SOC6302 Statistics for Sociologists

Monica Alexander

Week 3: Exploratory Data Analysis II (Data Visualization)

Announcements

What we will cover today:

- Data visualization principles
- Important types of graphs

Assignment 1 is out

Data visualization

Plot your data!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

- We started to compute some summary statistics above, and showed how summaries can be calculated by group and arranged in different ways to get a sense of differences across groups
- However, graphing/plotting your data is usually the best way to visualize patterns, trends, outliers, issues and other surprising points
- The most appropriate types of graph for your data depends on:
  - the type of variable you are interested in (quantitative or qualitative/categorical)
  - your research questions

Plot your data!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

- Before you start to do any statistical analysis, you should always plot your data
- Data visualization is a key part of EDA and essential in understanding the assumptions and outcomes of your eventual statistical analysis

Plot your data!!!!!!!!!!!!!!!!!!!!!!!!!!!

Here's a specific example. Imagine we have the following sets of datasets of (x,y) pairs

```
library(tidyverse)
library(datasauRus)
head(datasaurus_dozen)
```

```
## # A tibble: 6 x 3
##   dataset     x     y
##   <chr>   <dbl> <dbl>
## 1 dino     55.4  97.2
## 2 dino     51.5  96.0
## 3 dino     46.2  94.5
## 4 dino     42.8  91.4
## 5 dino     40.8  88.3
## 6 dino     38.7  84.9
```

How many observations?

```
datasaurus_dozen  %>%  count(dataset)
```

```
## # A tibble: 13 x 2
##    dataset        n
##    <chr>      <int>
##  1 away         142
##  2 bullseye     142
##  3 circle       142
##  4 dino         142
##  5 dots         142
##  6 h_lines      142
##  7 high_lines   142
##  8 slant_down   142
##  9 slant_up     142
## 10 star         142
## 11 v_lines      142
## 12 wide_lines   142
## 13 x_shape      142
```

Do some summaries for each dataset

```
datasaurus_dozen %>%
  group_by(dataset) %>%
  summarise(mean_x = mean(x),
            mean_y = mean(y),
            correlation = cor(x,y))
```

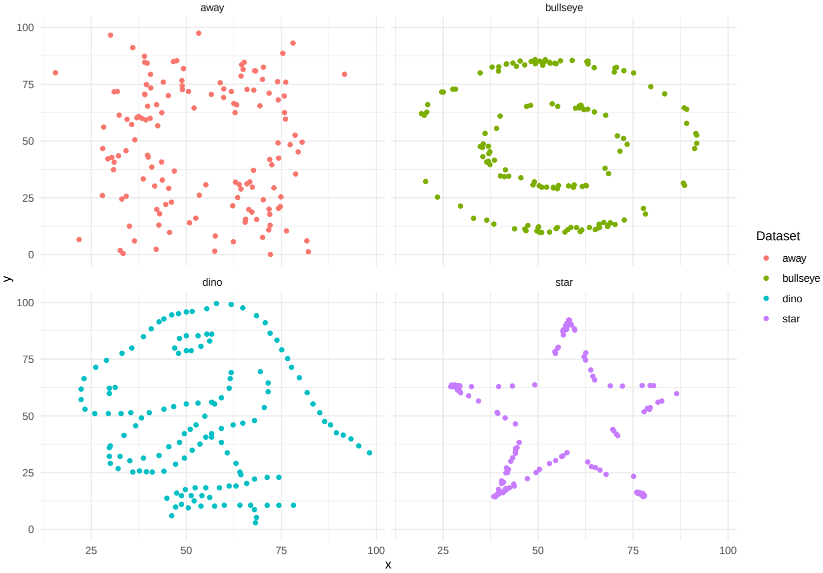
```
## # A tibble: 13 x 4
##    dataset    mean_x mean_y correlation
##    <chr>       <dbl>  <dbl>       <dbl>
##  1 away         54.3   47.8     -0.0641
##  2 bullseye     54.3   47.8     -0.0686
##  3 circle       54.3   47.8     -0.0683
##  4 dino         54.3   47.8     -0.0645
##  5 dots         54.3   47.8     -0.0603
##  6 h_lines      54.3   47.8     -0.0617
##  7 high_lines   54.3   47.8     -0.0685
##  8 slant_down   54.3   47.8     -0.0690
##  9 slant_up     54.3   47.8     -0.0686
## 10 star         54.3   47.8     -0.0630
## 11 v_lines      54.3   47.8     -0.0694
## 12 wide_lines   54.3   47.8     -0.0666
## 13 x_shape      54.3   47.8     -0.0656
```

Summaries are very similar

```
datasaurus_dozen %>%
  group_by(dataset) %>%
  summarise(mean_x = mean(x),
            mean_y = mean(y),
            correlation = cor(x,y))
```

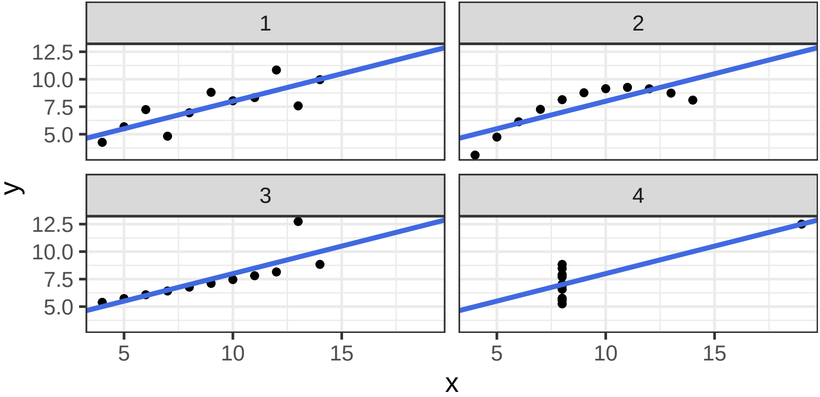
```
## # A tibble: 13 x 4
##    dataset    mean_x mean_y correlation
##    <chr>       <dbl>  <dbl>       <dbl>
##  1 away         54.3   47.8     -0.0641
##  2 bullseye     54.3   47.8     -0.0686
##  3 circle       54.3   47.8     -0.0683
##  4 dino         54.3   47.8     -0.0645
##  5 dots         54.3   47.8     -0.0603
##  6 h_lines      54.3   47.8     -0.0617
##  7 high_lines   54.3   47.8     -0.0685
##  8 slant_down   54.3   47.8     -0.0690
##  9 slant_up     54.3   47.8     -0.0686
## 10 star         54.3   47.8     -0.0630
## 11 v_lines      54.3   47.8     -0.0694
## 12 wide_lines   54.3   47.8     -0.0666
## 13 x_shape      54.3   47.8     -0.0656
```

But now let's plot



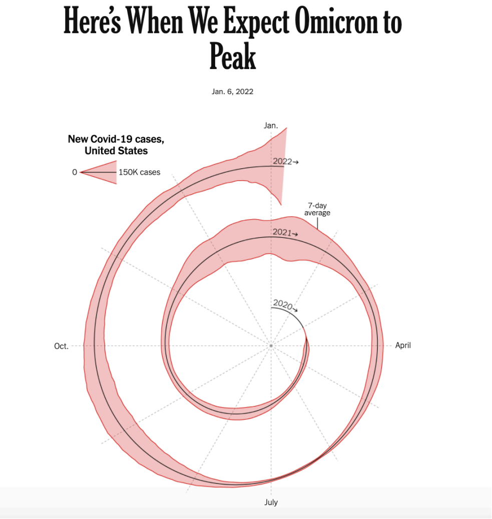
Anscombe's quartet

This is a modern version of a famous plot 'Anscombe's Quartet.' That plot conveys the same message about the importance of plotting the actual data and not relying on summary statistics.



What makes a good plot?

