Introduction to Data Analytics

Lab Sheet 2

Ian Nabney (partly based on a tutorial presented at the Alan Turing Institute)

1 More Tableau Fundamentals

Measures are data fields that contain numerical values, with which you can perform mathematical calculations. Dimensions are all other data fields (categorical, ordinal, date, etc.). The icons that Tableau uses for data types are shown in Table 1.

When you drag a field to the columns or rows shelf the field is shown as a *blue* or *green pill*. Blue pills are discrete/nominal variables (a finite number of values) whereas green pills are continuous (in principle they contain an infinite number of values). Most dimensions are discrete, but they can be continuous (dates can be discrete or continuous). Most measures are continuous, but they can be discrete (e.g., the number of items that a customer purchased).

The types of your data dictate the types of chart that you can create.

Note how Tableau uses the terminology ‘marks’ for what we have defined in the lectures as ‘channels’ (i.e. modifiers of the appearance of the basic marks.

Icon	Type
Abc	Text (string)
📅	Date
📅🕒	Date and Time
#	Numerical
T|F	Boolean
🌐	Geographic

Table 1: Tableau’s data types (see https://help.tableau.com/current/pro/desktop/en-us/datafields_typesandroles_datatypes.htm).

2 Exercises

2.1 Date and time data.

The granularity of *Date* and *Date/time* data, and whether or not it is treated as discrete (a blue pill) or continuous (green pill) can be changed interactively to help you explore your data. To illustrate this, open the “Sample – Superstore” dataset, and create a chart that shows *Order Date* vs. *Sales*. The date defaults to discrete at year granularity, but by clicking on the ‘+’ of the Order Date pill you can make the chart finer-grained (see Figure 1).

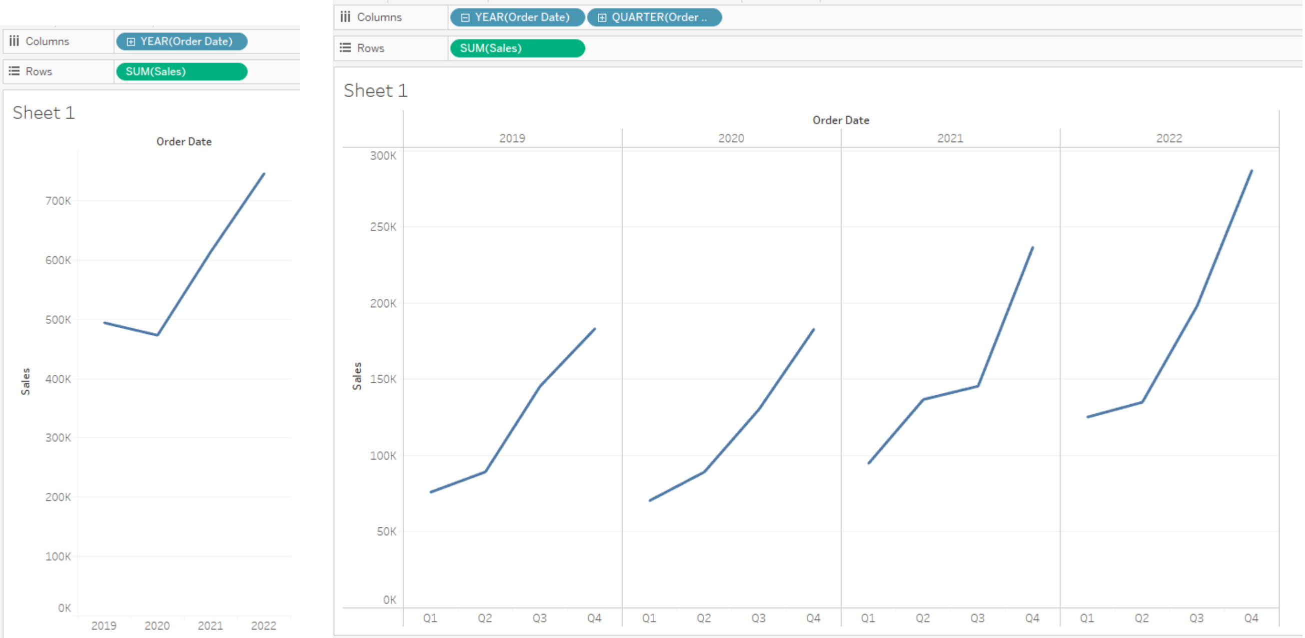


Figure 1: Sales data shown for discrete dates at the granularity of (a) a year and (b) a quarter.

Alternatively, you can select the drop-down menu of the Order Date pill and both change the granularity and display the data as continuous (see Figure 2).

