Advanced Quantitative Methods

Data Visualization

Instructor: Gregory Eady
Office: 18.2.10
Office hours: Fridays 13-15

Today

- Data visualization
- Implementation with `ggplot2` in R

Visualization

Why look at data?

Scatterplot

The mean and correlations in these figures are identical:

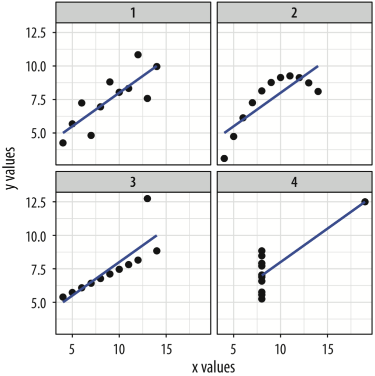


Figure 1.1: Plots of Anscombe's quartet.

Real case: Democracy & Inequality

THE EFFECT OF POLITICAL DEMOCRACY AND SOCIAL DEMOCRACY ON EQUALITY IN INDUSTRIAL SOCIETIES: A CROSS-NATIONAL COMPARISON*

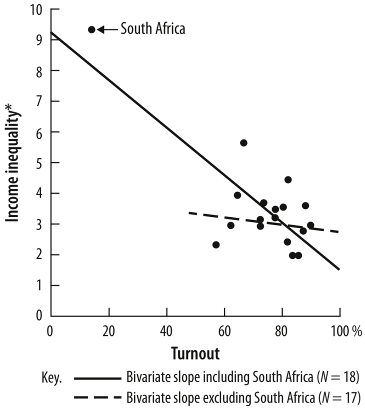
CHRISTOPHER HEWITT

University of Maryland, Baltimore County

American Sociological Review 1977, Vol. 42 (June):450–464

This paper considers the effect of political democracy on the stratification systems of non-communist industrial societies. In contrast to previous research, this study assumes that the effect of democracy will be incremental and, therefore, that the historical experience of democracy must be considered rather than the current political situation. Two hypotheses are suggested: that democracy itself will lead to equality and that only the election of socialist legislatures will lead to equality. The historical experience of democracy and that of socialist legislatures is related to five measures of inequality. It is concluded, after taking into account the level of economic development and the growth rate, that although democracy itself has little effect, the experience of democratic socialist parties is significantly related to variations in inequality. The stronger the democratic socialist parties, the more egalitarian is the contemporary class system.

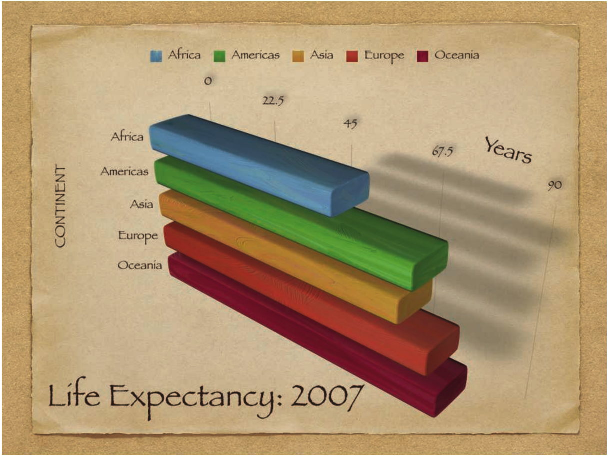
Scatterplot of turnout (democracy) and inequality



What makes a "bad" visualization?

1. Aesthetic concerns
2. Substantive concerns

Bad aesthetics



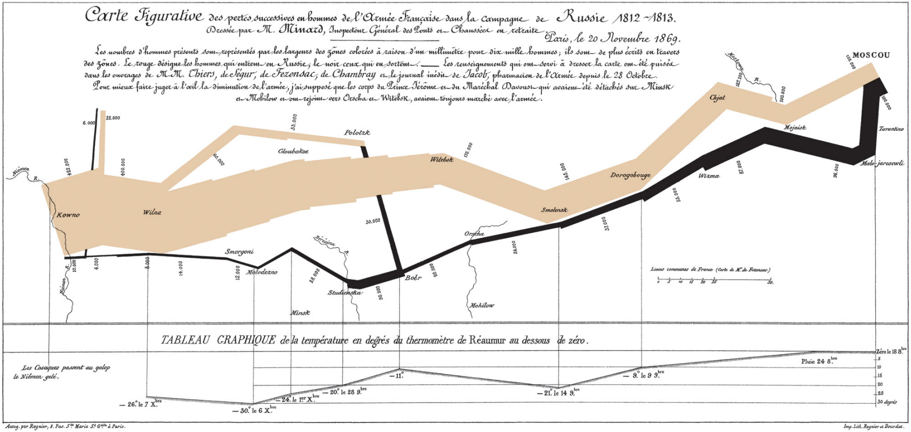
Problems

- Bars are difficult to read & compare
- Labels are duplicated
- 3D effects are distracting
- Drop shadows are useless

"Graphical excellence is the well-designed presentation of interesting data—a matter of substance, of statistics, and of design … [It] consists of **complex ideas communicated with clarity, precision, and efficiency** … [It] is that which gives to the viewer **the greatest number of ideas in the shortest time with the least ink in the smallest space** … And graphical excellence requires telling the truth about the data."

– Tufte (1983)

Napolean's Russian Campaign (Charles Minard, 1869)



However...

- Complex visualizations like this are relatively rare
- No clear compositional principles to develop from a Charles Minard-style visualization

Tufte & the conventional wisdom

- Maximize the "data-to-ink" ratio
  - Display the most data with the least about of ink
  - i.e. Simplify as much as possible

Unfortunately infographic-style visualizations have benefits

- Are easier to recall results, even if they are harder to interpret
- Are more memorable in general

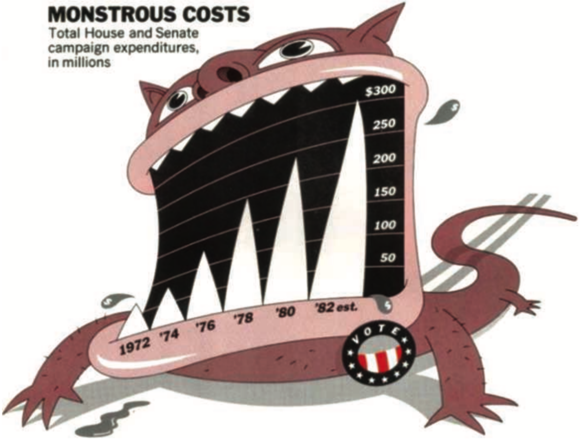


Figure 1.6: "Monstrous Costs" by Nigel Holmes (1982). Also a classic of its kind.