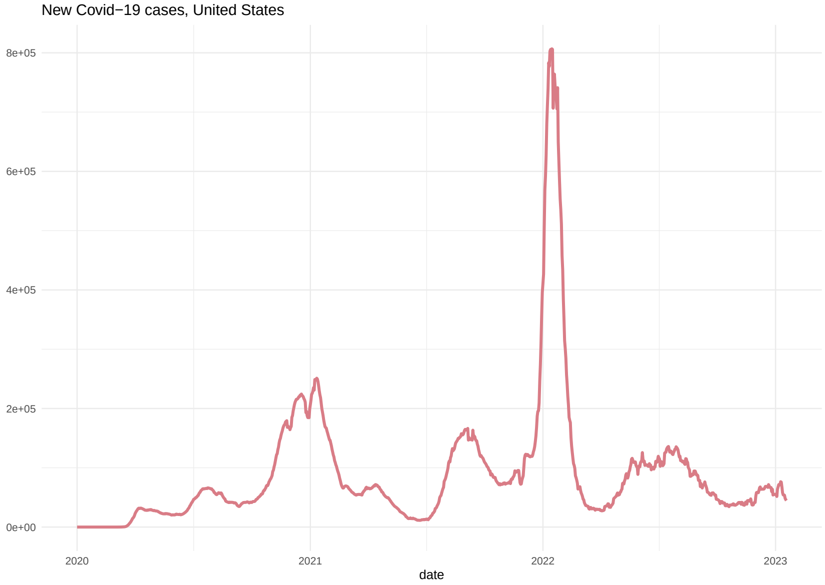
The spiral of doom



Here's When We Expect Omicron to Peak

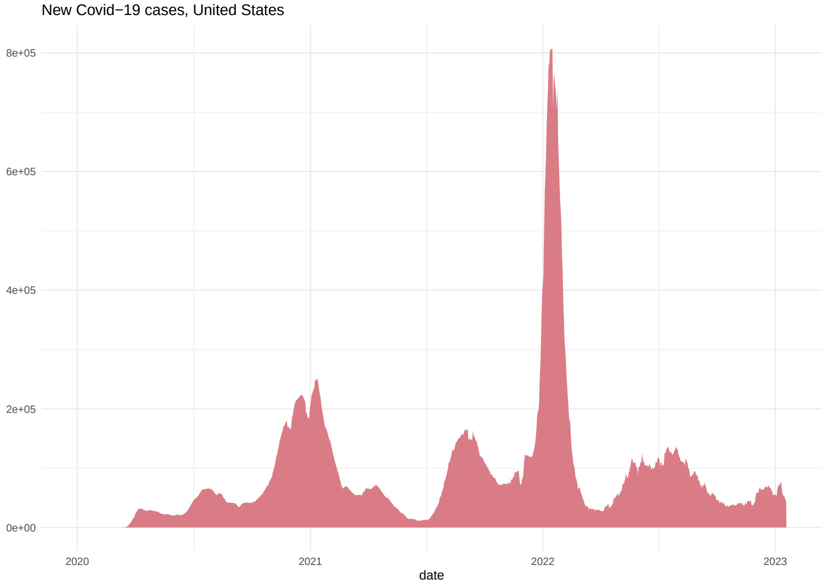
Jan. 6, 2022

Line plot

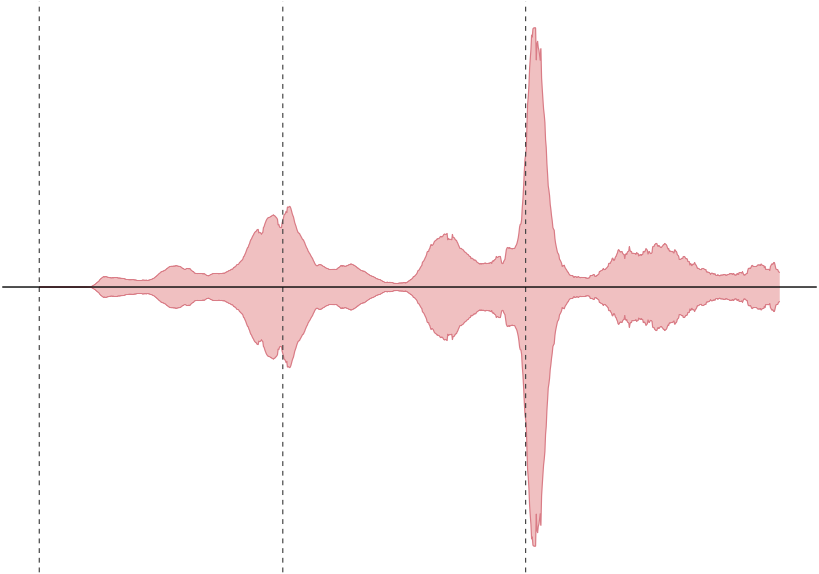


New Covid−19 cases, United States
8e+05
6e+05
4e+05
2e+05
0e+00
2020
2021
2022
2023
date

Area plot


New Covid−19 cases, United States
8e+05
6e+05
4e+05
2e+05
0e+00
2020
2021
2022
2023
date

Ribbons



What makes a good plot?

- Is the spiral the right graph to use?
- What does right mean?
- Does it effectively portray the information?
- Is it misleading?
- Is it easy to read?
- Is it memorable?

Data visualization principles

- Choose the right graph
- Know your audience
- Emphasize important patterns without being misleading
- Clear, effective designs

Choose the right graph

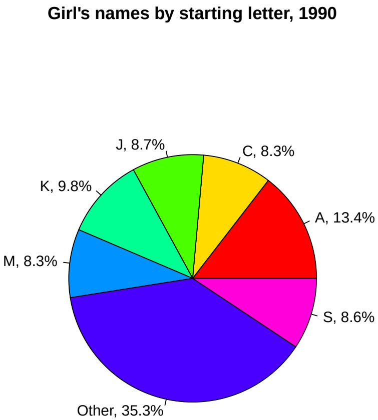
Choosing the right graph primarily depends on the type of variables that you are trying to visualize:

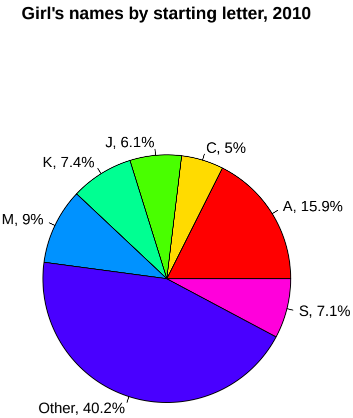
- Quantitative variables e.g. histograms, scatter plots
- Qualitative variables e.g. barcharts

Choose the graph based on the kind of data and the message to be conveyed.

- Do not use different graphs just for variety, as specific graphs convey certain types of information more effectively than others.
- If not required, do not use any chart — show only numbers.

Pie charts


Girl's names by starting letter, 1990
J, 8.7%
C, 8.3%
K, 9.8%
A, 13.4%
M, 8.3%
S, 8.6%
Other, 35.3%

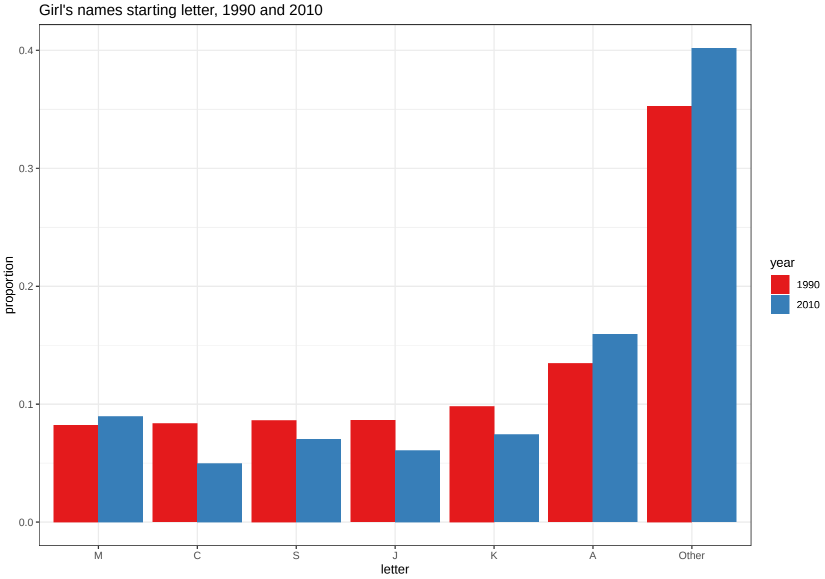

Girl's names by starting letter, 2010
J, 6.1%
C, 5%
K, 7.4%
A, 15.9%
M, 9%
S, 7.1%
Other, 40.2%

Pie charts

```
?pie
```

> *Pie charts are a very bad way of displaying information. The eye is good at judging linear measures and bad at judging relative areas. A bar chart or dot chart is a preferable way of displaying this type of data.*

Alternative



Know your audience

Graphs can be used for

- our own exploratory data analysis
- to convey a message to experts,
- to help tell a story to a general audience.

Make sure that the intended audience understands each element of the plot.

