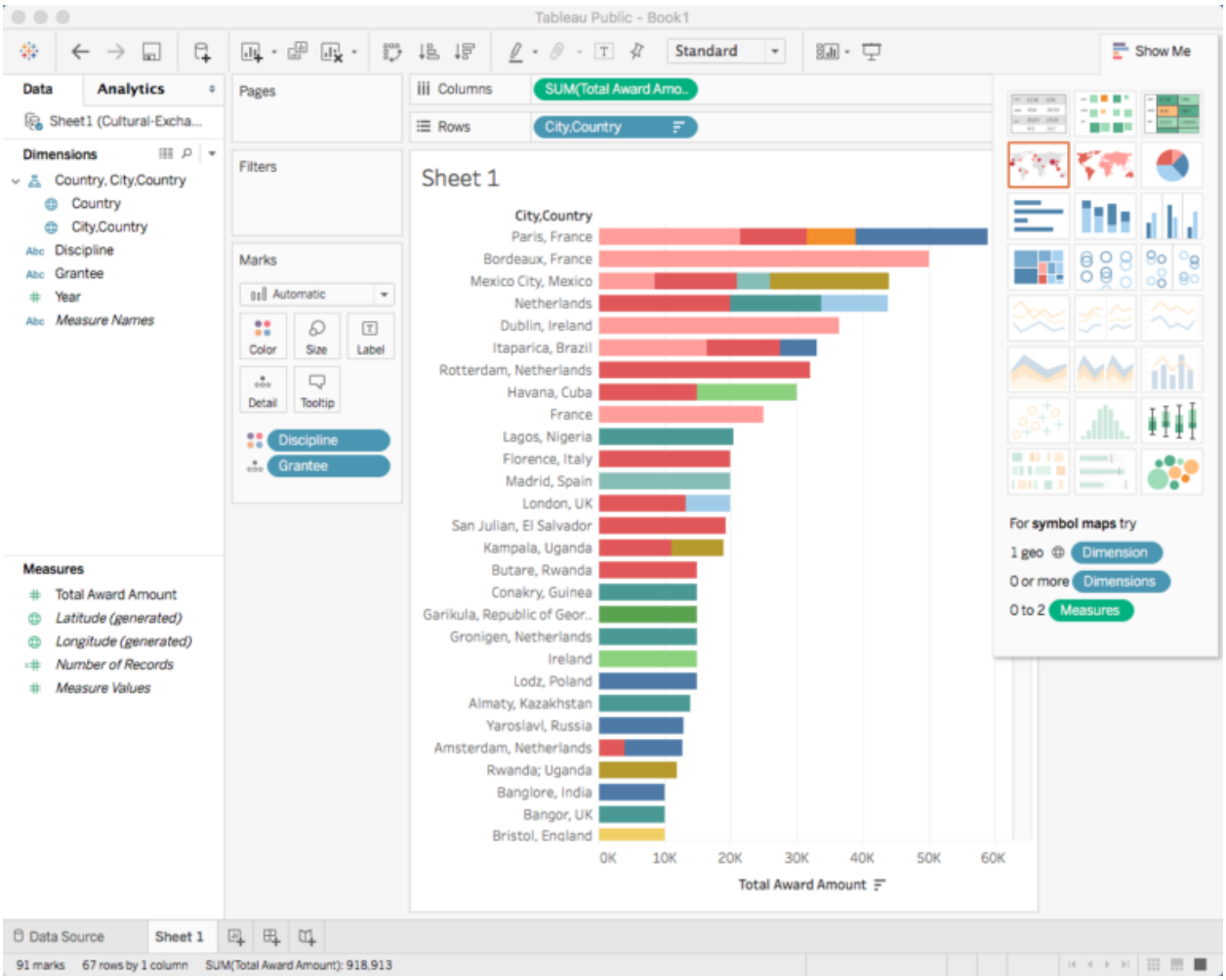


Tableau Public - Book1
Standard
Show Me
Data
Analytics
Sheet1 (Cultural-Excha...
Dimensions
Country, City,Country
Country
City,Country
Discipline
Grantee
Year
Measure Names
Measures
Total Award Amount
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Discipline
Grantee
Columns
SUM(Total Award Amo..
Rows
City,Country
Sheet 1
City,Country
Paris, France
Bordeaux, France
Mexico City, Mexico
Netherlands
Dublin, Ireland
Itaparica, Brazil
Rotterdam, Netherlands
Havana, Cuba
France
Lagos, Nigeria
Florence, Italy
Madrid, Spain
London, UK
San Julian, El Salvador
Kampala, Uganda
Butare, Rwanda
Conakry, Guinea
Garikula, Republic of Geor..
Gronigen, Netherlands
Ireland
Lodz, Poland
Almaty, Kazakhstan
Yaroslavl, Russia
Amsterdam, Netherlands
Rwanda; Uganda
Banglore, India
Bangor, UK
Bristol, England
0K
10K
20K
30K
40K
50K
60K
Total Award Amount
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
Data Source
Sheet 1
91 marks
67 rows by 1 column
SUM(Total Award Amount): 918,913

Figure 5.10:

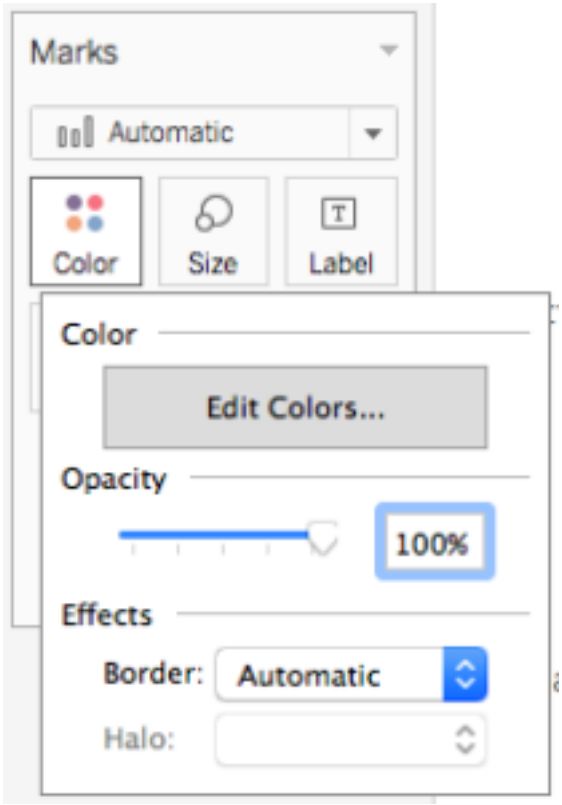


Figure 5.11:

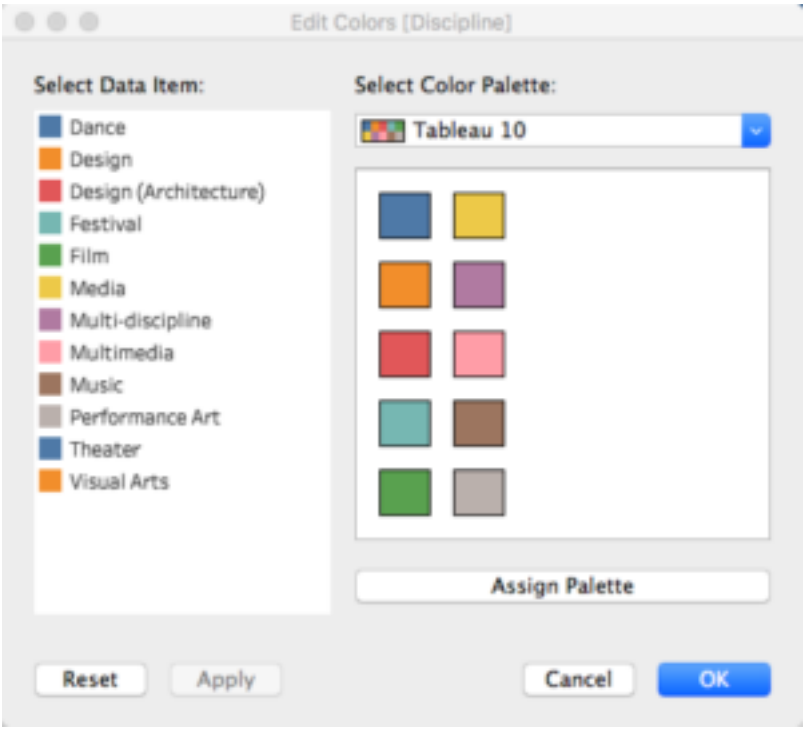


Figure 5.12:

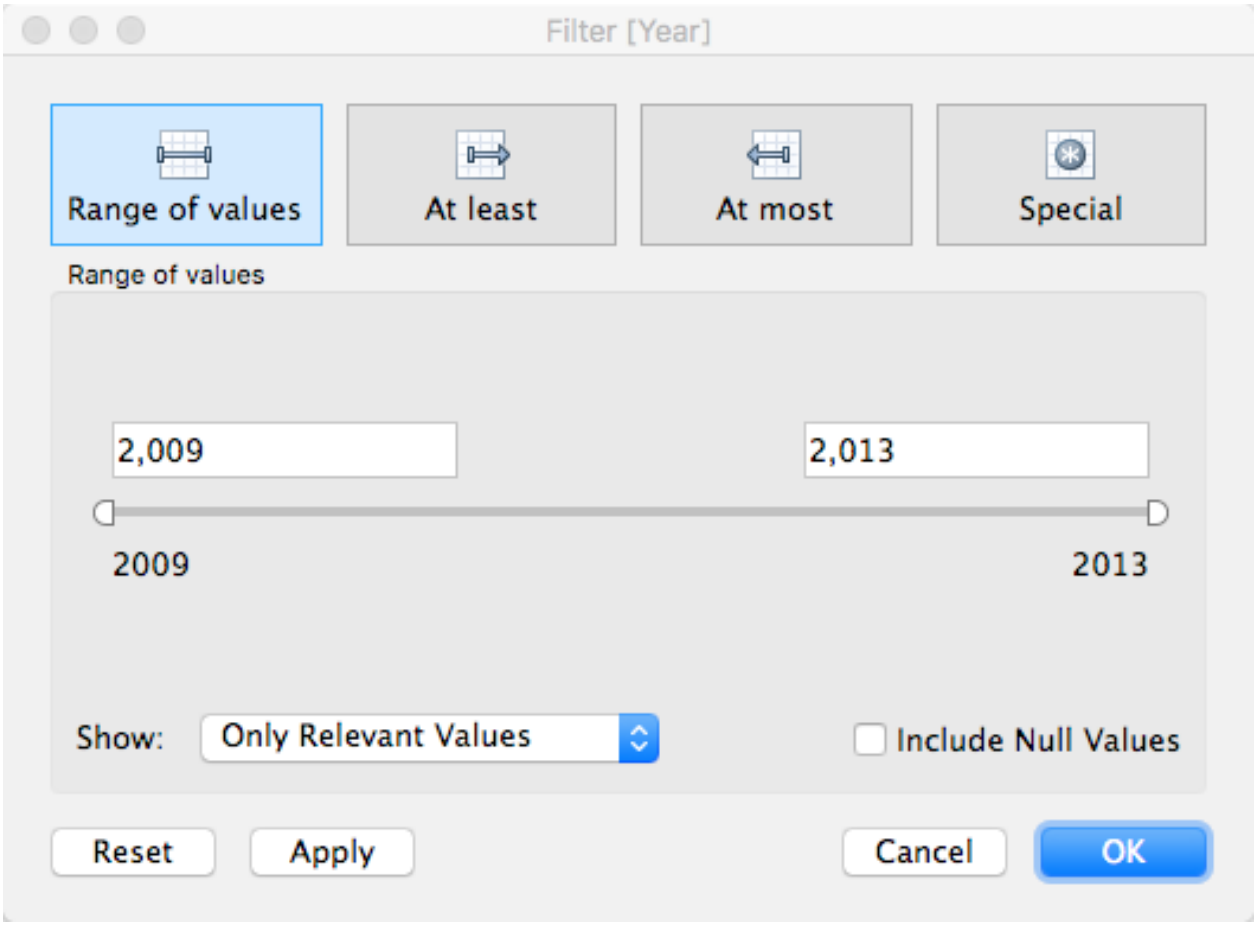


Figure 5.13:

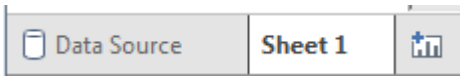


Figure 5.14:

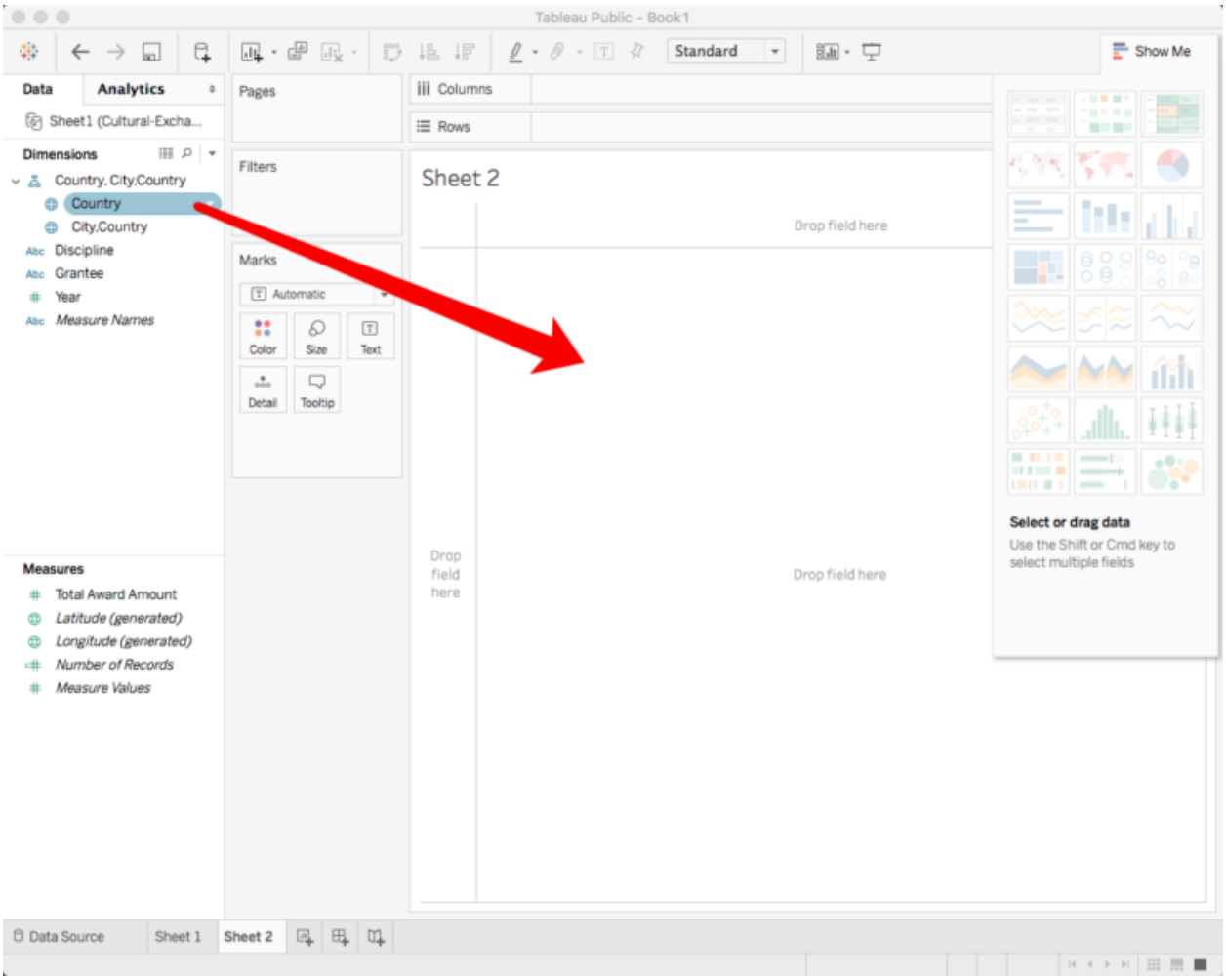


Tableau Public - Book1
Standard
Show Me
Data
Analytics
Sheet1 (Cultural-Excha...
Dimensions
Country, City,Country
Country
City,Country
Discipline
Grantee
Year
Measure Names
Measures
Total Award Amount
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Columns
Rows
Sheet 2
Drop field here
Drop field here
Drop field here
Select or drag data
Use the Shift or Cmd key to select multiple fields
Data Source
Sheet 1
Sheet 2

Figure 5.15:

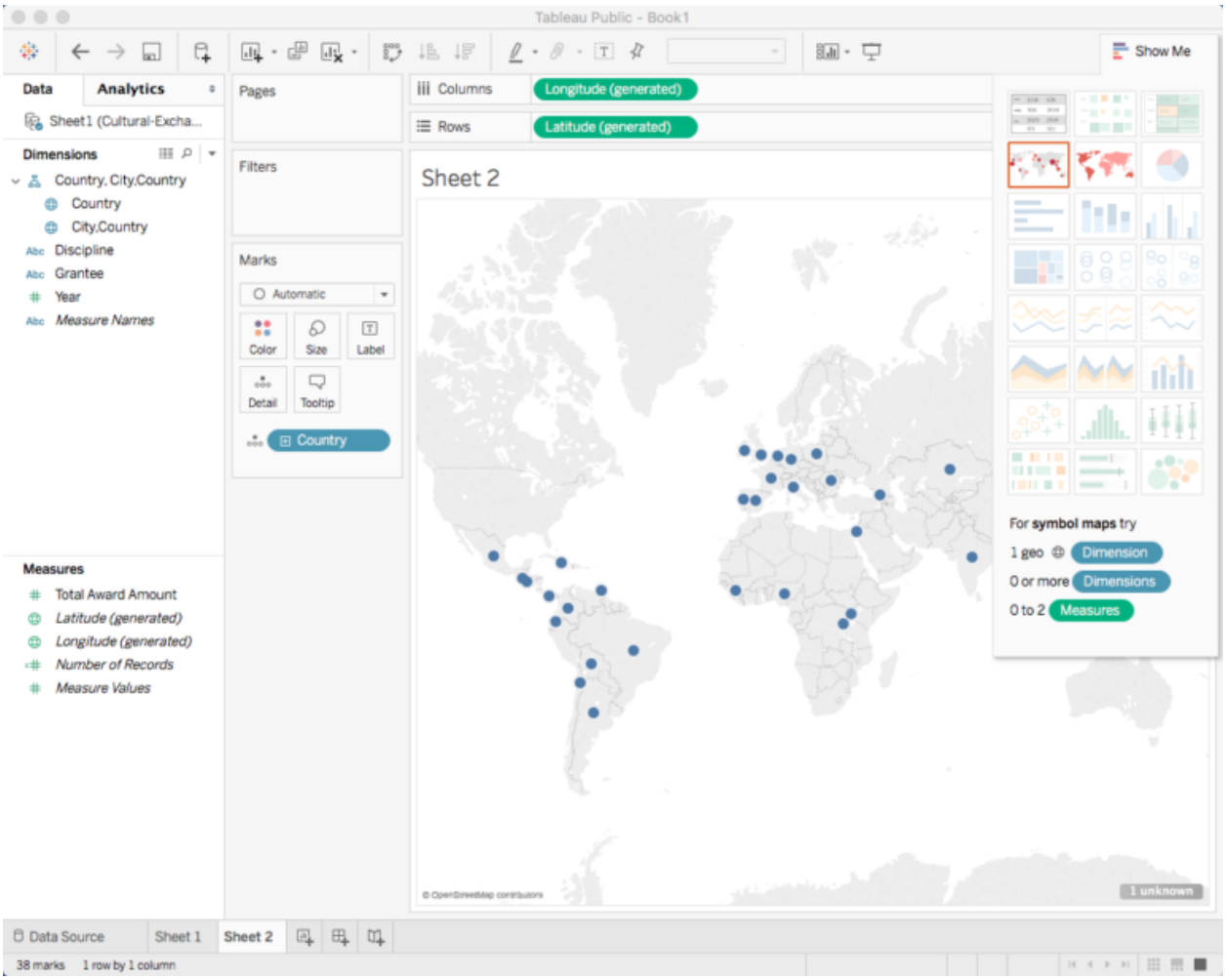


Tableau Public - Book1
Show Me
Data
Analytics
Sheet1 (Cultural-Excha...
Dimensions
Country, City,Country
Country
City,Country
Discipline
Grantee
Year
Measure Names
Measures
Total Award Amount
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Country
Columns
Longitude (generated)
Rows
Latitude (generated)
Sheet 2
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
1 unknown
Data Source
Sheet 1
Sheet 2
38 marks
1 row by 1 column

Figure 5.16:

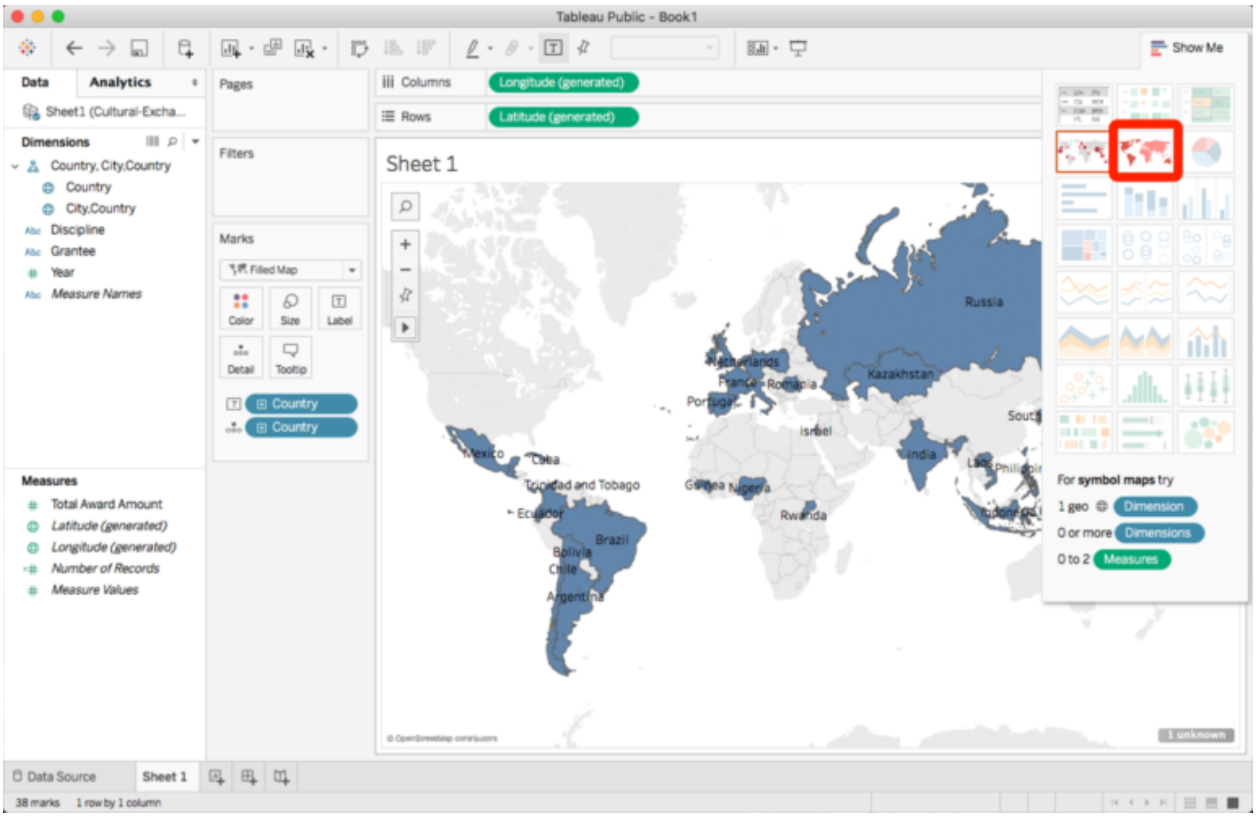


Tableau Public - Book1
Show Me
Data
Analytics
Sheet1 (Cultural-Excha...
Dimensions
Country, City,Country
Country
City,Country
Discipline
Grantee
Year
Measure Names
Measures
Total Award Amount
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Filled Map
Color
Size
Label
Detail
Tooltip
Country
Country
Columns
Longitude (generated)
Rows
Latitude (generated)
Sheet 1
Russia
Kazakhstan
Mexico
Cuba
Trinidad and Tobago
Ecuador
Brazil
Bolivia
Chile
Argentina
Rwanda
India
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
1 unknown
Data Source
Sheet 1
38 marks
1 row by 1 column

