Visualisation with ggplot2

2031 - Statistical Computing

Sam Clifford

2019-11-15

Introduction

About this practical session

In the lecture session we introduced visualisation with the histogram, x-y plots and other scatter plot techniques, and touched on Tufte's principles of graphical excellence.

This prac will investigate the visual display of data and what makes a good and a bad graph.

- Assumed skills
  - Writing R code into a script file
  - Identifying things that are visually pleasing
- Learning objectives
  - Identifying things that are informative
  - Being able to critique a graph
  - Understanding why and how data is encoded and decoded visually
  - Understanding the subjectivity of what is aesthetically pleasing
- Professional skills
  - Creating high quality graphics

Group formation

Organise yourselves into groups of 2-3 students to collaboratively solve the following exercises.

A reminder of expectations in the prac:

- Keep a record of the work being completed with a well-commented R script
- Allow everyone a chance to participate in the learning activities, keeping disruption of other students to a minimum while still allowing for fruitful discussion
- All opinions are valued provided they do not harm others
- Everyone is expected to do the work, learning seldom occurs solely by watching someone else do work

Activity 1 - Building an attempt at a plot

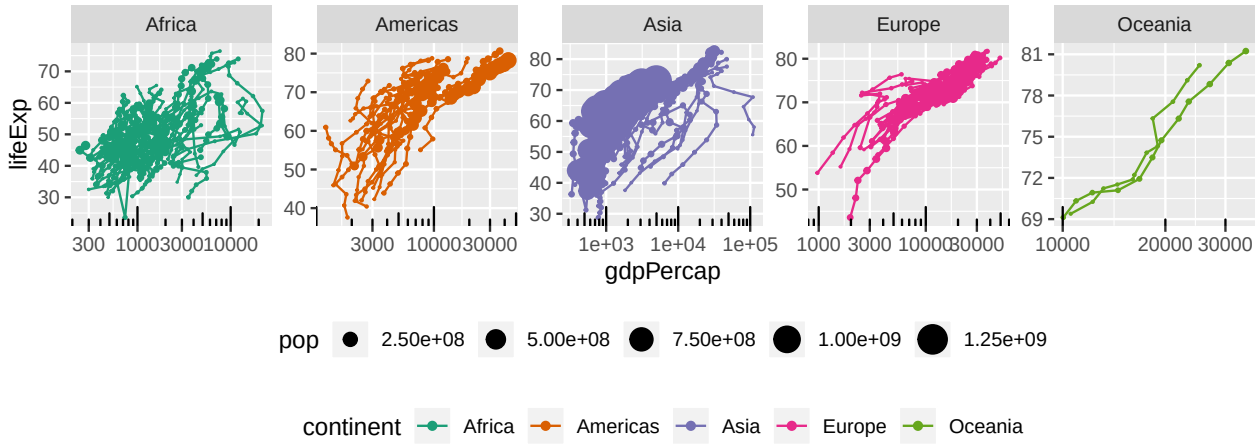
We will be looking at the gapminder data set as found in the gapminder package (Bryan 2017). This data has been collected from countries around the world and contains data on life expectancy, population and GDP per capita for 142 countries from 1952 to 2007.

Exercise: Copy and paste the code below to produce a plot showing how the relationship between GDP, life expectancy and population vary over time and continent. If you can't install the gapminder package, you can download the data from Moodle and load it with `read_csv()` from the readr package (loaded when tidyverse is loaded).

```
library(gapminder)
library(tidyverse)
data(gapminder)

ggplot(data = gapminder,
```

```
      aes(x = gdpPercap, y = lifeExp)) +
  geom_path(aes(group = country, color = continent)) +
  geom_point(aes(color = continent, size = pop)) +
  scale_color_brewer(palette = 'Dark2') + scale_x_log10() +
  annotation_logticks(sides = 'b') +
  facet_wrap( ~ continent, scales = 'free', nrow = 1) +
  scale_size_area() +
  theme(legend.position = 'bottom',
        legend.box = 'vertical')
```



Exercise: Discuss, within your group, what you think is good and bad about this plot. Does it conform to Tufte's principles of graphical excellence? Is it easy to interpret? Does it show the relationship we are interested in? List *three* important improvements that are needed for this graph to be useful.

