Econ 106

Lecture 5
Fall 2024

slides adapted from
https://jhudatascience.org/tidyversecourse/dataviz.html#about-this-course-3

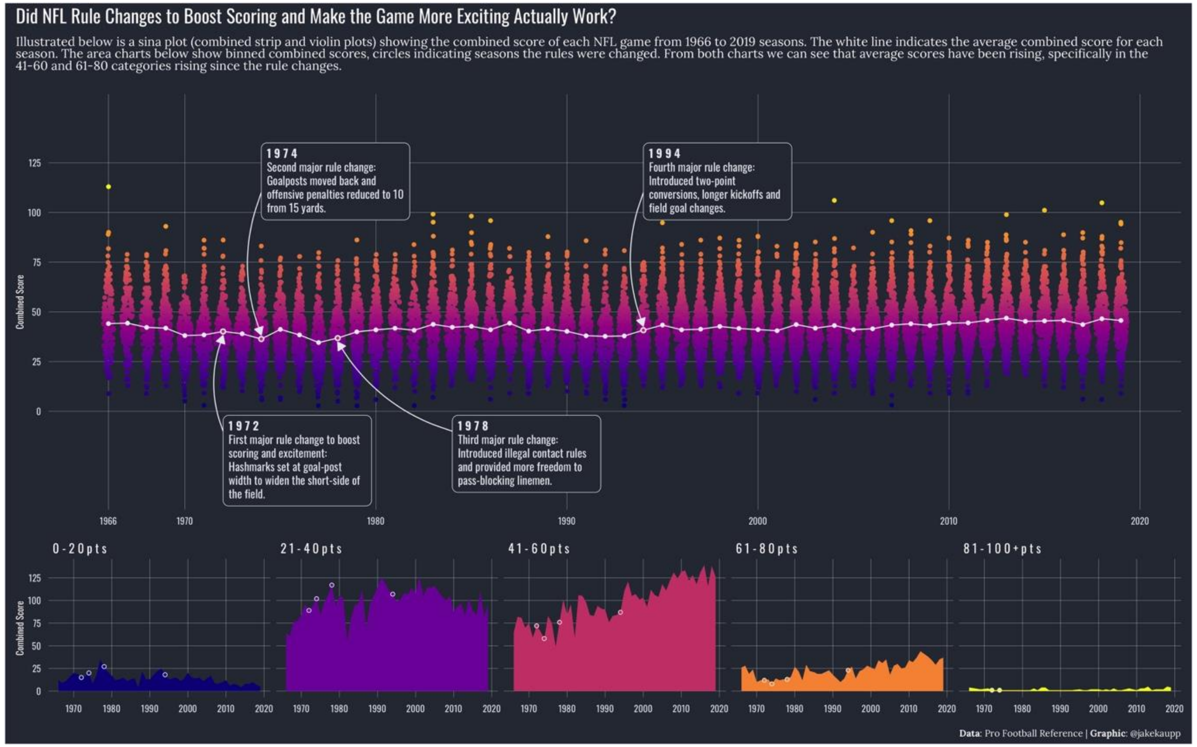
Reminders

- Lab 1 is due Sunday 11:59pm
- Poll everywhere scores are in the canvas gradebook (full credit if >50% correct)

https://pollev.com/vsovero

#tidytuesday

Football is happening now, right?



Did NFL Rule Changes to Boost Scoring and Make the Game More Exciting Actually Work?

Illustrated below is a sina plot (combined strip and violin plots) showing the combined score of each NFL game from 1966 to 2019 seasons. The white line indicates the average combined score for each season. The area charts below show binned combined scores, circles indicating seasons the rules were changed. From both charts we can see that average scores have been rising, specifically in the 41-60 and 61-80 categories rising since the rule changes.

1974 Second major rule change: Goalposts moved back and offensive penalties reduced to 10 from 15 yards.

1994 Fourth major rule change: Introduced two-point conversions, longer kickoffs and field goal changes.

1972 First major rule change to boost scoring and excitement: Hashmarks set at goal-post width to widen the short-side of the field.

1978 Third major rule change: Introduced illegal contact rules and provided more freedom to pass-blocking linemen.

Data: Pro Football Reference | Graphic: @jakekaupp

https://x.com/jakekaupp/status/1226556813476270080?s=20

Outline

- Visualization Background
- Introduction to ggplot2
  - Basic elements (data, aesthetics, geoms)
  - color as information

“A picture is worth a thousand words"

- Replace (or complement) ‘typical’ tables of data or statistical results with figures that are more compelling and accessible.
- Two main advantages of data visualization:
  - Facilitates comparisons
  - Helps identify trends