Furthermore, simplicity can go too far

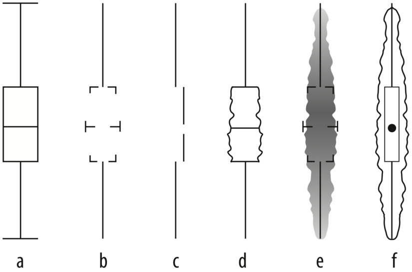
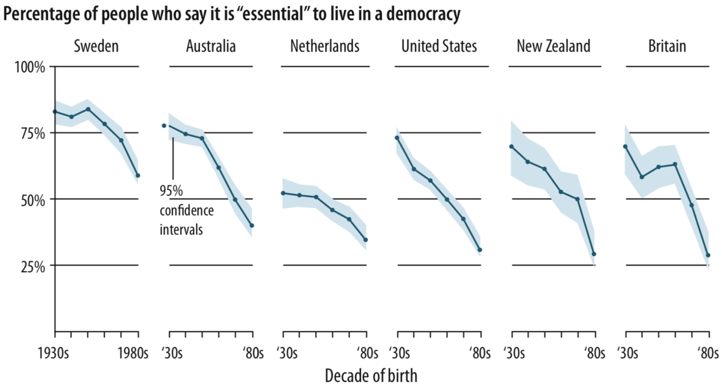


Figure 1.7: Six kinds of summary boxplots. Type (c) is from Tufte.

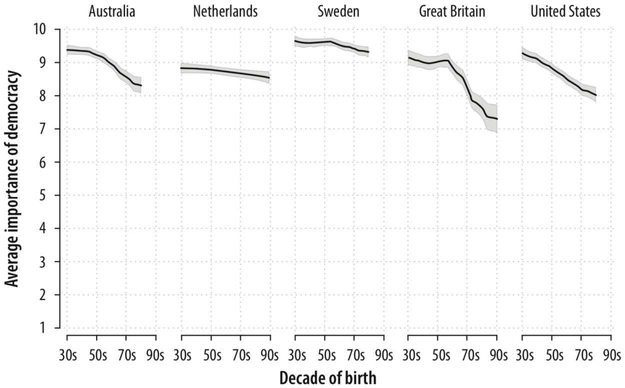
Tufte's own preferred boxplot (type "c" above) is the least understood in experiments

Bad data



The y-axis is percentage of people "agreeing"...

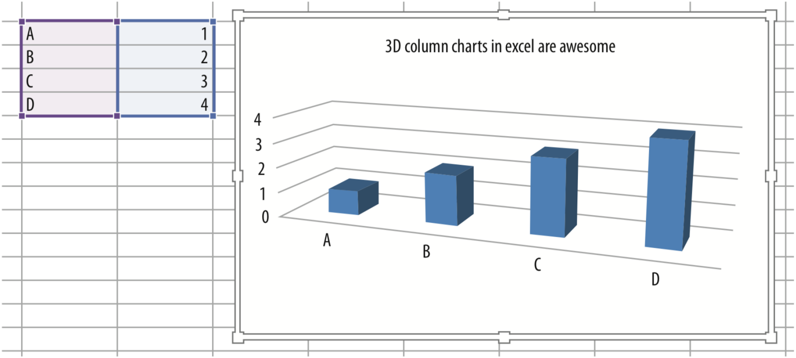
But on the original interval scale



Graph by Erik Voeten, based on WVS 5

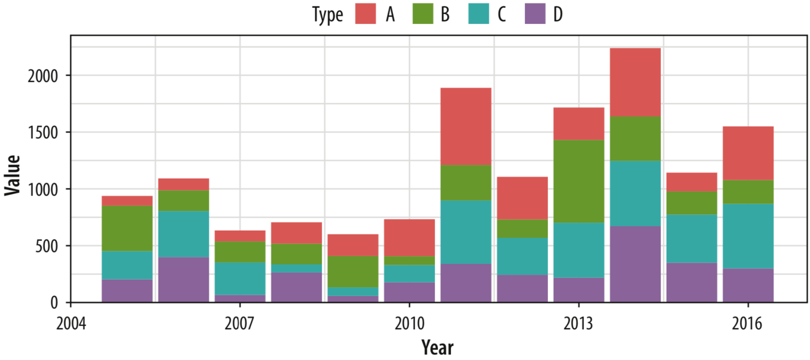
Problem is how the data are coded

Bad perceptions



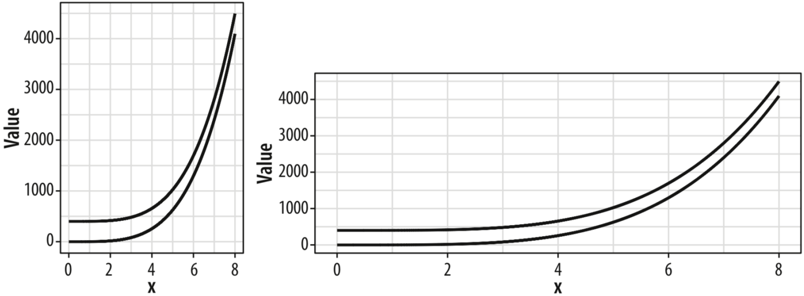
Values might be perceived as lower than their true value in the in the figure

Bad perceptions



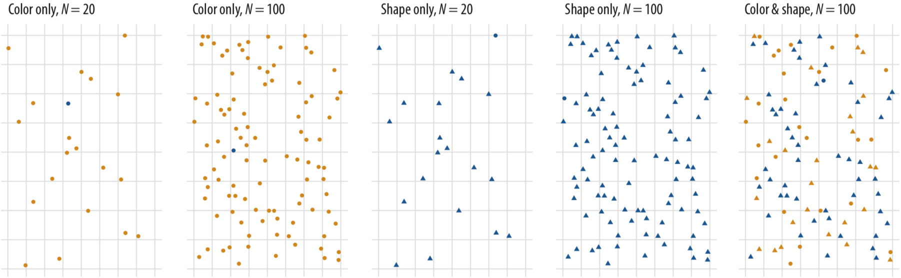
Flat (not 3D), but still difficult to interpret