Examples: spiral plot, log scales

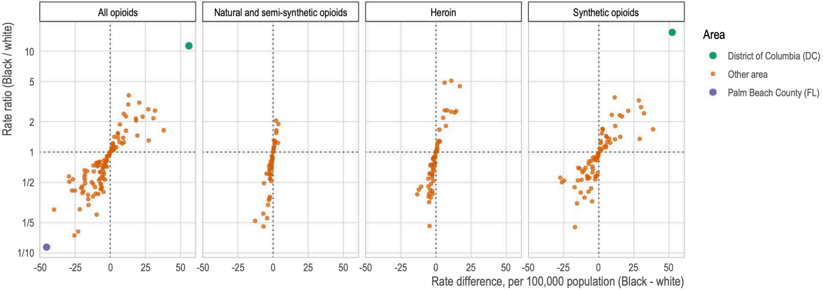
- Think of the color blind. In R, `viridis` and `brewer` palettes give colorblind-friendly options

Emphasize important patterns without being misleading

> *There is no such thing as information overload. There is only bad design. — Edward Tufte*

- Eliminate distractions
- Highlight the essential
- Use color and text strategically
- Avoid pseudo-3D plots

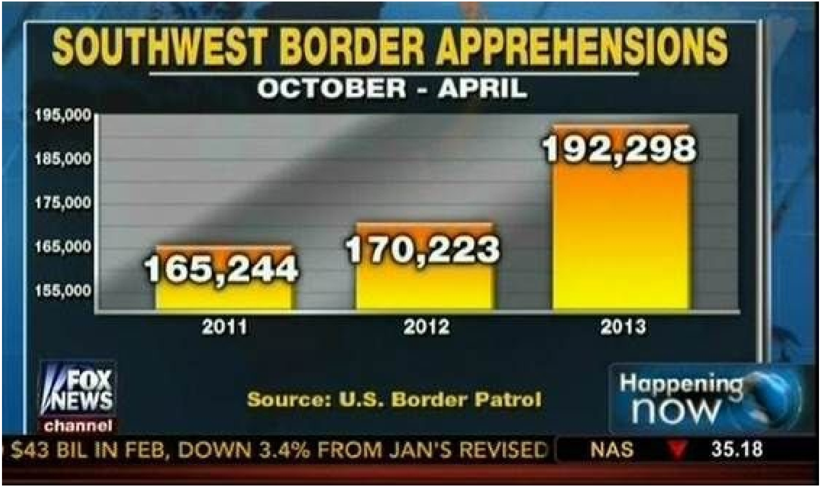
Highlight the essential



Source:
https://link.springer.com/article/10.1007/s11524-021-00573-8

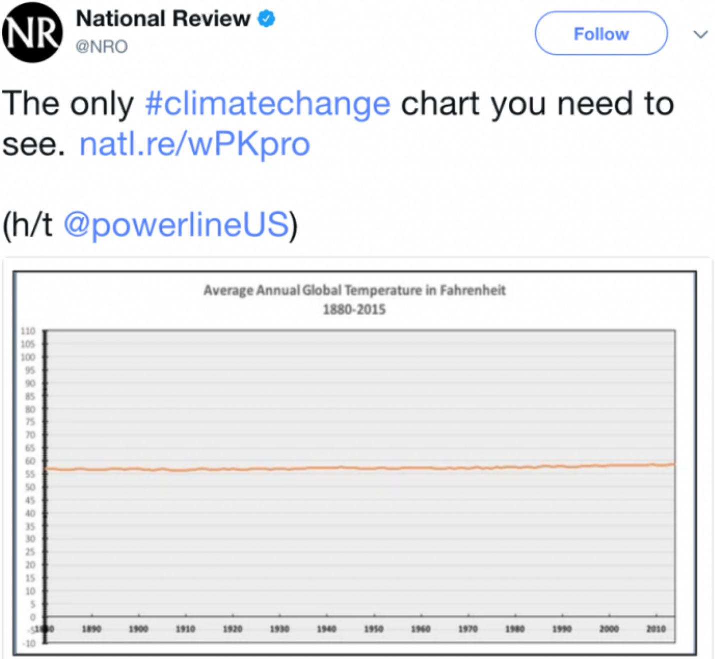
When to start the axis at zero?

When to start the axis at zero?



Source

When to start the axis at zero?



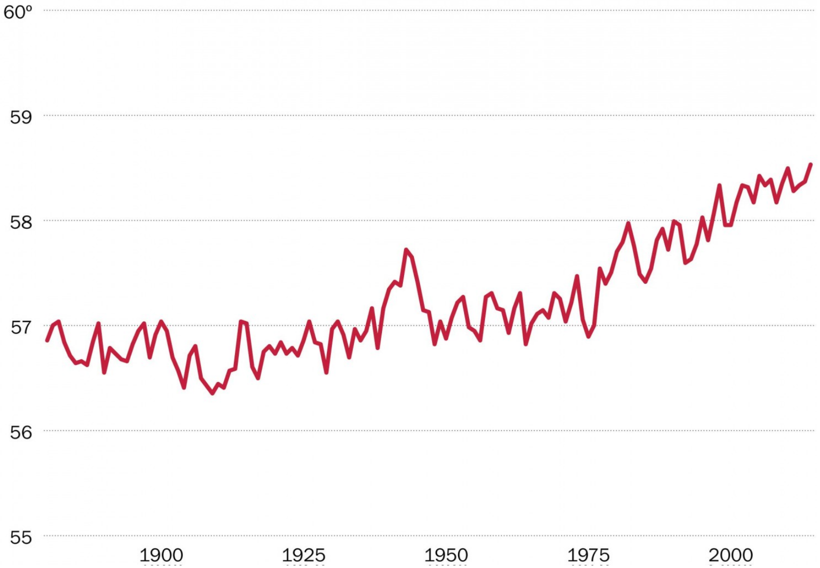
National Review @NRO

Follow

The only #climatechange chart you need to see. natl.re/wPKpro

(h/t @powerlineUS)

When to start the axis at zero?

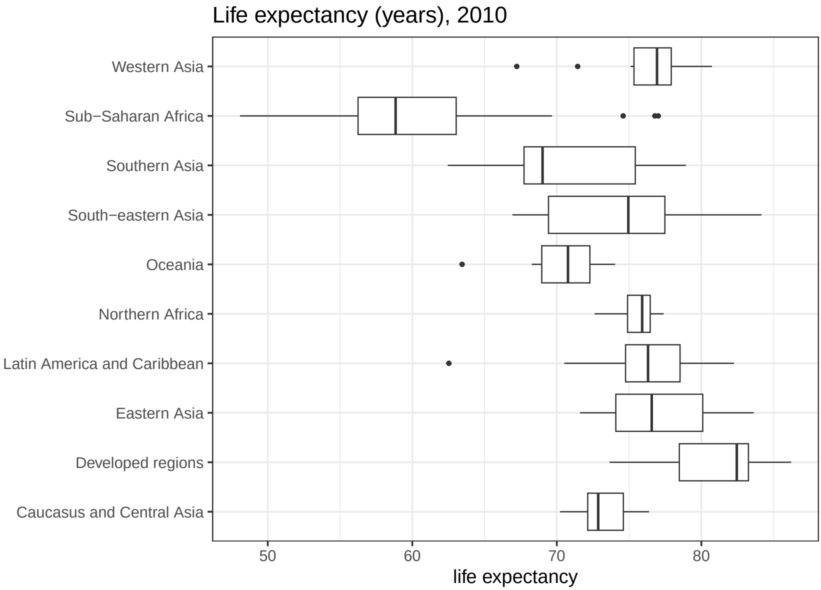


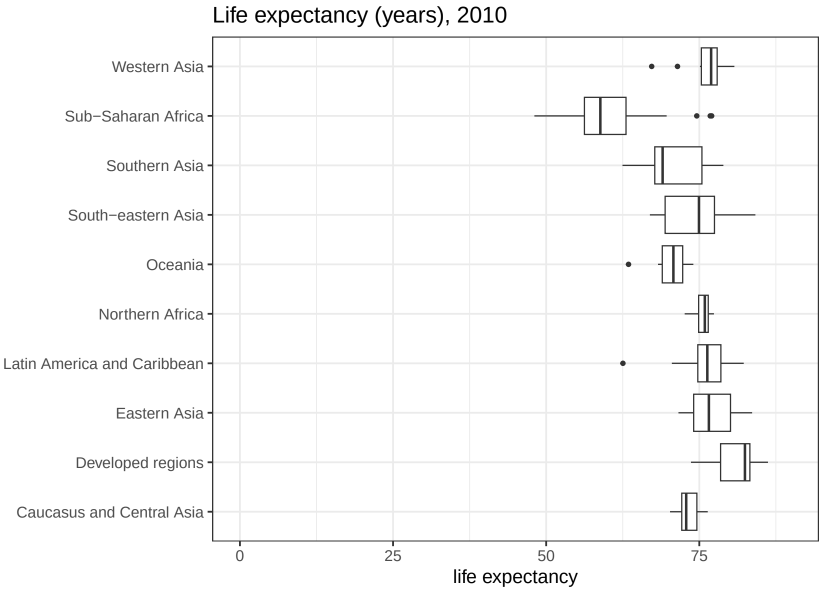
Average global temperature by year

Data from NASA/GISS.