Figure 5.17:

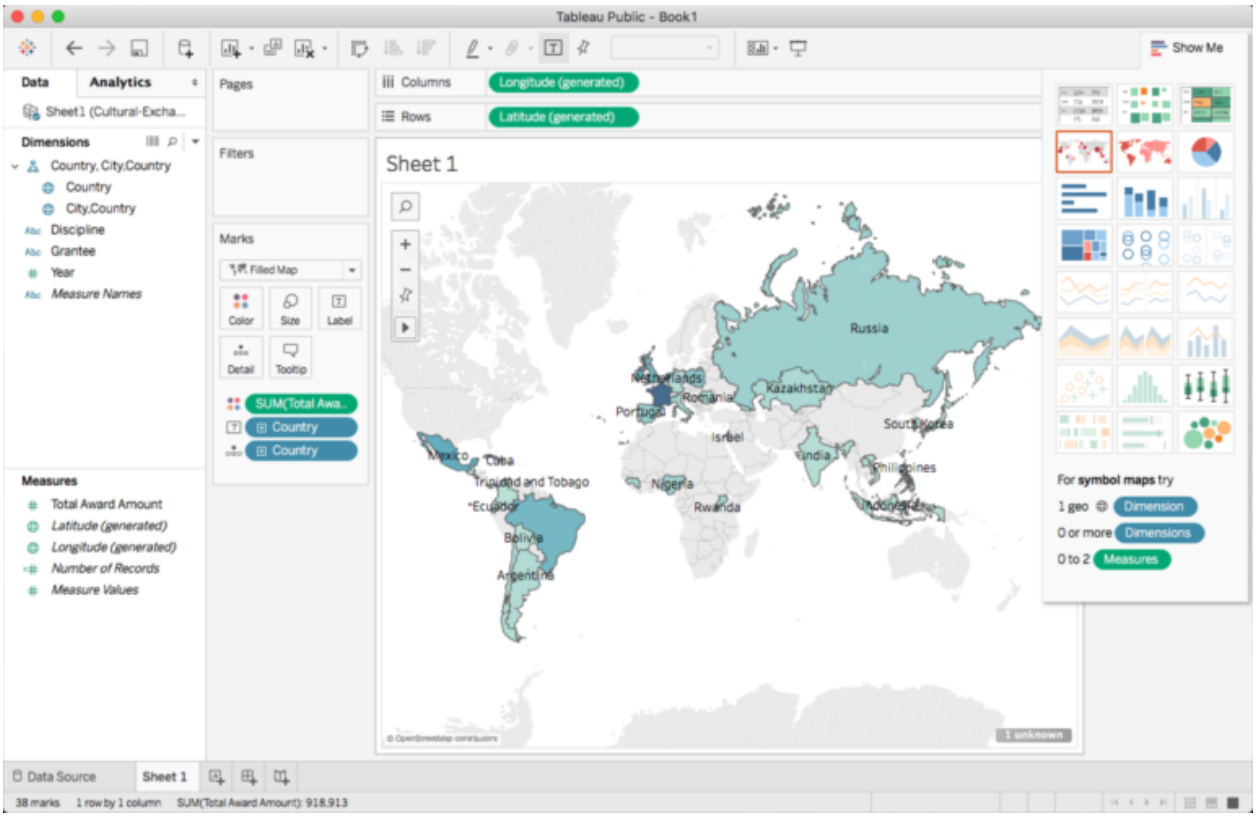


Figure 5.18:

Figure 5.19:

Step 12. Color the map

Drag 'Total Award Amount' onto 'Color' (the default colour range is gradient green, change it if you hate green!):

Step 13. Combine work sheets into a dashborad

Combining sheets in a 'New Dashboard' (icon below)

Drag from top left to 'Drop sheets here'

Optional: Share (if you have a tableau account). (based on http://dh101.humanities.ucla.edu/wp-content/uploads/2014/09/Tableau_Public_Tutorial.pdf)

5.2.3 B. Fuse tables

Sometimes (actually most of the time), the data is not stored in a single file.

For example, one file stores the ***state name*** and ***state population*** in Australia; another file stores the ***state name*** and ***state population density*** in Australia.

You want to explore the relationship between ***population*** and ***population density*** across different ***states*** in Australia.

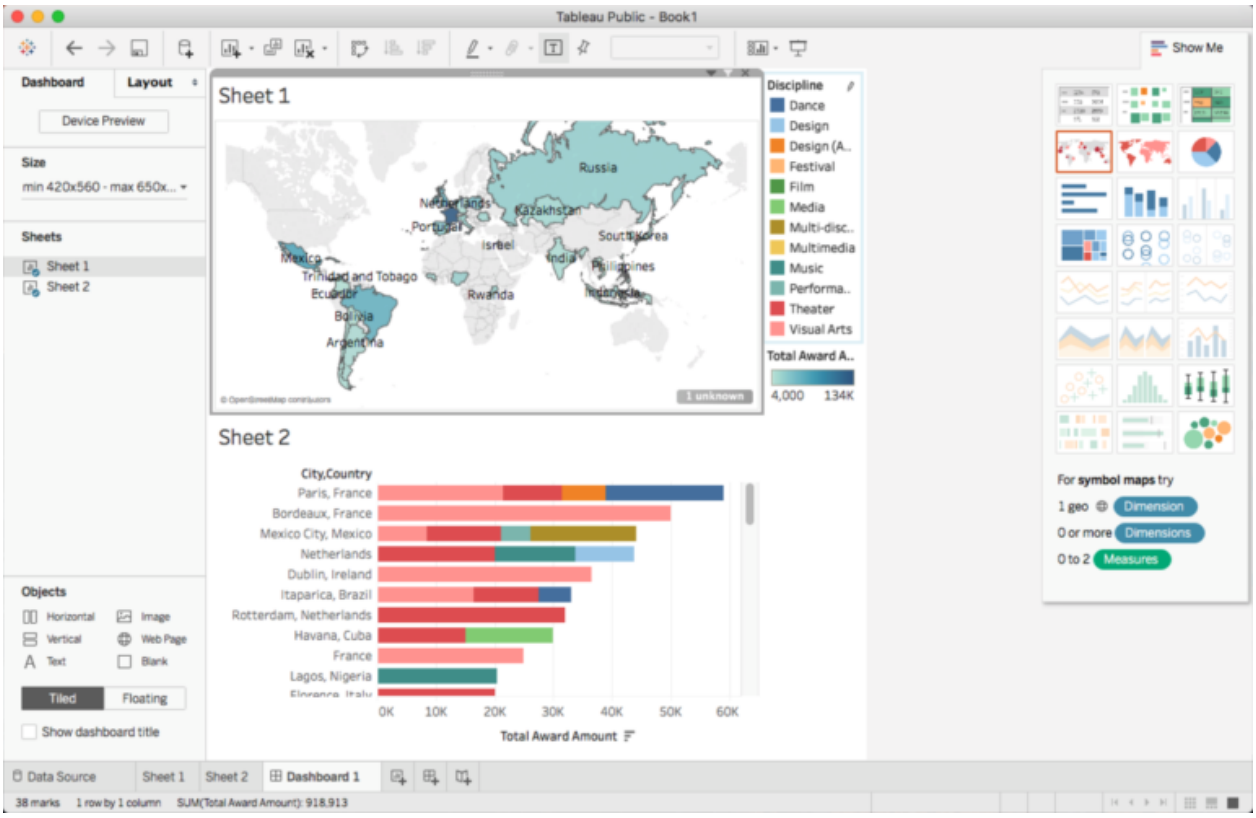


Tableau Public - Book1
Show Me
Dashboard
Layout
Device Preview
Size
min 420x560 - max 650x...
Sheets
Sheet 1
Sheet 2
Objects
Horizontal
Vertical
Text
Image
Web Page
Blank
Tiled
Floating
Show dashboard title
Sheet 1
Russia
Kazakhstan
Portugal
Israel
South Korea
India
Mexico
Trinidad and Tobago
Ecuador
Bolivia
Rwanda
Discipline
Dance
Design
Design (A..
Festival
Film
Media
Multi-disc..
Multimedia
Music
Performa..
Theater
Visual Arts
Total Award A..
4,000
134K
1 unknown
Sheet 2
City,Country
Paris, France
Bordeaux, France
Mexico City, Mexico
Netherlands
Dublin, Ireland
Itaparica, Brazil
Rotterdam, Netherlands
Havana, Cuba
France
Lagos, Nigeria
0K
10K
20K
30K
40K
50K
60K
Total Award Amount
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
Data Source
Sheet 1
Sheet 2
Dashboard 1
38 marks
1 row by 1 column
SUM(Total Award Amount): 918,913

Figure 5.20:

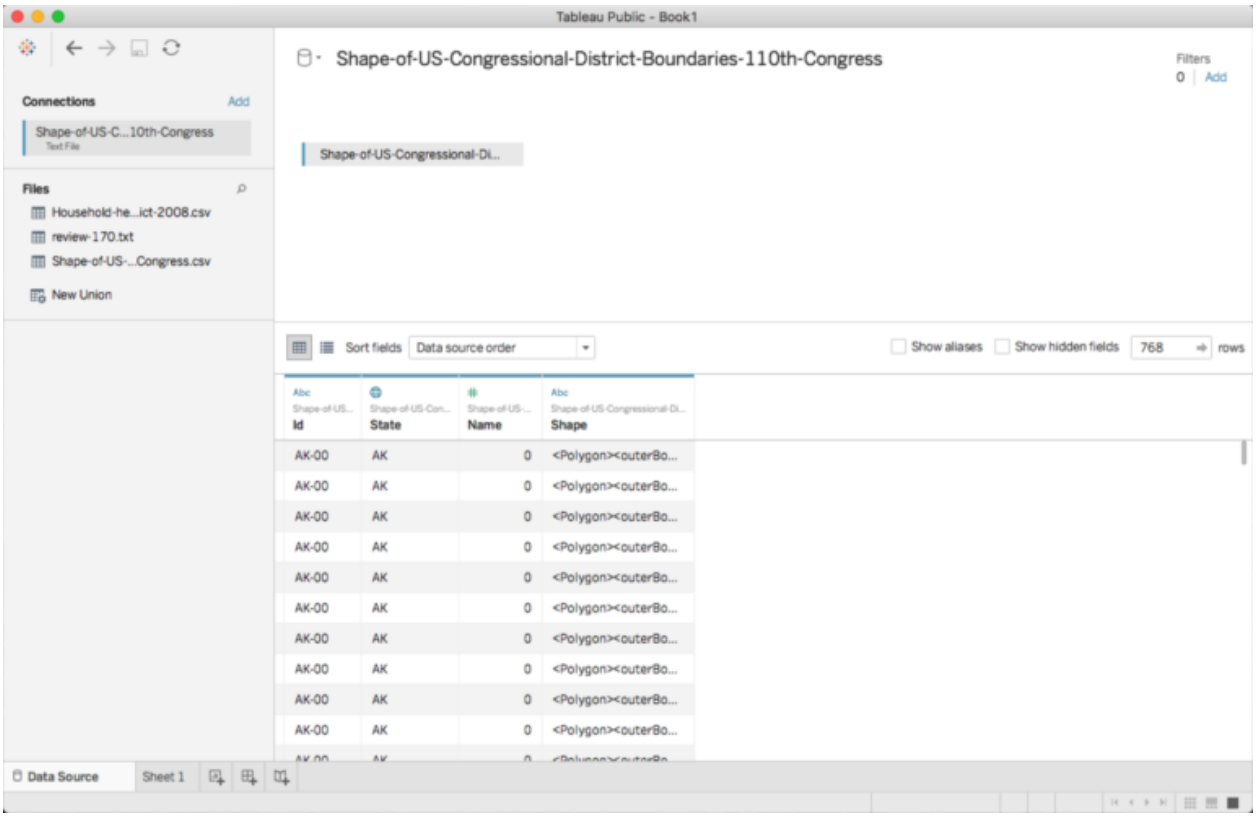


Figure 5.21:

For this very simple example, you could simply open Excel and copy and paste.