Bad perceptions



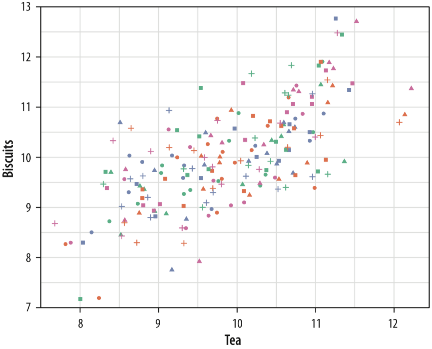
Same data, different aspect ratio

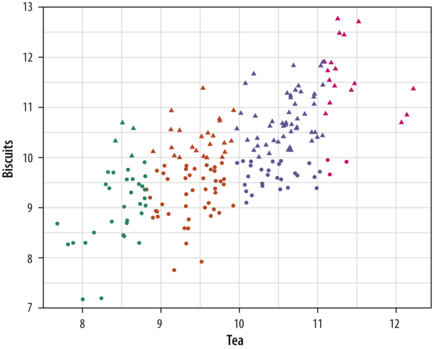
Mixing color and shape with a lot of data can be challenging



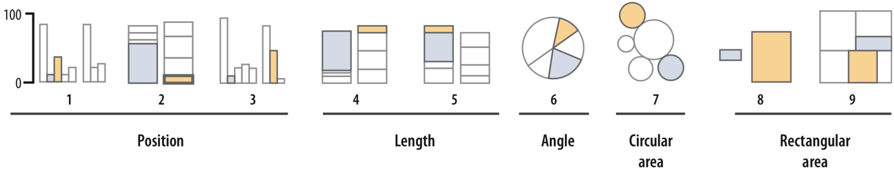
With very few data points, this might work

Be careful when adding many channels (left), unless there is substantial structure to the data (right)

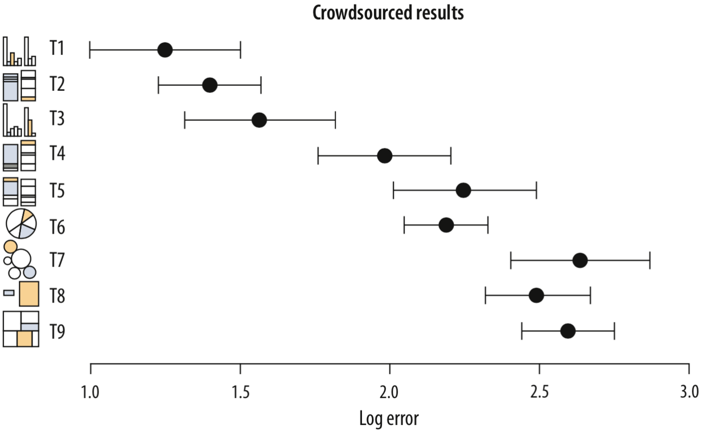




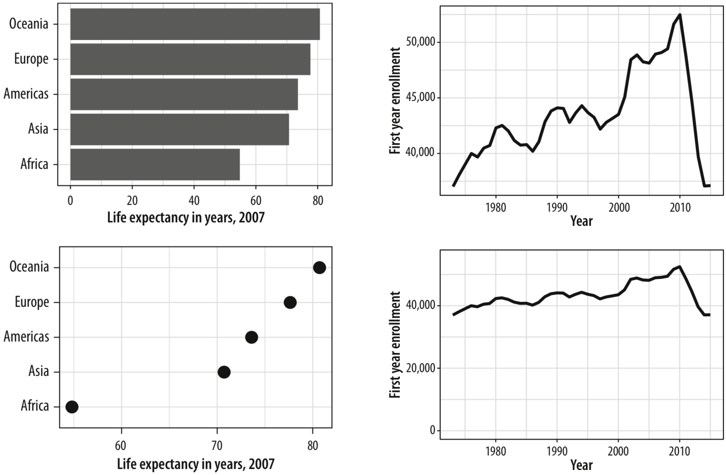
What types of figures are the most interpretable?



What types of figures are the most interpretable?



Honesty and good judgment



Honesty and good judgment

The New York Times

TheUpshot

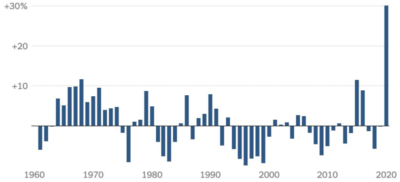
Murder Rose by Almost 30% in 2020. It's Rising at a Slower Rate in 2021.

The increase in U.S. murders this summer does not appear to be as large as the record spike last summer.

276

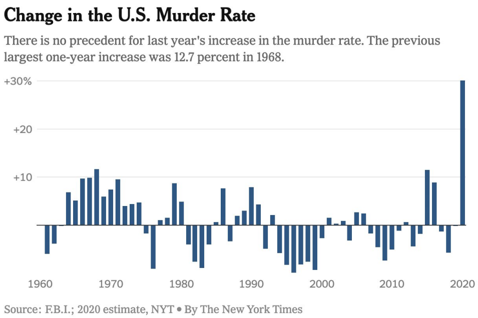
Change in the U.S. Murder Rate

There is no precedent for last year's increase in the murder rate. The previous largest one-year increase was 12.7 percent in 1968.