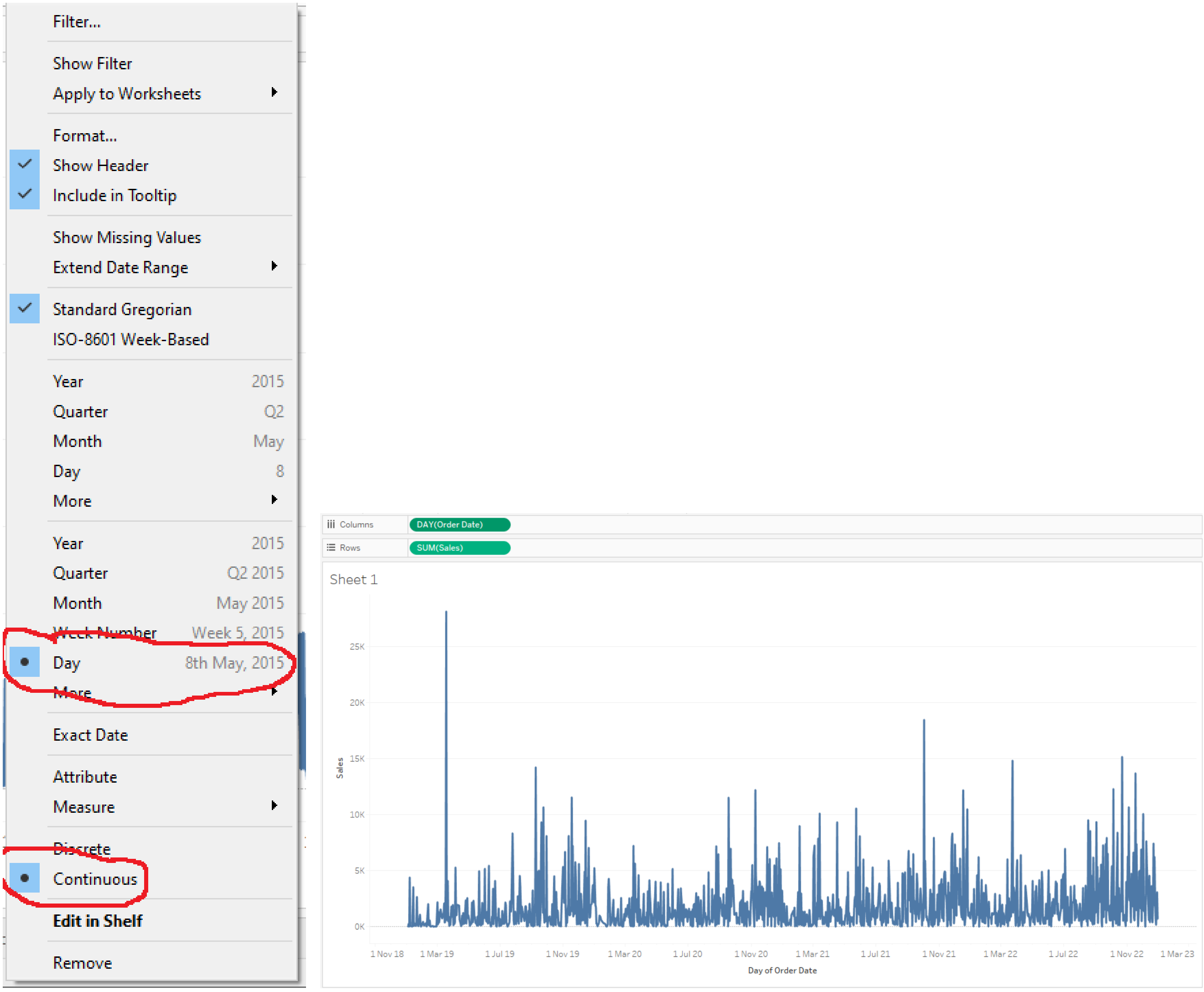


Figure 2: (a) Options for displaying date data, and (b) sales data shown for continuous dates at the granularity of a day.

2.2 Chart types

We will now discuss some more chart types and how they can be used. More will be covered in the exercises in lab 3.

Now close your workbook and open the “World Indicators” dataset.

2.2.1 Bar Chart

Create a new worksheet. Now, if you select Region vs. Internet Usage then Tableau creates a bar chart. *Show Me* uses an orange rectangle to indicate that a bar chart is the recommended chart type, but several other chart types are possible (see Figure 3).