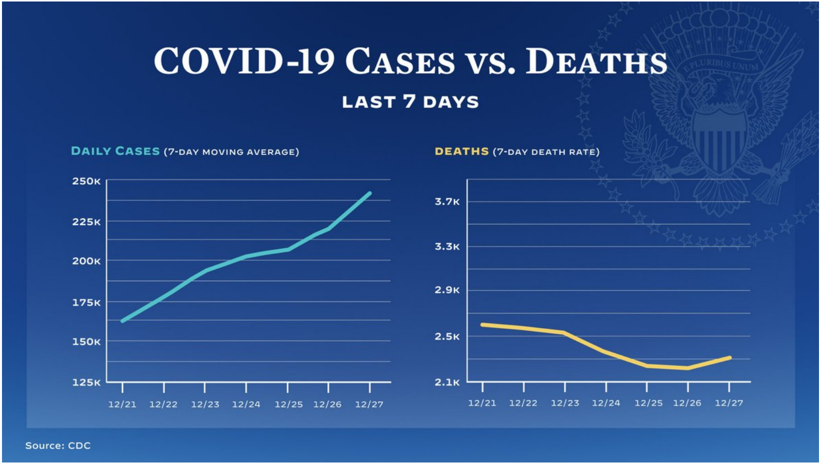
When to include zeroes

- With bar plots, we are implying the length is proportional to the quantities being displayed. By avoiding 0, relatively small differences can be made to look much bigger than they actually are.
- With line plots or plots that use position, it is not neccessary to start the axis at zero (and could be misleading)


Life expectancy (years), 2010
Western Asia
Sub−Saharan Africa
Southern Asia
South−eastern Asia
Oceania
Northern Africa
Latin America and Caribbean
Eastern Asia
Developed regions
Caucasus and Central Asia
50
60
70
80
life expectancy


Life expectancy (years), 2010
Western Asia
Sub−Saharan Africa
Southern Asia
South−eastern Asia
Oceania
Northern Africa
Latin America and Caribbean
Eastern Asia
Developed regions
Caucasus and Central Asia
0
25
50
75
life expectancy

Emphasize important patterns without being misleading



COVID-19 CASES VS. DEATHS

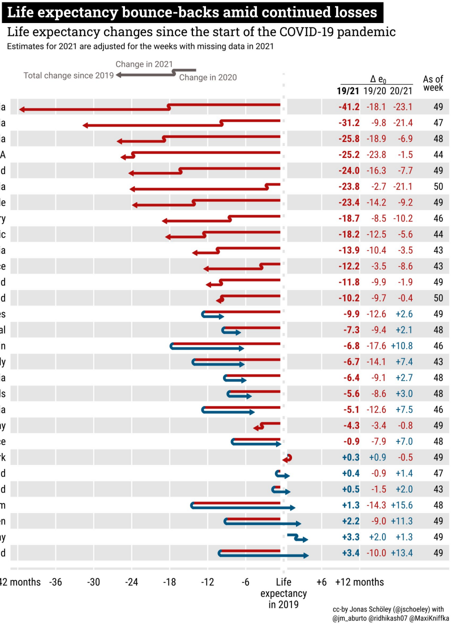
LAST 7 DAYS

DAILY CASES (7-DAY MOVING AVERAGE)

DEATHS (7-DAY DEATH RATE)

Source: CDC

Clear, effective designs



Life expectancy bounce-backs amid continued losses

Life expectancy changes since the start of the COVID-19 pandemic

Estimates for 2021 are adjusted for the weeks with missing data in 2021

	Δ e_0 19/21	19/20	20/21	As of week
Bulgaria	-41.2	-18.1	-23.1	49
Slovakia	-31.2	-9.8	-21.4	47
Lithuania	-25.8	-18.9	-6.9	48
USA	-25.2	-23.8	-1.5	44
Poland	-24.0	-16.3	-7.7	49
Estonia	-23.8	-2.7	-21.1	50
Chile	-23.4	-14.2	-9.2	49
Hungary	-18.7	-8.5	-10.2	46
Czech Republic	-18.2	-12.5	-5.6	44
Croatia	-13.9	-10.4	-3.5	43
Greece	-12.2	-3.5	-8.6	43
Northern Ireland	-11.8	-9.9	-1.9	49
Scotland	-10.2	-9.7	-0.4	50
England and Wales	-9.9	-12.6	+2.6	49
Portugal	-7.3	-9.4	+2.1	48
Spain	-6.8	-17.6	+10.8	46
Italy	-6.7	-14.1	+7.4	43
Austria	-6.4	-9.1	+2.7	48
Netherlands	-5.6	-8.6	+3.0	48
Slovenia	-5.1	-12.6	+7.5	46
Germany	-4.3	-3.4	-0.8	49
France	-0.9	-7.9	+7.0	48
Denmark	+0.3	+0.9	-0.5	49
Finland	+0.4	-0.9	+1.4	47
Iceland	+0.5	-1.5	+2.0	43
Belgium	+1.3	-14.3	+15.6	48
Sweden	+2.2	-9.0	+11.3	49
Norway	+3.3	+2.0	+1.3	49
Switzerland	+3.4	-10.0	+13.4	49

cc-by Jonas Schöley (@jschoeley) with
@jm_aburto @ridhikash07 @MaxiKniffka

Important types of graphs

Important types of graphs

- Histograms
- Bar charts
- Boxplots
- Line plots
- Scatter plots

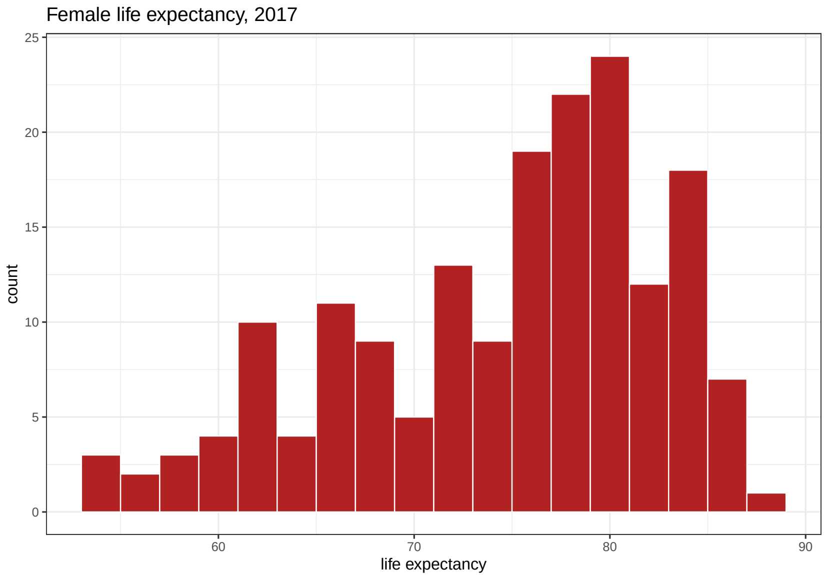
Example datasets used here

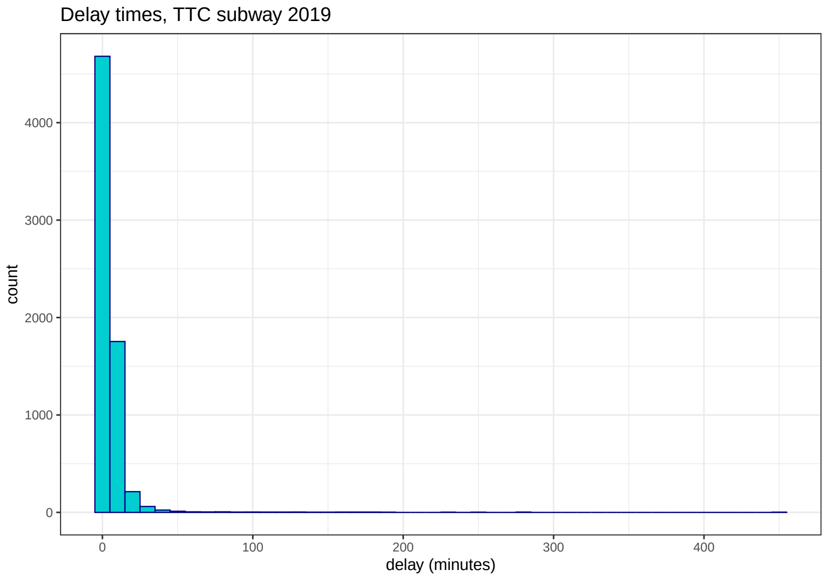
1. TTC subway delays (from last week)
2. Country-level indicators, 2009-2017