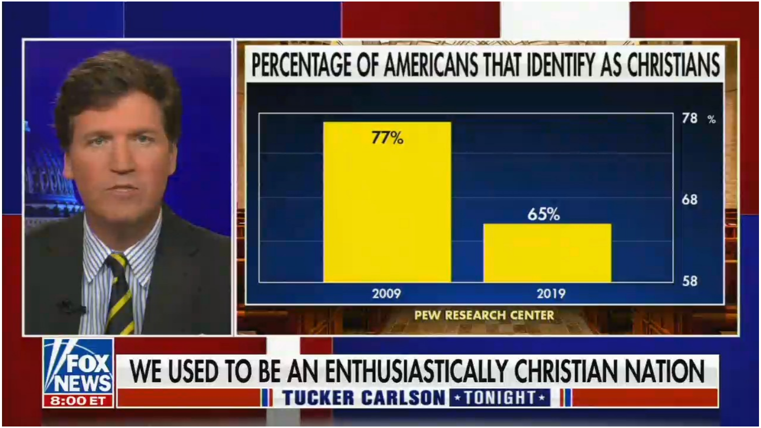


Source: F.B.I.; 2020 estimate, NYT • By The New York Times

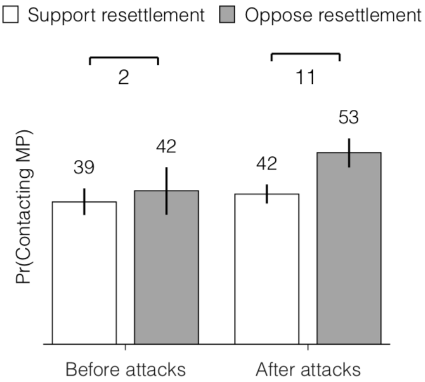
Honesty and good judgment



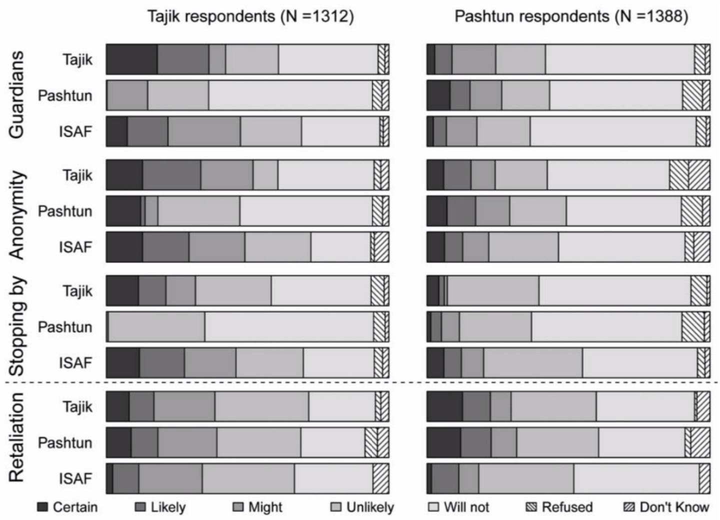
Honesty and good judgment



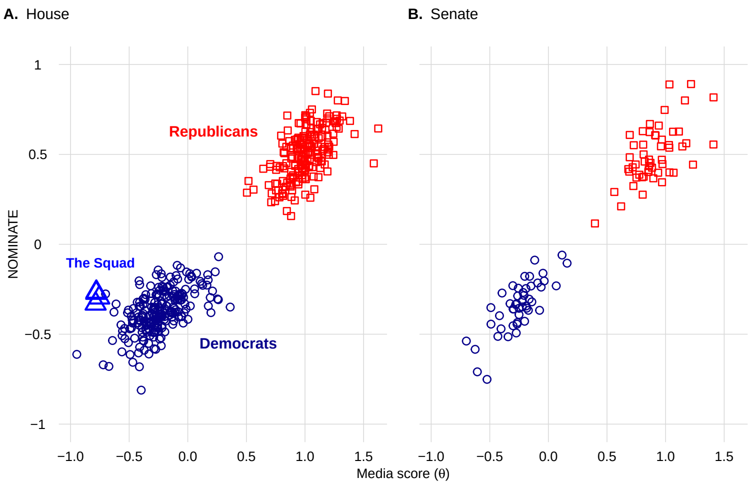
Bar graph `geom_bar(position = "dodge")`



Bar graph (stacked): `geom_bar(position = "stack")`

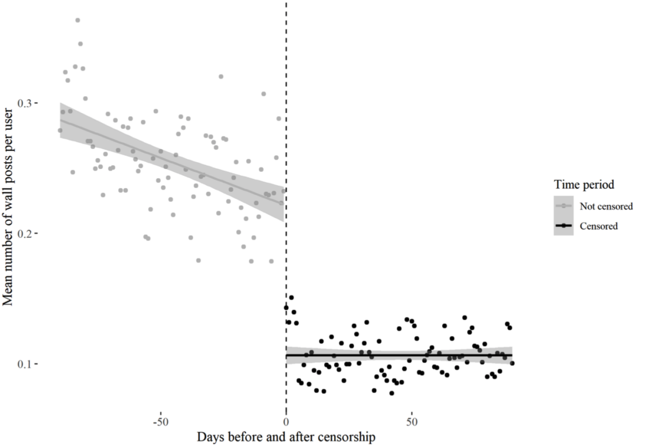


Scatterplot: geom_point()

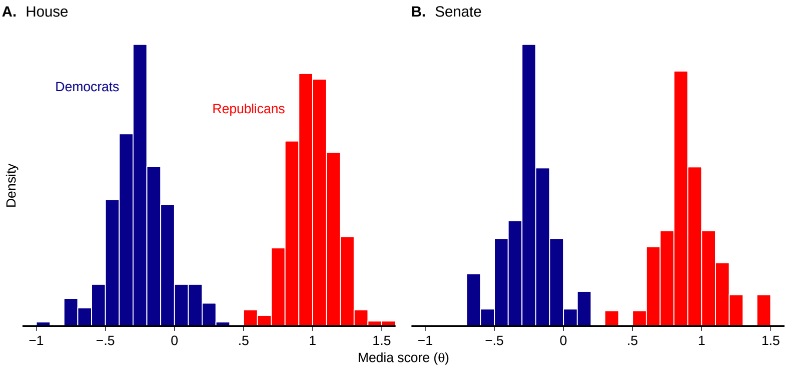


Scatterplot with regression line: geom_point() + geom_smooth()

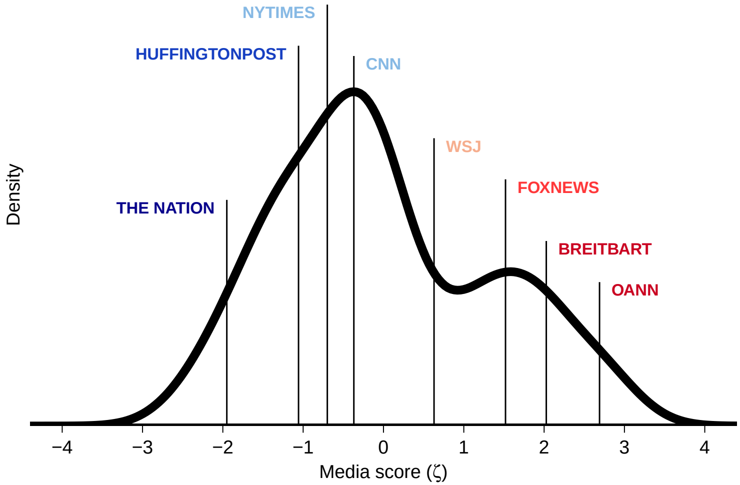
FIGURE 5. Regression discontinuity in posting activity 90 days before and after the ban (95% confidence interval)