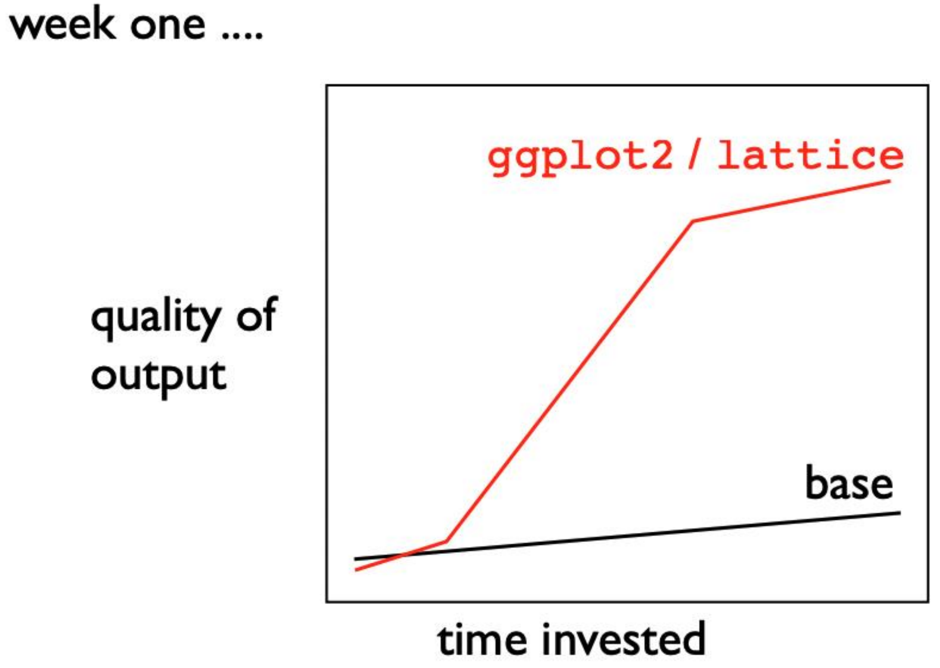
Why ggplot2?

- **Reproducibility**
- Part of the **tidyverse**
- **Pretty** by default
- Customizable

http://varianceexplained.org/r/why-I-use-ggplot2/

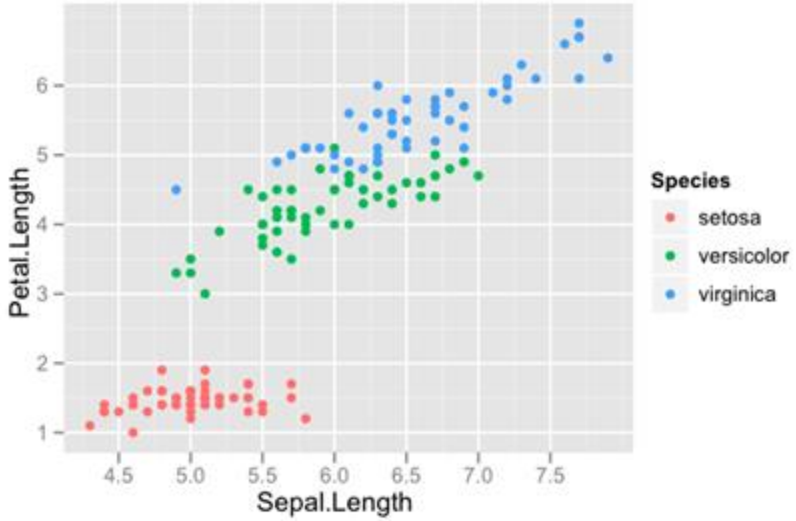
But first, some truth about ggplot2

- Full disclosure: it’s not the easiest to get the hang of
- Simple visualizations are easier using base R



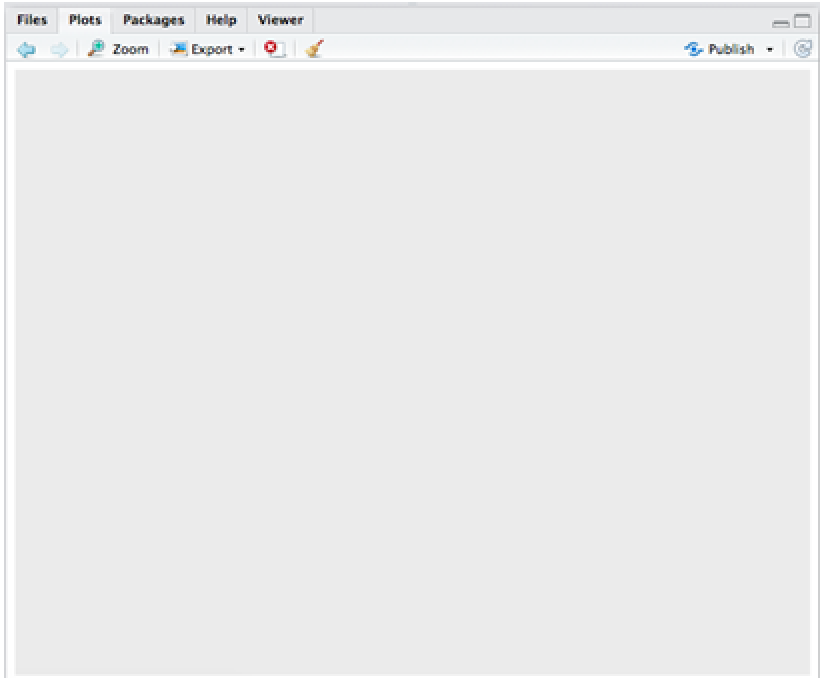
Basic Elements of a Data Visualization

1. **Data**: the data you want to plot
2. **Layout**: mapping variables on the plot
3. **Data display**: how you want the data to be visualized (points, lines, bars, etc.)