Histograms

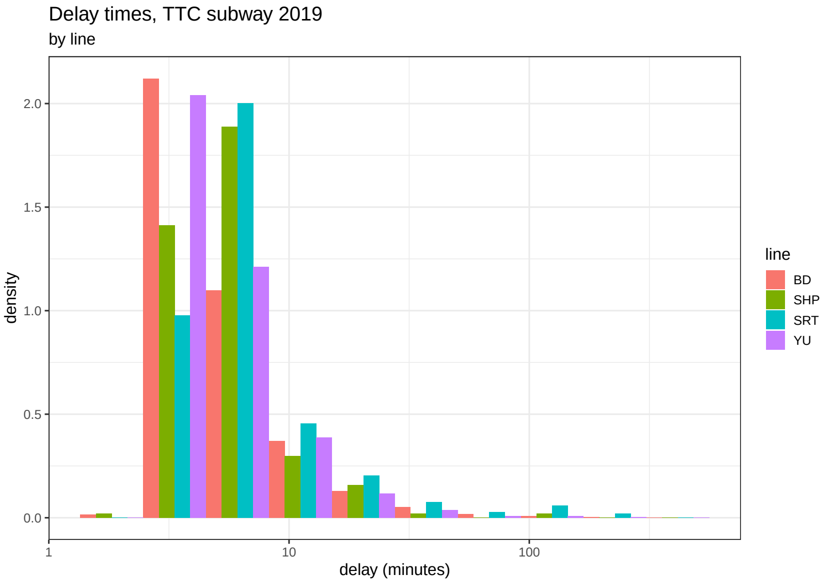
Shows the distribution of a **quantitative** variable

- Histograms show the frequency (count) of observations by value
- The range of values of a variables is divided into intervals ('bins') and then the number of observations in each bin is tabulated
- A histogram shows the count of observations in each bin with a rectangle of height equal to the count
- The x axis is the value bins, the y axis is the count/frequency (or proportion)


Female life expectancy, 2017
count
25
20
15
10
5
0
60
70
80
90
life expectancy


Delay times, TTC subway 2019
count
4000
3000
2000
1000
0
0
100
200
300
400
delay (minutes)

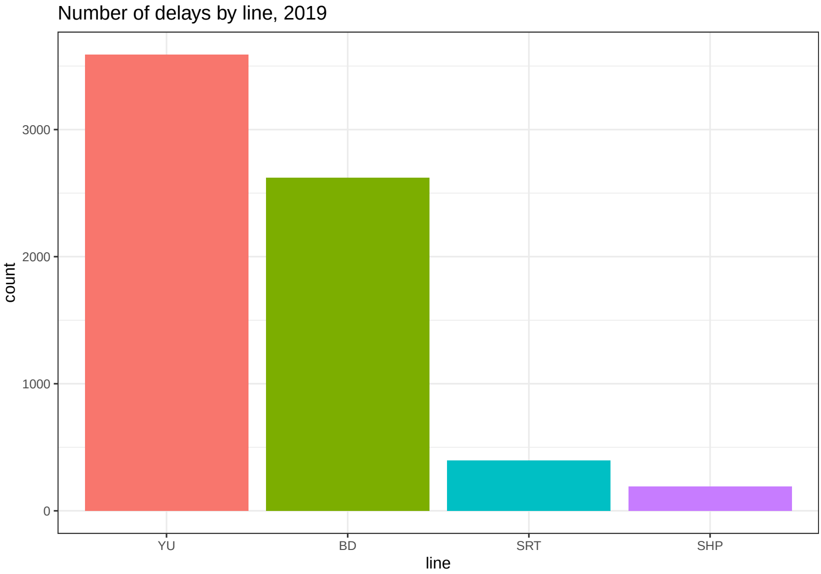
Making the histogram more informative



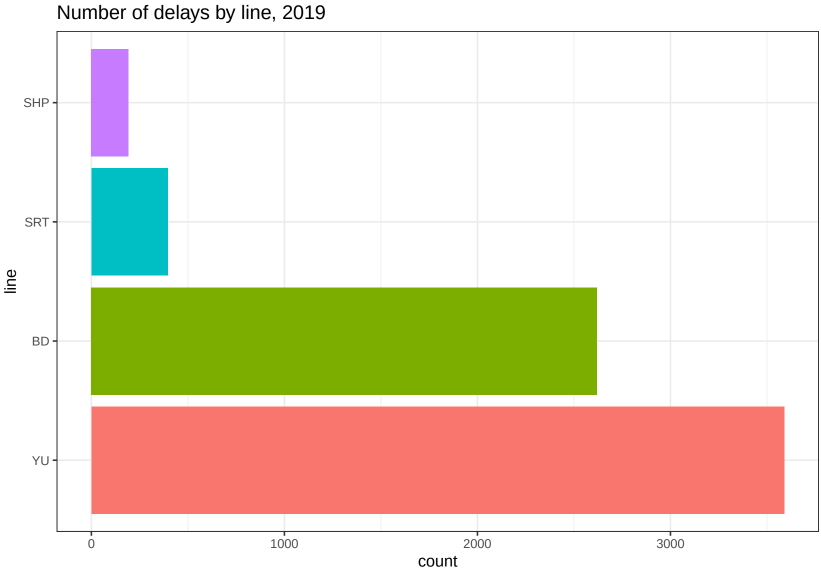
Bar charts

Shows summary measures across values of a **categorical** (qualitative) variable

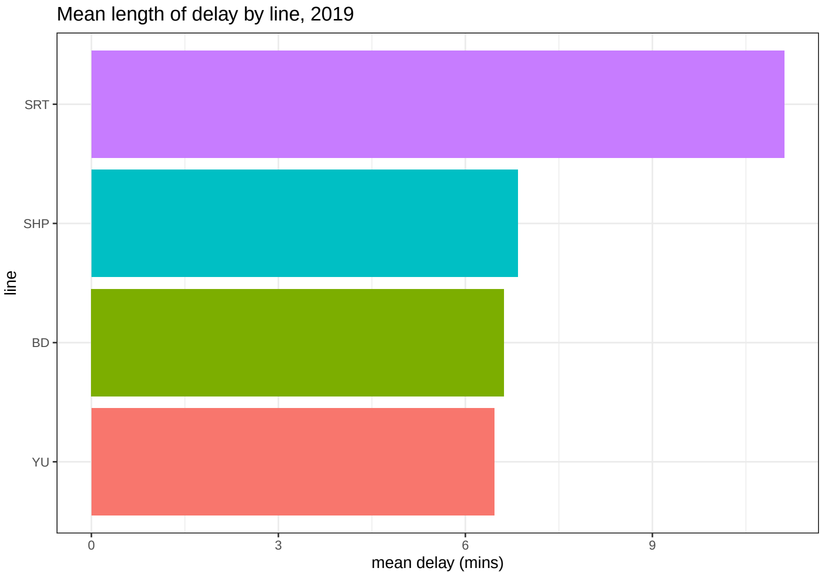
- Illustrate the value of a particular outcome in a particular category
- The ‘value’ can be counts, but could also be a summary measure (e.g. mean)
- The value is again shown by a rectangle of height equal to the value
- Bar carts can be plotted vertically or horizontally
- In the vertical setting, the x axis is the categories and the y axis is the value of the quantitative variable


