# The Normal Curve

## EDP 613

### Week 5

# Prepping a New R Script

1. Open up a blank R script using the menu path **File > New File > R Script**.

2. Save this script as `whatever.R` (replacing the term `whatever`) in your R folder. Remember to note where the file is!

3. After you have saved this file as `whatever.R`, go to the menu and this week try running running this shortcut to **Session > Set Working Directory > To Source File Location** at the top of your script

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

# Getting ready for this session

Get the files

> - `Box Office.csv`
> - `teampolview.csv`

and save it in the same location as this script.

> - Install the packages `viridis` and `patchwork`.
>
> - Load up `tidyverse` and `viridis`

This week try using `pacman` to do it

```
pacman::p_load(tidyverse,
               patchwork,
               viridis)
```

# Last week's R activity

# Load up data

```
boxoffice <- read_csv("Box Office.csv")
```

# Before we go on

Thes solutions are just one of **many** ways to get to the actual answer. Your work may and will likely vary.

```
boxoffice %>%
  arrange(Rank) %>%
  head(5) %>%
  summarize(mean_pos =
              mean(AllPos,
                   na.rm = TRUE)) %>%
  pull()
```

```
[1] 205.2
```

```r
boxoffice %>%
  arrange(Rank) %>%
  tail(5) %>%
  summarize(mean_neg =
              mean(AllNeg,
                   na.rm = TRUE)) %>%
  pull()
```

```
[1] 33.2
```

```
boxoffice %>%
  group_by(year) %>%
  count(name = "number of movies") %>%
  ungroup()
```

```
# A tibble: 55 × 2
    year `number of movies`
   <dbl>              <int>
 1  1937                  1
 2  1939                  1
 3  1940                  2
 4  1942                  1
 5  1950                  1
 6  1953                  1
 7  1955                  1
 8  1956                  1
 9  1961                  1
10  1964                  1
# … with 45 more rows
```

## Save as a variable

```
boxoffice_annualnum <-
  boxoffice %>%
  group_by(year) %>%
  count(name = "number of movies") %>%
  ungroup()
```

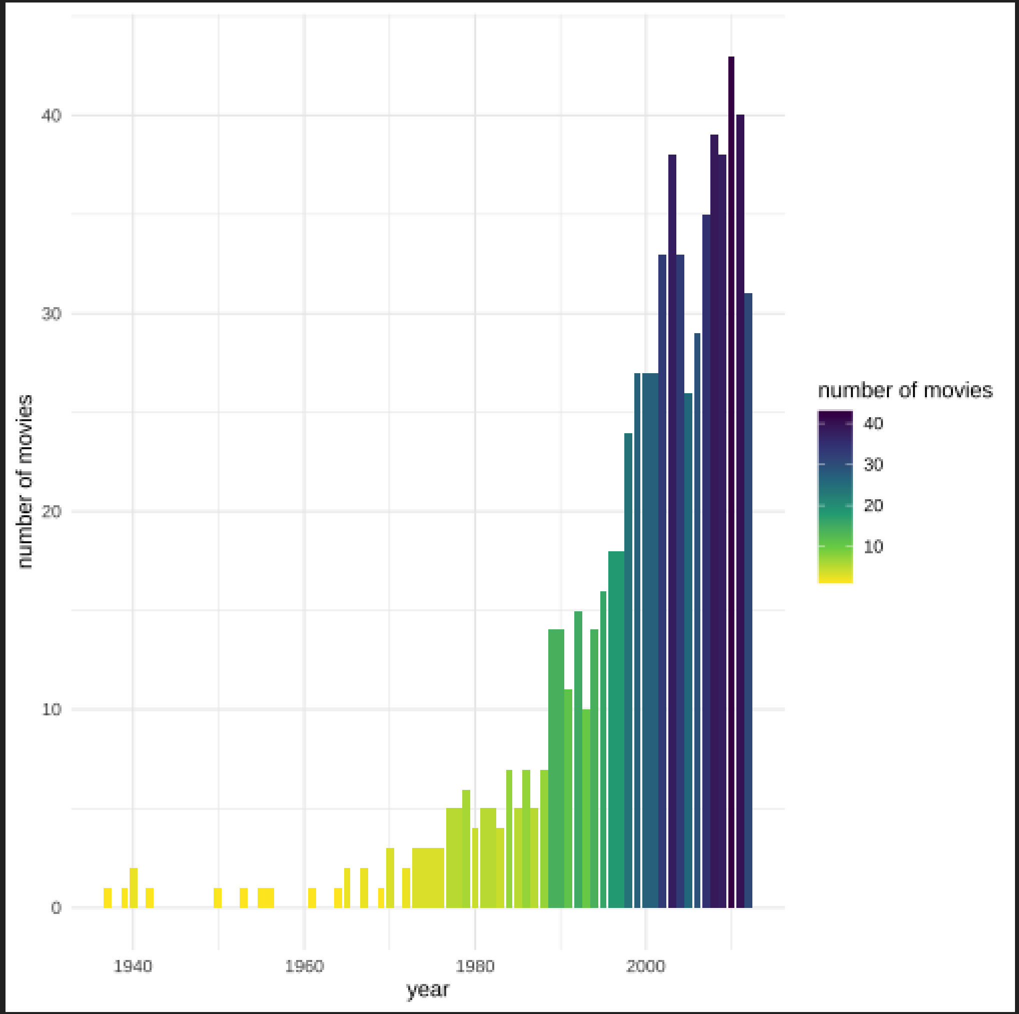
```r
boxoffice_annualnum %>%
  summarize(median =
              median(`number of movies`,
                     na.rm = TRUE)) %>%
  pull()
```