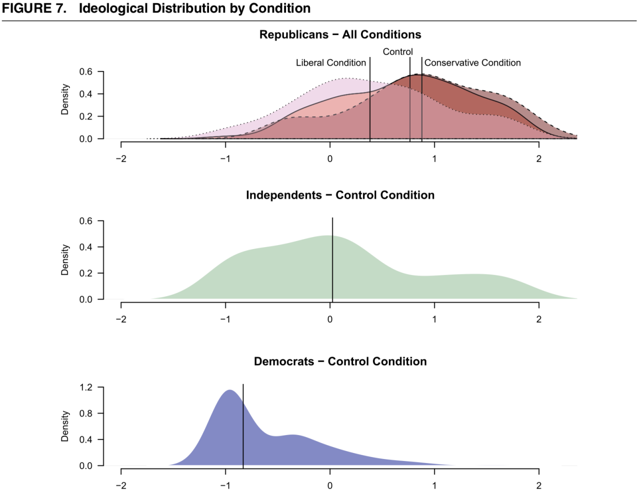
Histogram: geom_histogram()



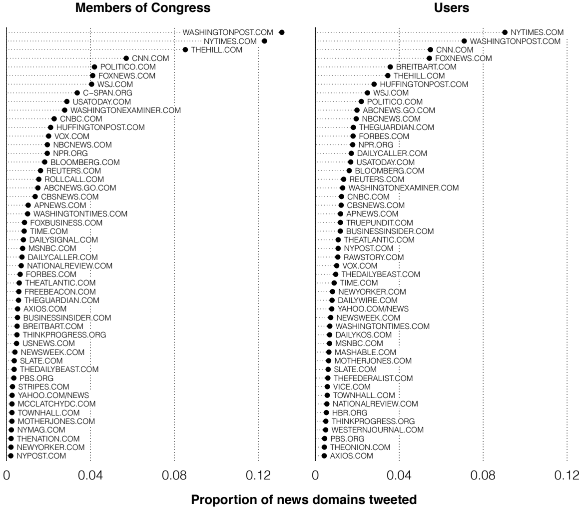
Density plot: `geom_density()`



Density plot: `geom_density()`



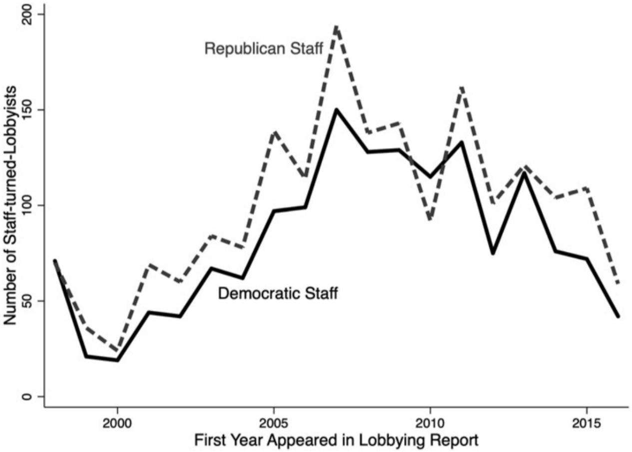
Dot plot: geom_point()



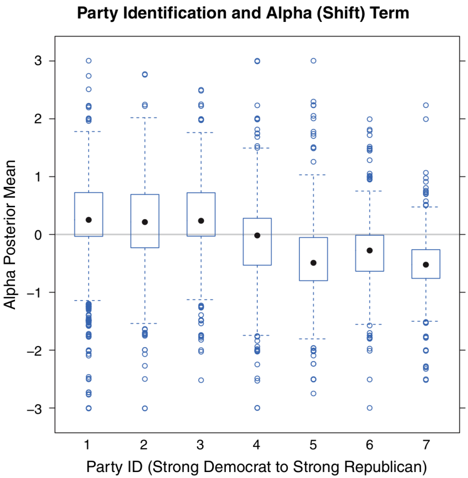
Dot plot (for coefficients): geom_point() + geom_segment()

Line plot: `geom_line()`

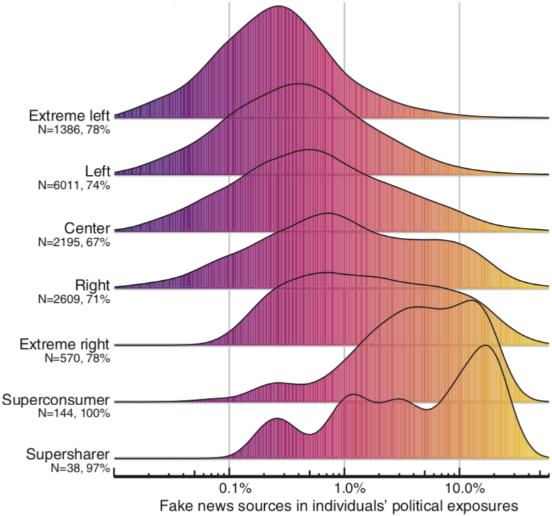
FIGURE 1. Number of Congressional Staffers-Turned-Lobbyists, 1998–2016



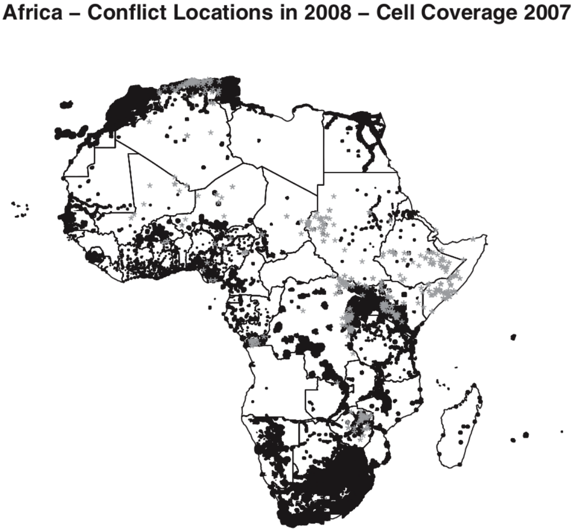
Boxplot: geom_boxplot()