However, for real, large data, it takes ages to do so, as you have to link them correctly record (row) by record (row).

TP provides the functionality to fuse multiple data sets by linking records (rows) through the "linking property" you specify for each data set.

Now we are going to introduce the procedures.

Download the example data sets, they are 'Shape of US Congressional District Boundaries, 110th Congress' and 'Household heating by Congressional District – 2008'.

Step 1. Load the first data set

Open TP first.

Drag the first data set "Shape-of-US-Congressional-District-Boundaries-110th-Congress.csv" and drop it into the TP windows.

Step 2. Load the second data set

Drag the second data set "Household-heating-by-Congressional-District-2008.csv" and drop it into the previous TP windows.

Step 3. Configure the fuse

You need to specify the way you want to fuse the data sets.

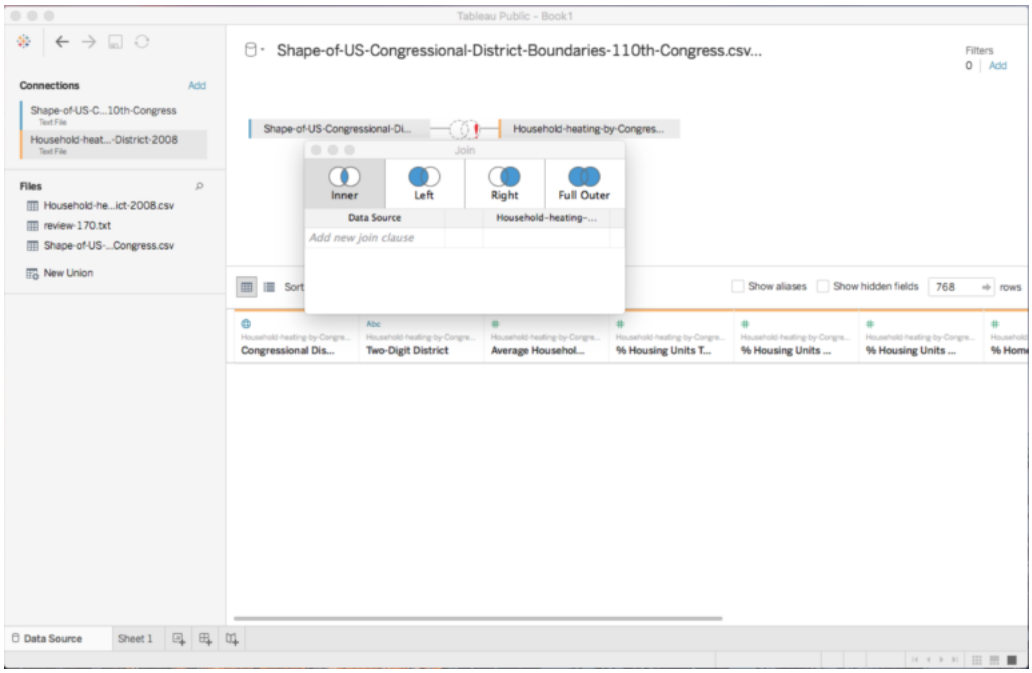


Figure 5.22:

- left data: first data set, Shape-of-US-Congressional-District-Boundaries-110th-Congress.csv and
- right data: second data set, Household-heating-by-Congressional-District-2008.csv

The 4 different ways to fuse are (TP shows them with visulisations – Venn diagram!):

- Inner, only fuse the rows (with the specified property) where the same value exists in both the left data and right data, drop the rows where the value only exists in one side.
- Left, fuse the rows (with the specified property) that exist in the left data, and drop the row if it is in the right data but not in the left data.
- Right, fuse the rows (with the specified property) that exist in the right data, and drop the rows if it is in the left data but not in the right data.
- Full order, fuse the rows (with the specified property) that exist in either side.

You also need to specify the linking properties for each side. In this case, let's choose

- "***id***" for the left
- "***Two-Digital District***" for the right

And "inner" model to fuse.

The fused table is shown in the "data view".

Close the above window.

And now you can create your "sheet" and use properties from both table to implement your visulisations!

Try different fuse models, and how many rows you can get in each way?

5.2.4 C. Choropleth Map

Choropleth map is a map with regions filled by different colors representing different properties.

One use case is using different color to present different type.

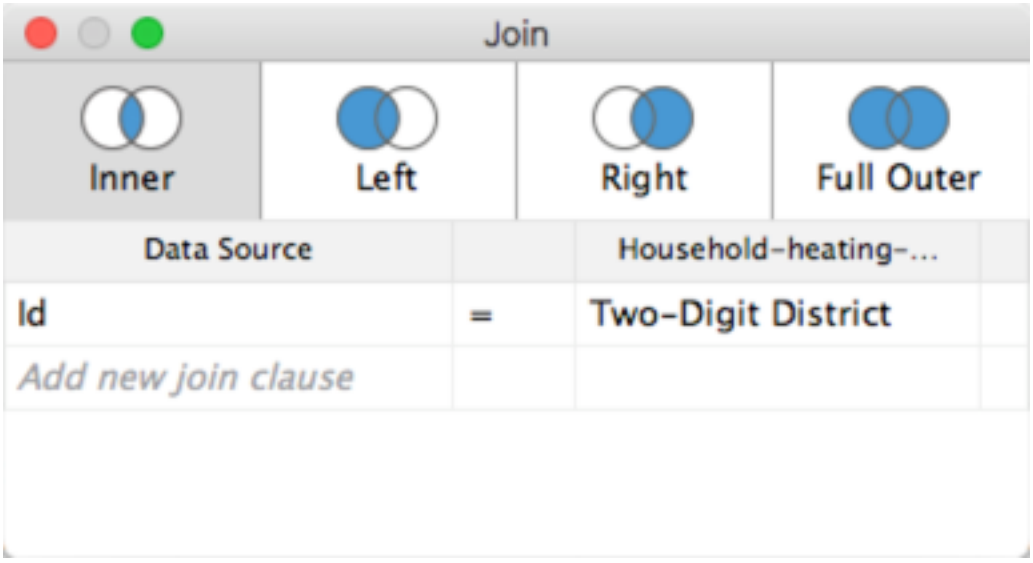


Figure 5.23:

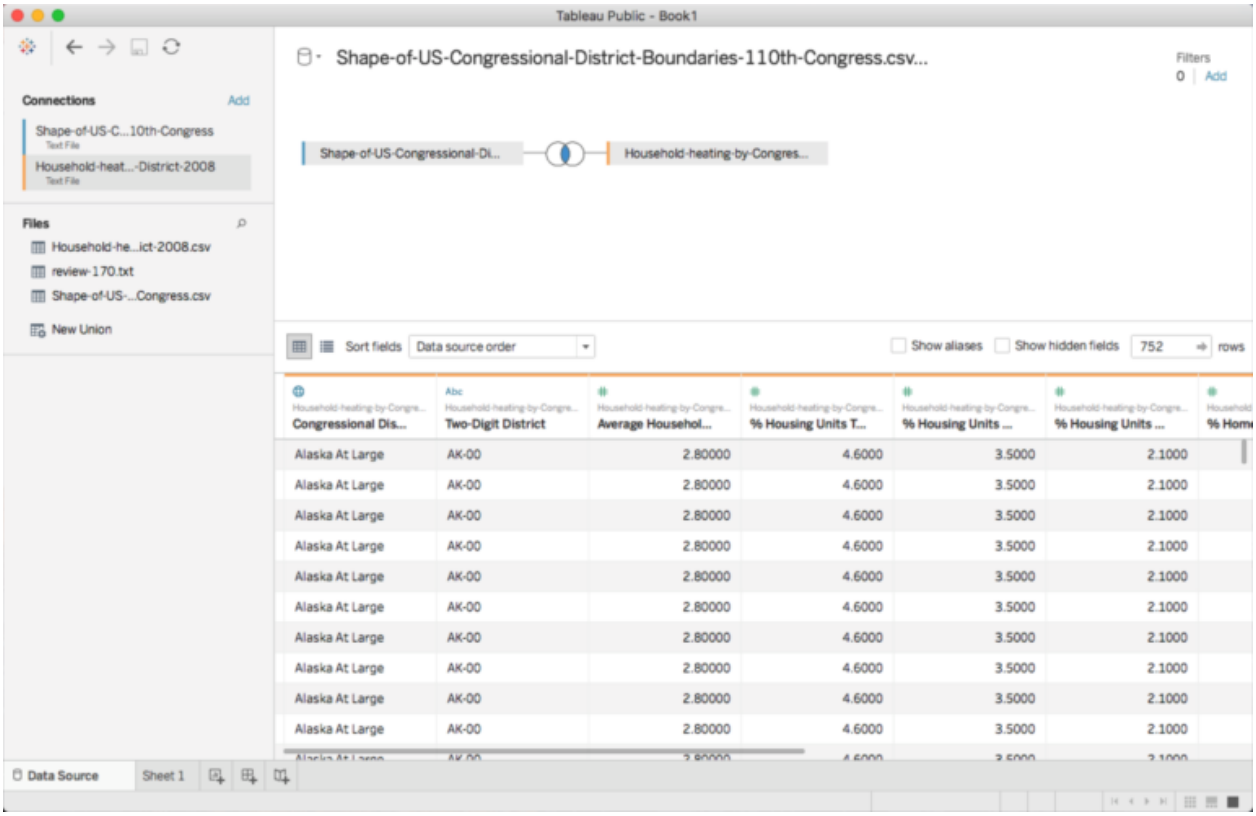


Figure 5.24:

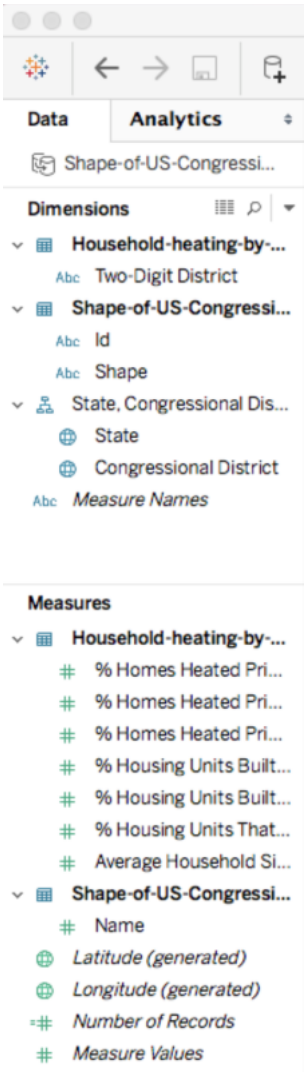

Data
Analytics
Shape-of-US-Congressi...
Dimensions
Household-heating-by-...
Two-Digit District
Shape-of-US-Congressi...
Id
Shape
State, Congressional Dis...
State
Congressional District
Measure Names
Measures
Household-heating-by-...
% Homes Heated Pri...
% Homes Heated Pri...
% Homes Heated Pri...
% Housing Units Built...
% Housing Units Built...
% Housing Units That...
Average Household Si...
Shape-of-US-Congressi...
Name
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values

Figure 5.25:

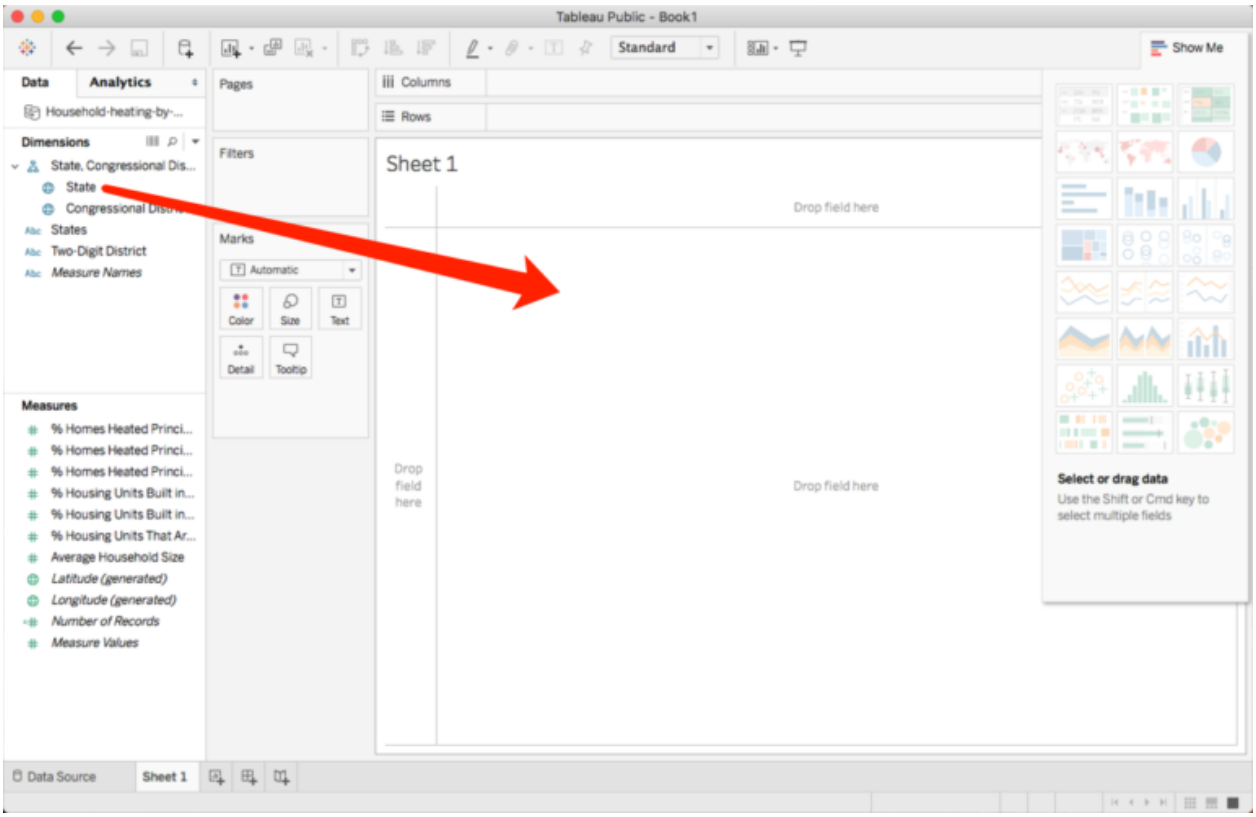


Figure 5.26:

Another is using gradient color to present quantitative data.

You will learn more details in Module 3 of FIT5147.

Step 1. Data always go first

Download the data from Household-heating-by-State-2008.csv and "drag & drop".

Make sure "State" column is marked as "geographic/locations".

Step 2. Create the map

Create a new "Sheet", and drag "States" to the center.

If you can see your map os US, congratulations! And you might choose to go to **Step 3** directly.

If your map cannot be generated properly, or you are looking at Australia, do not panic. This is not your fault, but TP has done something wrong.

Let's fix it. Click on the top toolbar: **Map => Edit Locations**

You should look at the following window:

TP locate all your data into Australia……Maybe it is because of the self-locate service on your machine.

Anyway, change it to "United States" and press "OK".

Step 3. Color the map

Drag "% Housing Units That Are Mobile Homes" to "Color" in "Marks" section.

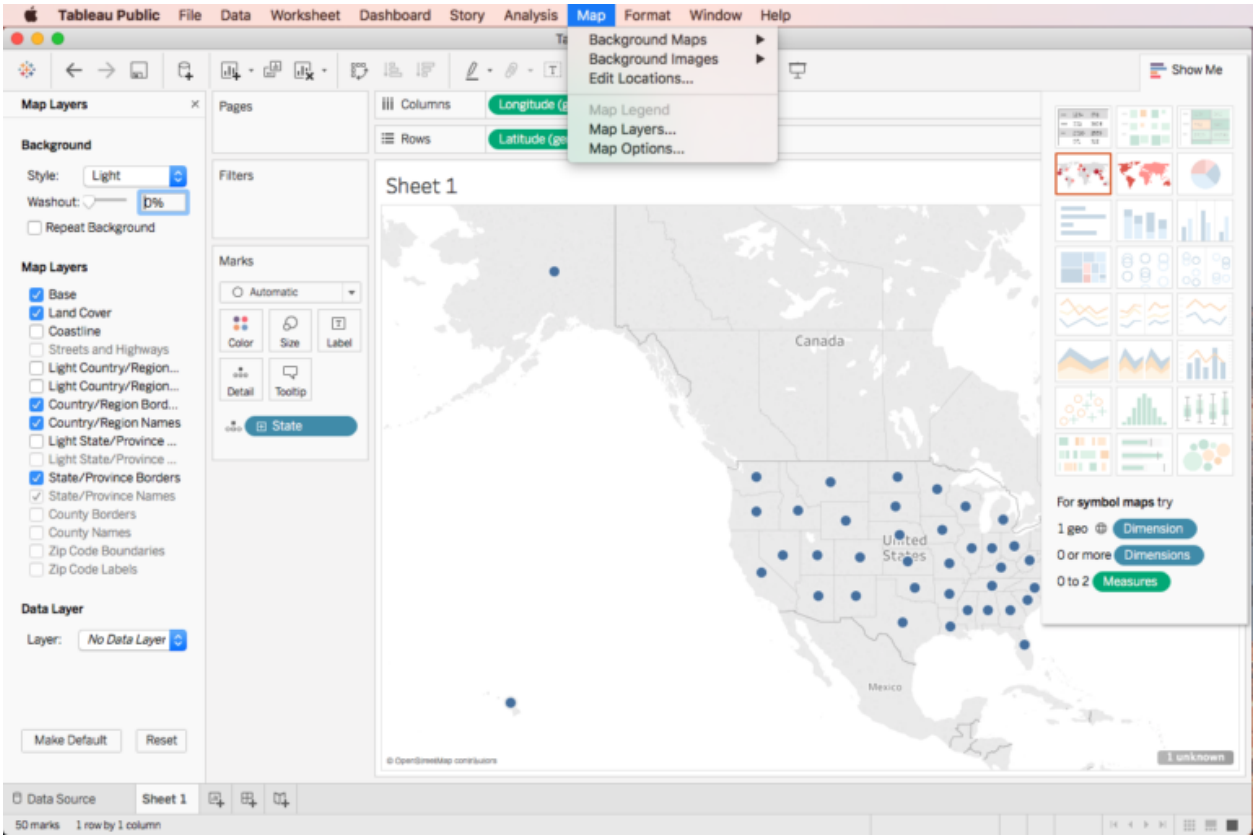


Tableau Public
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Window
Help
Background Maps
Background Images
Edit Locations...
Map Legend
Map Layers...
Map Options...
Show Me
Map Layers
Background
Style:
Light
Washout:
0%
Repeat Background
Map Layers
Base
Land Cover
Coastline
Streets and Highways
Light Country/Region...
Light Country/Region...
Country/Region Bord...
Country/Region Names
Light State/Province ...
Light State/Province ...
State/Province Borders
State/Province Names
County Borders
County Names
Zip Code Boundaries
Zip Code Labels
Data Layer
Layer:
No Data Layer
Make Default
Reset
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
State
Columns
Longitude (g
Rows
Latitude (ge
Sheet 1
Canada
United States
Mexico
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
1 unknown
Data Source
Sheet 1
50 marks
1 row by 1 column

Figure 5.27:

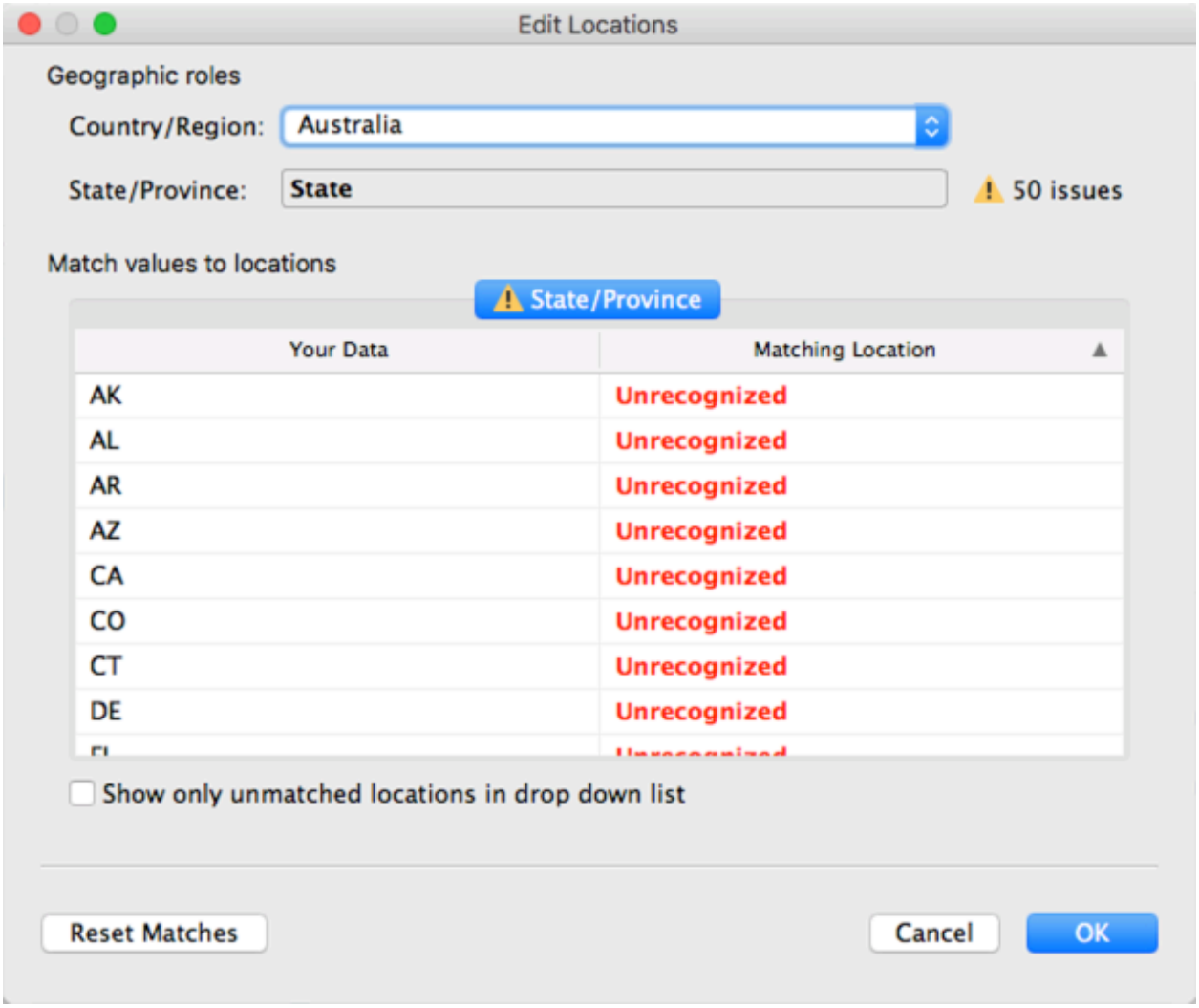

Edit Locations
Geographic roles
Country/Region:
Australia
State/Province:
State
50 issues
Match values to locations
State/Province
Your Data
Matching Location
AK
Unrecognized
AL
Unrecognized
AR
Unrecognized
AZ
Unrecognized
CA
Unrecognized
CO
Unrecognized
CT
Unrecognized
DE
Unrecognized
Show only unmatched locations in drop down list
Reset Matches
Cancel
OK

Figure 5.28:

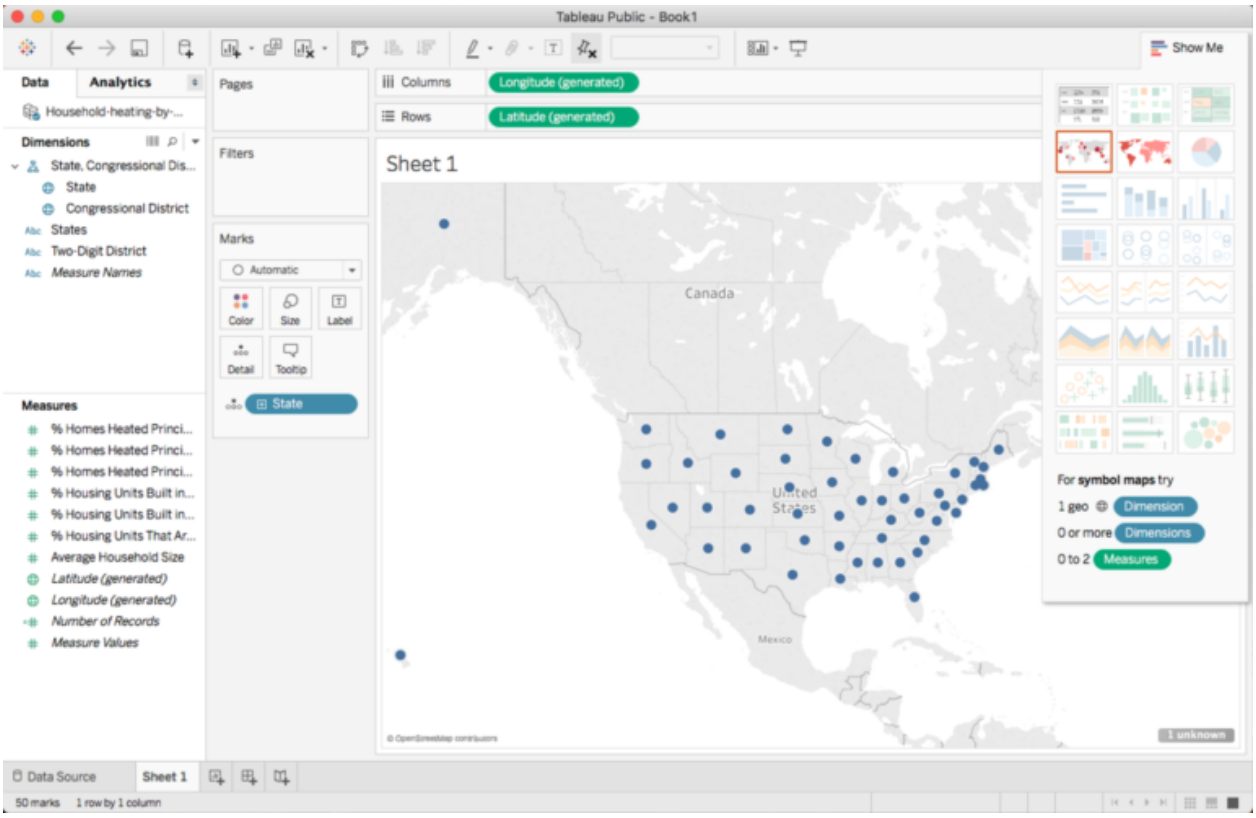


Tableau Public - Book1
Show Me
Data
Analytics
Household-heating-by-...
Dimensions
State, Congressional Dis...
State
Congressional District
States
Two-Digit District
Measure Names
Measures
% Homes Heated Princi...
% Homes Heated Princi...
% Homes Heated Princi...
% Housing Units Built in...
% Housing Units Built in...
% Housing Units That Ar...
Average Household Size
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
State
Columns
Longitude (generated)
Rows
Latitude (generated)
Sheet 1
Canada
United States
Mexico
For symbol maps try
1 geo
Dimension
0 or more
Dimensions
0 to 2
Measures
1 unknown
Data Source
Sheet 1
50 marks
1 row by 1 column

Figure 5.29:

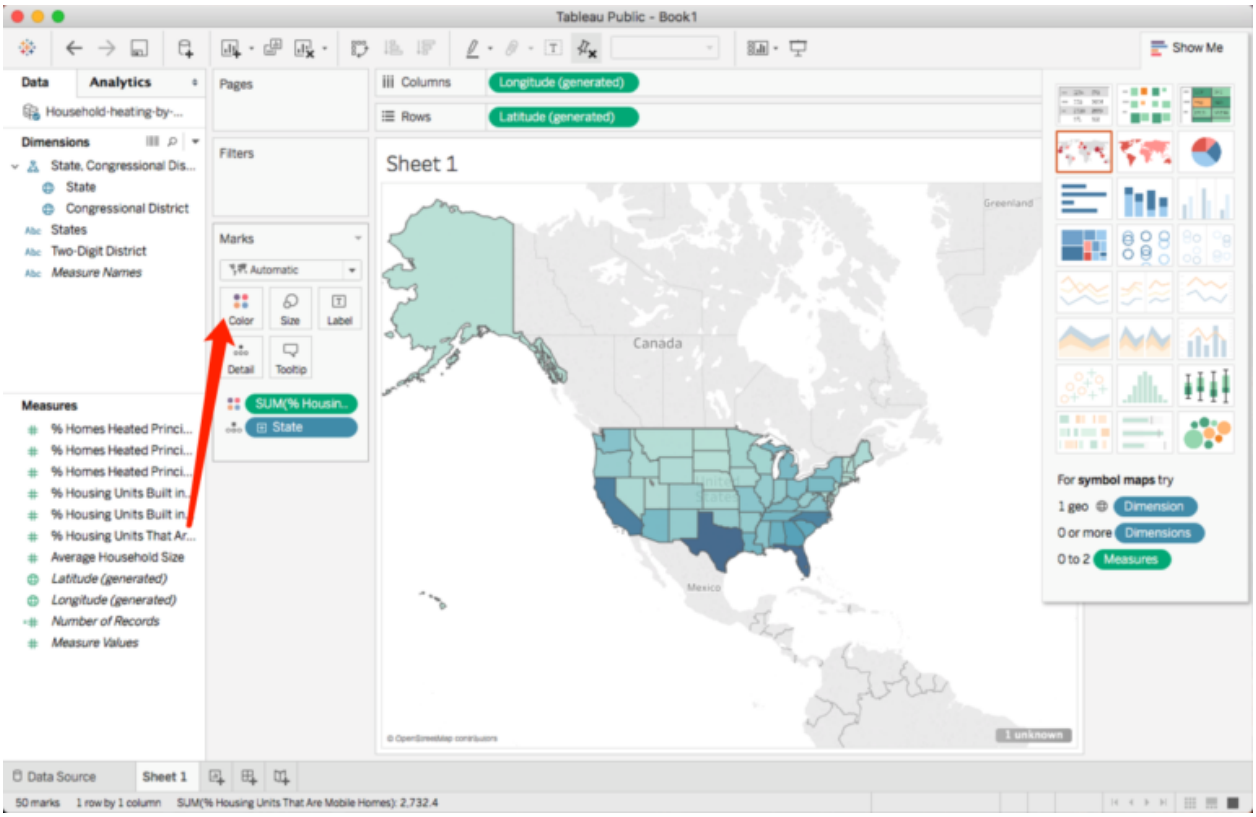


Figure 5.30:

Rename it if you think this name is so long...

Step 4. More map options

Click the top toolbar: **Map => Map Layers**

Try different options, what do they mean?

You can even add a colored data layer! You may want to remove your previous color first.

Where does the data come from?

Remember the "generated"? Investigate what can/cannot be generated?

Now its time to move on and see how to do this in R.

5.3 Activity: Exploring & Visualising Data using R

By Kimbal Marriott

Updated 28 January 2018

5.3.1 Data on Maps with R

You will need R and RStudio, you will also need to install the following libraries (and dependencies):

- ggmap

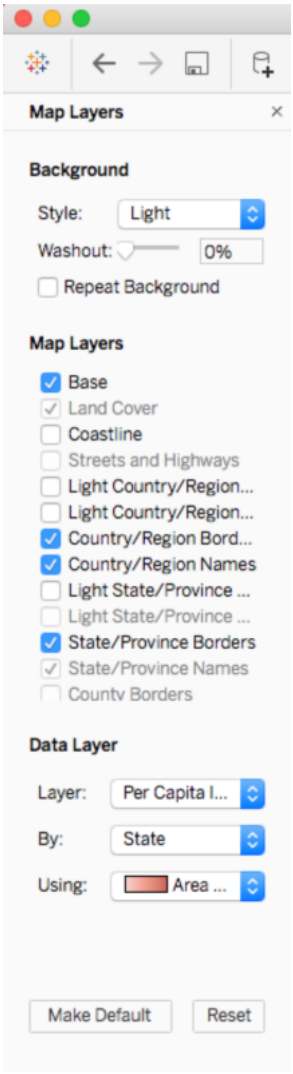

Map Layers
Background
Style:
Light
Washout:
0%
Repeat Background
Map Layers
Base
Land Cover
Coastline
Streets and Highways
Light Country/Region...
Light Country/Region...
Country/Region Bord...
Country/Region Names
Light State/Province ...
Light State/Province ...
State/Province Borders
State/Province Names
County Borders
Data Layer
Layer:
Per Capita I...
By:
State
Using:
Area ...
Make Default
Reset

Figure 5.31:

- ggplot2
- maps
- mapproj

If you are not sure how to do this and want to explore some basic knowledge about R.

DO TAKE A LOOK AT:

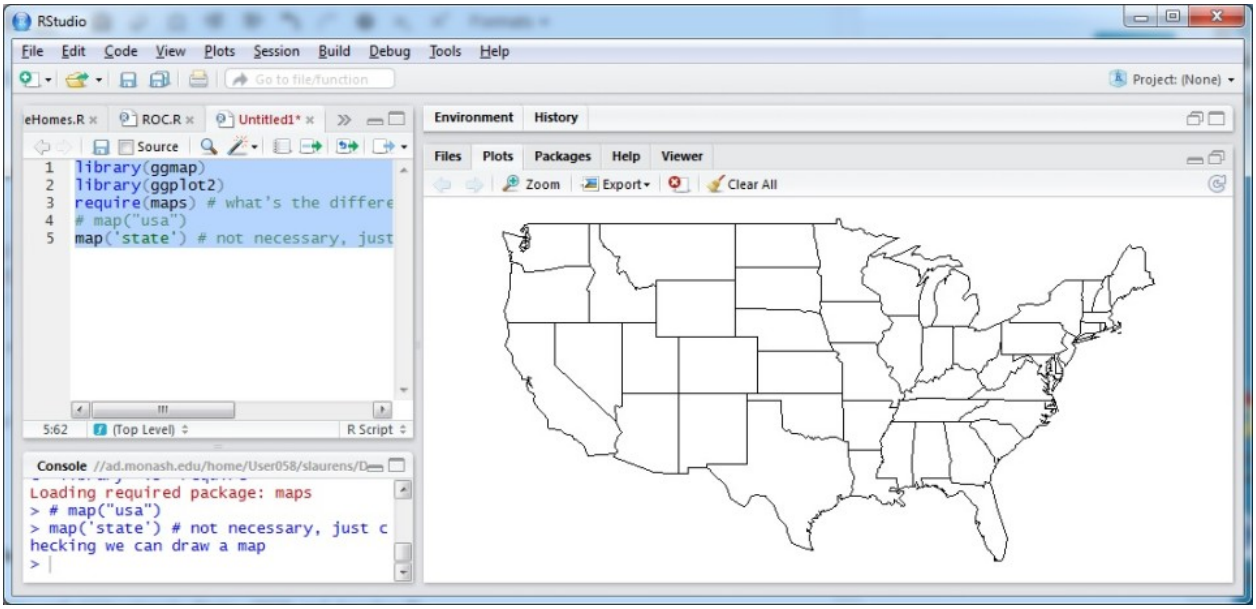
- Introduction to RStudio
- Introduction to R programming
- Torfs & Brauer's "A (Very) Short Introduction to R" [pdf]

Step 1

Run RStudio, install libraries, try the following code (copy, paste, run):

```
library(ggmap)
library(ggplot2)
library(maps)
library(mapproj)
# or you can try e.g. require(maps)
# map("usa")
map('state') # not necessary, just checking we can draw a map
```

You should see a basic map of the US states:



If an error appear as "Plot region too large".

