Data Cleaning & Analysis Coding Sample

Yiqing (Yuki) Chen

2025-9-24

This independent work is an exploratory study focusing around the 2018 elections for Senate in the United States. I explored data from three secondary data source (1) the website FiveThirtyEight's election forecast, (2) the context for the 2018 election from MIT Data Lab, and (3) historic Senate results from MIT Data Lab.

```
library(tidyverse)
library(dplyr)
library(lubridate)
library(lmtest)
library(sandwich)
```

Working with FiveThirtyEight's election forecast data

I will explore three research questions in the following analysis:

Research Question 1: What is 538's deluxe forecast over time for the odds of victory of candidates in the two major parties (Republican and Democrat) in Wisconsin?

Research Question 2: What is the distribution of estimated voteshare for Kyrsten Sinema, the Democratic nominee in Arizona? And how it varied over time?

Research Question 3: Who is the candidate who is neither a Republican or Democrat who has the highest probability of victory at any point in any of 538's models?

```
setwd("~/Downloads/DIME RA Application_Yuki Chen/R")
df_elections_forecast <- read_csv("Raw Data/senate_seat_forecast.csv") |>
  as_tibble()
```

Data Quality Checks

```
# Standardize string columns
df_elections_forecast <- df_elections_forecast %>%
  dplyr::mutate(
    party = dplyr::recode(party,
                          "D"="Democrat","R"="Republican","I"="Independent",
                          .default = party),
    model = factor(model, levels = c("classic","deluxe","lite")), # ordered factor
    state = toupper(state) # keep all state codes in uppercase
  )
```

```r
# Check missingness
colSums(is.na(df_elections_forecast))
```

```
##    forecastdate           state           class         special       candidate
##               0               0               0               0               0
##           party       incumbent           model win_probability       voteshare
##            4116               0               0               0               0
##   p10_voteshare   p90_voteshare
##               0               0
```

```r
# part of "party" information is missing.

# Range checks for numeric variables
summary(df_elections_forecast[, c("win_probability","voteshare",
                                  "p10_voteshare","p90_voteshare")])
```

```
##  win_probability    voteshare     p10_voteshare   p90_voteshare
##  Min.   :0.0000   Min.   : 1.50   Min.   : 0.36   Min.   : 2.95
##  1st Qu.:0.0000   1st Qu.:12.20   1st Qu.: 6.78   1st Qu.:18.32
##  Median :0.0731   Median :42.08   Median :36.99   Median :47.69
##  Mean   :0.3629   Mean   :36.27   Mean   :31.61   Mean   :41.05
##  3rd Qu.:0.8790   3rd Qu.:52.56   3rd Qu.:47.61   3rd Qu.:57.61
##  Max.   :1.0000   Max.   :77.27   Max.   :71.43   Max.   :85.11
```

```r
# Check unique identifiers
# From data source, I learned that the dataset includes FiveThirtyEight's model forecast
↪  for candidates in each senate race by day, in each of their "classic", "deluxe", and
↪  "lite" models. I want to test if the combination of forecastdate-state-party-model
↪  uniquely identifies the observations.
potential_unique_ids <-
  df_elections_forecast %>%
  distinct(forecastdate, state, party, model) %>%
  nrow()

paste0("Percentage of unique observations: ",
       (potential_unique_ids/
          (df_elections_forecast %>% nrow()))*100,
       "%")
```

```
## [1] "Percentage of unique observations: 91.7045815257645%"
```

The combination of forecastdate-state-party-model cannot uniquely identifies the observations.

```r
# To check which states have duplicates
duplicates <-
  df_elections_forecast %>%
  group_by(forecastdate, state, party, model) %>%
  summarize(n = n()) %>% filter(n > 1)
```

```
## `summarise()` has grouped output by 'forecastdate', 'state', 'party'. You can
## override using the `.groups` argument.
```

```
print("The states with duplicates on these variables are:")
```

```
## [1] "The states with duplicates on these variables are:"
```

```
print(table(duplicates$state))
```

```
##
##  CA  MN  MS
## 294 882 588
```

The three states with duplicates on these variables are California, Minnesota, and Mississippi.