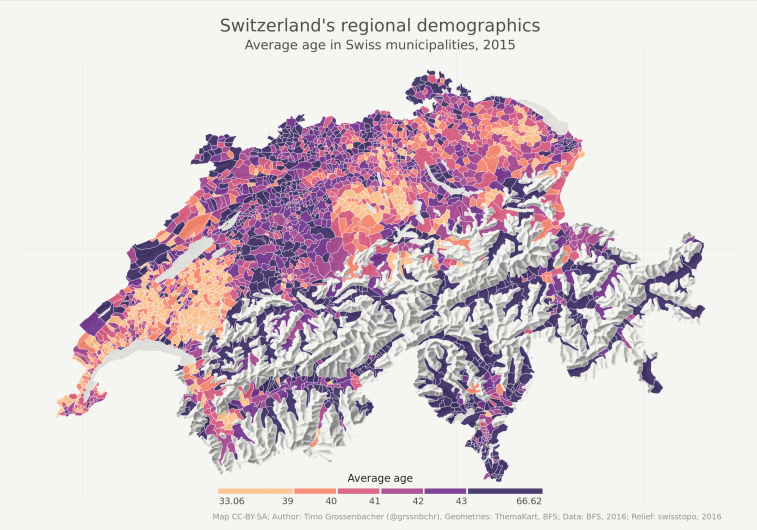
Ridge plot: `geom_density_ridges()`



Maps: `geom_polygon()`



Maps (choropleth): `geom_polygon()`



ggplot2 in practice

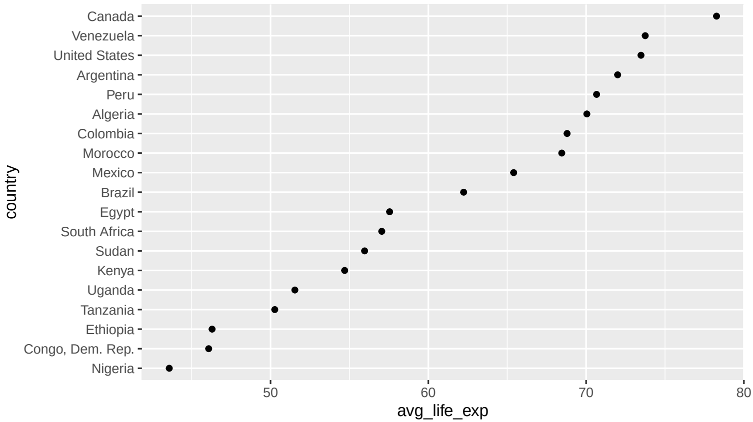
```
library(tidyverse) # contains ggplot2
library(gapminder) # load gapminder data

# Create data.frame for graph
# (1) Filter for only observations in Africa and the Americas with a
# population greater than 25 million people
# (2) Group by country
# (3) and take the average life expectancy across all years (for each country)
#     and keep the continent variable from each observation
# (4) Order the resulting country variable by its average life expectancy
# (5) Recode value of continent variable "Americas" to "North & South America"
# (6) Order the resulting continent variable manually, first by "Africa" and
#     then by countries in "North & South America"
G1 <- gapminder %>%
      filter(continent %in% c("Africa", "Americas") & pop > 25000000) %>%
      group_by(country) %>%
      summarize(continent = unique(continent),
                avg_life_exp = mean(lifeExp)) %>%
      mutate(country = fct_reorder(country, avg_life_exp),
             continent = recode(continent,
                                "Americas" = "North & South America"),
             continent = fct_relevel(continent, c("Africa",
                                     "North & South America")))
```

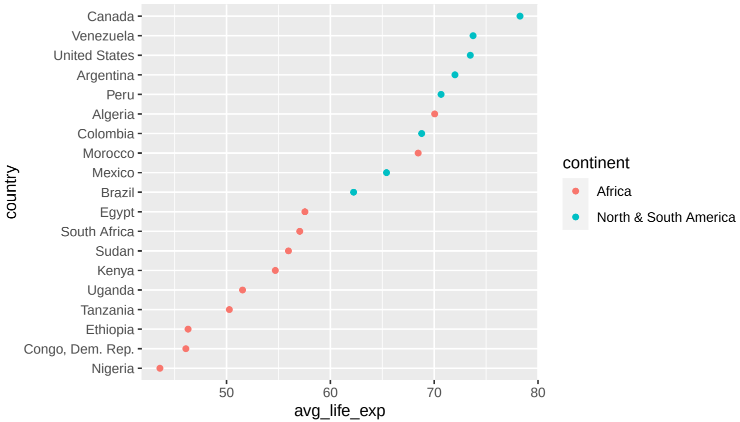
ggplot2 in practice

```
> G1
# A tibble: 19 x 3
   country           continent               avg_life_exp
   <fct>             <fct>                          <dbl>
 1 Algeria           Africa                          70.0
 2 Argentina         North & South America           72  
 3 Brazil            North & South America           62.2
 4 Canada            North & South America           78.3
 5 Colombia          North & South America           68.8
 6 Congo, Dem. Rep.  Africa                          46.1
 7 Egypt             Africa                          57.5
 8 Ethiopia          Africa                          46.3
 9 Kenya             Africa                          54.7
10 Mexico            North & South America           65.4
11 Morocco           Africa                          68.5
12 Nigeria           Africa                          43.6
13 Peru              North & South America           70.7
14 South Africa      Africa                          57.0
15 Sudan             Africa                          56.0
16 Tanzania          Africa                          50.3
17 Uganda            Africa                          51.5
18 United States     North & South America           73.5
19 Venezuela         North & South America           73.7
```

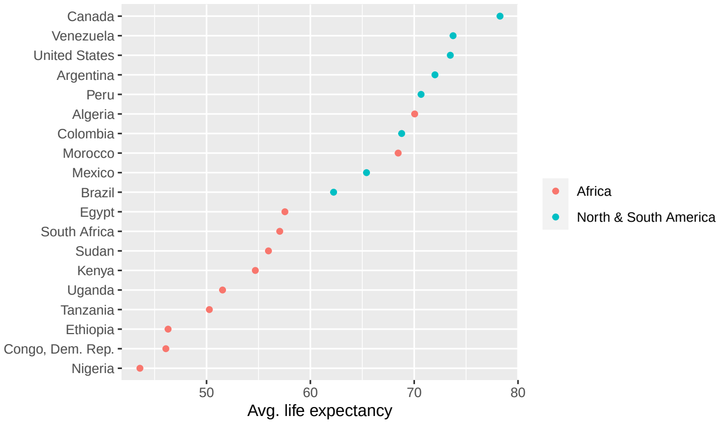
```
#Create a basic scatter plot
ggplot(G1, aes(x = avg_life_exp, y = country)) +
  geom_point()
```