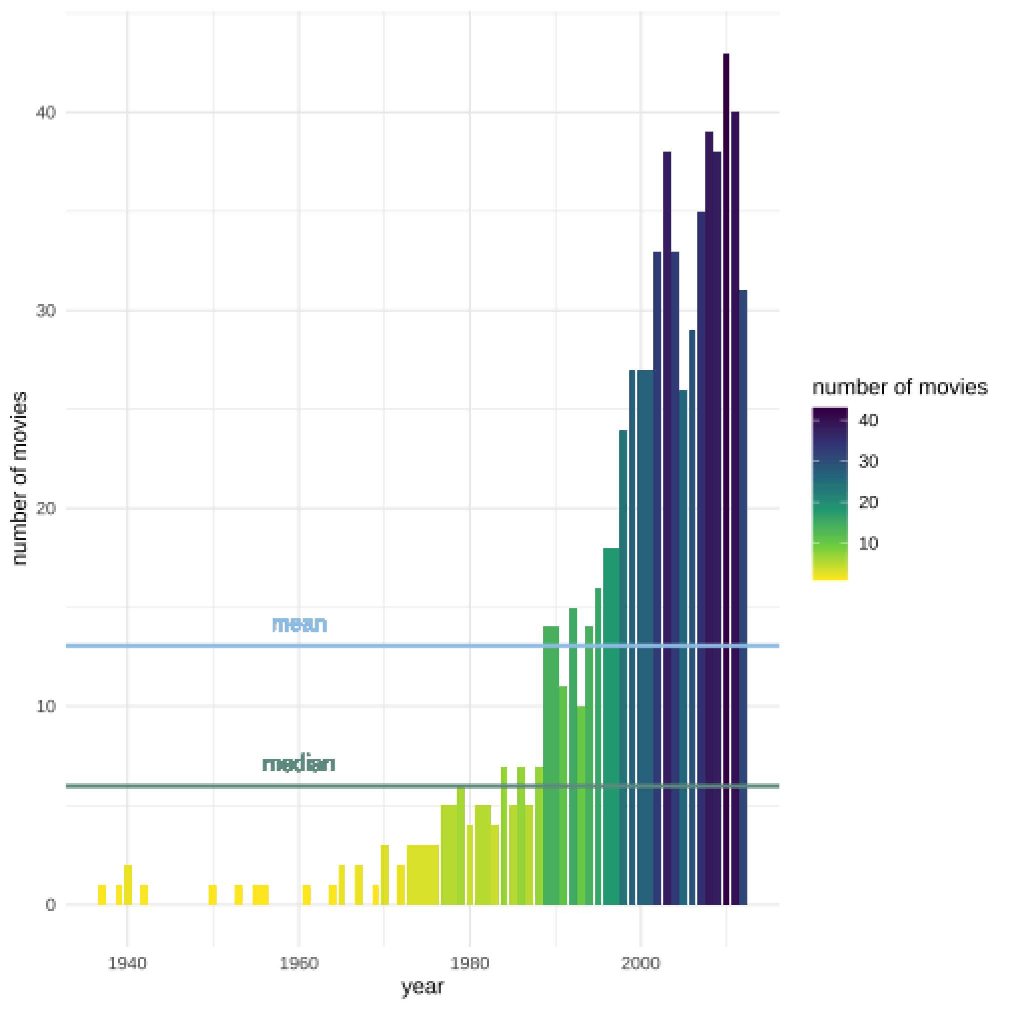
```
[1] 6
```

```r
boxoffice_annualnum %>%
  summarize(mean =
              mean(`number of movies`,
                   na.rm = TRUE)) %>%
  pull()
```

```
[1] 13.05455
```

```r
ggplot(boxoffice_annualnum,
       aes(year,
           `number of movies`,
           fill = `number of movies`)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  scale_fill_viridis_c(direction = -1)
```





mean
median
number of movies
0
10
20
30
40
1940
1960
1980
2000
year
number of movies
40
30
20
10

```
boxoffice %>%
  group_by(year) %>%
  tally(name = "number of movies") %>%
  ungroup() %>%
  filter(`number of movies` ==
           max(`number of movies`))
```

```
# A tibble: 1 × 2
   year `number of movies`
  <dbl>              <int>
1  2010                 43
```

```r
boxoffice %>%
  group_by(year) %>%
  summarise(`number of movies` = n()) %>%
  ungroup() %>%
  filter(`number of movies` ==
           max(`number of movies`))
```

```
# A tibble: 1 × 2
   year `number of movies`
  <dbl>              <int>
1  2010                 43
```

```r
boxoffice %>%
  group_by(year) %>%
  mutate(`number of movies` = n()) %>%
  ungroup() %>%
  distinct(year, .keep_all=TRUE) %>%
  filter(`number of movies` ==
           max(`number of movies`)) %>%
  select(year, `number of movies`)
```