The next a few steps are to investigate the issue of duplicates, and identify other variables to uniquely identify each row:

```
# Filter the tibble to a single day and model
df_elections_forecast_filtered <-
  df_elections_forecast %>%
  filter(forecastdate == "2018-08-01",
         model == "classic",
         state == "CA" |
           state == "MN" |
           state == "MS")

df_elections_forecast_filtered %>% nrow()
```

```
## [1] 15
```

When I restrict to states California, Minnesota, and Mississippi on date 2018-08-01, with model classic, there are a total of 15 rows.

```
# To investigate what other variables can be combined with forecastdate-state-party-model
↪  to uniquely identify the observations
df_elections_forecast_filtered_MN_MS <-
  df_elections_forecast_filtered %>% filter(state %in% c("MN", "MS"))

class(df_elections_forecast_filtered)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
# Printing the whole data frame
print(df_elections_forecast_filtered_MN_MS)
```

```
## # A tibble: 13 x 12
##    forecastdate state class special candidate        party      incumbent model
##    <date>       <chr> <dbl> <lgl>   <chr>            <chr>      <lgl>     <fct>
##  1 2018-08-01   MN        1 FALSE   Amy Klobuchar    Democrat   TRUE      class~
```

```
##  2 2018-08-01   MN        1 FALSE   Jim Newberger    Republican FALSE     class~
##  3 2018-08-01   MN        1 FALSE   Others           <NA>       FALSE     class~
##  4 2018-08-01   MN        2 TRUE    Tina Smith       Democrat   FALSE     class~
##  5 2018-08-01   MN        2 TRUE    Karin Housley    Republican FALSE     class~
##  6 2018-08-01   MN        2 TRUE    Others           <NA>       FALSE     class~
##  7 2018-08-01   MS        1 FALSE   Roger F. Wicker  Republican TRUE      class~
##  8 2018-08-01   MS        1 FALSE   David Baria      Democrat   FALSE     class~
##  9 2018-08-01   MS        1 FALSE   Others           <NA>       FALSE     class~
## 10 2018-08-01   MS        2 TRUE    Cindy Hyde-Smith Republican FALSE     class~
## 11 2018-08-01   MS        2 TRUE    Mike Espy        Democrat   FALSE     class~
## 12 2018-08-01   MS        2 TRUE    Chris McDaniel   Republican FALSE     class~
## 13 2018-08-01   MS        2 TRUE    Tobey Bartee     Democrat   FALSE     class~
## # ... with 4 more variables: win_probability <dbl>, voteshare <dbl>,
## #   p10_voteshare <dbl>, p90_voteshare <dbl>
```

```
# Checking for uniquely identified rows using the following combination:
df_elections_forecast_filtered_MN_MS %>%
  distinct(forecastdate, state, party, model, candidate, class) %>%
  nrow() /
  df_elections_forecast_filtered_MN_MS %>%
  nrow()
```

```
## [1] 1
```

```
# It worked. Now test that with the whole data
df_elections_forecast %>%
  filter(state != "CA") %>%
  distinct(forecastdate, state, party, model, candidate, class) %>%
  nrow()/
  df_elections_forecast %>%
  filter(state != "CA") %>%
  nrow()
```