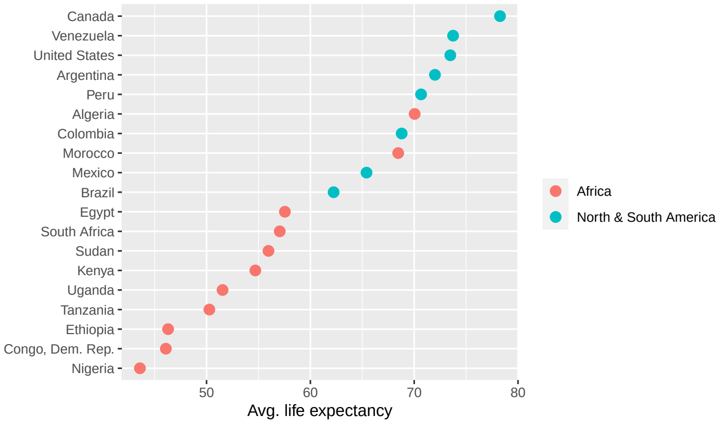
```
#Add an arugment to color the points by continent
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point()
```



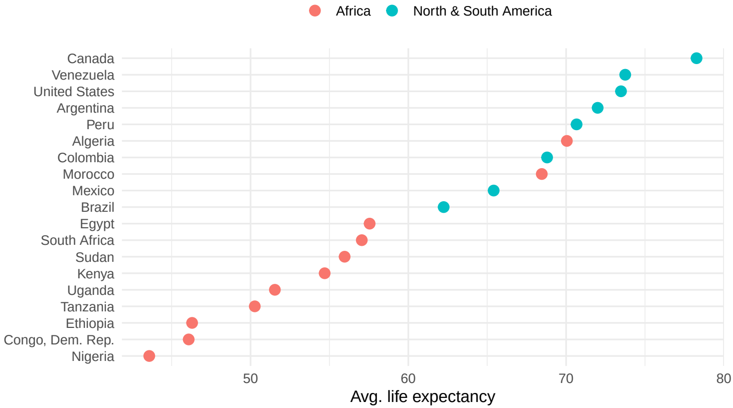
```r
#Add a layer to add and remove labels
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point() +
  labs(x = "Avg. life expectancy", y = "", color = "")
```



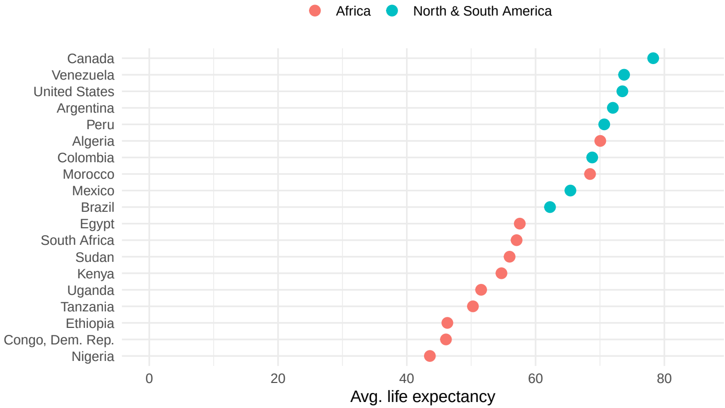
```
#Add an argument with specification of the point sizes
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point(size = 3) +
  labs(x = "Avg. life expectancy", y = "", color = "")
```



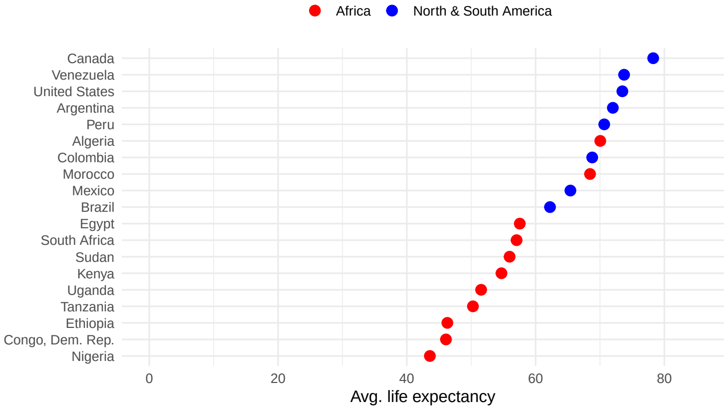
```r
#Add a layer specifying the theme, and another layer
#positioning the legend at the top of the plot
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point(size = 3) +
  labs(x = "Avg. life expectancy", y = "", color = "") +
  theme_minimal() +
  theme(legend.position = "top")
```



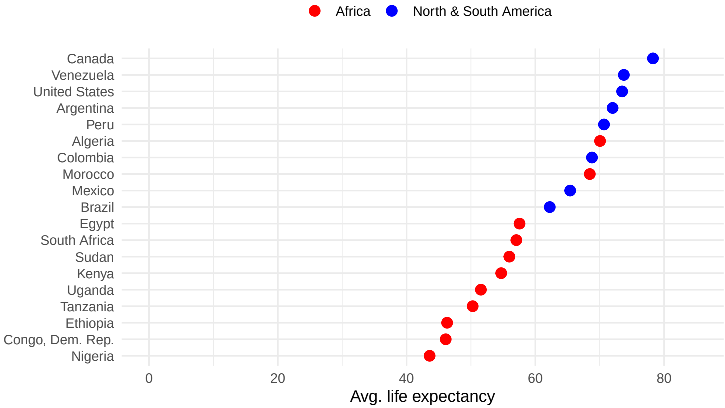
```r
#Add a layer that specifies the range of the x-axis
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point(size = 3) +
  coord_cartesian(xlim = c(0, 85)) +
  labs(x = "Avg. life expectancy", y = "", color = "") +
  theme_minimal() +
  theme(legend.position = "top")
```