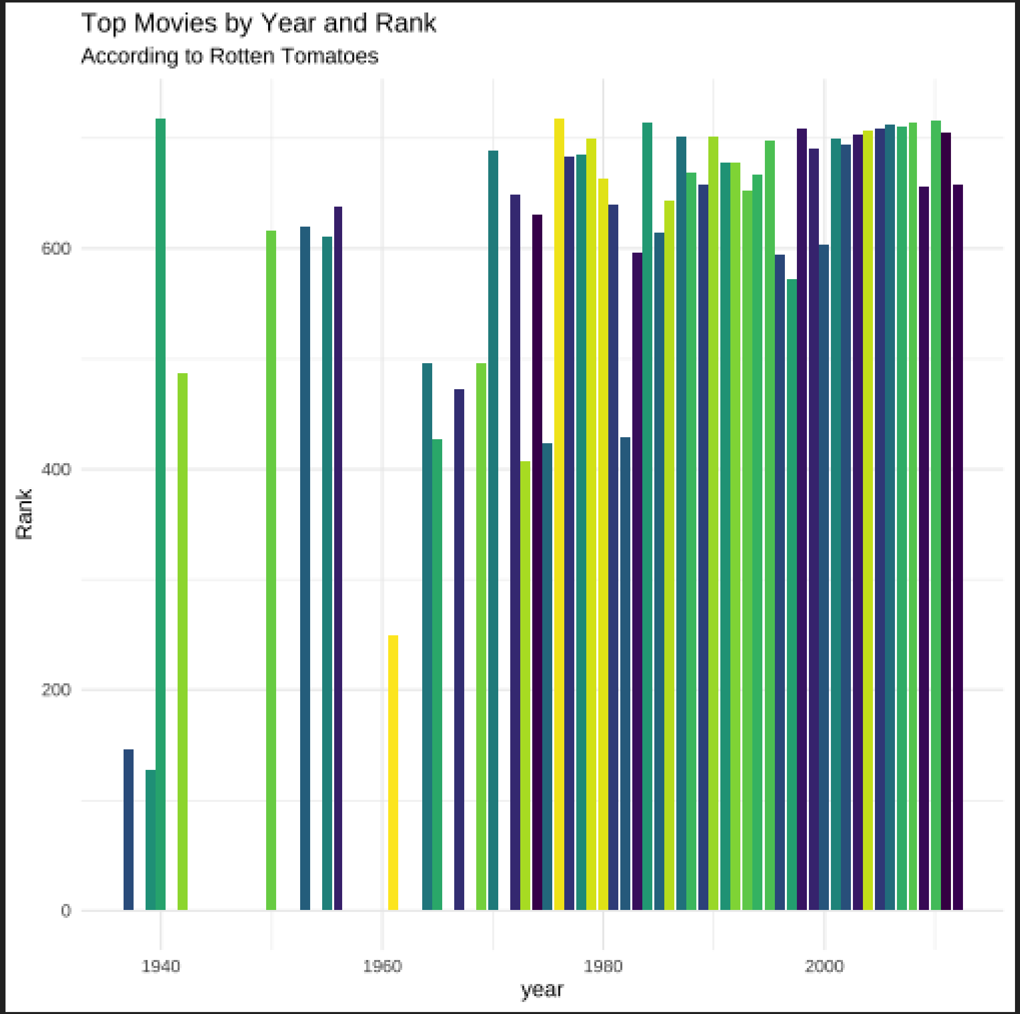
```
# A tibble: 1 × 2
   year `number of movies`
  <dbl>              <int>
1  2010                 43
```

```
boxoffice %>%
  group_by(year) %>%
  filter(Rank == max(Rank)) %>%
  select(Rank, Movie, year)%>%
  arrange(-year) %>%
  ungroup()
```