1. Specify data

ggplot(data = diamonds)

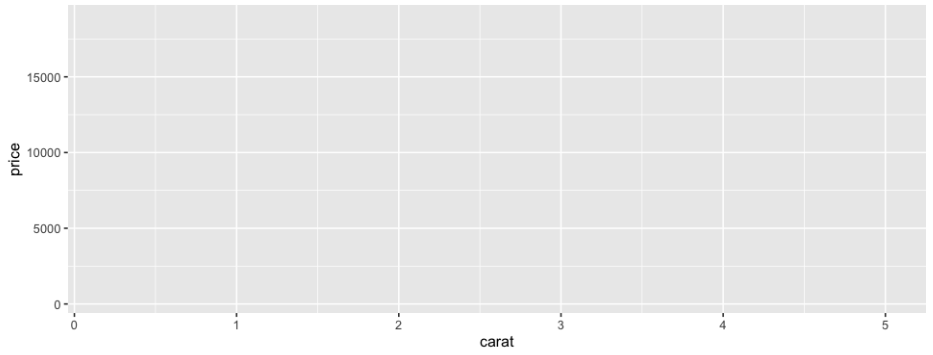


- **ggplot**(): Creates a plot object
- **data** specifies what data table you will use
- **Output**: blank plot

2. Specify Layout

```
ggplot(data = diamonds,
          mapping = aes(x = carat, y = price))
```

- **mapping** argument specifies what should go on the x and y axes
  - **x** = x axis variable
  - **y** = y axis variable
- **aes**() function is required whenever you reference specific variables in your data



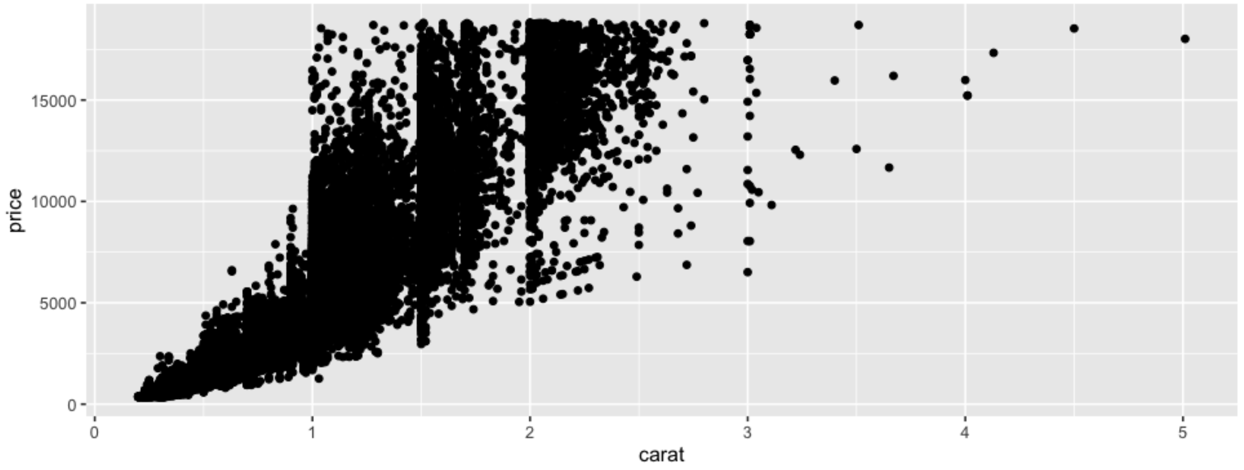
- **Output**: plot with axes, no data

3. Specify Data Display

```
ggplot(data = diamonds,
         mapping = aes(x = carat,  y = price)) +
geom_point()
```

- requires:
  - **+ operator**
  - **geom_point**()

- **Output**: scatterplot



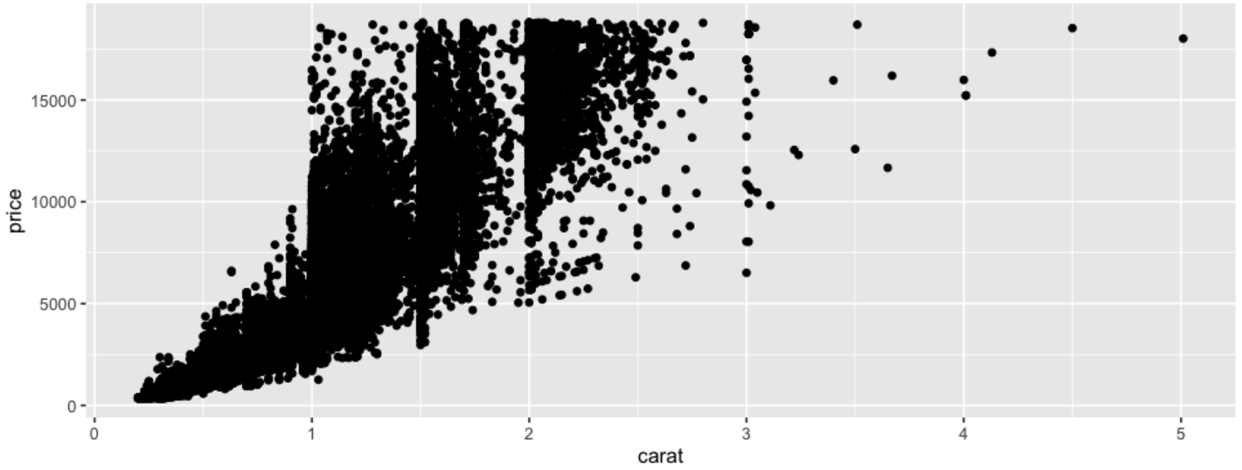
ggplot2 functions

- **ggplot**(): creates a ggplot object
- **aes**() function is required whenever you reference specific variables in your data
- **geom_XXX**(): draws points/lines etc.
- **+**: adds components to plot
  - Modular structure

Scatter Plot

https://pollev.com/vsovero

```
ggplot(data = diamonds,
        mapping = aes(x = carat,  y = price)) +
geom_point()
```