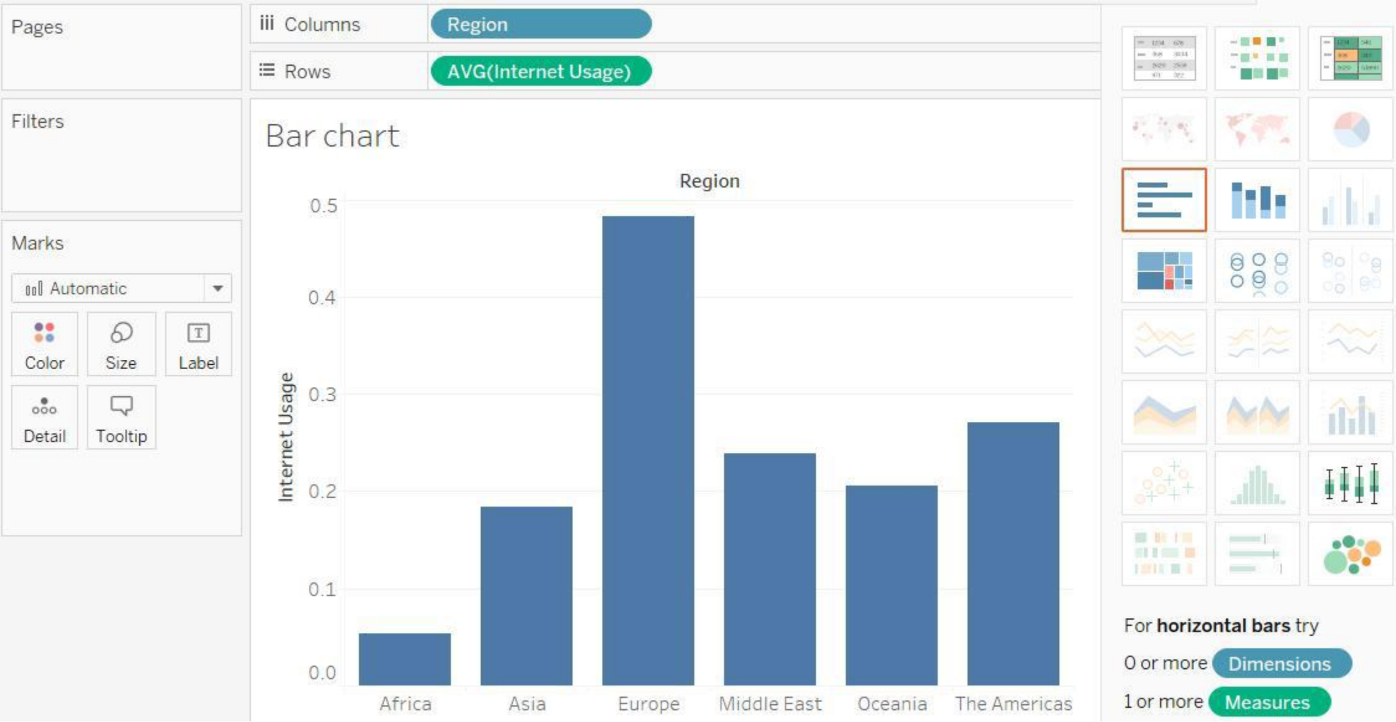
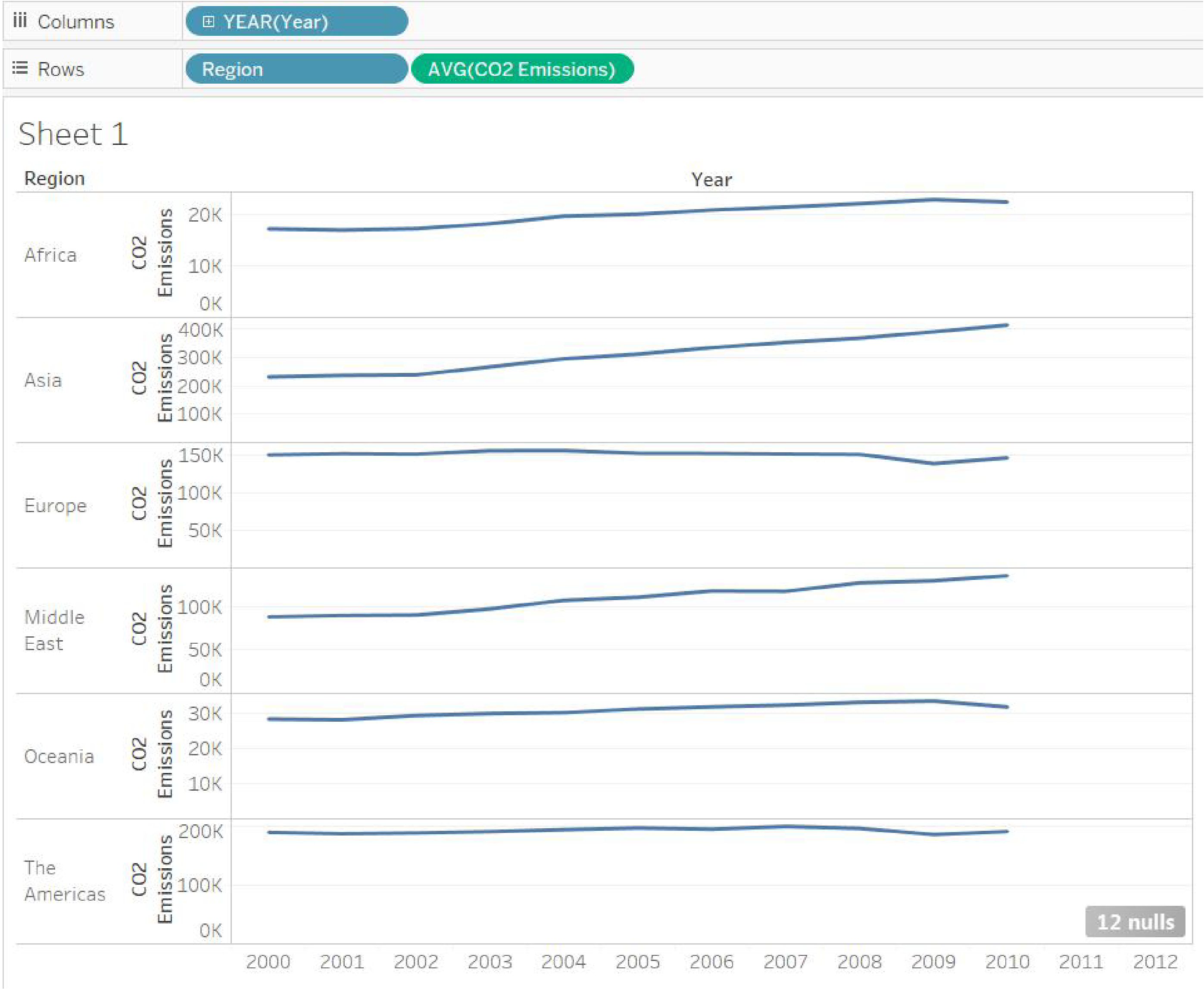


Figure 3: A bar chart, which in this case is the recommended Show Me chart type.

2.2.2 Line chart (independent axes, dual axis)

Start a new worksheet and create a line chart of *Year* vs. *Region* and average *CO2 Emissions*. Tableau creates a stack of six line charts (one for each Region) and, by default, all of the axes have the same range. That helps you to make absolute comparisons between the regions, but Africa’s emissions are so low in comparison with other regions, that you cannot see whether or not they are changing.

Double-click within the area of any of the CO2 Emissions axes to open the *Edit Axis* dialog window, and choose *Independent axis ranges for each row or column*. Now you can see that Africa’s emissions are rising, like those of Asia and the Middle East (see Figure 4).