Answer: The plot shows each country's data connected by a line so it's clear that individual country's GDP and life expectancy changes over time in a sequence, although we don't know which direction along the line we are travelling forward in time. By separating each continent out, we can see the trends geographically, such as Europe being clustered quite closely and Africa being very scattered (lots of variation across space and time, rather than a nice orderly procession from bottom left to top right). Logarithmic axis helps put the lines at approximately 45 degree angles rather than as difficult to read logarithmic curves. The large points obscure the data. Continent is mapped to both colour and facet. It's hard to make comparisons across continents because the x and y scales are different, so we have to think very hard about whether Asia is richer or poorer than Europe, on average, and we may miss that there is very little overlap in the Oceania and Africa values.

Three things which might be worth changing:

1. Same axis scaling
2. Show less data (either fewer variables or fewer continents/years)
3. Put time on the x axis

Exercise: As a group, discuss what you think each line of code in the above block does. You may wish to answer as comments in your code (everything after a # is a comment) or in a separate document.

Answer:

```
library(gapminder) # load package with data
library(tidyverse) # load package to manipulate and visualise data
data(gapminder)    # load the gapminder data

# make a plot with the gapminder data
ggplot(data = gapminder,
       # put GDP as X variable and LE as Y
```

```
      aes(x = gdpPercap, y = lifeExp)) +
  # draw the X-Y pairs as a line, ordered by appearance in data frame
  # group the lines by country and colour them by the continent
  geom_path(aes(group = country, color = continent)) +
  # draw points at each X-Y pair, coloured by continent and
  # with their size based on the POP variable
  geom_point(aes(color = continent, size = pop)) +
  # change the colour scheme from the default for any colour aesthetics
  # make a logarithmic scale on the X axis
  scale_color_brewer(palette = 'Dark2') + scale_x_log10() +
  # put logarithmic ticks on the bottom X axis showing that we don't have
  # a uniform scaling
  annotation_logticks(sides = 'b') +
  # put each continent on its own set of axes, with all plots in one row
  facet_wrap( ~ continent, scales = 'free', nrow = 1) +
  # the size of the points, based on the POP variable, should have an area
  # proportional to value of POP, rather than their radius
  scale_size_area() +
  # put the plot legend at the bottom and stack the variable keys vertically
  theme(legend.position = 'bottom',
        legend.box = 'vertical')
```

Activity 2 – Making a better graph

Based on the ideas discussed, build a graph which your group believes better shows the relationship between life expectancy and GDP. Think first about what story you want your plot to tell; are you interested in trends over space and/or time? Are you interested in a particular continent or even just one country?

You may choose to either modify the code given above or create your own graph from scratch. Make sure your code is written in your script file with appropriate comments.

Some things you may wish to consider:

- fixing up the axis labels
- a relevant title
- a different theme
- different plotting geometries
- different aesthetic options for colour, shape, etc.

You may wish to sketch the graph by hand before attempting to write the R code to generate it. This will help you and your group come to an agreement about the plot you want to make and will help the tutors understand what you're aiming for when you ask them for help.

If you get stuck, look at the ggplot2 documentation or ask a tutor.