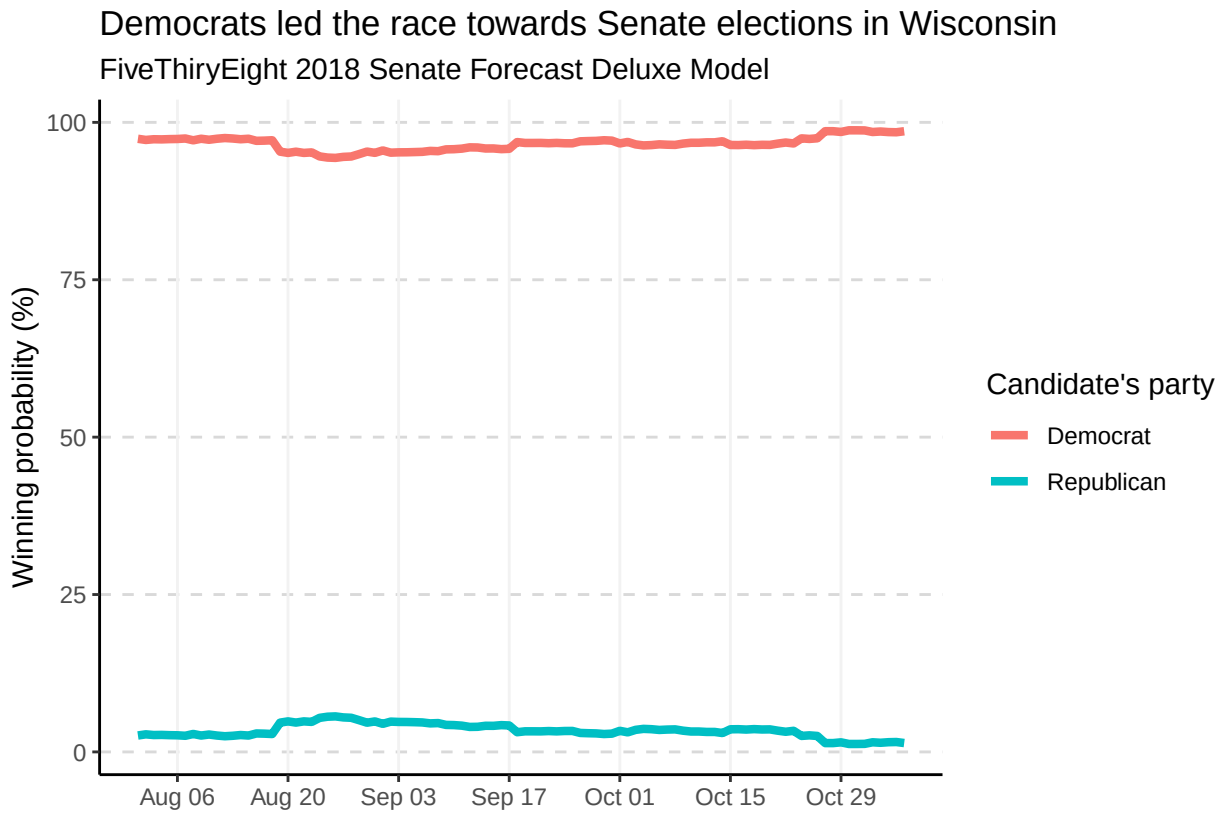
```
## [1] 1
```

The six variables: forecastdate, state, party, model, candidate, and class can be used to uniquely identify each row. The reason for having to add both is that there are three candidates in Mississippi that appear as "Others".

Research Question 1: What is 538's deluxe forecast over time for the odds of victory of candidates in the two major parties (Republican and Democrat) in Wisconsin?

```
# Convert the forecastdate variable to a date and create a column of year.
df_elections_forecast <-
  df_elections_forecast %>%
  mutate(forecastdate = as_date(forecastdate),
         year = year(forecastdate))

# Create line graph
df_elections_forecast %>%
```