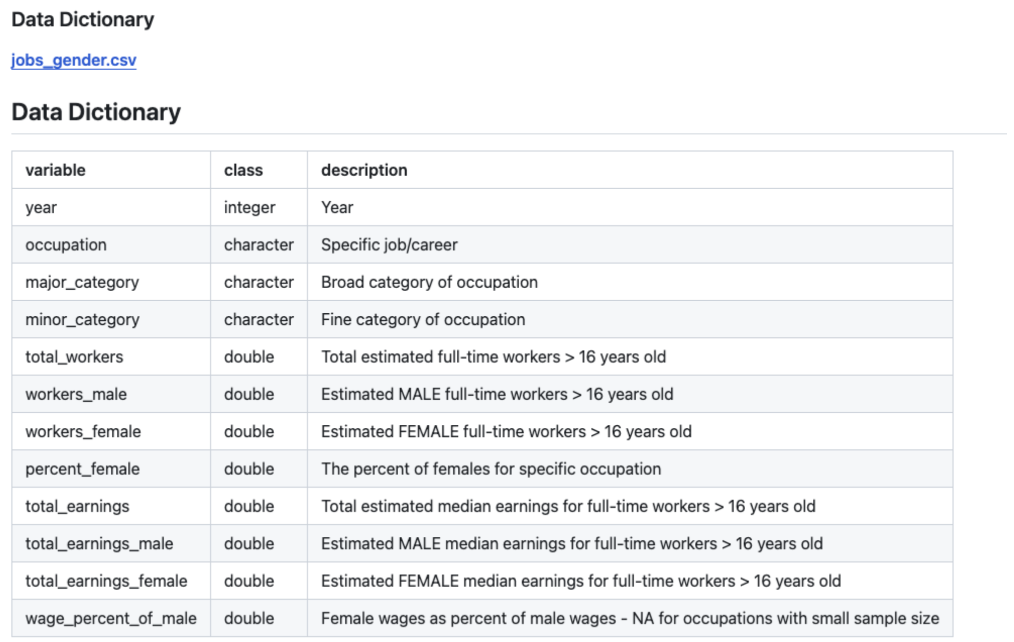


Data Example

We are going to work with the gender gap data:

```
jobs_gender <-
read_csv("https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2019/2019-03-05/jobs_gender.csv")
```

tidytuesday data



Data Dictionary

jobs_gender.csv

Data Dictionary

variable	class	description
year	integer	Year
occupation	character	Specific job/career
major_category	character	Broad category of occupation
minor_category	character	Fine category of occupation
total_workers	double	Total estimated full-time workers > 16 years old
workers_male	double	Estimated MALE full-time workers > 16 years old
workers_female	double	Estimated FEMALE full-time workers > 16 years old
percent_female	double	The percent of females for specific occupation
total_earnings	double	Total estimated median earnings for full-time workers > 16 years old
total_earnings_male	double	Estimated MALE median earnings for full-time workers > 16 years old
total_earnings_female	double	Estimated FEMALE median earnings for full-time workers > 16 years old
wage_percent_of_male	double	Female wages as percent of male wages - NA for occupations with small sample size

Exercise

Create a scatter plot with total_earnings on the x-axis and wage_percent_of_male on the y-axis

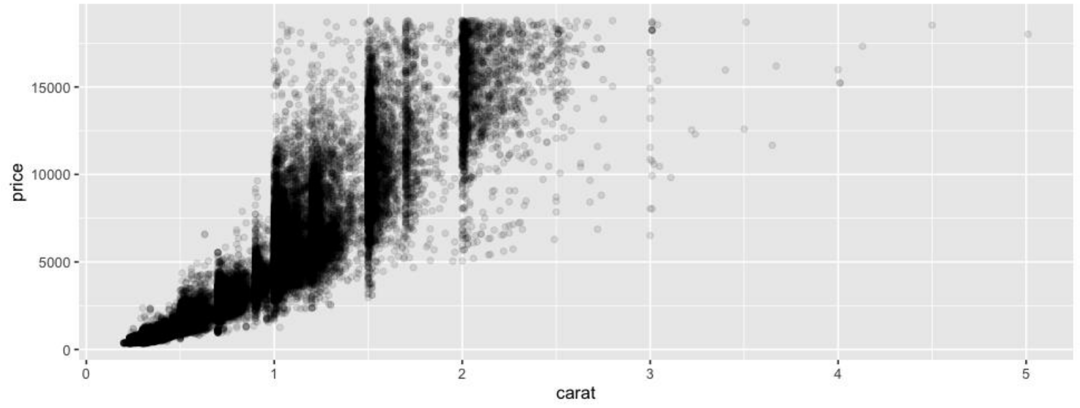
Adjusting Plot Settings

- **color**: color of 1-d objects
- **fill**: fill color of 2-d objects
- **linetype**: how lines should be drawn (solid, dashed, dotted, etc.)
- **shape**: shape of markers in scatter plots
- **size**: how large objects appear
- **alpha**: transparency of objects (value between 0 and 1)

Transparency

```
ggplot(data = diamonds,
        mapping= aes(x = carat,  y =price)) +
geom_point(alpha = 0.1)
```

- Add argument to **geom_point**()
- Reduce transparency of points
- **Input**: **alpha** = 0.1
  - 1/10 opacity
  - Range: 0-1