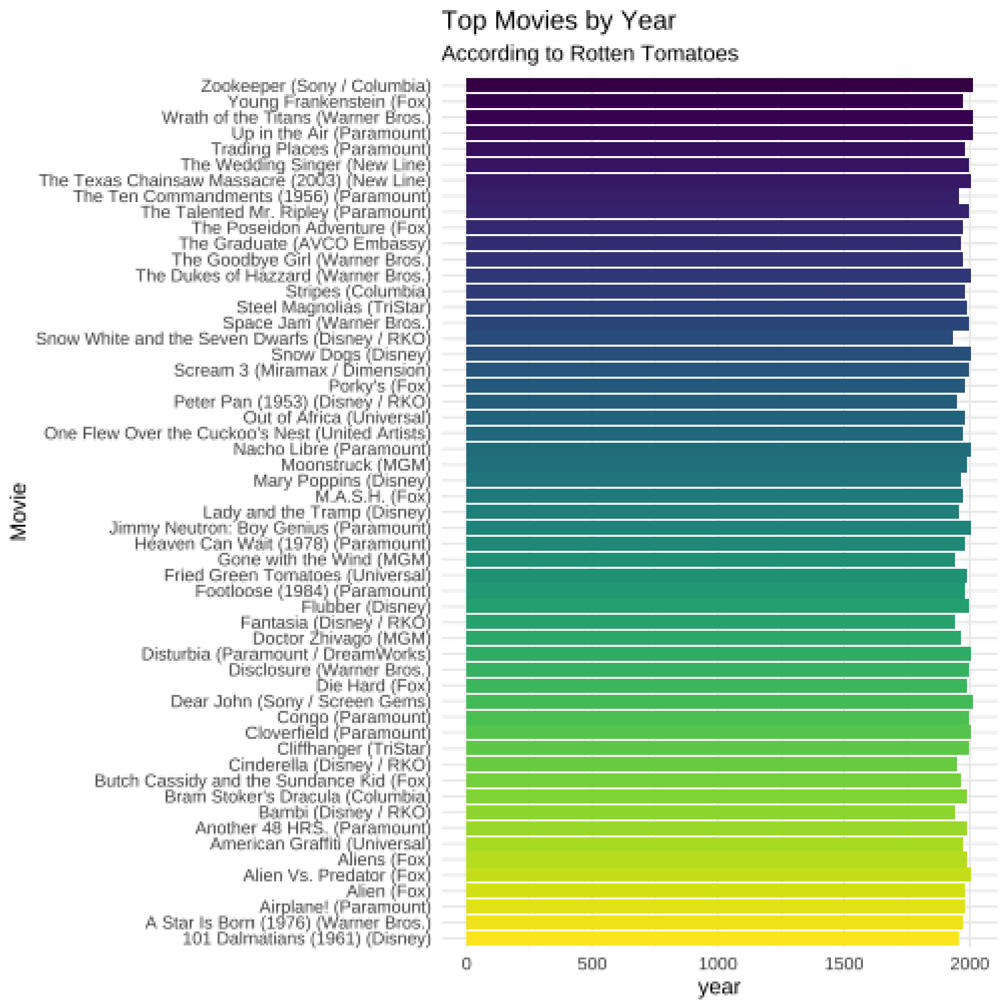
```
# A tibble: 55 × 3
    Rank Movie                                                  year
   <dbl> <chr>                                                 <dbl>
 1   658 Wrath of the Titans (Warner Bros.)                     2012
 2   705 Zookeeper (Sony / Columbia)                            2011
 3   716 Dear John (Sony / Screen Gems)                         2010
 4   656 Up in the Air (Paramount)                              2009
 5   714 Cloverfield (Paramount)                                2008
 6   711 Disturbia (Paramount / DreamWorks)                     2007
 7   712 Nacho Libre (Paramount)                                2006
 8   708 The Dukes of Hazzard (Warner Bros.)                    2005
 9   706 Alien Vs. Predator (Fox)                               2004
10   704 The Texas Chainsaw Massacre (2003) (New Line)          2003
# … with 45 more rows
```

```r
top_movie_year <-
  boxoffice %>%
  group_by(year) %>%
  filter(Rank == max(Rank)) %>%
  select(Rank, Movie, year)%>%
  arrange(-year) %>%
  ungroup()
```

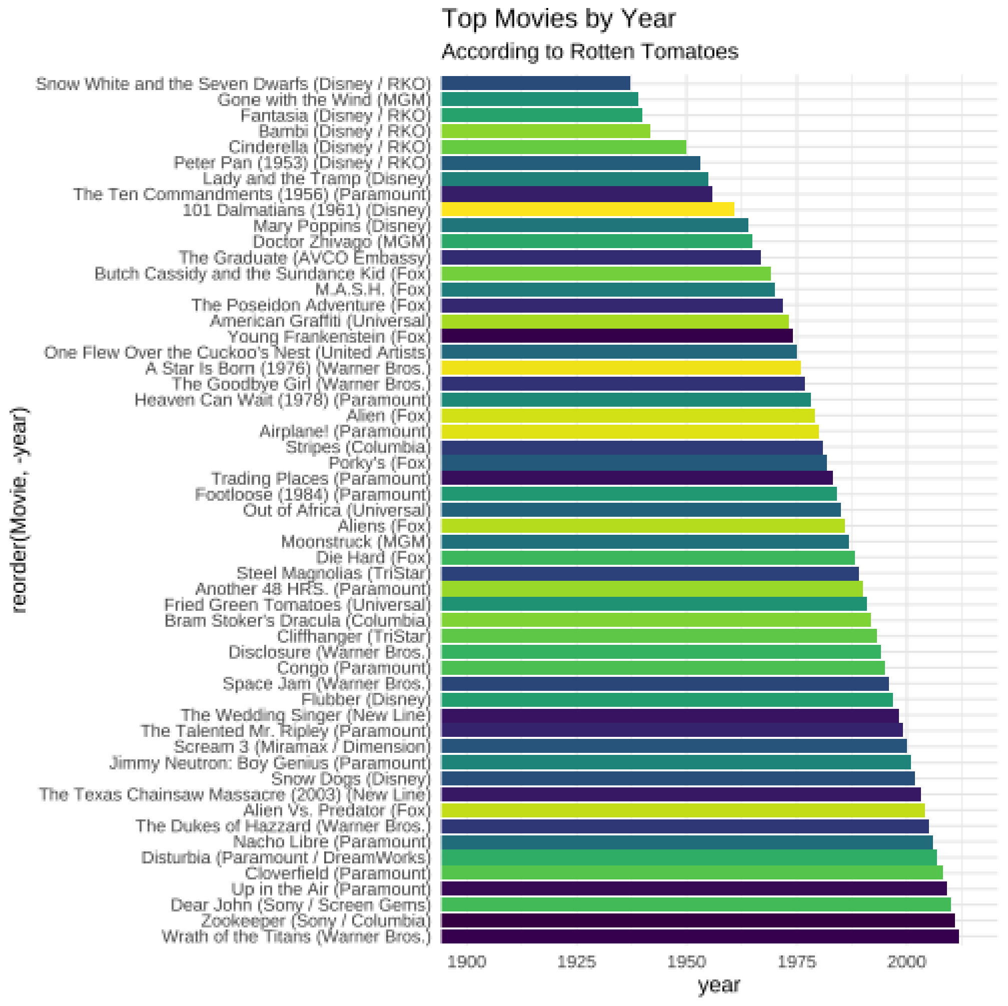
```r
ggplot(top_movie_year,
       aes(year,
           Rank,
           fill = Movie)) +
  geom_bar(stat = "identity",
           show.legend = FALSE) +
  theme_minimal() +
  scale_fill_viridis_d(direction = -1) +
  labs(title = "Top Movies by Year and Rank",
       subtitle = "According to Rotten Tomatoes")
```



```
ggplot(top_movie_year,
       aes(year,
           Movie,
           fill = Movie)) +
  geom_bar(stat = "identity",
           show.legend = FALSE) +
  theme_minimal() +
  scale_fill_viridis_d(direction = -1) +
  labs(title = "Top Movies by Year",
       subtitle = "According to Rotten Tomatoes")
```



```
ggplot(top_movie_year,
       aes(year,
           reorder(Movie, -year),
           fill = Movie)) +
  geom_bar(stat = "identity",
           show.legend = FALSE) +
  theme_minimal() +
  scale_fill_viridis_d(direction = -1) +
  labs(title = "Top Movies by Year",
       subtitle = "According to Rotten Tomatoes") +
  coord_cartesian(xlim = c(1900, 2015))
```