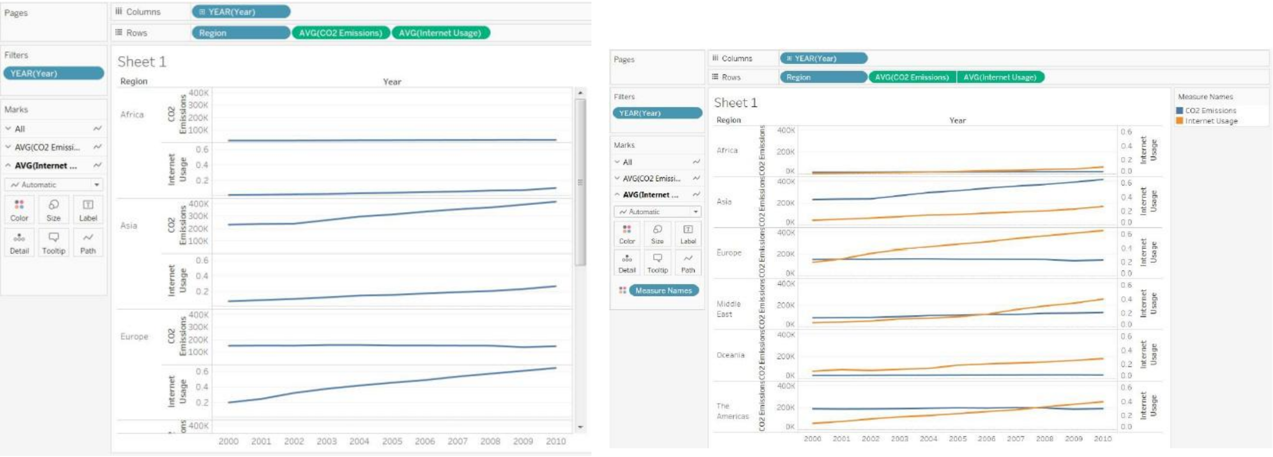


Columns: YEAR(Year)

Rows: Region, AVG(CO2 Emissions)

Figure 4: Using an independent axis range to show changes in each region's CO2 emissions.

Go back to a uniform axis range (press *Undo*) and add average *Internet Usage* to the Rows shelf. This creates a stack of two line charts for each region, which clearly displays each measure but requires a lot of display area, so you will probably have to scroll the worksheet to see all of the charts. An alternative is to display both measures in a dual-axis chart by selecting either *Dual Axis* from the Internet Usage pill or the *dual lines* chart type from *Show Me* (see Figure 5).



Columns: YEAR(Year)

Rows: Region, AVG(CO2 Emissions), AVG(Internet Usage)

Filters: YEAR(Year)

Figure 5: CO2 emissions and internet usage in each region visualised using (a) stacked charts and (b) dual-axis charts.

2.3 Maps

Tableau can create choropleth (filled) maps, and a variety of visualizations that superimpose other types of chart and networks onto a map background (see https://help.tableau.com/current/pro/desktop/en-us/maps_build.htm). All of these map visualizations require geocoded data, which is complex and can be problematic.

One critical factor is that the workbook locale must be set to English (United Kingdom) as otherwise, Tableau will not recognise UK geographic information. This can be found under the File menu. (This has caused some non-UK students problems last year, since they had Automatic (user locale) selected. If your computer is set up in another country, then the automatic locale will be based on the local country geography)

To specify that a dimension or measure is a *Geographic* data type (see Table 1), click on the dimension/measure and select the appropriate *Geographic Role*. However, **be warned** that Tableau's default handling of UK geography is incomplete/flawed:

- Metropolitan counties are missing (e.g., West Yorkshire) because Tableau uses the Metropolitan Boroughs instead (Bradford, Calderdale, Kirklees, Leeds & Wakefield).
- Tableau only recognises the first segment of a postcode (the *outcode*, e.g., LS2) not the whole postcode (e.g., LS2 9JT).

To experience these problems first-hand:

- Download the "GP surgeries in Yorkshire.xlsx" spreadsheet from Blackboard and save it in a convenient directory. Open the spreadsheet in Tableau.
- Choose the "County and Postcode" sheet
- In a worksheet, double-click on County and notice that
  - No boundaries are displayed
  - There are three labelled points on the map
- Double-click on Postcode – full and notice that nothing is filled on the map (pause while the new view is computed). All that changes is that the variable is included on details and that the tooltips on the map for each region now give a single postcode. Now remove the pill from the marks.

In Tableau, you can double-click fields to quickly add them to a view. When you double-click a geographic field specifically, it's automatically added to Detail on the Marks card, and a map view is created with marks for each location listed in the field.

Fortunately, Tableau can create maps from spatial files and people have created workarounds for some of the deficiencies of the built-in geographic knowledge.

3 Maps (again)

We are going to work again at maps in Tableau and concentrate on some of the additional challenges that UK (rather than US) data presents.

Two key principles of Tableau mapping on UK data:

1. Tableau doesn't always immediately understand when a field represents a geographical location. You may need to tell Tableau that the data is geographical, and that it relates to the UK.
2. Tableau doesn't work directly with full postcodes, e.g. SW1A 1AA; but it inherently understands the *postal sector* or *outcode*, e.g. SW1A. You can pre-transform the data, or create a calculated field to strip out only the postal sector (Calculated fields will be covered in a later worksheet).