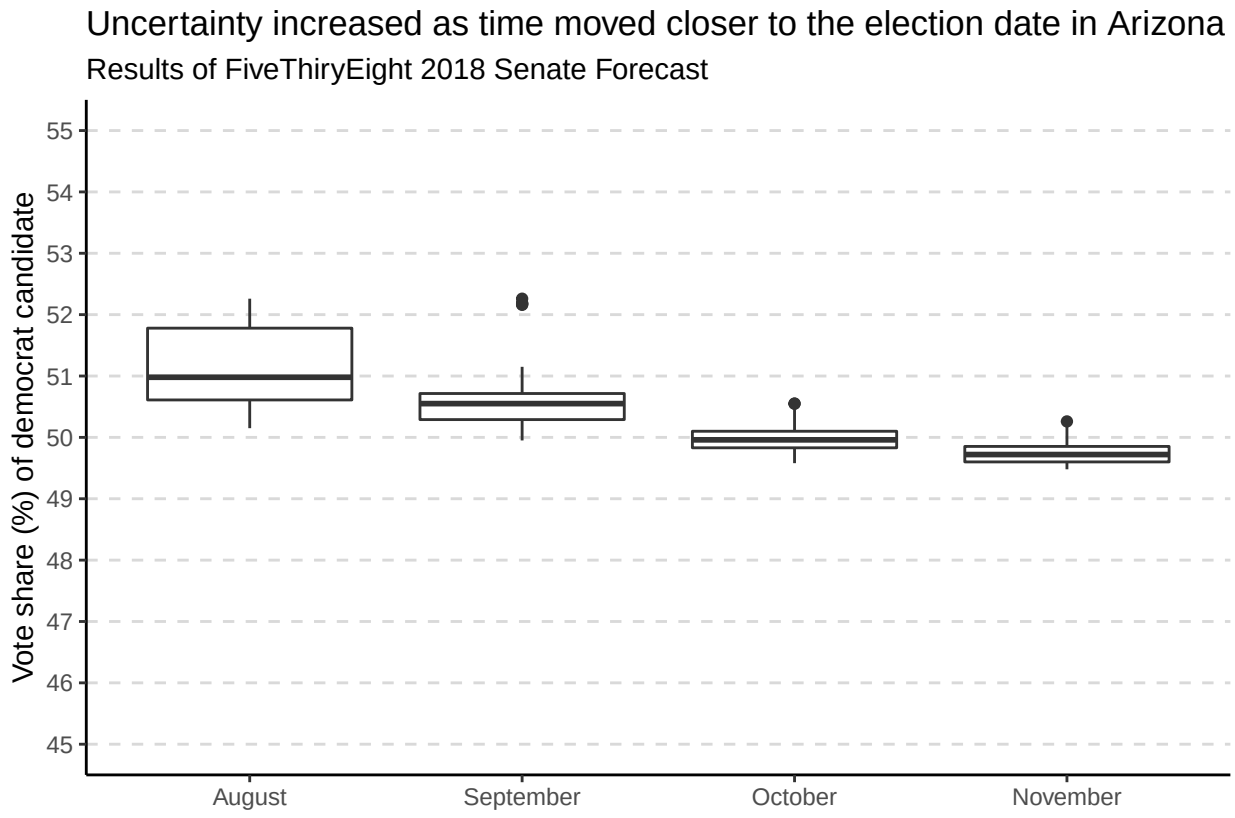
```
  filter(model == "deluxe" & state == "WI") %>%
  ggplot() +
  geom_line(aes(x=forecastdate, y=win_probability, color=party), size=1.5) +
  theme_classic() +
  theme(panel.grid.minor = element_blank(),
        panel.grid.major.y = element_line(color = "grey85", linetype = "dashed"),
        panel.grid.major.x = element_line(color = "grey95")) +
  scale_x_date(date_breaks = "2 weeks",
               date_labels = "%b %d") +
  scale_y_continuous(labels = function(x) format(x*100, decimal.mark = ".")) +
  labs(x="",
       y="Winning probability (%)",
       color="Candidate's party",
       title="Democrats led the race towards Senate elections in Wisconsin",
       subtitle = "FiveThiryEight 2018 Senate Forecast Deluxe Model") -> plot_rq1

print(plot_rq1)
```




```
ggsave("line_538_win_probability_wisconsin_2018.png", plot = plot_rq1, width = 6, height
↪  = 4, dpi = 300)
```