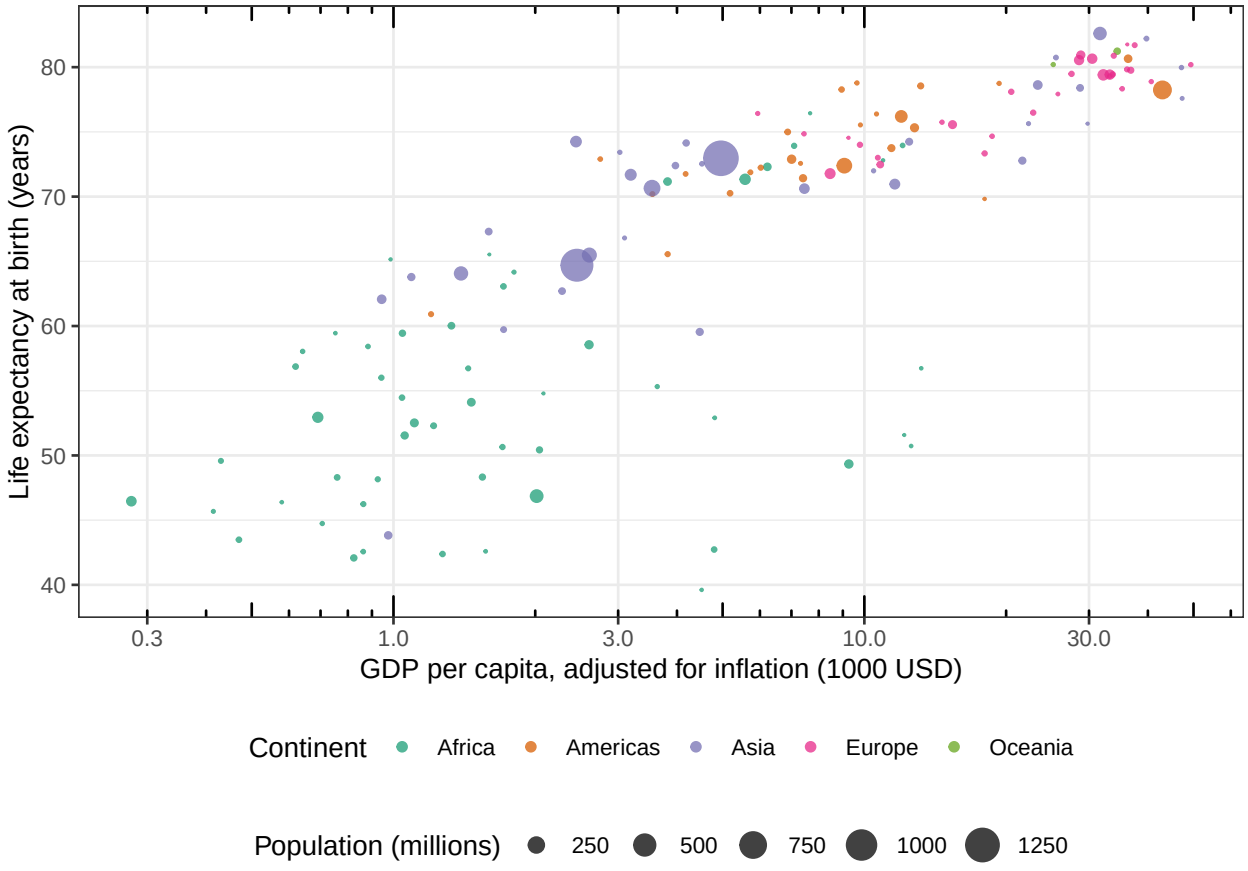
Exercise: Make a plot, save it to your computer and write comments in your code or standalone document that outline what the changes you made were and why.

Answer: There are many ways to do this, but a scatter plot which is static in time and shows the variability across space can help us avoid showing too much. Essentially we run out of things to reasonably change to show all dimensions of the data.

1. Reduce number of variables shown
2. Remove faceting, relying on colour to show difference
3. Clarity around units of variables
4. Human friendly axis labels

```
p <- ggplot(data = filter(gapminder, year == 2007),
       aes(x = gdpPercap/1000, y = lifeExp)) +
  geom_point(aes(group  = country,#
                 size   = pop*1e-6,
                 color  = continent),
             alpha = 0.75) +
  scale_color_brewer(palette = "Dark2", name = "Continent") +
  scale_x_log10() +
  scale_size_area(name = "Population (millions)") +
  theme_bw() + theme(legend.position = "bottom", legend.box = "vertical") +
  xlab("GDP per capita, adjusted for inflation (1000 USD)") +
  ylab("Life expectancy at birth (years)") +
    theme(panel.grid.minor.x = element_blank()) +
    annotation_logticks(sides = "bt")

p
```



Activity 3 – Making a worse graph

Make a new graph as in the previous activity but make it as bad as possible while still attempting to honestly show the information (i.e. don't add things to the plot which can't be derived from the variables in the plot).