You just need to adjust the **right bottom** section of RStudio larger.

Step 2 Read the modified data file (state names & codes have been added).

Download the file at Household-heating-by-State-2008.csv, and put it in your working directory.

```
data <- read.csv("Household-heating-by-State-2008.csv", header=T)
head(data)
names(data)
```

Step 3 Names are a bit much, simplify the one we're interested in:

```
names(data)[4] <- "MobileHomes"
names(data)
```

Step 4 Now group the Mobile Home data by State, calculating the average:

```
ag <- aggregate(MobileHomes ~ States, FUN = mean, data = data)
```

Look at the paprameters of aggregate.

- `MobileHomes ~ States` means group the Mobile Home data by State
- `FUN = mean` means calculate their averages,

How many states should there be?

```
head(ag)
dim(ag)
```

Not going to worry about the first '#N/A' state for now, delete it if you like

Step 5 Get map data (built-in map data in ggplot2 package)

```
m.usa <- map_data("state") # we want the states
head(m.usa)
dim(m.usa) # more info than we need
```

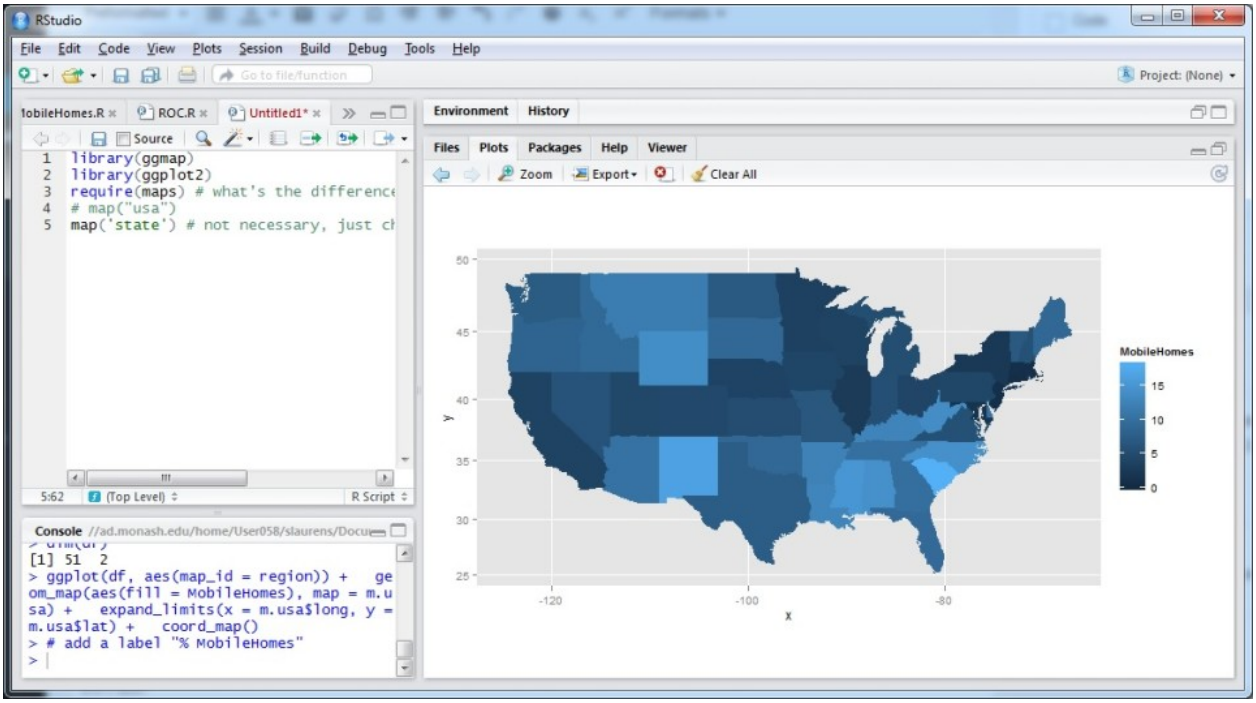
Step 6 Force our states to lowercase to match the map, keep also one column of data

```
df <- data.frame(
  region = tolower(ag$States),
  MobileHomes = ag$MobileHomes,
  stringsAsFactors = F
)
dim(df)
```

50 states?

Step 7 Now plot it on the map (look at all those layers...)

```
# Create a empty canvas
ggplot(df, aes(map_id = region)) +
  # draw the grid
  expand_limits(x = m.usa$long, y = m.usa$lat) +
  # draw a us map, fill = MobileHomes means color the map according to MobileHomes property
  geom_map(aes(fill = MobileHomes), map = m.usa) +
  # fix the ratio of the x and y axises, to match a map
  coord_map()
```



QUESTION: Compare R with Tableau Public for data visualisation – **which do you prefer and why?** Consider also the data wrangling to prepare and process the data.

5.4 Activity: Exploring & Visualising Data with D3

By Yalong Yang

Updated 6 March 2018

In this activity, you are going to use D3 to build a choropleth map.

Do not worry, it will be easier than you expect…

5.4.1 Introduction

D3. (**Data-Driven Documents**) is a HTML5/SVG + JavaScript based data visualisation toolkit. It allows you to create dynamic and interactive data visualisation that can be viewed in a web browser. D3 is a JavaScript library. JavaScript is a programming language for web pages. It is used to create web pages from dynamic content and change them in response to user interaction. Along with HTML and CSS it is one of the pillars of modern web page design.

D3 has its origins in Protovis. D3 is extremely powerful and allows you to create almost any kind of visualisation you can imagine. You can have a look at some of the examples: https://github.com/mbostock/d3/wiki/Gallery

If you look at the code you can see that this power comes at a price: D3 is quite complicated and relies on several different concepts. However, in this activity, we are not going to dive into all the details. Instead, you can have some hands-on experience of using D3. Later in the unit you will learn more about D3. Or if you cant wait there is a free comprehensive online tutorials of D3:

D3 Tutorials by Scott Murray

If you are unfamiliar with JavaScript you might like to look at the W3C tutorial on JavaScript https://www.w3schools.com/js/

5.4.2 Preparation

- Download the map file. This file contains the details of maps of different states in USA. Basically, it keeps the records of geographic shapes, like lines, points and etc. (optional) Check out http://geojson.org/ for more details.
- Download the data License: CC Attribution 4.0 International (CC BY 4.0) To be consistent, we are going to use the same data we used in our R activity. D3 (or even JavaScript) is not the ideal tool for data processing, so here we use the aggregated data which was calculated by the aggregate function in R.
- Set up a static web server For security reason, browsers cannot access local files directly, so a static web server needs to be set up to allow our D3 script to load data. The **easiest way** to do this is to download http://brackets.io/. This is a free text editor with live preview feature which runs a static web server at the back end. However you can use your favourite text editor and skip this step if you already know how to set up a static web server. (Optional)If you prefer to launch a static server, you can do this with:
  - Install Python (on Windows, Python should be installed by default for most Linux and Mac OS)
  - Adding Python to your system path (on Windows)
  - Use a terminal (command line) to enter your project directory/folder (with cd command)
  - Running the command: for python 2: `python -m SimpleHTTPServer 8000` for python 3: `python -m http.server 8000`
  - Then you can open http://localhost:8000/ in your browser to access your D3 visualisations.

5.4.3 Set up an HTML page

Using the menu File -> New to create an empty file, save it to the folder you keep all the data you downloaded, with the name us-map.html (of course change the name if you do not like it).

Let's create a basic HTML page with:

```
<!DOCTYPE html>
<html>
    <head>
        <title>US Mobile Homes choropleth map</title>
    </head>
    <body>
        <svg></svg>
    </body>
</html>
```

If you are using Brackets, click the "lightning" icon on the right to view this very new webpage.

As expected, you will see a blank page, however, if you are careful enough you can find the text on the tab of the browser is "US Mobile Homes choropleth map".

5.4.4 Initialise svg canvas

We need a canvas to draw our visualisation.

Luckily, we already defined our svg element in the HTML. (SVG is the HTML5 standard for vector graphics)

We will take advantage of D3's

- selection function to access the svg element

- attr function to modify the attributes of the svg element, and
- style function to modify the style of the svg element

First, we need to reference D3 library in our webpage, add

```
<script src="https://d3js.org/d3.v4.min.js"></script>
```

to the body section of our HTML.

Create another script section for our own JavaScript code, and initialise the svg element. The whole HTML should look like:

```
<!DOCTYPE html>
<html>
    <head>
        <title>US Mobile Homes choropleth map</title>
    </head>
    <body>
        <svg></svg>

        <script src="https://d3js.org/d3.v4.min.js"></script>

        <script>
            var width = 960;
            var height = 500;

            var svg = d3.select("svg")
                .attr("width", width)
                .attr("height", height)
                .style("border", "1px solid");
        </script>
    </body>
</html>
```

Refresh the webpage in your browser: you should now see our canvas.

Of course, if you are experienced with web development, you know this process could be done without javascript and you could simply write the SVG code

Here we want to demonstrate some basic functions of D3.

5.4.5 Reading data

We will take advantage of D3's request functions to load data. (Optional) More details at: https://github.com/d3/d3/blob/master/API.md#requests-d3-request

We can use d3.json and d3.csv function to load our map and data respectively.

```
<!DOCTYPE html>
<html>
    <head>
        <title>US Mobile Homes choropleth map</title>
    </head>
    <body>
        <svg></svg>

        <script src="https://d3js.org/d3.v4.min.js"></script>

        <script>
```

```
            var width = 960;
            var height = 500;

            var svg = d3.select("svg")
                .attr("width", width)
                .attr("height", height)
                .style("border", "1px solid");

            d3.json("us-states.txt", function(mapData){
                d3.csv("MobileHomes.csv", function(data) {

                });
            });
        </script>
    </body>
</html>
```

5.4.6 Draw the map

Let's focus on the map first.

The map file is in GeoJson format and contains pure geographic information.

To draw a map, we need to convert the geographic information (latitude, longitude) to 2D coordinates for the screen (x, y).

This process is called *map projection*, you will learn (a lot) more details in the future in this unit.

D3 provides lots of different map projections! Probably the most comprehensive collection available in Javascript. (Optional) https://github.com/d3/d3-geo-projection

We use the following code to define the map projection we are going to use later.

```
// Map projection
var projection = d3.geoAlbersUsa()
  // move the center of the map to the center of our canvas
  .translate([width / 2, height / 2])
  // scale things down so see entire US
  .scale([1000]);

// Define path generator
// path generator that will convert GeoJSON to SVG paths
var path = d3.geoPath()
  // tell path generator to use the previous map projection
  .projection(projection);
```

After this, drawing the map is extremely easy:

```
var states = svg.selectAll("path")
  // bind the geographic data to svg elements
  .data(mapData.features)
  // create one "path" svg element for each datum
  .enter().append("path")
  // using the map projection to convert geographic information to screen coordinates
  .attr("d", path)
  // change the style properties for the svg
  .style("stroke", "black")
```

```
  .style("stroke-width", "1")
  .style("fill", "white");
```

Now, refresh your webpage, say hello to the US Map.

If you cannot see anything, probably it is again a trick about working directory.

Please use the menu on the top with "**File => Open Folder**" and choose the directory you kept all your files.

Then try the "lightning" button again.

5.4.7 Color the map

We need a bit data processing first.

To make the color of different states proportional to its value, we need to know:

- the value for each state
- the min and max value among all states

This can be achieved by:

```
var country2value = {};
var minValue = Infinity;
var maxValue = -1;
data.forEach(function(d){
    var thisValue = d["MobileHomes"];
    country2value[d["States"]] = thisValue;

    minValue = Math.min(minValue, thisValue);
    maxValue = Math.max(maxValue, thisValue);
});
```

We also need a color mapping to map the values to colors.