3.1 Cities and towns

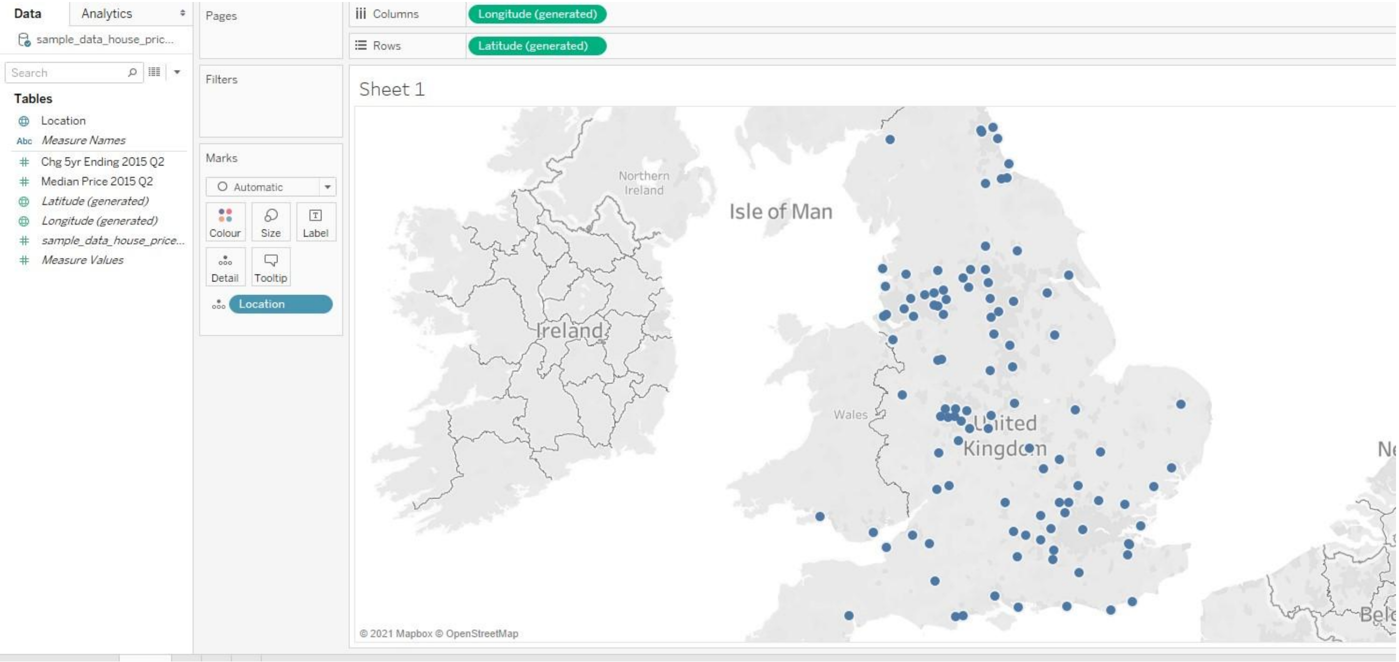
Tableau has inbuilt 'understanding' of around 750 towns and cities in the UK. It refers to these all as cities. While every self-respecting British citizen knows of course that city status is complex and historical and often requires a cathedral, the American-born Tableau doesn't care about this distinction. Cities are represented as a single location, so can be mapped as a point but not as an area. Cities in England, Scotland, Wales and Northern Ireland are all included.

Download the sample_data_house_prices dataset from Blackboard and open it in Tableau. Create a new worksheet.

The *Location* field has not been recognised by Tableau as having a geographical role; it is being treated as normal string datatype. You can see this because it is preceded by the marker Abc which denotes a text variable. If the field had originally been named ‘City’ or ‘Town’ or similar, Tableau would have identified it as geographic automatically.

So now give the *Location* field a geographical context. Right click on the field, hover over ‘geographical role’ and click on ‘City’. The icon next to the field will change to a globe, representing the geographical datatype[1].

Note that in the ShowMe palette, maps and symbol maps are the only two types of visualisation available. If you now click on the symbol map option, a simple graph of the UK is produced with the towns contained in the dataset shown as points. The longitude and latitude are generated as columns and rows (because Tableau knows the locations of all the towns in the UK), and the detail is the *Location* field (which can be seen when hovering over the points). Note that the dataset only includes some towns in the UK and two in Wales (none in Northern Ireland or Scotland).



Columns: Longitude (generated)

Rows: Latitude (generated)

Marks: Detail – Location

Figure 6: location of towns in dataset.

To start the process of understanding the price movements, drag the measure *Median Price* to the size shelf and the measure *Chg 5yr* (percentage change in house prices) to the colour shelf. Figure 8 shows the result. The cities with negative growth are now immediately obvious as the orange points. We can see that Cardiff and parts of the North West saw negative growth; the Midlands and much of the North was moderately positive, and parts of the South-East and London saw the strongest growth.

[1] If any locations are not known to Tableau, you can fix that in a number of ways. Information can be found here https://help.tableau.com/current/pro/desktop/en-gb/maps_editlocation.htm