Number of delays by line, 2019
count
3000
2000
1000
0
YU
BD
SRT
SHP
line

Same but horizontal



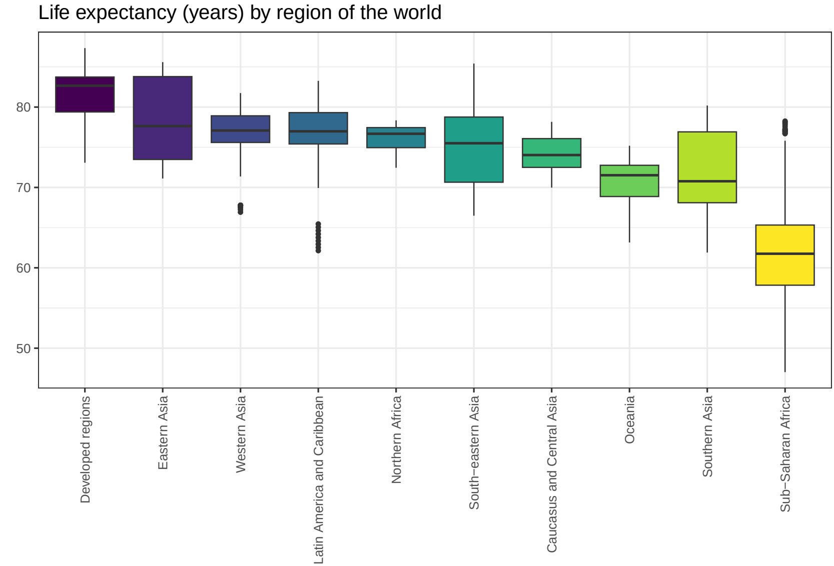
Showing mean delay time



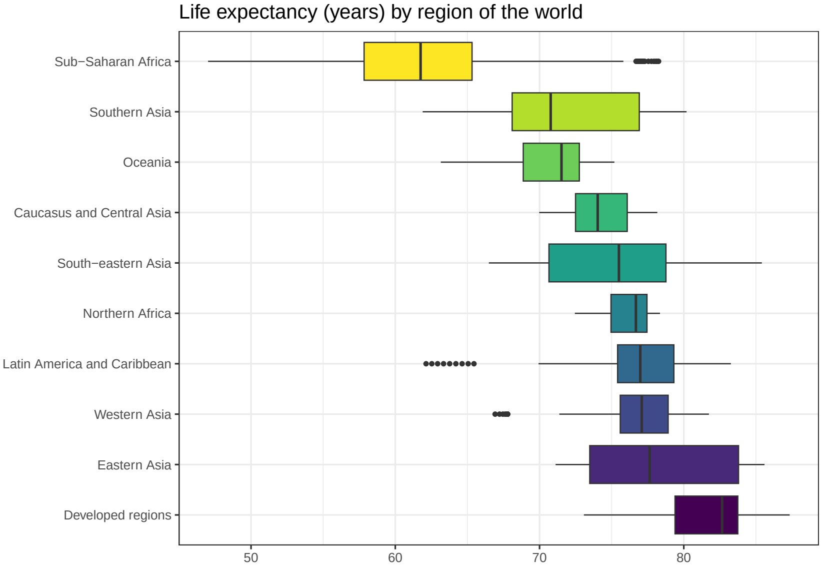
Box plots

Good for showing summaries of **quantitative** variables across different **categorical** groups.

- Visualizing quartiles (25/50/75 percentiles) of quantitative data
- Boxes show the IQR and median
- Whiskers show values outside IQR (in R/`ggplot`, default is 1.5*IQR)
- Outliers may be shown with individual dots
- In the vertical case, the `x` axis is the categories and the `y` axis is the quantitative variable


Life expectancy (years) by region of the world
80
70
60
50
Developed regions
Eastern Asia
Western Asia
Latin America and Caribbean
Northern Africa
South−eastern Asia
Caucasus and Central Asia
Oceania
Southern Asia
Sub−Saharan Africa

Could also do horizontal

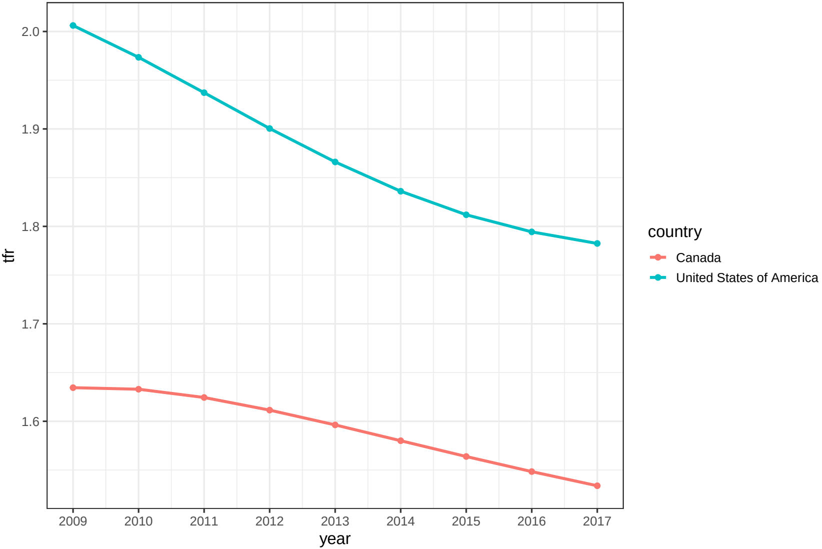


Line plots

Best used to describe values of a **quantitative** variable (on y axis) across sequential values of another **quantitative** variable on the x axis

- Plots a series of values of a quantitative variable connected together by a line
- Useful to visualize trends over time

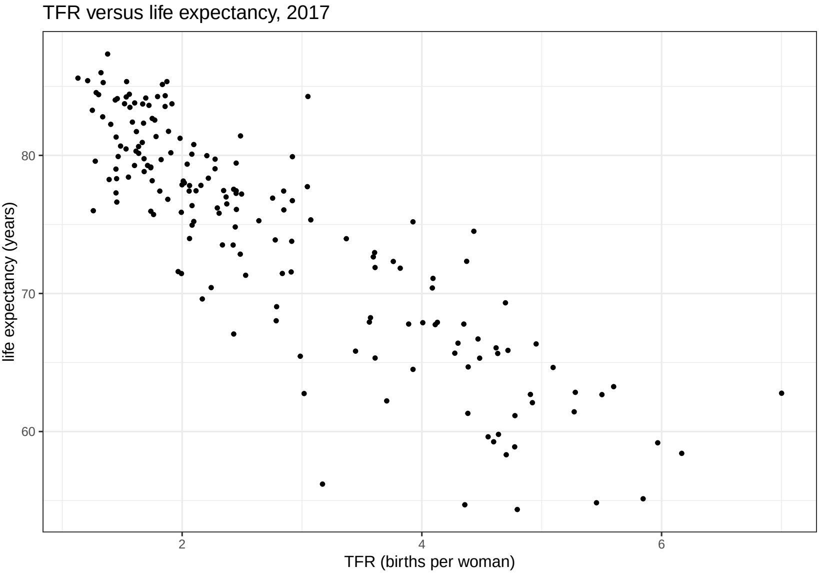
Total fertility rate, Canada and the US

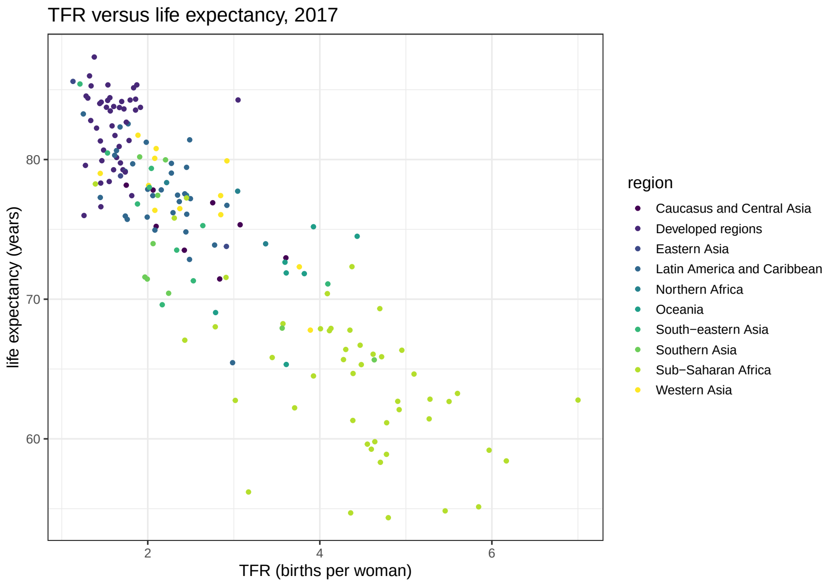


Scatter plots

Shows relationship between two different **quantitative** variables

- Uses dots to represent values for two different **quantitative** values
- The position of each dot on the x and y axis indicates values for an individual data point
- Extremely useful in visualizing the relationship between two quantitative variables