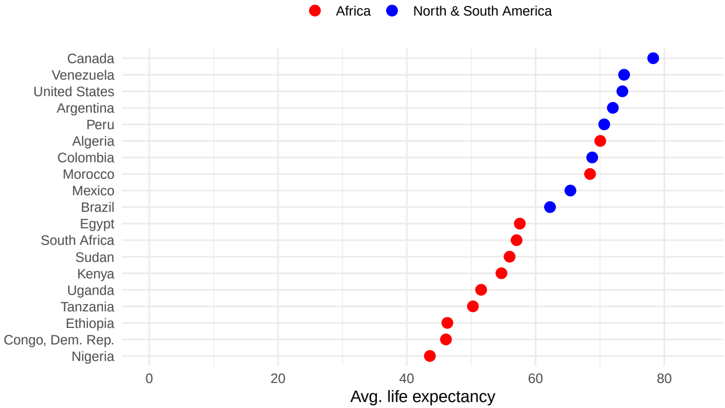
```
#Add a layer that manually adjusts the continent colors
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point(size = 3) +
  coord_cartesian(xlim = c(0, 85)) +
  labs(x = "Avg. life expectancy", y = "", color = "") +
  scale_color_manual(values = c("Africa" = "red",
                                "North & South America" = "blue")) +
  theme_minimal() +
  theme(legend.position = "top")
```



```r
pdf("My_Graph.pdf", 7, 4) # Or png("...", 700, 400)
ggplot(G1, aes(x = avg_life_exp, y = country, color = continent)) +
  geom_point(size = 3) +
  coord_cartesian(xlim = c(0, 85)) +
  labs(x = "Avg. life expectancy", y = "", color = "") +
  scale_color_manual(values = c("Africa" = "red",
  "North & South America" = "blue")) +
  theme_minimal() +
  theme(legend.position = "top")
dev.off() #Save the plot (Option A)
```



```r
#Save the plot (Option B - using ggplot's own function)
ggsave("My_Graph.pdf",
       device = "pdf", width = 7, height = 4)
```



Exercise

- Download the .R exercise file from the course website and fill in the missing sections

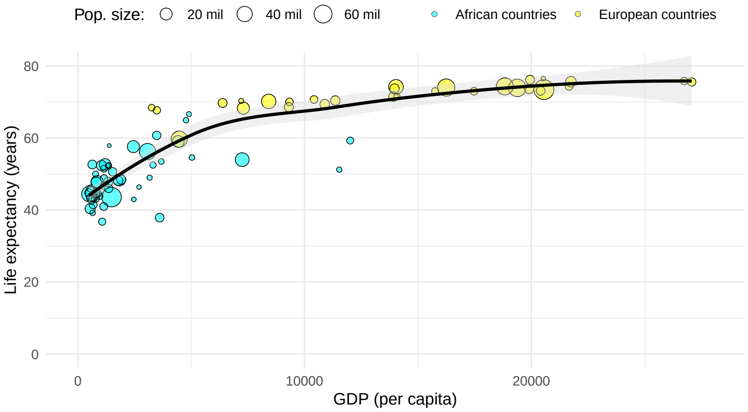
Exercise 1 solution

```
G2 <- gapminder %>%
      filter(continent %in% c("Europe", "Africa")) %>%
      mutate(continent = recode(continent, "Europe" = "European countries",
                                           "Africa" = "African countries")) %>%
      group_by(country) %>%
      summarize(continent = unique(continent),
                avg_life_exp = mean(lifeExp),
                population = mean(pop),
                gdp = mean(gdpPercap)) %>%
      arrange(avg_life_exp)

head(G2) # To see the lowest life expectancy
tail(G2) # To see the highest life expectancy

pdf("Exercise_1_Graph.pdf", 6, 4)
ggplot(G2, aes(x = gdp, y = avg_life_exp,
               size = population, fill = continent)) +
  geom_point(shape = 21, alpha = 0.6, stroke = 0.3) +
  geom_point(shape = 21, stroke = 0.3, fill = NA) +
  stat_smooth(color = "black", fill = "grey85", size = 1) +
  coord_cartesian(xlim = c(0, 28000), ylim = c(0, 80)) +
  labs(x = "GDP (per capita)", y = "Life expectancy (years)",
       size = "Pop. size:", fill = "") +
  scale_fill_manual(values = c("African countries" = "cyan",
                               "European countries" = "yellow")) +
  scale_size(breaks = c(20000000, 40000000, 60000000),
             labels = c("20 mil", "40 mil", "60 mil")) +
  theme_minimal() +
  theme(legend.position = "top")
dev.off()
```

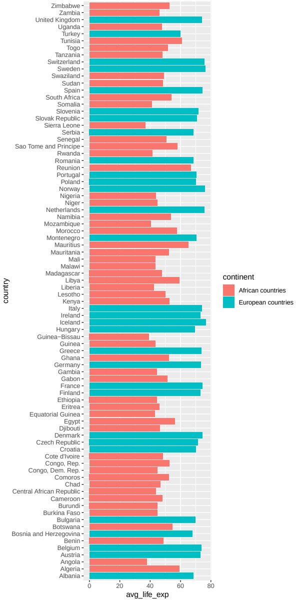
Exercise 1 graph



Exercise 2 solution

```
pdf("Exercise_2_Graph.pdf", 6, 12)
ggplot(G2, aes(x = country, y = avg_life_exp, fill = continent)) +
  geom_col() +
  coord_flip()
dev.off()
```

Exercise 2 graph

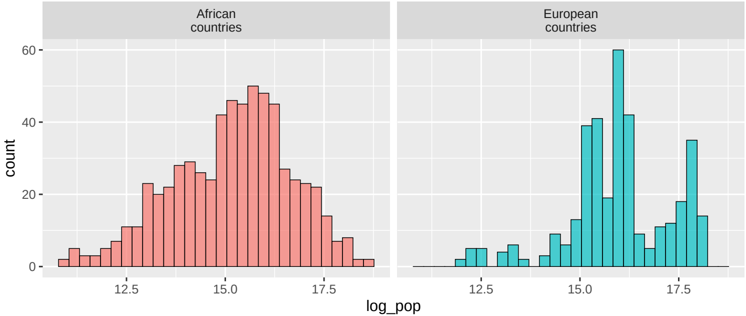


Exercise 3 solution

```
G3 <- gapminder %>%
      filter(continent %in% c("Europe", "Africa")) %>%
      mutate(continent = recode(continent, "Europe" = "European\ncountries",
                                           "Africa" = "African\ncountries"),
             log_pop = log(pop))

pdf("Exercise_3_Graph.pdf", 7, 3.25)
ggplot(G3, aes(x = log_pop, fill = continent)) +
  geom_histogram(alpha = 0.7, color = "black", size = 0.25) +
  facet_wrap(~ continent) +
  theme(legend.position = "none")
dev.off()
```

Exercise 3 graph



Exercise 4 solution

```
G4 <- gapminder %>%
      filter(country %in% c("Denmark", "Canada", "United States"))

pdf("Exercise_4_Graph.pdf", 6, 3.25)
ggplot(G4, aes(x = year, y = gdpPercap, color = country, shape = country)) +
  geom_line() +
  geom_point()
dev.off()
```

Exercise 4 graph