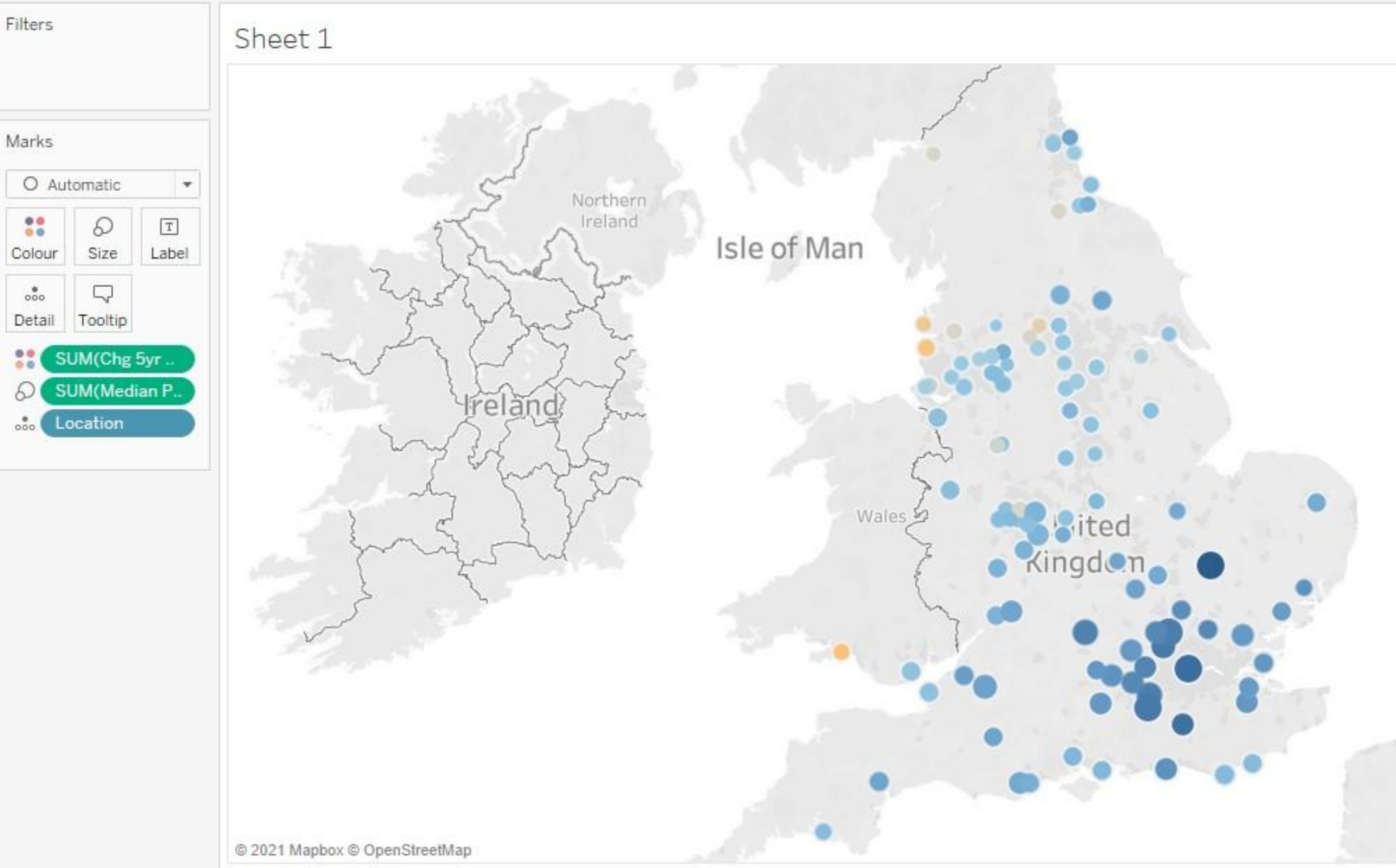


Figure 7: price as size, change as colour.

You can make the map less gray by experimenting with the background. Click on the Map menu, then 'Map Layers…' and try playing around with the layers, style and washout until you get something that appeals to you. Adding roads as a map layer often helps people identify locations more easily, and this is even more true when we are zoomed in to a single city – the network of roads typically converging on the centre or circling around it acts as a reference base. Place names can be helpful but can also be obscured by points in a busy dataset. Try to avoid crowding the background layer with too much information or too great a depth of colour, as the actual data points will lose clarity.

3.2 Postcodes

This exercise[2] uses a large extract of house price data from London. Close your old worksheets, download the Excel spreadsheet London Housing Data Extract from Blackboard and load it into Tableau (this may take up to a minute since it is a big file). Have a look at the Excel file to familiarise yourself with the variables and data types.

You should see a list of attributes and values beneath. The eighth column is labelled *Outcode* (and is of text type) while the ninth column is labelled *Postcode* (and is of geographic type). This might seem strange, since Tableau can interpret outcodes and not full postcodes[3].

Now click on Sheet 1. To make life simpler (and this exercise quicker), we will take an extract of the data. Right-click on the data icon and select *Extract data*. Leave the defaults in place and click the *Extract* button and save the file. The Tableau interface changes slightly to remind you that this is a data extract (see Figure 8).

Now we want to visualise this data. As noted above *Postcode* (and *Postcode Area*) are the only two variables with geographic type. If you double-click on *Postcode*, Tableau tries to create a meaningful map,

[2] Partially based on one developed by Tableau Tim. For his YouTube channel go to https://www.youtube.com/c/TableauTim He is a very engaging and clear demonstrator of techniques in Tableau. For example, this video https://www.youtube.com/watch?v=GGbFamljUhs explains how to change the language used in Tableau.

[3] To find out more about UK postcodes and their structure, the Wikipedia page is very informative: https://en.wikipedia.org/wiki/Postcodes_in_the_United_Kingdom This is not essential knowledge for this unit.

but all you will see is the world map with no information plotted on it. In the bottom right hand corner is a grey box stating >48K unknown. Clearly this is a failure.

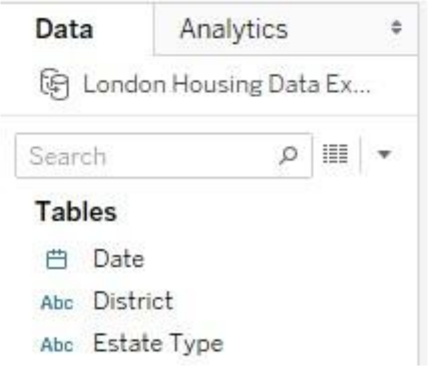


Figure 8: change in icon for data – two linked drums denote a data extract.

Click on the undo arrow and then try double-clicking the Postcode Area attribute. You should get the same result. Undo it again. If you click on Outcode and select Describe from the drop-down menu, you will see a definition of the field with some sample values: E1, E10 etc. These are the first part of the postcode designating regions within London and are the values we want to work with. Now close the pop-up window.

To fix this, we have to assign the right geographic role to the field. This role is Zip Code/Postcode (unsurprisingly) and this can be achieved through the same context menu as before. Now double-click on Outcode, and go to the drop-down menu below Marks (which reads *Automatic* at the moment) and select *Map* instead. As you move the mouse over each region on the map, a tooltip tells you the corresponding outcode[4]. The result should look something like Figure 9. Again, Tableau has automatically generated the correct longitude and latitude of the outcode because of its understanding of outcodes (all the information is stored in an internal database).