TFR versus life expectancy, 2017
life expectancy (years)
80
70
60
2
4
6
TFR (births per woman)


TFR versus life expectancy, 2017
life expectancy (years)
80
70
60
TFR (births per woman)
2
4
6
region
Caucasus and Central Asia
Developed regions
Eastern Asia
Latin America and Caribbean
Northern Africa
Oceania
South−eastern Asia
Southern Asia
Sub−Saharan Africa
Western Asia

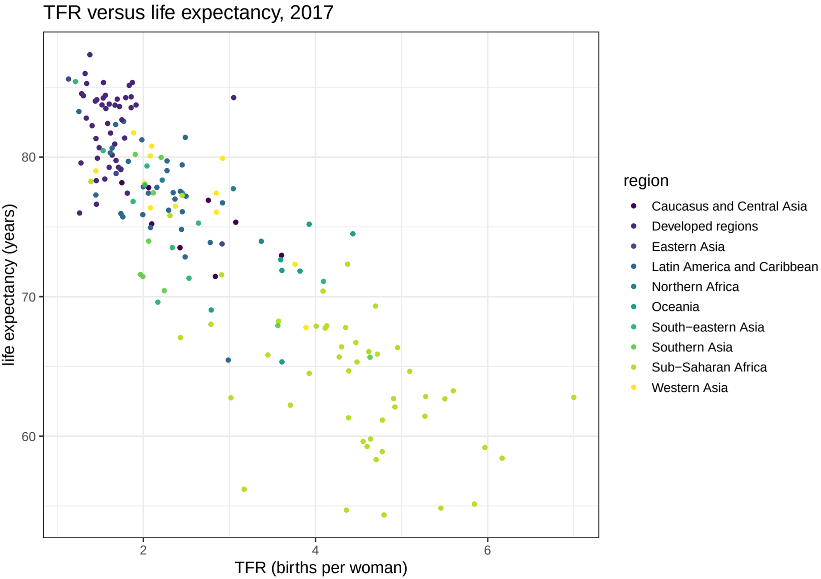
Introduction to `ggplot`

ggplot

- ggplot is the graphing package that goes with the tidyverse in R
- Very powerful to make a wide range of graphics
- Every graph so far this lecture was done in ggplot
- ggplot code works in layers, with each layer adding complexity
  - start with defining dataset and different variables
  - add on type of plot
  - scales
  - layout (facets)
  - themes, fonts, sizes...

More practice in lab, but here's a starting example

Reproducing the TFR verus life expectancy chart, colored by region



Data

```r
# read in the data
country_ind <- read_csv("../../data/country_indicators.csv")
country_ind
```

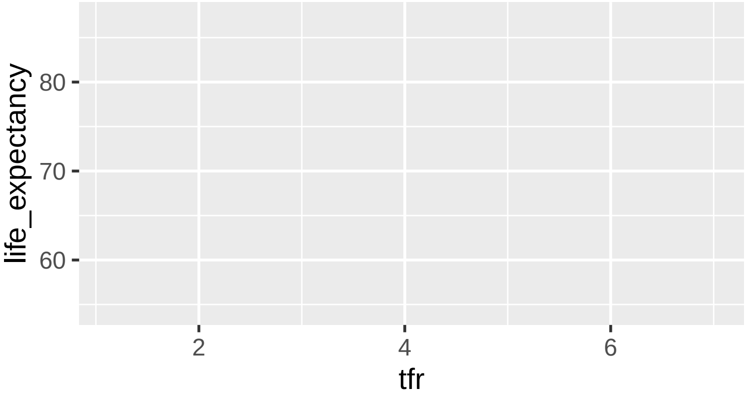
```
## # A tibble: 1,584 x 9
##    country_code country     region      year   tfr life_~1 child~2 mater~3   gdp
##    <chr>        <chr>       <chr>      <dbl> <dbl>   <dbl>   <dbl>   <dbl> <dbl>
##  1 AFG          Afghanistan Southern ~  2009  6.18    61.9    93.9     993 1502.
##  2 AFG          Afghanistan Southern ~  2010  5.98    62.5    90.0     954 1672.
##  3 AFG          Afghanistan Southern ~  2011  5.77    63      86.3     905 1627.
##  4 AFG          Afghanistan Southern ~  2012  5.56    63.5    82.9     858 1773.
##  5 AFG          Afghanistan Southern ~  2013  5.36    64.0    79.6     810 1808.
##  6 AFG          Afghanistan Southern ~  2014  5.16    64.5    76.6     786 1796.
##  7 AFG          Afghanistan Southern ~  2015  4.98    64.9    73.8     701 1767.
##  8 AFG          Afghanistan Southern ~  2016  4.80    65.3    71.2     673 1757.
##  9 AFG          Afghanistan Southern ~  2017  4.63    65.7    68.8     638 1758.
## 10 ALB          Albania     Developed~  2009  1.65    79.0    16.7      20 9525.
## # ... with 1,574 more rows, and abbreviated variable names 1: life_expectancy,
## #   2: child_mort, 3: maternal_mort
```

```r
# filter to just be 2017
country_ind_2017 <- country_ind %>% filter(year==2017)
```

A blank canvas

aes stands for aesthetic and tells ggplot the main characteristics of your plot (x, y, and if the color or fill vary by group)

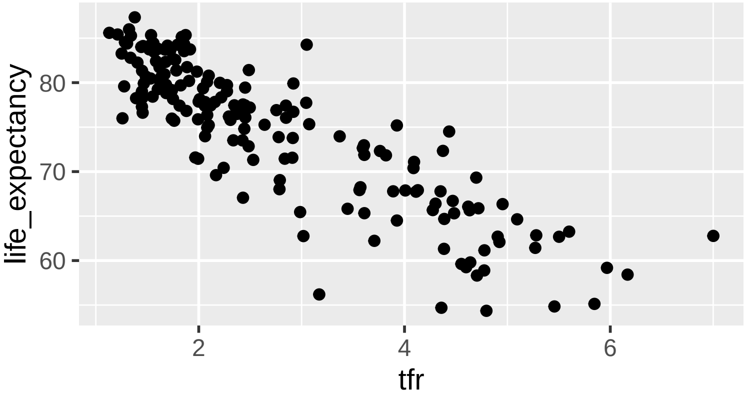
```
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy))

#print
plot1
```



Add the points

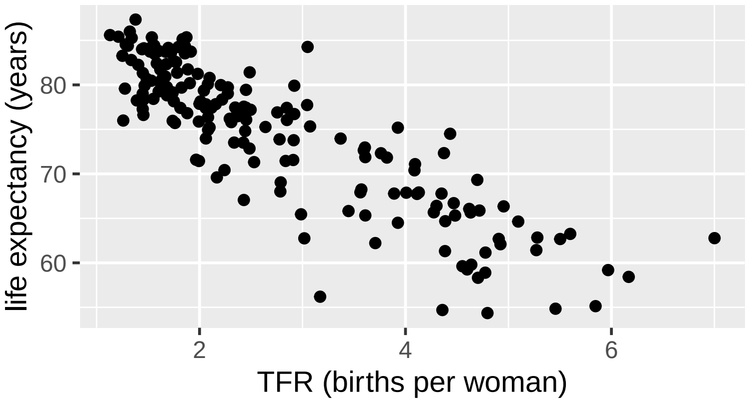
Add layers with ggplot using the +

```
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy)) +
  geom_point()

plot1
```



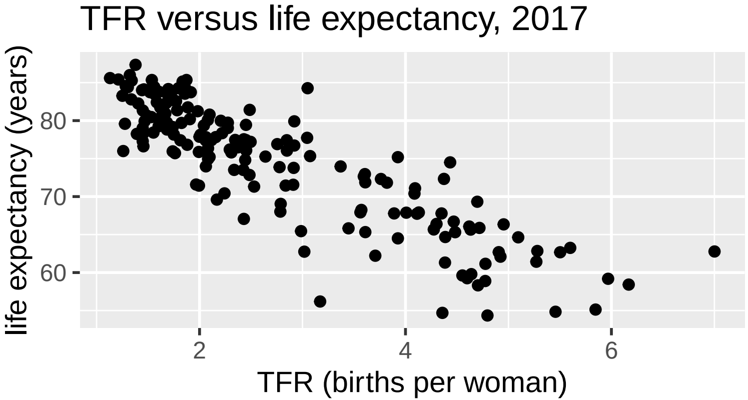
Tidy up labels

```
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy)) +
  geom_point()+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")

plot1
```