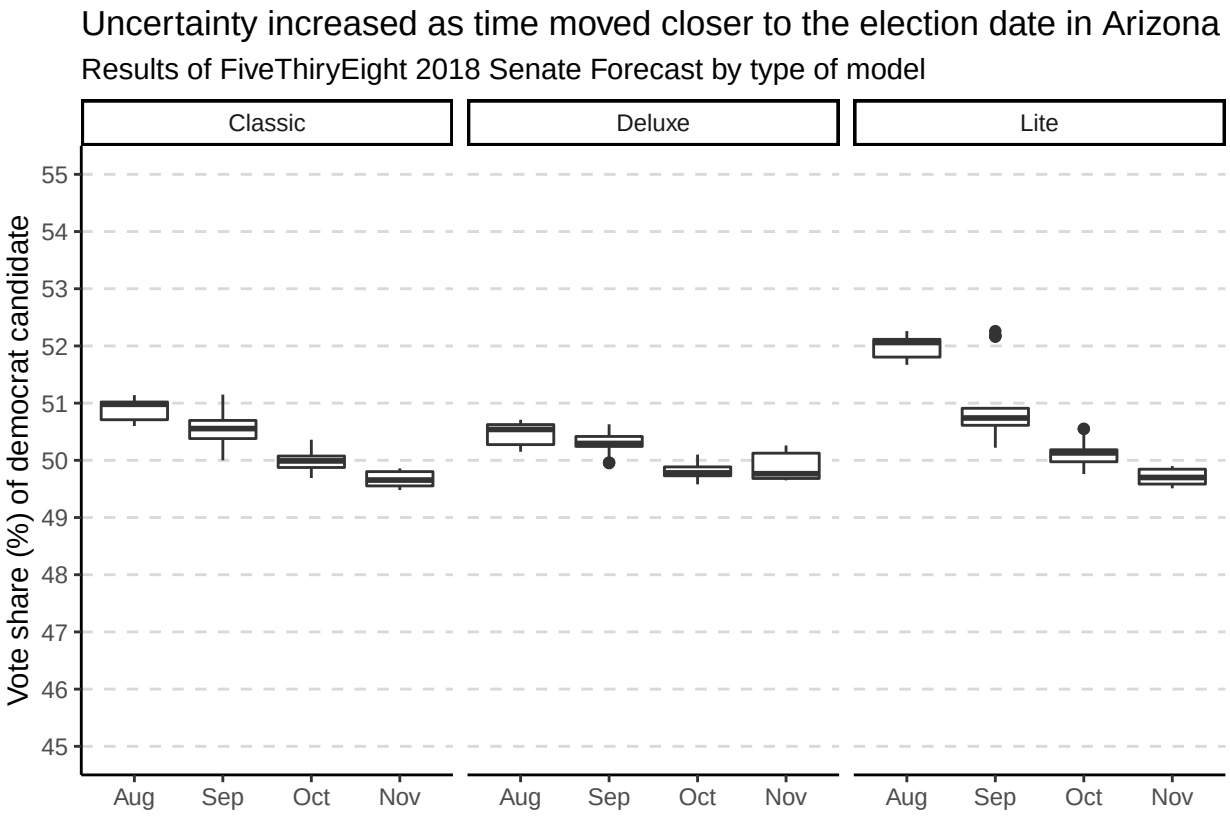
Research question 2: What is the distribution of estimated voteshare for Kyrsten Sinema, the Democratic nominee in Arizona? And how it varied over time?

```
# Creating a month variable equal to the month of the forecast:
df_elections_forecast <-
  df_elections_forecast %>%
  mutate(month = month(forecastdate, label = TRUE, abbr = FALSE))

# Creating a tibble that only stores 538 model estimates for Sinema:
tibble_sinema <- as_tibble(df_elections_forecast %>%
                             filter(party == "Democrat" & state == "AZ"))

# Creating a boxplot for all estimates of Sinema's voteshare by month:
tibble_sinema %>%
  ggplot() +
  geom_boxplot(aes(x=as.factor(month), y=voteshare)) +
  theme_classic() +
  theme(panel.grid.minor = element_blank(),
        panel.grid.major.y = element_line(color = "grey85", linetype = "dashed")) +
  scale_y_continuous(limits = c(45, 55),
                     n.breaks = 10) +
  labs(x="",
       y="Vote share (%) of democrat candidate",
       title="Uncertainty increased as time moved closer to the election date in
       ↪  Arizona",
       subtitle = "Results of FiveThiryEight 2018 Senate Forecast")  -> plot_rq2

print(plot_rq2)
```




```
ggsave("box_voteshre_arizona.png", plot = plot_rq2, width = 6, height = 4, dpi = 300)
```

```
# Creating a facet_wrap by model type:
tibble_sinema %>%
  mutate(month = month(forecastdate, label = TRUE),
         model = str_to_title(model)) %>%
  ggplot() +
  geom_boxplot(aes(x=as.factor(month), y=voteshare)) +
  facet_wrap(vars(model)) +
  theme_classic() +
  theme(panel.grid.minor = element_blank(),
        panel.grid.major.y = element_line(color = "grey85", linetype = "dashed")) +
  scale_y_continuous(limits = c(45, 55),
                     n.breaks = 10) +
  labs(x="",
       y="Vote share (%) of democrat candidate",
       title="Uncertainty increased as time moved closer to the election date in
       ↪  Arizona",
       subtitle = "Results of FiveThiryEight 2018 Senate Forecast by type of model") ->
       ↪  plot_rq2_2

print(plot_rq2_2)
```




```
ggsave("box_voteshre_arizona_class_deluxe_lite.png", plot = plot_rq2_2, width = 6, height
↪  = 4, dpi = 300)
```