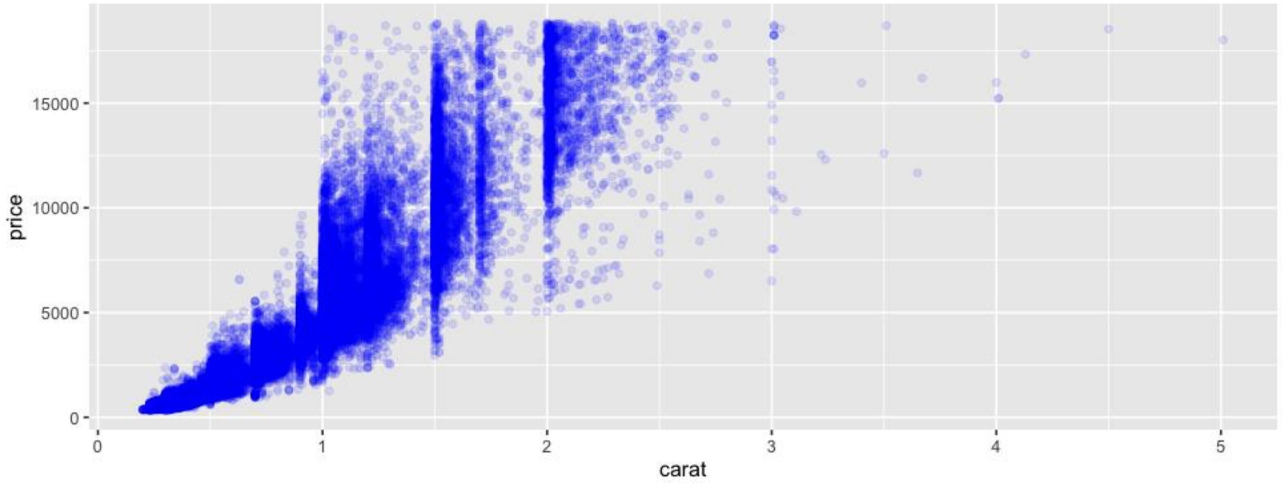
Color

https://pollev.com/vsovero

```
ggplot(data = diamonds,
            mapping= aes(x = carat, y = price))+
geom_point(alpha = 0.1,
             color = "blue")
```

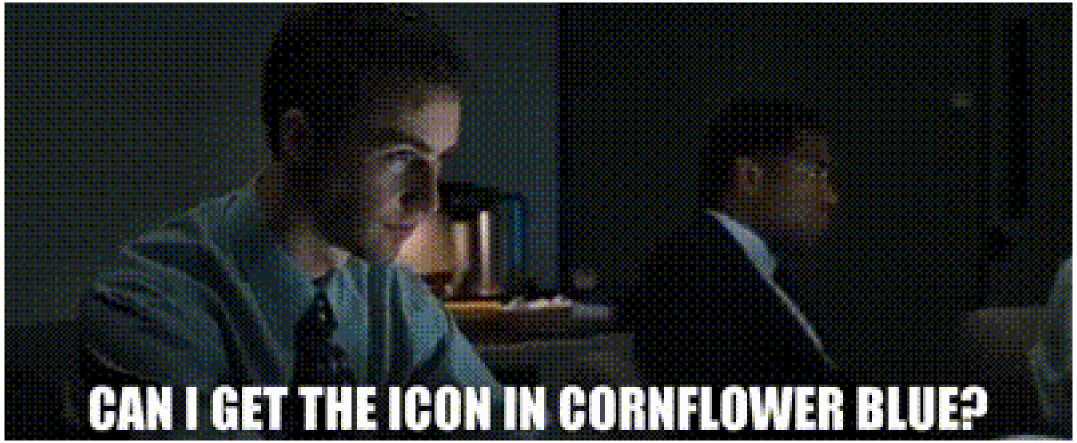
- Change point colors to blue
- **Input**: color argument
- **Output**: blue points

Color reference chart:
http://sape.inf.usi.ch/quick-reference/ggplot2/colour



Exercise: scatter plots and color

Your boss requires that all scatter plots use triangle shapes and the cornflower blue color (“cyan”). Adjust your scatter plot of total_earnings and wage_percent_of_male accordingly.



Dplyr and ggplot

- Oftentimes you will use dplyr to create a new data frame, then plot the results using ggplot
- Remember to put in the name of the new data frame in your ggplot()

```
very_good_cut<-diamonds%>%
filter(cut== "Very Good")


ggplot(data = very_good_cut,
                mapping= aes(x = carat, y = price))+
geom_point(alpha = 0.1,
                 color = "blue")
```

Dplyr and ggplot

- Oftentimes you will use dplyr to create a new data frame, then plot the results using ggplot
- You can also "pipe" in the results directly into ggplot (removing the data argument inside ggplot())

```
diamonds%>%
filter(cut== "Very Good") %>%
ggplot(mapping= aes(x = carat, y = price))+
geom_point(alpha = 0.1,
                      color = "blue")
```

Exercise

- Filter for occupations in computer, engineering, and science
- create a scatter plot of total_earnings and wage_percent_of_male

Next up: Line Graphs

- Line graphs are probably the hardest graph to generate correctly (not look like a hot mess)
- To get it right, most data requires wrangling (get your data ready before ggplot) or grouping (within ggplot)

Gapminder Data

- part of gapminder package
- For 185 countries in the world, the package provides values for life expectancy, GDP per capita, every year from 1960 to 2016.