Consider the principles of graphical excellence and how can we go against them to make a truly terrible plot. Think about what was bad about the plot provided earlier. Consider abusing the ability to map graphical options (e.g. color, fill, line

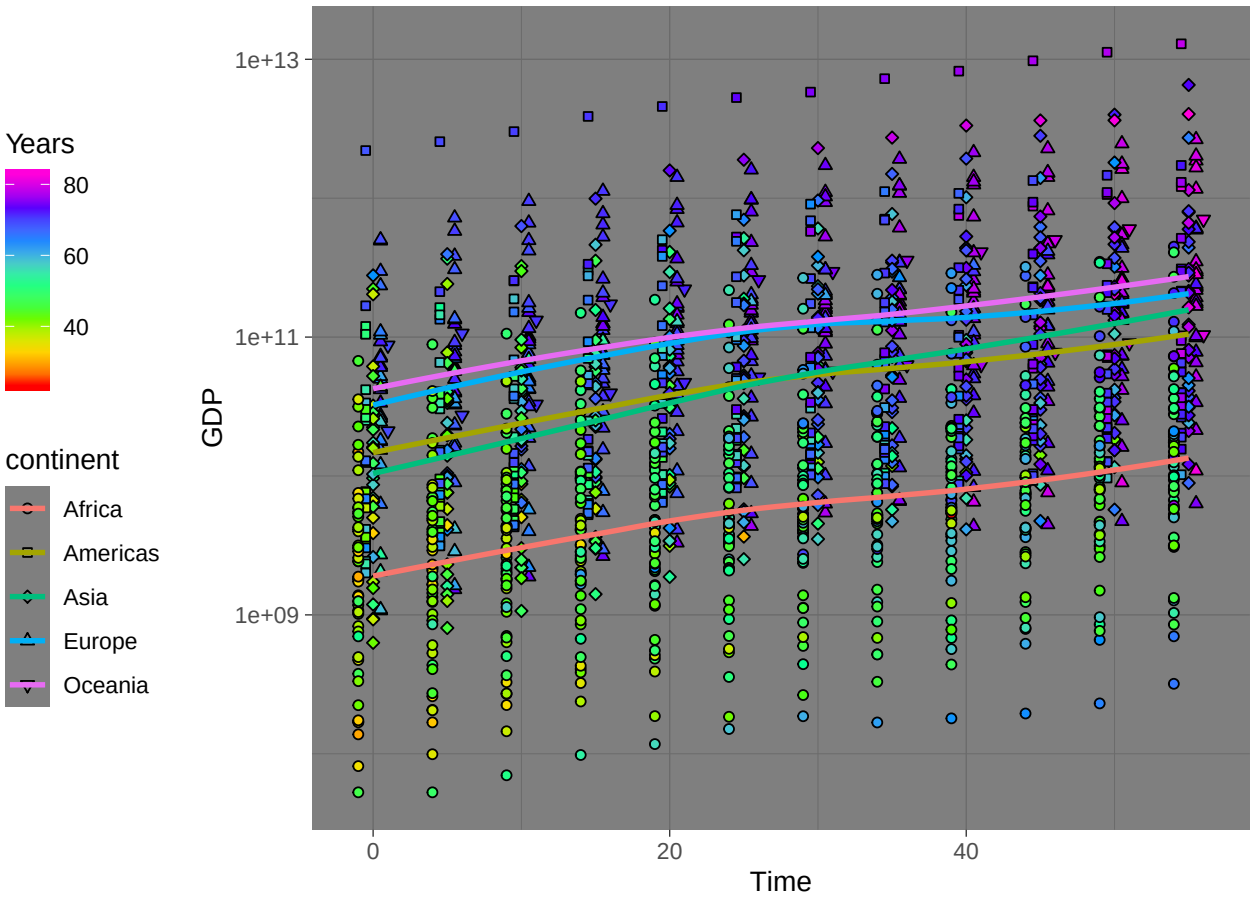
type, point size) to our variables of interest.

Exercise: Make a plot, save it to your computer and write comments in your code or standalone document that outline what the changes you made were and why.