Ok now on to the normal curve!

# Load up data

```
nfl_pol <- read_csv("teampolview.csv")
```

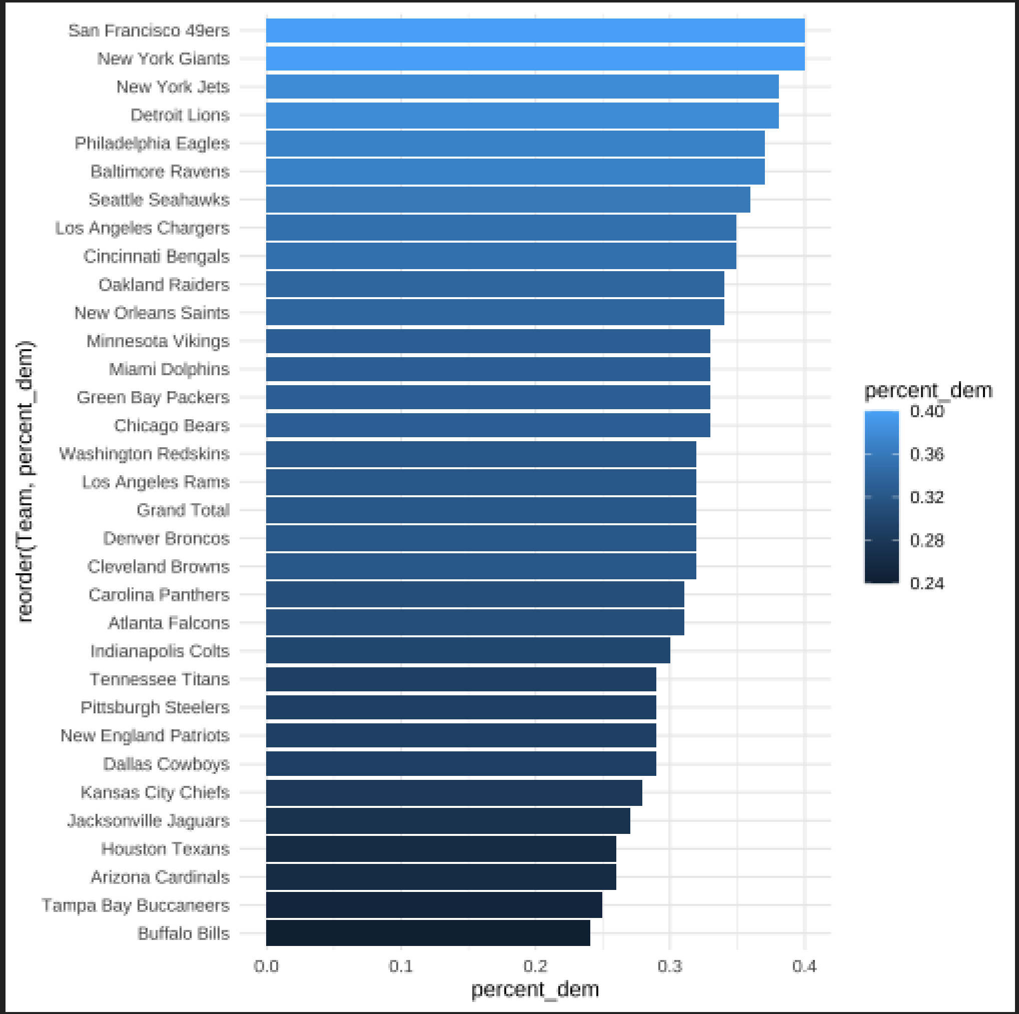
```r
nfl_pol %>%
  select(Team, `Total Respondents`, `Total Democrats
  rowwise(Team) %>%
  mutate(`Total Republicans` = sum(c(Republican,`Othe
  select(-c(Republican,`Other Republican`)) %>%
  mutate(percent_dem = round(`Total Democrats`/`Total
  mutate(percent_rep = round(`Total Republicans`/`To
```

```
# A tibble: 33 × 6
# Rowwise:  Team
   Team               `Total Responde… `Total Democrat… `Total Republic… percent_dem
   <chr>                         <dbl>            <dbl>            <dbl>       <dbl>
 1 Arizona Cardinals               148               39               32        0.26
 2 Atlanta Falcons                 188               59               44        0.31
 3 Baltimore Ravens                150               56               27        0.37
 4 Buffalo Bills                    92               22               16        0.24
 5 Carolina Panthers               164               51               45        0.31
 6 Chicago Bears                   285               94               55        0.33
 7 Cincinnati Bengals              106               37               32        0.35
 8 Cleveland Browns                105               34               28        0.32
 9 Dallas Cowboys                  438              128              129        0.29
10 Denver Broncos                  313              100               87        0.32
# … with 23 more rows, and 1 more variable: percent_rep <dbl>
```