Research Question 3: Who is the candidate who is neither a Republican or Democrat who has the highest probability of victory at any point in any of 538's models?

```
df_elections_forecast %>%
  filter(party != "D" & party != "R") %>%
  arrange(desc(win_probability))
```

```
## # A tibble: 24,237 x 14
##    forecastdate state class special candidate            party  incumbent model
##    <date>       <chr> <dbl> <lgl>   <chr>                <chr>  <lgl>     <fct>
##  1 2018-08-01   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  2 2018-08-01   NY        1 FALSE   Kirsten E. Gillibrand Democ~ TRUE      clas~
##  3 2018-08-03   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  4 2018-08-04   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  5 2018-08-05   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  6 2018-08-06   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  7 2018-08-08   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  8 2018-08-09   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
##  9 2018-08-10   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
## 10 2018-08-11   HI        1 FALSE   Mazie K. Hirono      Democ~ TRUE      clas~
```

```
## # ... with 24,227 more rows, and 6 more variables: win_probability <dbl>,
## #   voteshare <dbl>, p10_voteshare <dbl>, p90_voteshare <dbl>, year <dbl>,
## #   month <ord>
```

The candidate is Bernard Sanders.

Next, I'm interested in restricting my attention to just Democrats and Republicans. But it might make my analysis complicated, because independent/third party candidates might actually be consequential in a few states. The next steps are to understand how prevalent this issue is (ie non-Republicans/Democrats with a nontrivial predicted vote share.

```
# Filter the sample to exclude Republicans and Democrats
# By candidate, calculate the mean estimated vote share in the 538 model and mean
↪  probability of victory, across all observations
# Display the candidate, their mean vote share, their party and state, sorted in
↪  descending order by candidate with the highest mean estimated vote share

df_elections_forecast %>%
  filter(party != "D" & party != "R") %>%
  group_by(candidate) %>%
  mutate(voteshare_mean = mean(voteshare),
         win_probability_mean = mean(win_probability)) %>%
  select(candidate, voteshare_mean, party, state) %>%
  arrange(desc(voteshare_mean)) %>%
  distinct(candidate, .keep_all = TRUE)
```

```
## # A tibble: 83 x 4
## # Groups:   candidate [83]
##    candidate             voteshare_mean party       state
##    <chr>                          <dbl> <chr>       <chr>
##  1 Mazie K. Hirono                 75.7 Democrat    HI
##  2 Bernard Sanders                 68.0 Independent VT
##  3 John Barrasso                   67.5 Republican  WY
##  4 Benjamin L. Cardin              66.5 Democrat    MD
##  5 Kirsten E. Gillibrand           66.3 Democrat    NY
##  6 Sheldon Whitehouse              64.5 Democrat    RI
##  7 Elizabeth Warren                63.2 Democrat    MA
##  8 Christopher Murphy              61.5 Democrat    CT
##  9 Angus S. King Jr.               60.9 Independent ME
## 10 Maria Cantwell                  60.7 Democrat    WA
## # ... with 73 more rows
```

```
# Create a tibble, restricting my sample to candidates who ever have both (a) a greater
↪  than 10% chance of winning a race, and (b) a less than 90% chance of winning a race.
↪  Exclude Minnesota because it's not part of the research scope.
competitive_538 <- df_elections_forecast %>%
  filter(win_probability > 0.1 & win_probability < 0.9,
         state != "MN")
```

Working with three datasets: FiveThirtyEight's election forecast, the context for the 2018 election, and historic Senate results

I will explore three research questions in the following analysis:

Research Question 4: How does the election context data compare to the results from New York Times?

Research Question 5: How did democrat party's eletoral support change between 2016 and 2018?

Research Question 6: What is the association of (i) the "shift right" from 2012 to 2016 against (ii) the share voting for a republican candidate in 2018?