Answer: While the graph is already quite bad, we can make it worse by removing the focus on the relationship between GDP and LE, shifting one of them out of the x and y variables.

```
p_bad <- ggplot(data = gapminder,
                aes(x = year - min(year), y = gdpPercap*pop)) +
  geom_point(aes(fill = lifeExp, shape = continent,
                 group = continent),
             position = position_dodge(width = 2.5)) +
  #facet_wrap(~country, scales = "free_y") +
  scale_y_log10() +
  scale_shape_manual(values = 21:25) +
  theme_dark() +
  theme(legend.position = "left") +
  scale_fill_gradientn(colours = rainbow(7), name = "Years") +
  ylab("GDP") +
  xlab("Time") +
  geom_smooth(aes(color = continent), se=FALSE)

p_bad
```



Here we've made the following changes:

1. Put time since first observation on x axis without units
2. Put GDP on the y axis, an interesting choice given we want to look chiefly at how life expectancy changes over time
3. Put coloured points and coloured lines on the graph, but the colours aren't related
4. Continent is mapped to both shape and line colour
5. Put life expectancy's legend label as "Years"
6. Whitespace is added on the left of the plot around the legend, a waste of space
7. No context for what GDP is (it's total national GDP, equal to per capita GDP multiplied by total population), and USD `1e13` is hard to contextualise (it should be increasing in steps of 10^3, e.g. a million, billion, trillion for the major breaks)
8. The position changes in continent help make it a little clearer that there are multiple continents within a year, but it ends up looking like measurements were taken a year earlier or later than they were.
9. The dark colour scheme is too similar in intensity to some of the blue and purple points, obscuring them from our vision
10. Chosen a rainbow colour scheme, which is notorious for having poor properties that visually focus on yellows, are not colourblind friendly and do not convert to greyscale easily on account of multiple regions having the same intensity.

Activity 4 – Group discussion

Have participants present their best (and/or worst) graph from the last activities. What did they identify as good and bad and how has each group attempted to present the relationship?

Tidy up

Make sure you save your R script, and anything else you have produced and ensure everyone in your group has a copy. Email your worst graph to Dr Sam Clifford. Leave the room in a better condition than you found it.

Further reading

A lot of the key ideas in data visualisation arose with Tufte (1983), and are summarised by Pantoliano (2012). Tufte's website is well worth exploring, particularly the discussion on how the visual presentation of information could have helped avert the *Challenger* disaster (Tufte 1997).

Some of the history of data visualisation is summarised well by Friendly (2005) and Friendly (2006).

For some more guidance on using ggplot2 for data visualisation, check Chapter 3 of Wickham and Grolemund (2017), the RStudio cheatsheets (RStudio 2012), and Chang (2017).

References

Bryan, Jennifer. 2017. *Gapminder: Data from Gapminder*. https://CRAN.R-project.org/package=gapminder.

Chang, Winston. 2017. *R Graphics Cookbook: Practical Recipes for Visualizing Data*. 2nd ed. O'Reilly Media. http://www.cookbook-r.com/Graphs/.

Friendly, M. 2005. "Milestones in the History of Data Visualization: A Case Study in Statistical Historiography." In *Classification: The Ubiquitous Challenge*, edited by C. Weihs and W. Gaul, 34–52. New York: Springer. http://www.math.yorku.ca/SCS/Papers/gfkl.pdf.

———. 2006. "A Brief History of Data Visualization." In *Handbook of Computational Statistics: Data Visualization*, edited by C. Chen, W. Härdle, and A Unwin. Vol. III. Heidelberg: Springer-Verlag. http://www.datavis.ca/papers/hbook.pdf.

Pantoliano, Mike. 2012. "Data Visualization Principles: Lessons from Tufte." 2012. https://moz.com/blog/data-visualization-principles-lessons-from-tufte.

RStudio. 2012. "RStudio Cheat Sheets." 2012. https://www.rstudio.com/resources/cheatsheets/.

Tufte, Edward. 1983. *The Visual Display of Quantitative Information*. Graphics Press.

Tufte, E. R. 1997. “Visual and Statistical Thinking.” In *Visual Explanations: Images and Quantities, Evidence and Narrative*. Graphics Press. https://www.edwardtufte.com/tufte/books_textb.

Wickham, Hadley, and Garrett Grolemund. 2017. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O’Reilly Media. http://r4ds.had.co.nz.