Title

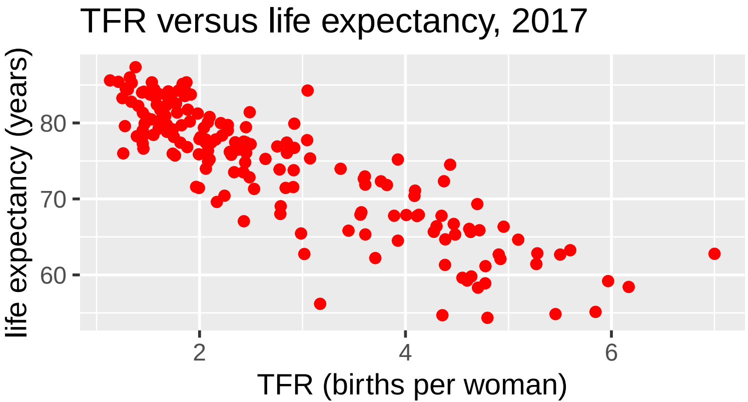
```
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy)) +
  geom_point()+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")+
  ggtitle("TFR versus life expectancy, 2017")

plot1
```



Change color of points

to see all colors, type `colors()`

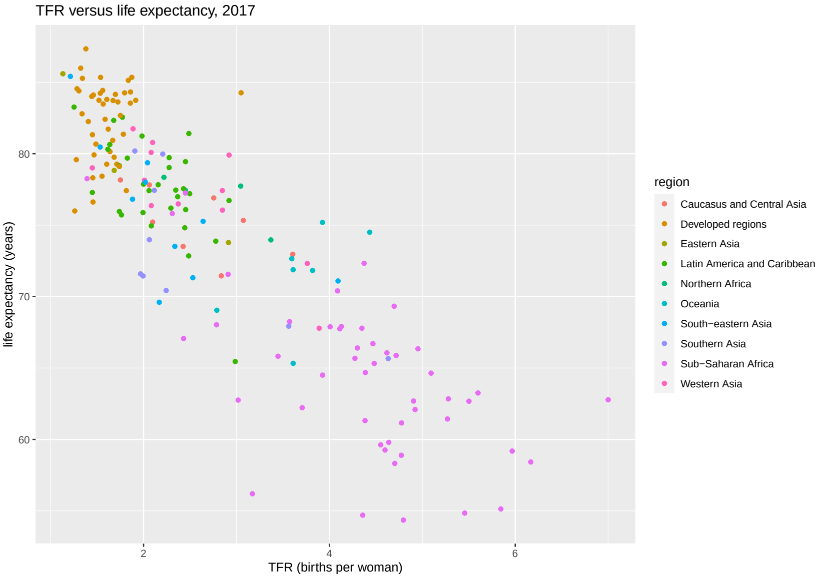
```r
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy)) +
  geom_point(color = "red")+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")+
  ggtitle("TFR versus life expectancy, 2017")

plot1
```



Coloring by group

This goes in the `aes()` because it **depends on the data**

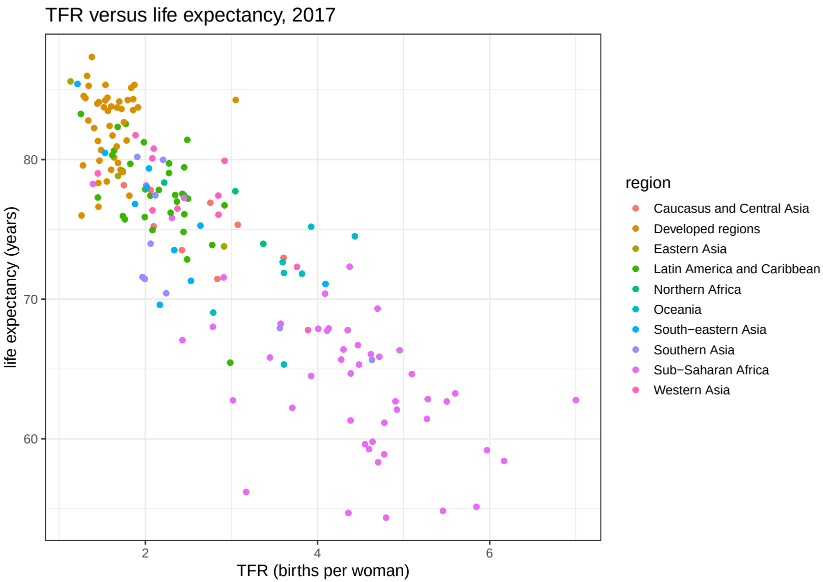
```r
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy, color = region)) +
  geom_point()+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")+
  ggtitle("TFR versus life expectancy, 2017")

plot1
```



TFR versus life expectancy, 2017
life expectancy (years)
60
70
80
TFR (births per woman)
2
4
6
region
● Caucasus and Central Asia
● Developed regions
● Eastern Asia
● Latin America and Caribbean
● Northern Africa
● Oceania
● South−eastern Asia
● Southern Asia
● Sub−Saharan Africa
● Western Asia

Change theme (optional) and size of points

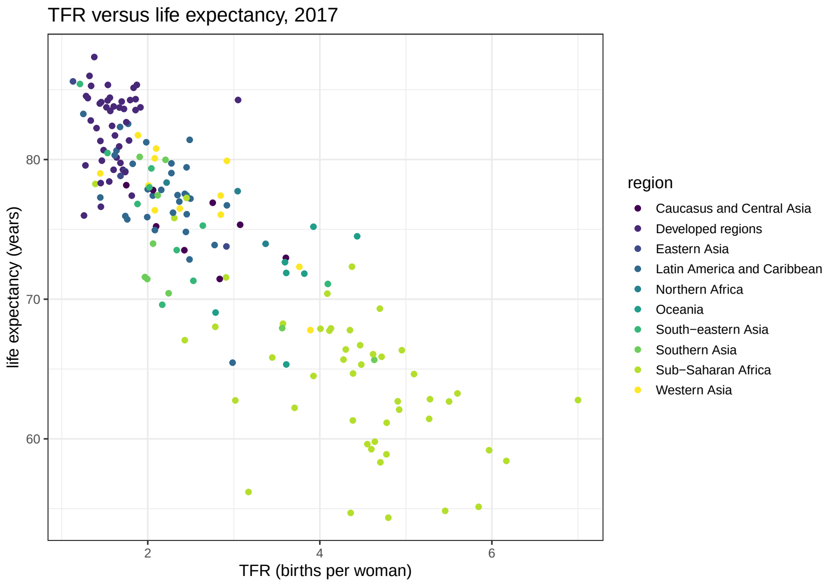
```r
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy, color = region)) +
  geom_point(size =2)+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")+
  ggtitle("TFR versus life expectancy, 2017")+
  theme_bw(base_size = 14)

plot1
```

TFR versus life expectancy, 2017
life expectancy (years)
80
70
60
2
4
6
TFR (births per woman)
region
Caucasus and Central Asia
Developed regions
Eastern Asia
Latin America and Caribbean
Northern Africa
Oceania
South−eastern Asia
Southern Asia
Sub−Saharan Africa
Western Asia

Change color scheme

`viridis` and `brewer` both good options

```
plot1 <- ggplot(data = country_ind_2017, aes(x = tfr, y = life_expectancy, color = region)) +
  geom_point(size =2)+
  xlab("TFR (births per woman)")+
  ylab("life expectancy (years)")+
  ggtitle("TFR versus life expectancy, 2017")+
  theme_bw(base_size = 14)+
  scale_color_viridis_d()

plot1
```



TFR versus life expectancy, 2017

Summary

- EDA and data visualization is often just as informative and important as statistical analysis
- It is essential to understand the structure of your data, missing-ness, any outliers/issues, and the raw patterns in your data before deciding on your statistical analysis
- Plot, plot, plot
- Practice, practice, practice

Plots:

- Bar charts for categorical/qualitative variables
- Histograms, boxplots for one quantitative variable (potentially across multiple categories)
- Line plots and scatter plots for two quantitative variables (line plot when one is sequential)