DATA

Link to the original data in .csv format:
https://github.com/fivethirtyeight/data/blob/76c471a9124d690ba92709ca21cbfcdde226b44e/polls/pres_pollaverages_1968-2016.csv

R CODE

```
library(lubridate)
library(tidyverse)
library(repr)
library(infer)
library(cowplot)
library(broom)
library(dplyr)
library(data.table)

# LOADING AND TIDYING DATA
data_raw <- read.csv("data/pres_poll_avgs_1968-2016.csv")

# only keep relevant columns
data_tidy <- select(data_raw, cycle:pct_estimate,election_date)

# rename columns
data_tidy <- rename(data_tidy, model_date = modeldate)
data_tidy <- rename(data_tidy, election_year = cycle)

# change model_date and election_date from char to date format for easier manipulation
data_tidy$election_date <- mdy(data_tidy$election_date)
data_tidy$model_date <- mdy(data_tidy$model_date)

# Extract month and day from date column
data_tidy$month <- month(data_tidy$model_date)
data_tidy$day <- day(data_tidy$model_date)
# Filter out rows where the date is before April 9th
filtered_data_out <- data_tidy |>
  filter(month > 3 & day > 8)
```

```
# show the earliest date with data recorded for each election to ensure the right
subset of data was taken

first_dates_by_election_year <- filtered_data_out |>

  group_by(election_year) |>

  slice_head(n = 1) |>

  select(election_year, model_date)


# creating a .csv file for the filtered_data

write.csv(filtered_data_out, "data/filtered_data.csv", row.names = FALSE)


# GROUPING THE DATA AND REMOVING UNNEEDED COLUMNS

all_years <- read.csv("data/filtered_data.csv")


# Grouping states by division

grouped_by_division <- all_years |>

    filter(!(state %in% c("ME-1", "ME-2", "NE-1", "NE-2", "NE-3", "National"))) |>

  mutate(division = case_when(

    state %in% c("Connecticut", "Maine", "Massachusetts", "New Hampshire", "Rhode
Island", "Vermont") ~ "New England",

    state %in% c("New Jersey", "New York", "Pennsylvania") ~ "Middle Atlantic",

    state %in% c("Illinois", "Indiana", "Michigan", "Ohio", "Wisconsin") ~ "East North
Central",

    state %in% c("Iowa", "Kansas", "Minnesota", "Missouri", "Nebraska", "North
Dakota", "South Dakota") ~ "West North Central",

    state %in% c("Delaware", "District of Columbia", "Florida", "Georgia", "Maryland",
"North Carolina", "South Carolina", "Virginia", "West Virginia") ~ "South Atlantic",

    state %in% c("Alabama", "Kentucky", "Mississippi", "Tennessee") ~ "East South
Central",

    state %in% c("Arkansas", "Louisiana", "Oklahoma", "Texas") ~ "West South Central",

    state %in% c("Arizona", "Colorado", "Idaho", "Montana", "Nevada", "New Mexico",
"Utah", "Wyoming") ~ "Mountain",

    state %in% c("Alaska", "California", "Hawaii", "Oregon", "Washington") ~
"Pacific"))


# remove any unneeded columns

grouped_by_division <- grouped_by_division |>
```

```
                select(election_year, candidate_name, pct_estimate,
division,model_date)



# CALCULATING PER DISTRICT POLLING AVERAGE FOR EACH CANDIDATE IN EACH ELECTION
# Define the function to calculate averages by year (unchanged)
calculate_average_by_year <- function(dataframe, year) {
  filtered_data <- filter(dataframe, election_year == year)
  average_pct_estimates <- aggregate(
    pct_estimate ~ candidate_name + division,
    data = filtered_data,
    FUN = mean)
  average_pct_estimates$election_year <- year  # Add year information
  return(average_pct_estimates)}

# Create an empty dataframe to store combined results
all_year_averages <- data.frame()  # Start with an empty dataframe

# Iterate through years and append results to the dataframe
years <- c(2016, 2012, 2008, 2004, 2000, 1996, 1992, 1988, 1984, 1980, 1976, 1972)
for (year in years) {
  year_df <- calculate_average_by_year(grouped_by_division, year)
  all_year_averages <- rbind(all_year_averages, year_df)  # Append using rbind}

# creating a .csv file for the computed division averages
write.csv(all_year_averages, "data/division_averages.csv", row.names = FALSE)
# CALCULATING PER DISTRICT POLLING AVERAGE FOR EACH CANDIDATE IN EACH ELECTION
# Define the function to calculate averages by year (unchanged)
calculate_average_by_year <- function(dataframe, year) {
  filtered_data <- filter(dataframe, election_year == year)
  average_pct_estimates <- aggregate(
    pct_estimate ~ candidate_name + division,
    data = filtered_data,
```

```
    FUN = mean)
  average_pct_estimates$election_year <- year  # Add year information
  return(average_pct_estimates)}

# Create an empty dataframe to store combined results
all_year_averages <- data.frame()  # Start with an empty dataframe

# Iterate through years and append results to the dataframe
years <- c(2016, 2012, 2008, 2004, 2000, 1996, 1992, 1988, 1984, 1980, 1976, 1972)
for (year in years) {
  year_df <- calculate_average_by_year(grouped_by_division, year)
  all_year_averages <- rbind(all_year_averages, year_df)  # Append using rbind}

# creating a .csv file for the computed division averages
write.csv(all_year_averages, "data/division_averages.csv", row.names = FALSE)

division_averages <- read.csv("data/division_averages.csv")
# CALCULATING A WEIGHTED AVERAGE OF EACH CANDIDATE'S POLLING SCORES BASED ON DIVISION
# Each district's electoral vote total is the sum of all their state's electoral votes
south_atlantic <- 104
pacific <- 81
east_north_central <- 72
mid_atlantic <- 61
west_south_central <- 61
mountain <- 49
west_north_central <- 43
east_south_central <- 34
new_england <- 33
total <- 538

# Define weights dictionary
weights <- c(
    "South Atlantic" = south_atlantic / total,
```

```
    "Pacific" = pacific / total,
    "East North Central" = east_north_central / total,
    "Middle Atlantic" = mid_atlantic / total,
    "West South Central" = west_south_central / total,
    "Mountain" = mountain / total,
    "West North Central" = west_north_central / total,
    "East South Central" = east_south_central / total,
    "New England" = new_england / total)

# Calculate weighted average per candidate and year
weighted_averages <- division_averages %>%
  group_by(candidate_name, election_year) %>%
  summarize(
    weighted_pct_estimate = sum(pct_estimate * weights[division]),
    .N = n())

weighted_averages$weighted_pct_estimate <-
round(weighted_averages$weighted_pct_estimate, digits = 2)
weighted_averages <- weighted_averages |> arrange(election_year)
weighted_averages

# creating a .csv file for the weighted averages
write.csv(weighted_averages, "data/weighted_averages.csv", row.names = FALSE)

options(repr.plot.width = 12, repr.plot.height = 10)
line_graph <- ggplot(summary_stats) +
  geom_line(aes(x = election_year, y = mean_weighted_ratings, color = election_year))+
  scale_x_continuous(breaks = seq(1972, 2016, 4)) +
  scale_y_continuous(breaks = seq(29, 46, 2)) +
  xlab("Election Years") +
  ylab("Mean Weighted Ratings per Election") +
  theme(
    text = element_text(size = 19),
```

```
    plot.title = element_text(face = "bold"),
    axis.title = element_text(face = "bold"))

line_graph
```