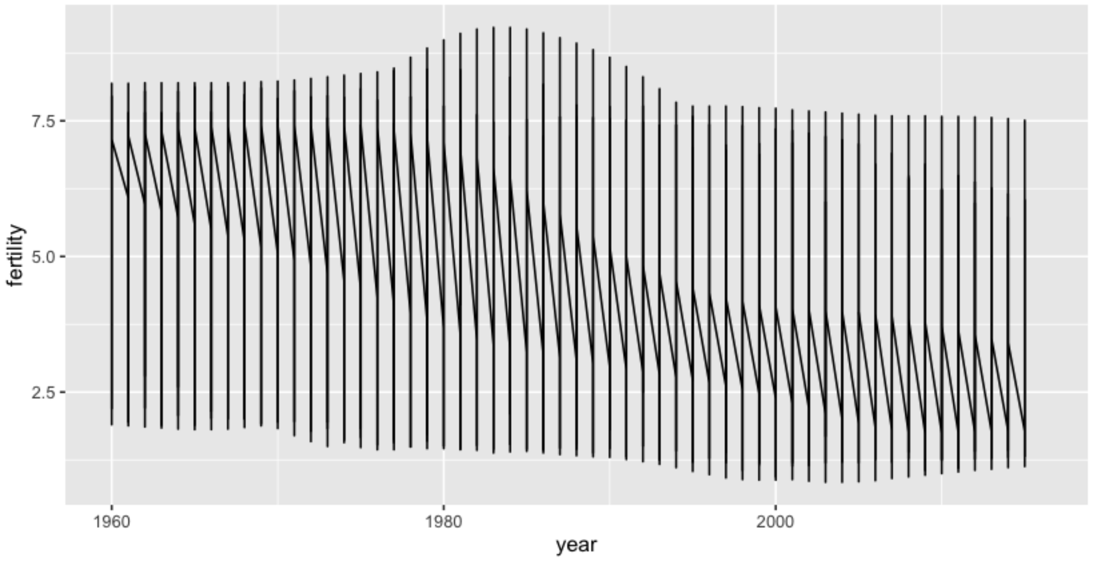
```
library(dslabs)

data(gapminder)
```

Fertility over Time

```
ggplot(data = gapminder,
            mapping=aes(x=year, y=fertility)) +
 geom_line()
```

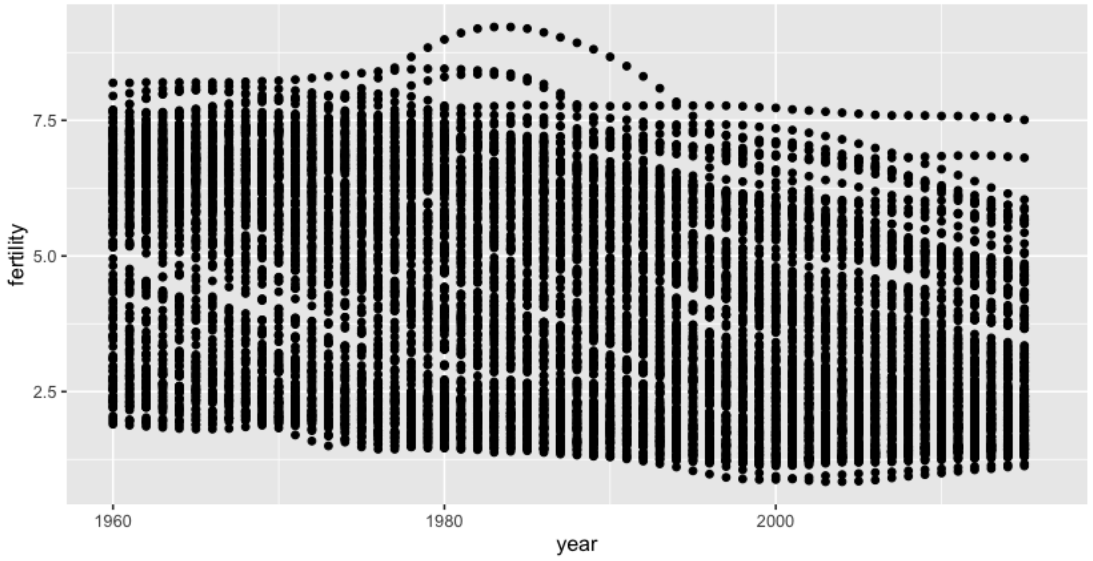
Well, this doesn't look right.
What happened?



Fertility over Time (scatter plot)

- Displaying the data as a scatter plot can help us figure out what's going on
- There are many values of fertility for each year (one for each country)
- It doesn't make sense to draw a single line through all these points

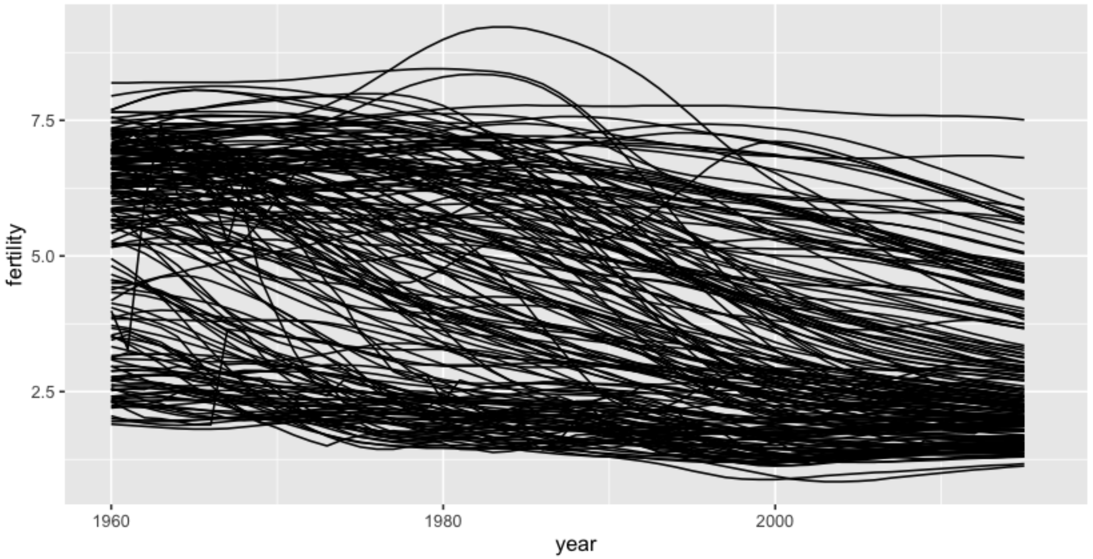
```
ggplot(data = gapminder,
             mapping=aes(x=year, y=fertility)) +
 geom_point()
```