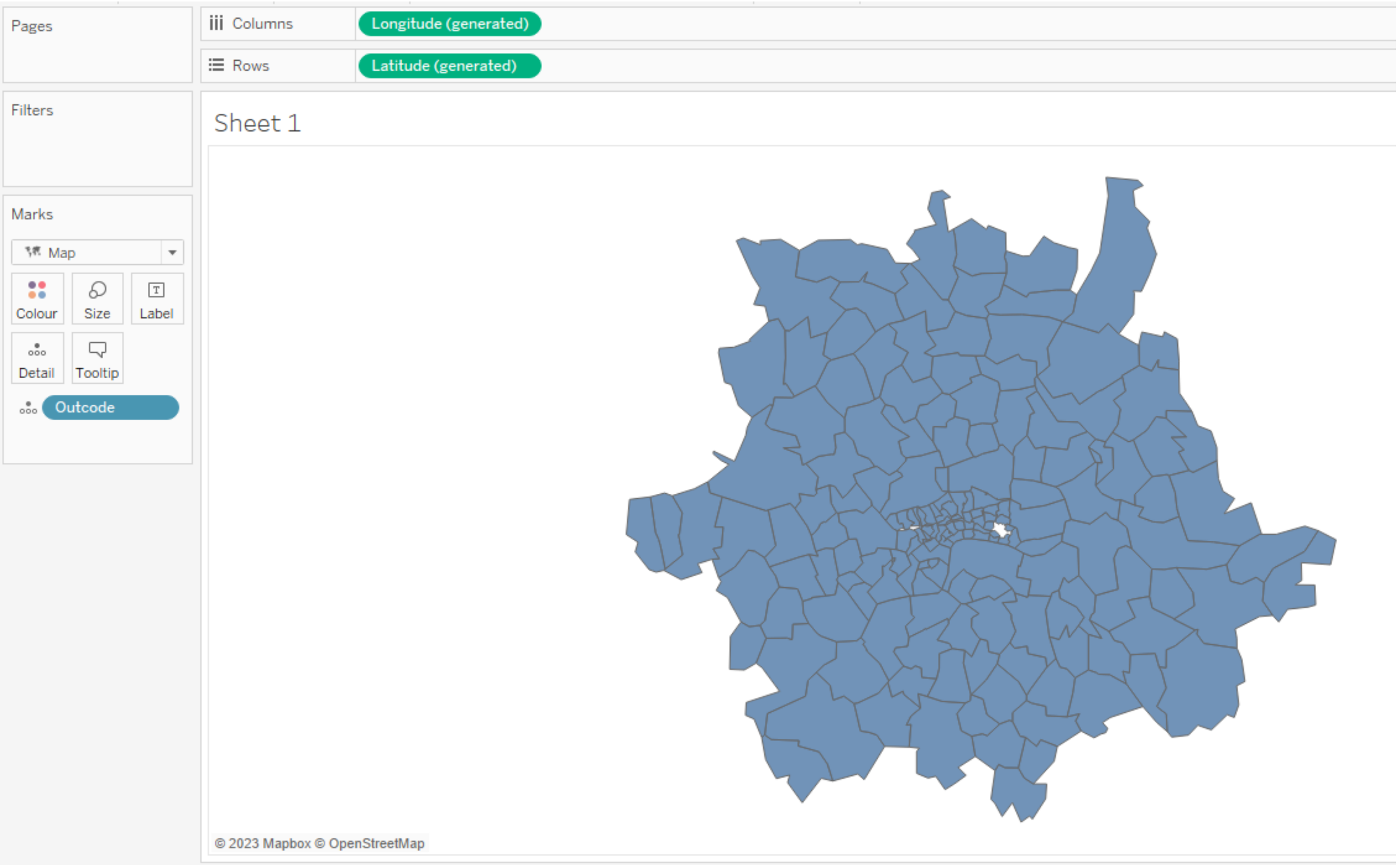


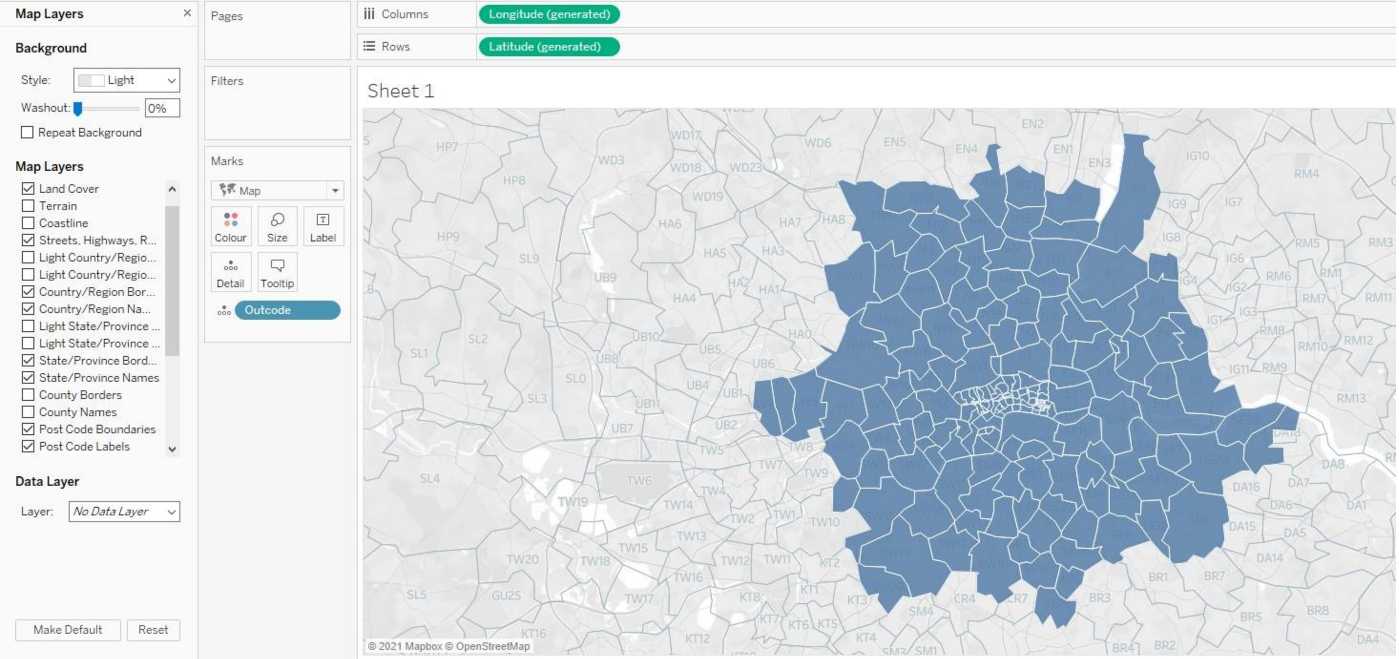
Figure 9. Choropleth of London outcodes.