One good possible option is to use the d3-scale-chromatic library. (Optional) https://github.com/d3/d3-scale-chromatic

Let's reference this library first:

```
<script src="https://d3js.org/d3-scale-chromatic.v1.min.js"></script>
```

Here, to make choropleth map for comparing values, we want to map the continuous values to Sequential colors.

This kind of color mapping accepts parameters of numbers from 0 to 1 and mapping them to related colors.

Usually light colors stand for small values, and dark ones for large values.

Two mappings need to be created:

- map values to the range of 0-1
- map the above outputs to colors

```
var value2range = d3.scaleLinear()
    .domain([minValue, maxValue])
    .range([0, 1])
var range2color = d3.interpolateBlues;
```

Thanks for your patience, now it is the time to finish with just a few more line of code.

```
states.style("fill", function(d){
    return range2color(value2range(country2value[d.properties.name]));
});
```

The whole web page should look like:

```
<!DOCTYPE html>
<html>
    <head>
        <title>US Mobile Homes choropleth map</title>
    </head>
    <body>
        <svg></svg>

        <script src="https://d3js.org/d3.v4.min.js"></script>
        <script src="https://d3js.org/d3-scale-chromatic.v1.min.js"></script>

        <script>
            var width = 960;
            var height = 500;

            var svg = d3.select("svg")
                .attr("width", width)
                .attr("height", height)
                .style("border", "1px solid");

             // Map projection
            var projection = d3.geoAlbersUsa()
              // move the center of the map to the center of our canvas
              .translate([width / 2, height / 2])
              // scale things down so see entire US
              .scale([1000]);

            // Define path generator
            // path generator that will convert GeoJSON to SVG paths
            var path = d3.geoPath()
              // tell path generator to use the previous map projection
              .projection(projection);


            d3.json("us-states.txt", function(mapData){

                var states = svg.selectAll("path")
                  // bind the geographic data to svg elements
                  .data(mapData.features)
                  // create one "path" svg element for each datum
                  .enter().append("path")
                  // using the map projection to convert geographic
                  // information to screen coordinates
                  .attr("d", path)
                  // change the style properties for the svg
                  .style("stroke", "black")
                  .style("stroke-width", "1")
                  .style("fill", "white");
```

```
                d3.csv("MobileHomes.csv", function(data) {

                    var country2value = {};
                    var minValue = Infinity;
                    var maxValue = -1;
                    data.forEach(function(d){
                        var thisValue = d["MobileHomes"];
                        country2value[d["States"]] = thisValue;

                        minValue = Math.min(minValue, thisValue);
                        maxValue = Math.max(maxValue, thisValue);
                    });

                    var value2range = d3.scaleLinear()
                        .domain([minValue, maxValue])
                        .range([0, 1])

                    var range2color = d3.interpolateBlues;

                    states.style("fill", function(d){
                        return range2color(
                          value2range(country2value[d.properties.name])
                        );
                    });
                });
            });
        </script>
    </body>
</html>
```

QUESTION: Compare D3 with R and Tableau Public for data visualisation – which do you prefer and why?

Chapter 6

Designing Data Visualisations

By Kimbal Marriott

Updated 24 February 2017

One of the most widely used frameworks for understanding the design and evaluation of data visualisations was developed by Tamara Munzner. This has three parts:

- **What** is the kind of data to be visualised?
- **Why** is the data being visualised–what task does the user wish to perform?
- **How** is the data visually represented and what interaction is provided

Questions come from application and do not care about the ***how*** (at first).

6.1 What: Kinds of data

Different sorts of visualisation are appropriate for different kinds of data. I'll use the following classification for different kinds of datasets and different kinds of data. In this unit we will be mainly interested in four kinds of dataset:

- *Tabular dataset*: By this I mean data that is conceptually organised into a table with each row corresponding to a different data point or item and each column corresponds to different attributes of the data. This is the sort of data that traditional statistics works with and that R is designed for.
- *Network dataset*: This is data that consists of nodes or items and links between these nodes which represent different kinds of abstract relationships such as 'reports-to' or 'is-married-to'. The items can contain attributes. A *hierarchy* is a kind of network dataset.
- *Spatial dataset*: This is data in which items are associated with a geographic location or region and this geographic key is a natural way in which to organise and understand the data.
- *Textual dataset*: this kind of data set consists of sequences of words and punctuation.

These are not the only kinds of way data can be organised but they are the most common kinds of dataset used in data science and data visualisation. One other way of organising data used in scientific visualisation is the *field* in which data is sampled from a continuous, conceptually infinite domain. An example of a data organised as a field is an X-ray image.

Attributes in data items are simple values that can be measured or logged. They can be

- *Categorical*: data that does not have an inherent ordering. It is often organised into a hierarchy. Examples include names.
- *Ordered*: data that can be ranked or ordered. It has two subtypes
  - *Ordinal*: data that can be ranked but for which the difference between items does not make arithmetic sense. Examples include clothes' sizes (small, medium, large) and survey response

scales such as one that allows respondents to select from a 5-point scale such as 'disagree strongly', 'disagree', 'neutral', 'agree', 'agree strongly'.

- *Quantitative*: data that has a magnitude supporting arithmetic comparison. For instance height or weight. This may be integer or a real number. Time is an important example of quantitative data. Sometimes quantitative data is split into *interval* vs *ratio* data but this distinction is usually not that important. The difference is that ratio data has a natural 0 while interval data does not. Thus length is an example of ratio data while date is an example of interval data.

Ordered data can either be *sequential*, in which case there is a minimum and maximum value, or *diverging* in which case it can be understood as two sequences going in opposite directions, e..g. like/dislike scales from -5 to 5 are diverging around the neutral value of 0. Ordered data can also be cyclic where the values wrap around. Time measurements are often cyclic, e.g. months in the year.

The data being displayed is not only the original data but maybe data, such as statistical values, computed from the original data.

6.2 Why: The tasks

When designing a visualisation it is fundamental to understand what are the high-level goals and tasks it needs to support. There have been many classifications of these.

Discovery: The fundamental task that an analyst wants to achieve is to derive insight or knowledge from the data. The desired outcome might be to formulate a hypothesis ("discover the unexpected") or to test a hypothesis ("confirm the expected"). The hypotheses may take the form of cause-and-effect relationships, trends, correlations or clusters.

Presentation: This goal refers to presenting insight or knowledge that has already been found to some intended audience.

Enjoy: This goal refers to creating data visualisations that are intended to entice casual users to engage with the data. This might be for instance an attractive infographic or a visualisation like Name Voyager which shows the evolution of children's names.

We shall focus on the discovery task. This relies on performing a series of analytical tasks:

- *Search* for elements that satisfy certain properties, if they exist. This might be locating a known data point, filtering the data, or finding outliers.
- *Identify* the properties of a single data item
- *Compare* or *rank* elements
- *Visually* identify patterns in some subset of elements. Examples include trends, correlations, clusters or categories.
- Calculate *derived* properties not originally in the data. These may be data transformations, data aggregations or may be statistical properties such as regression lines or clusters

Analytical tasks rely on presentation tasks:

- *Visualise* by mapping elements and their attributes to visual variables to create a view
- *Manipulate (or configure)* a view by navigating and selecting subsets of elements

There are other user tasks that support analytical tasks

- *Annotate* visual elements with text or graphical elements
- *Record* visualisation elements so that they can be preserved and accessed outside the analytics tool. For example, Tableau records a graphical history, with snapshots showing how the current visualisation was reached
- *Revisit* an earlier visualisation or relocate an element or pattern that was previously found by the analyst

6.3 How: The vis idioms

Once you know **what** you wish to visualise and **why** you want to visualise, you need to decide **how** best to visualise it. Munzner divides this into:

- **(Visual) Encoding**: how data is mapped to visual and spatial variables in each view;
- **Manipulate**: how the user interacts with the visualisation and data
- **Facet**: How different views are arranged and combined
- **Reduction and Statistical Analysis**: The different ways for aggregating, filtering and statistical analysis of the data.

6.4 Summary

In this topic we have introduced Munzner's **What-Why-How** framework for understanding data visualisation design. In subsequent modules we will look in more detail at common visual encodings and analysis for different kinds of data.

REFERENCES AND FURTHER READING

This topic is based upon

Munzner, Tamara. Chapters 11-14 of *Visualization Analysis and Design.* CRC Press, 2014.

Pretorius, A. Johannes, Helen C. Purchase, and John T. Stasko. Tasks for multivariate network analysis. In *Multivariate Network Visualization*, pp. 77-95. Springer International Publishing, 2014.

Ward, Matthew O., Georges Grinstein, and Daniel Keim. Chapters 11-13 of *Interactive data visualization: foundations, techniques, and applications (2nd Ed).* CRC Press, 2015.

Further reading

- Chapters 2-3 of Munzner, 2014.

Chapter 7

Activity: Five Design-Sheet Methodology

By Kimbal Marriott

Updated 14 February 2017

It is not easy to design effective data visualisations. Like other kinds of design activities such as architecture, graphic design or software design a good approach is to generate, explore and evaluate a wide variety of different design solutions. Sketching and lo-fidelity prototyping allows the designer to explore ideas quickly without getting bogged down in technical details.

The *Five Design-Sheet Methodology (FdS)* was introduced as a way of formalising this approach to design and for teaching design to new data visualisation designers. As you might expect from the name FdS requires the designer to produce 5 design sheets. It is client focussed and relies on an initial discussion with the client to understand the requirements and then discussing design ideas (Sheets 2, 3, 4) with the client.

Preparation

Determine **what** kind of data is to be visualised and **why** it is to be visualised. Get paper, a ruler and coloured pens and pencils ready to start the design and to work out the **how**.

Sheet 1: The Ideas Sheet

This is the initial brainstorming sheet. Think about different possible ideas for the design and jot them down. Relax, no idea is bad at this stage. The aim is to fully explore the design space,

1. Generate as many **ideas** as possible, somewhere between 10 and 20. This can be done individually or in a group. Sketch the designs on paper. At this stage they will be mini-ideas, partial half-baked solutions.
2. The next step is to **filter** the mini-ideas, removing duplicates and irrelevant or impossible concepts.
3. Group the mini-ideas into **categories**. Think about if there are some missing categories.
4. **Combine & refine** the mini-ideas into more complete solutions. Think about which visualisations complement one another.
5. **Question** and reflect on what you have created. Do the designs solve the original problem? What are their advantages and disadvantages?Identify which 3 ideas will be explored in more detail in Sheets 2,3 and 4. Choose the best idea and two designs which are also good but which are totally different to it.

Sheet 2, 3 & 4: Alternative Designs

Now refine the three designs identified in Sheet 1. Produce a sheet for each design with 5 panels

1. **Information**: appropriate meta-information such as title, author and data
2. **Layout**: sketch the final visualisation or interface
3. **Operations**: list the operations the visualisation interface supports using Action -> Result pairs.
4. **Focus/Parti**: Detail the core concept in the visualisation or interface
5. **Discussion**: Discuss the advantages, disadvantages, feasibility of the design. This might be done with the client

Sheet 5: Realisation

This is the designers final design detailing the concept to be delivered. This contains enough information about the design, functionality and interactions for it to be implemented. It might include

- **Description** of the main design patterns, algorithms or data structures and references
- Underlying **maths** and calculations used in the design such as layout dimensions
- **Software** requirements, dependencies etc
- **Estimates** of cost, time etc
- Any **other** requirements or resources needed.

For more information about FdS take a look at the Design Sheet website and the paper introducing it *J.C.Roberts, C.J.Headleand, P.D.Ritsos*, "Sketching Designs Using the Five Design-Sheet Methodology", *Transactions on Visualization and Computer Graphics, IEEE, 22(1), Jan. 2016.*

Activity

Now its time to try your hand at FdS. Work in pairs.

You have been approached by Monash to design an app that helps students plan their selection of units.

What: You have access to all of the information in the handbook for units and courses including requirements for course completion, location and time of unit offerings, unit dependencies and prohibitions as well as previous SETU evaluations of the units.

Why: The app should help students select units and understand the dependencies between units.

Now use the FdS methodology to work out **How**. You can work in groups of two or three people to create the 5 sheets and then present these to the class.