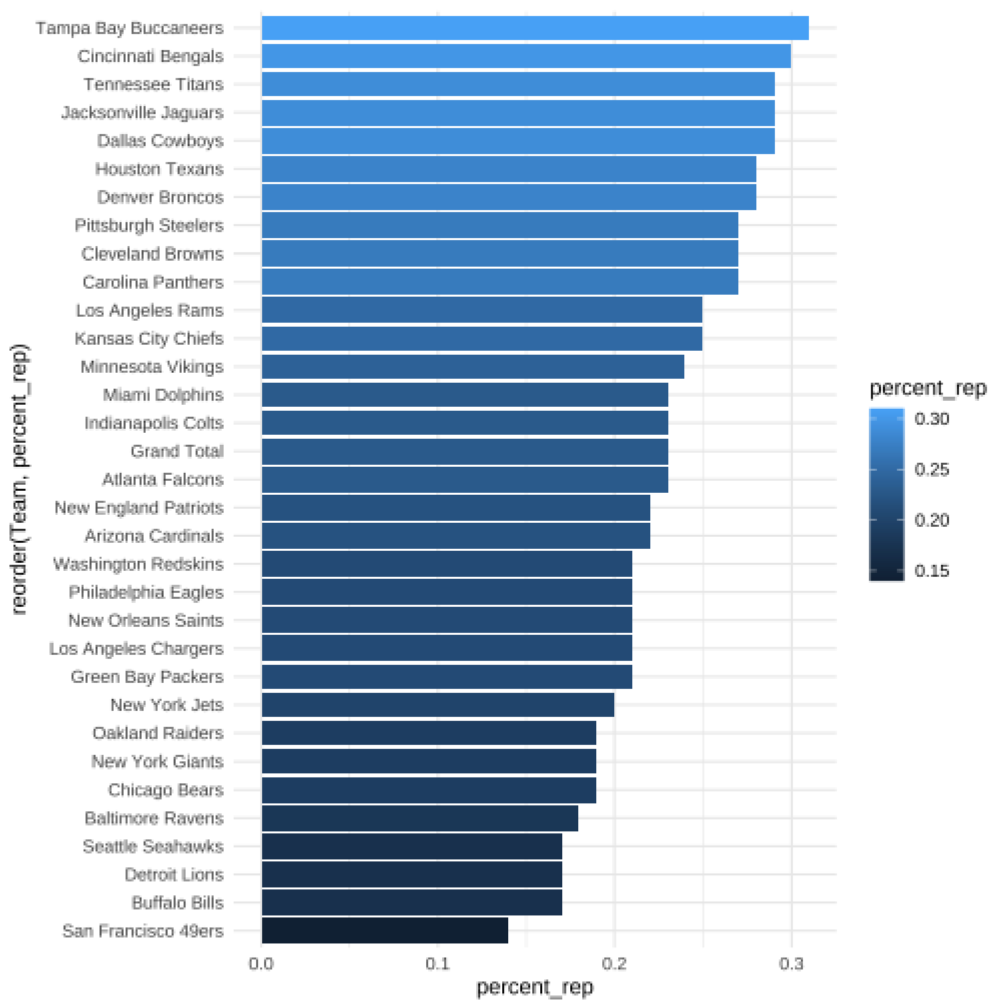
# Give it a variable

```r
nfl_percentages <-
  nfl_pol %>%
  select(Team, `Total Respondents`, `Total Democrats`, Republican, `Other Republican`) %>%
  rowwise(Team) %>%
  mutate(`Total Republicans` = sum(c(Republican,`Other Republican`))) %>%
  select(-c(Republican,`Other Republican`)) %>%
  mutate(percent_dem = round(`Total Democrats`/`Total Respondents`,2)) %>%
  mutate(percent_rep = round(`Total Republicans`/`Total Respondents`,2))
```

```r
ggplot(nfl_percentages,
       aes(reorder(Team, percent_dem),
                                percent_dem,
                                fill = percent_dem)) +
  geom_bar(stat="identity") +
  coord_flip() +
  theme_minimal()
```



```r
ggplot(nfl_percentages, aes(reorder(Team, percent_re
                            percent_rep,
                            fill = percent_rep)) +
  geom_bar(stat="identity") +
  coord_flip() +
  theme_minimal()
```