Fertility over Time (line for each country)

```
ggplot(data = gapminder,
            mapping=aes(x=year, y=fertility)) +
geom_line(aes(group=country))
```

- We need to tell ggplot to create a line for each country using the group argument



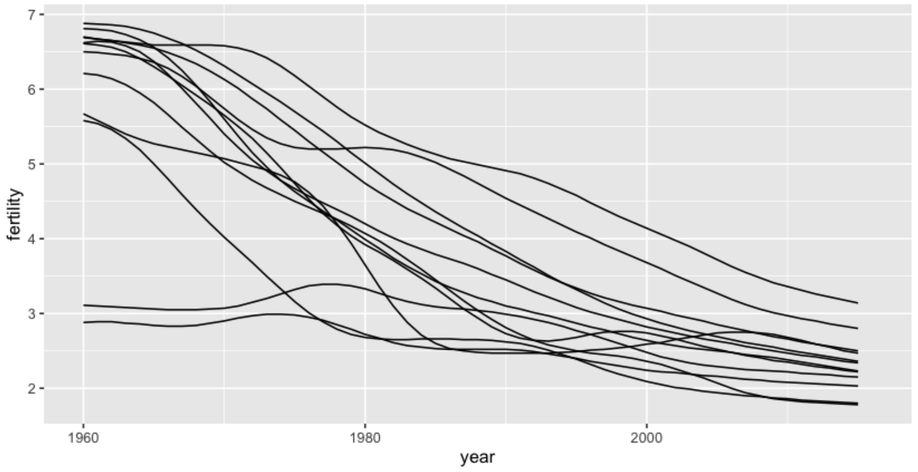
Ok, but too many countries!

Fertility over Time (line for each country)

- First, only select countries in South America, then ggplot

```
South_America <- gapminder %>%
   filter(region=="South America")


ggplot(data = South_America,
            mapping=aes(x=year, y=fertility)) +
 geom_line(aes(group=country))
```

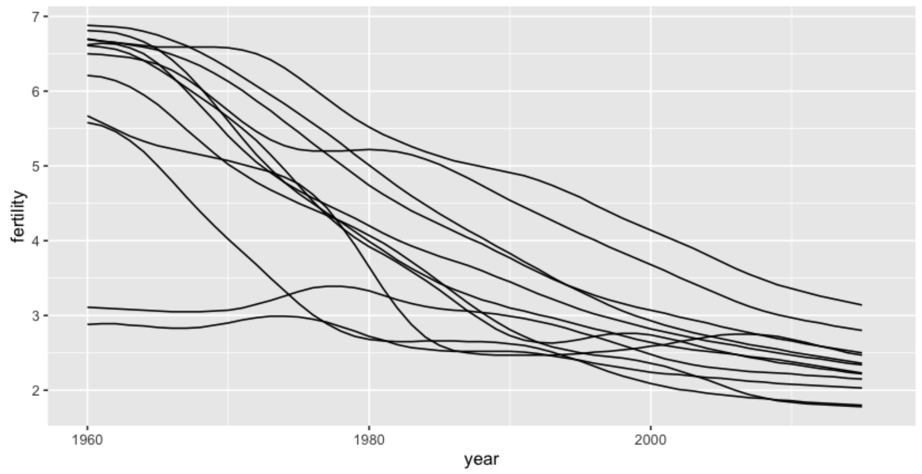


https://pollev.com/vsovero

Fertility over Time (line for each country)

- We can use pipes to wrangle and ggplot all at once

```
    gapminder %>%
  filter(region=="South America") %>%
ggplot(mapping=aes(x=year, y=fertility)) +
 geom_line(aes(group=country))
```



Exercise: line graphs

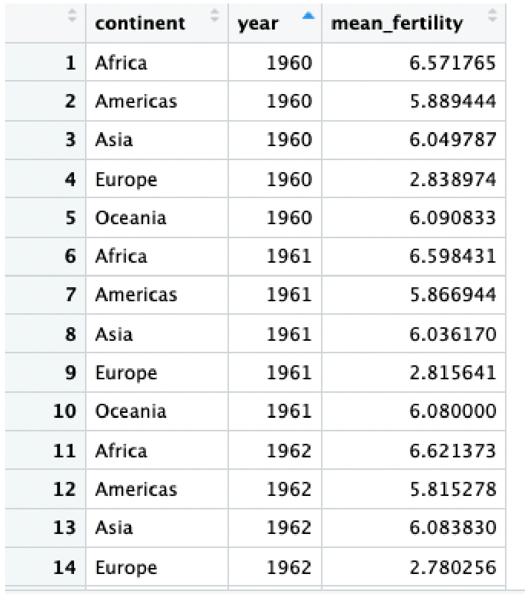
Create a line graph of occupations in architecture and engineering occupations showing percent_female by year

https://pollev.com/vsovero

Fertility over Time (line for each continent)

- We need to collapse the data to continent by year using **group_by**() and **summarize**()

```
continent_summary<-gapminder %>%
group_by(continent, year) %>%
summarize(mean_fertility=mean(fertility, na.rm=TRUE)
```