Now, let’s make this map more readable. As suggested above, adding the road layer is helpful to orient people who know the city, and I suggest you also add postcode boundaries and postcode labels as this

[4] For what it is worth, I was brought up from the age of 11 in NW6. See if you can find that area on the map.

helps to see which postcodes are included in the data and which are not. Finally, set the borders to white. You do this by selecting the colour mark, clicking on the Border dropdown and then selecting a white box (or other colour if you prefer). You should see something like Figure 10 (though some of this detail may not be visible – what worked last year doesn't seem to work for me this year).

Close the Layer palette and return to the Data menu. Let's plot some data: move the variable *Price* to the Colour mark. This is OK but the default value inserted by Tableau is the sum of all the prices (in each outcode). This skews the analysis since it is dependent on the total number of properties for sale in that locality. Instead, we are more interested in the average price. Luckily, the average is one of the built-in Tableau functions that can be applied to a variable. Left click on the green pill for *Price* and go down to the *Measure* sub-menu (which shows *Measure (Sum)* since sum has been chosen. Change that to *Average* instead. It should look something like Figure 11.



Map Layers

Background

Style: Light

Washout: 0%

☐ Repeat Background

Map Layers

☑ Land Cover
☐ Terrain
☐ Coastline
☑ Streets, Highways, R...
☐ Light Country/Regio...
☐ Light Country/Regio...
☑ Country/Region Bor...
☑ Country/Region Na...
☐ Light State/Province ...
☐ Light State/Province ...
☑ State/Province Bord...
☑ State/Province Names
☐ County Borders
☐ County Names
☑ Post Code Boundaries
☑ Post Code Labels

Data Layer

Layer: No Data Layer

Columns: Longitude (generated)

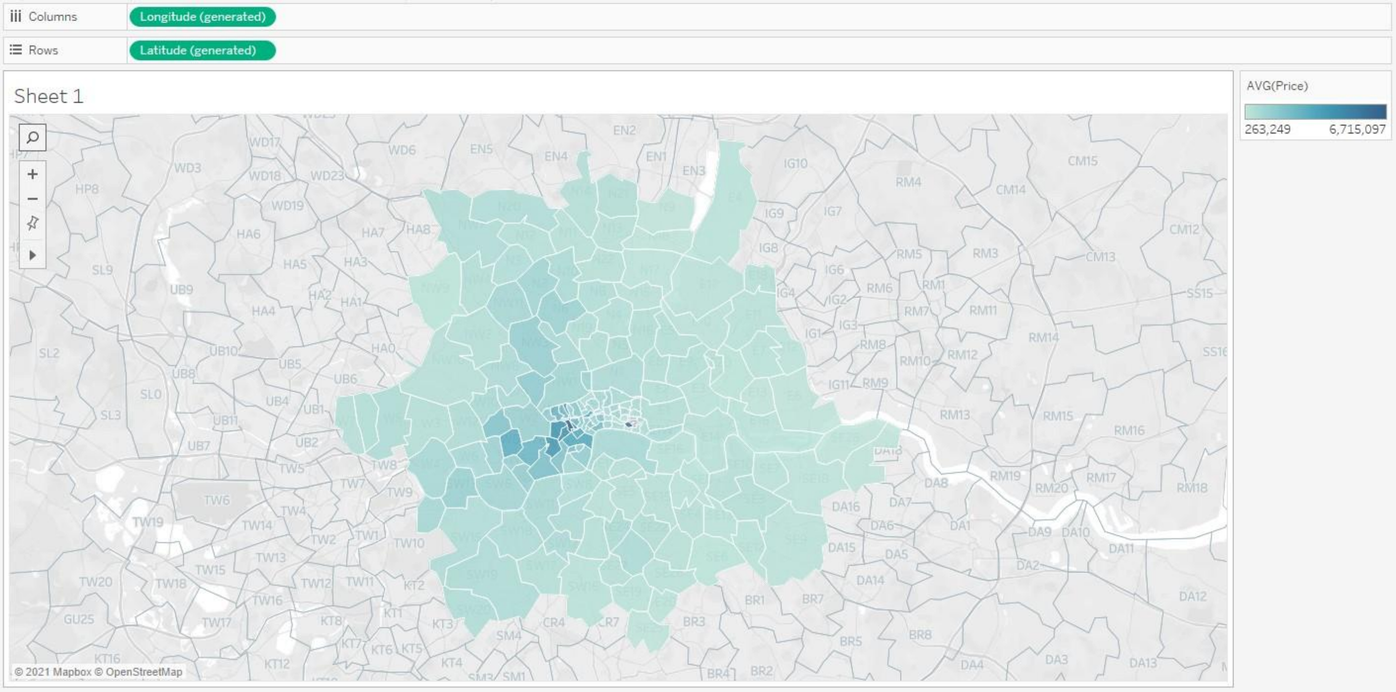
Rows: Latitude (generated)

Marks: Map

Outcode

Figure 10: London map with postcode (outcode) regions with white boundaries.

Note that the legend has an extremely large range (263,249 to 6,715,097) driven by two postcodes with exceptionally high prices (use mouse-over to find them). Doing more with this requires more sophisticated Tableau features, so will be left until later or for your experimentation.



Columns: Longitude (generated)

Rows: Latitude (generated)

Sheet 1

AVG(Price)

263,249 6,715,097

© 2021 Mapbox © OpenStreetMap

Figure 11: London map with average property price in each postcode region.

There is more to be said on the subject of UK maps (e.g. https://www.theinformationlab.co.uk/2015/06/01/uk-filled-map-geocoding-pack-for-tableau/) but this is much more complicated and will be left for later.