# Let's compare them!

But first we need to assign variables

```
p1 <- ggplot(nfl_percentages, aes(reorder(Team, percent_dem),
                            percent_dem,
                            fill = percent_dem)) +
  geom_bar(stat="identity") +
  coord_flip() +
  theme_minimal()
```

```
p2 <- ggplot(nfl_percentages, aes(reorder(Team, percent_rep),
                            percent_rep,
                            fill = percent_rep)) +
  geom_bar(stat="identity") +
  coord_flip() +
  theme_minimal()
```

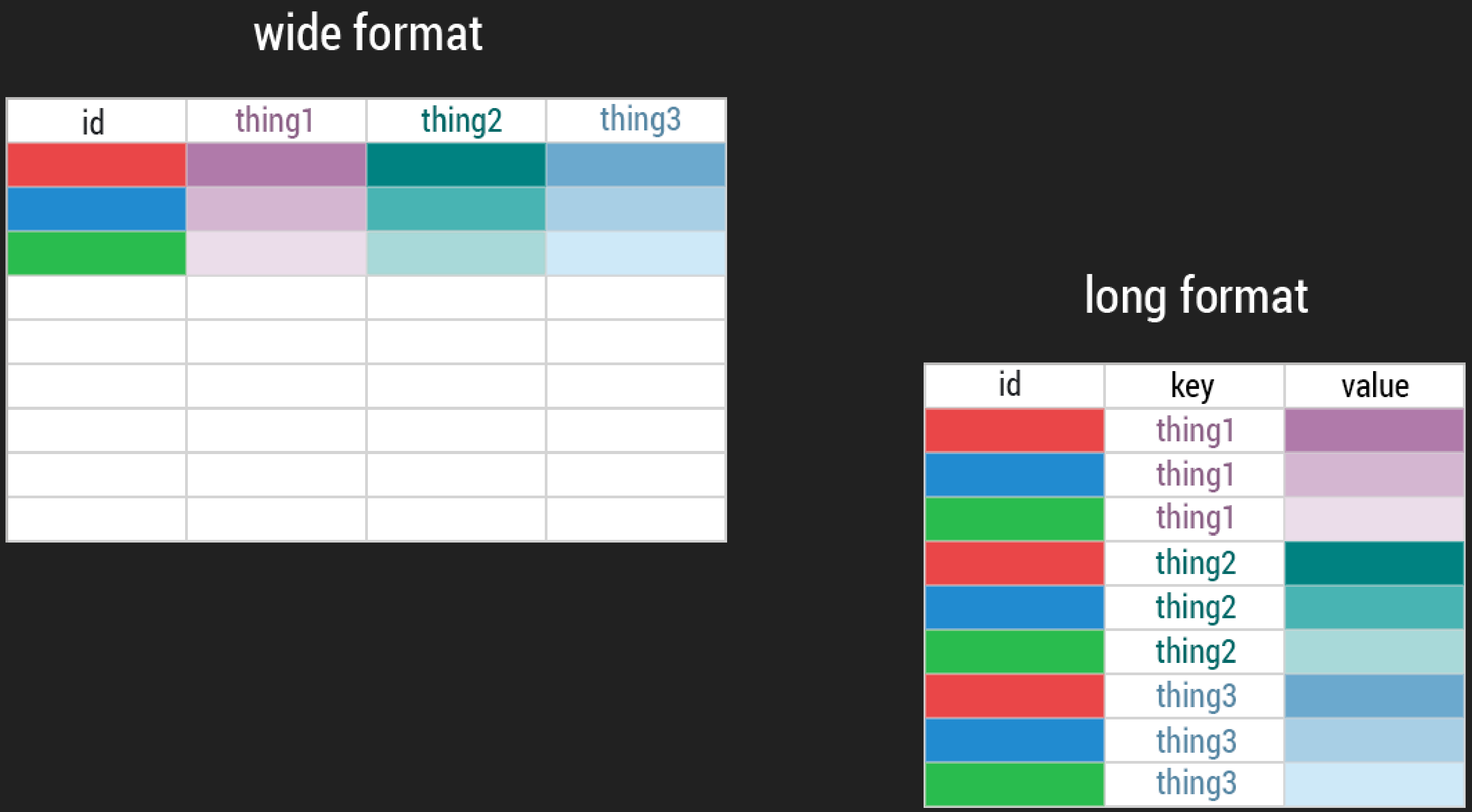
# Patch it together using `Patchwork`

```
p1 + p2
```

# A better way

That's not really a comparison...at least not teamwise! Let's try something different

# More Data Wrangling: Going from wide to long using `pivot_longer`



wide format

| id | thing1 | thing2 | thing3 |
|---|---|---|---|
| 🟥 | 🟪 | 🟩 | 🟦 |
| 🟦 | 🟪 | 🟩 | 🟦 |
| 🟩 | 🟪 | 🟩 | 🟦 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

long format

| id | key | value |
|---|---|---|
| 🟥 | thing1 | 🟪 |
| 🟦 | thing1 | 🟪 |
| 🟩 | thing1 | 🟪 |
| 🟥 | thing2 | 🟩 |
| 🟦 | thing2 | 🟩 |
| 🟩 | thing2 | 🟩 |
| 🟥 | thing3 | 🟦 |
| 🟦 | thing3 | 🟦 |
| 🟩 | thing3 | 🟦 |