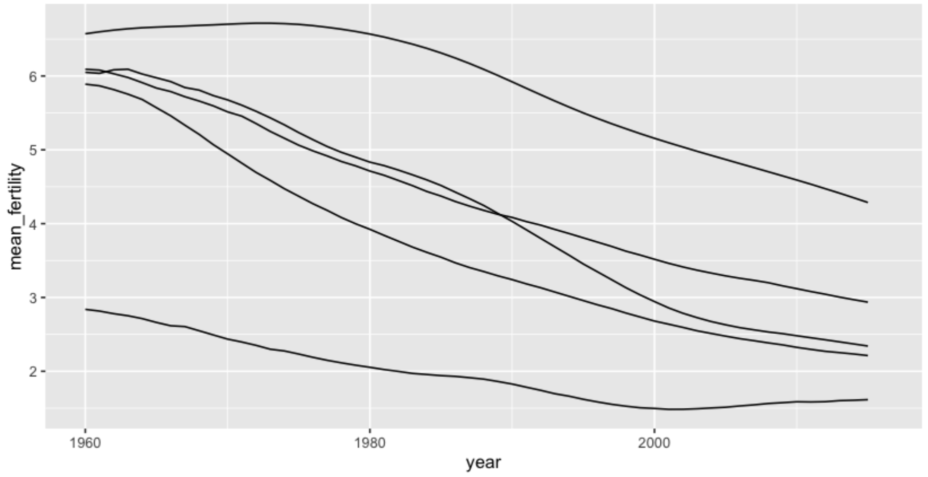


	continent	year	mean_fertility
1	Africa	1960	6.571765
2	Americas	1960	5.889444
3	Asia	1960	6.049787
4	Europe	1960	2.838974
5	Oceania	1960	6.090833
6	Africa	1961	6.598431
7	Americas	1961	5.866944
8	Asia	1961	6.036170
9	Europe	1961	2.815641
10	Oceania	1961	6.080000
11	Africa	1962	6.621373
12	Americas	1962	5.815278
13	Asia	1962	6.083830
14	Europe	1962	2.780256

Fertility over Time (line for each continent)

- Then we ggplot using the group argument

```
ggplot(data=continent_summary,
mapping=aes(x=year, y=mean_fertility)) +
 geom_line(aes(group=continent))
```



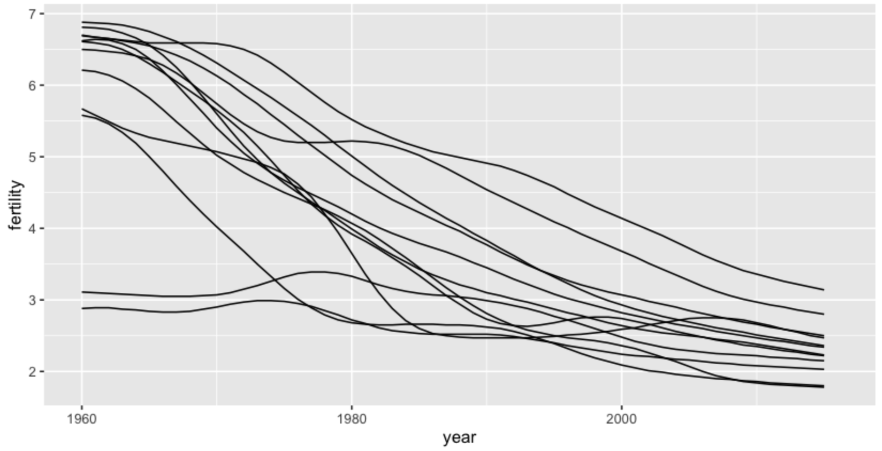
Exercise

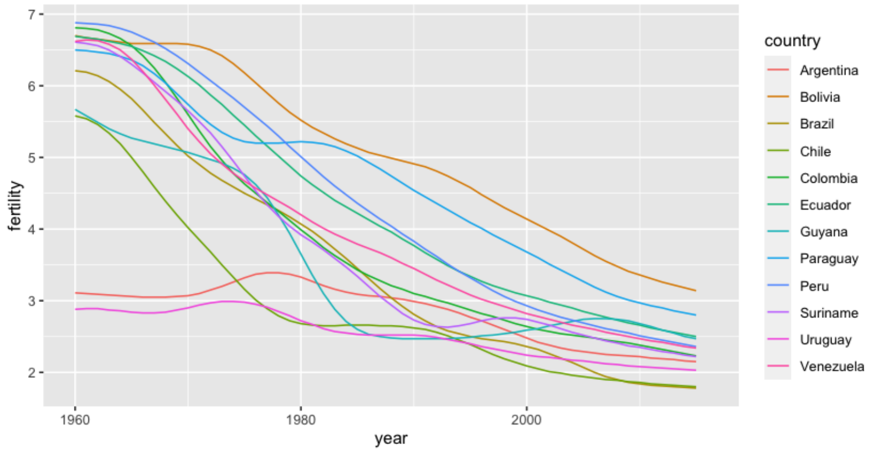
- calculate the mean percent_female by year and by minor_category for occupations in computer, engineering and science
- plot as a line graph

Color Mapping

```
gapminder %>%
   filter(region=="South America") %>%
ggplot(mapping=aes(x=year, y=fertility)) +
 geom_line(aes(group=country))
```

```
gapminder %>%
   filter(region=="South America") %>%
ggplot(mapping=aes(x=year, y=fertility)) +
 geom_line(aes(color=country))
```





Exercise

- calculate the mean percent_female by year and by minor_category for occupations in computer, engineering and science
- plot as a line graph
- color the lines by minor_category

https://pollev.com/vsovero