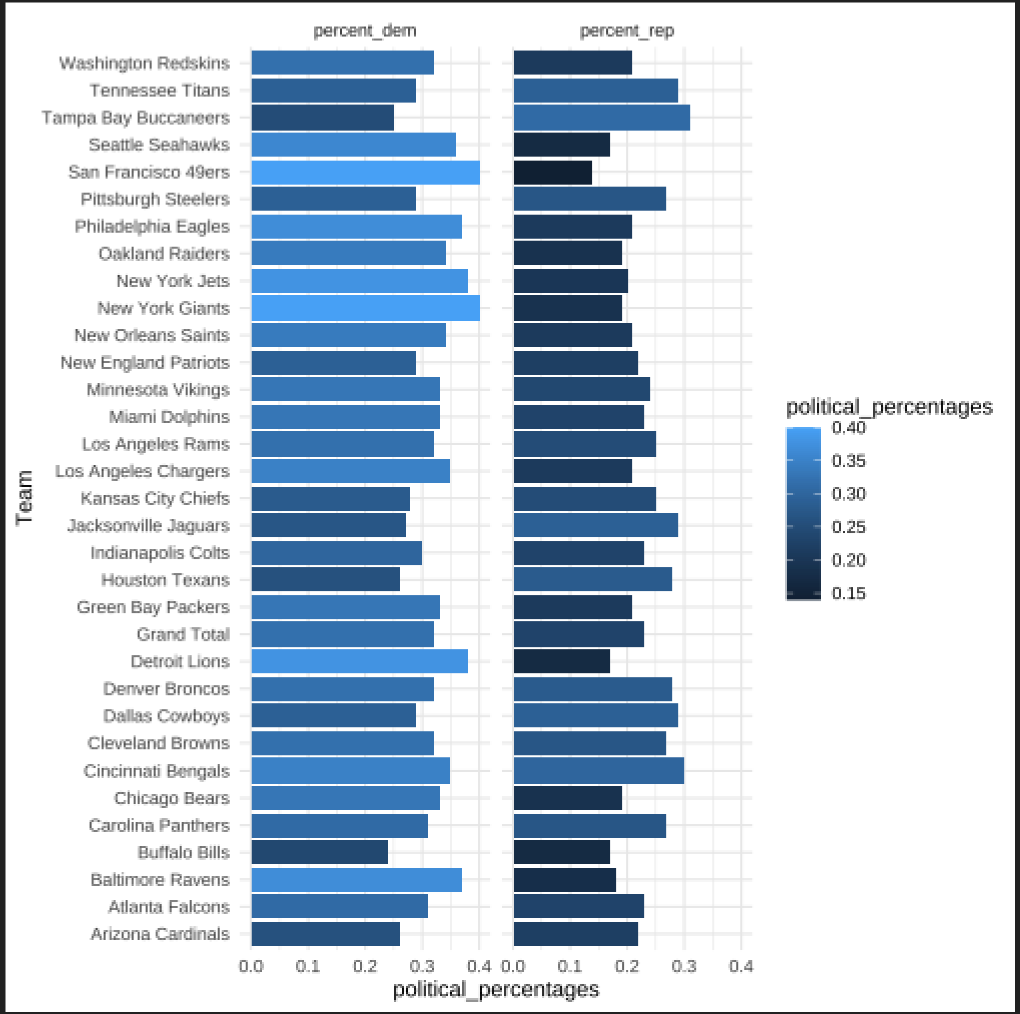
```r
nfl_percentages %>%
  pivot_longer(c(percent_dem, percent_rep),
               names_to = "type",
               values_to = "political_percentages")
```

```
# A tibble: 66 × 6
   Team              `Total Responden… `Total Democrats` `Total Republic… type
   <chr>                         <dbl>             <dbl>            <dbl> <chr>
 1 Arizona Cardinals               148                39               32 perce…
 2 Arizona Cardinals               148                39               32 perce…
 3 Atlanta Falcons                 188                59               44 perce…
 4 Atlanta Falcons                 188                59               44 perce…
 5 Baltimore Ravens                150                56               27 perce…
 6 Baltimore Ravens                150                56               27 perce…
 7 Buffalo Bills                    92                22               16 perce…
 8 Buffalo Bills                    92                22               16 perce…
 9 Carolina Panthers               164                51               45 perce…
10 Carolina Panthers               164                51               45 perce…
# … with 56 more rows, and 1 more variable: political_percentages <dbl>
```

# Give it a variable

```r
nlf_percentages_long <- nfl_percentages %>%
  pivot_longer(c(percent_dem, percent_rep),
               names_to = "type",
               values_to = "political_percentages")
```

```
ggplot(nlf_percentages_long, aes(Team,
                                 political_percentag
                            fill =  political_percen
  geom_bar(stat="identity") +
  coord_flip() +
  theme_minimal() +
  facet_wrap(.~type)
```



# Your turn

Try these on your own

1. Compare how the different ethnicities within each political party differ.
2. Compare how each specific ethnicity between each political party differ.
3. Which ethnicity in each political party is the most conservative? the most liberal?

# That's it for today!