Research Question 7: Design a function to analyze the trend of democratic and republican vote share across states from 2012 to 2018 (use Arizona as an example)

```
df_senate_elections <- read.csv("Raw Data/1976-2020-senate.csv") |>
  as_tibble()

df_election_context <- read.csv("Raw Data/election-context-2018.csv") |>
  as_tibble()
```

Data Quality Checks

```
colSums(is.na(df_election_context))
```

```
##                  state                 county                   fips
##                      0                      0                      0
##                trump16              clinton16            otherpres16
##                      0                      0                      0
##                romney12                obama12            otherpres12
##                      0                      0                      0
##                demsen16              repsen16             othersen16
##                   1172                   1172                   1172
##              demhouse16             rephouse16           otherhouse16
##                    252                    252                    252
##                demgov16              repgov16             othergov16
##                    2513                   2513                   2513
##                repgov14              demgov14             othergov14
##                    966                    966                    966
##       total_population                   cvap              white_pct
##                      3                      3                      3
##              black_pct           hispanic_pct           nonwhite_pct
##                      3                      3                      3
##        foreignborn_pct             female_pct      age29andunder_pct
##                      3                      3                      3
##      age65andolder_pct          median_hh_inc       clf_unemploy_pct
##                      3                      3                      3
##             lesshs_pct        lesscollege_pct      lesshs_whites_pct
##                      3                      3                      3
## lesscollege_whites_pct              rural_pct          ruralurban_cc
##                      3                      1                      1
```

Before analysis, I check duplicates: hypothesize that state and county uniquely identifies the observations

```
# Create a context_duplicates tibble, which includes all state-county combinations that
↪  appear more than once in the dataset.
context_duplicates <-
```

```
  df_election_context %>%
  group_by(state, county) %>%
  mutate(freq = n()) %>%
  filter(freq > 1) %>%
  select(state, county, freq)

# Do a semi_join of df_election_context and this context_duplicates tibble to observe all
↪  duplicate cases in df_election_context.
df_election_context %>%
  semi_join(context_duplicates, by=c("county", "state")) %>%
  distinct(state)
```

```
## # A tibble: 1 x 1
##   state
##   <chr>
## 1 Virginia
```

```
# duplicates are all in Virginia
```

Research Question 4: How does the election context data compare to the results from New York Times?

Aggregate the df_election_context data to the state level that includes: (a) the total votes for Trump in 2016 (b) the total votes for Clinton in 2016 (c) the total votes for Romney in 2012 (d) the total votes for Obama in 2012 (e) the total population (f) the percent white (g) the percent black (h) the percent hispanic

```
# create a tibble at the state level
state_level_context <-
  df_election_context %>%
  filter(is.na(total_population) == FALSE) %>%
  group_by(state) %>%
  summarize(trump16_total = sum(trump16),
            clinton16_total = sum(clinton16),
            romney12_total = sum(romney12),
            obama12_total = sum(obama12),
            population_total = sum(total_population),
            white_percent = sum(white_pct * total_population) /
              sum(total_population),
            black_percent = sum(black_pct * total_population) /
              sum(total_population),
            hispanic_percent = sum(hispanic_pct * total_population) /
              sum(total_population))
```

I want to analyze how these aggregate totals for Trump and Clinton in Alabama and Arizona compare to results from the NY Times.

```
# Compare total votes of Trump and Clinton for Alabama and Arizona
state_level_context %>%
  select(trump16_total, clinton16_total, state)
```

```
## # A tibble: 50 x 3
##    trump16_total clinton16_total state
##            <int>           <int> <chr>
##  1       1318250          729547 Alabama
##  2       1252401         1161167 Arizona
##  3        684872          380494 Arkansas
##  4       4483810         8753788 California
##  5       1202484         1338870 Colorado
##  6        673215          897572 Connecticut
##  7        185127          235603 Delaware
##  8         12723          282830 District of Columbia
##  9       4617886         4504975 Florida
## 10       2089104         1877963 Georgia
## # ... with 40 more rows
```

Trump's votes in Alabama according to NY Times are 1,318,255, Clinton's votes are 729,547. Trump's votes in Arizona according to NY Times are 1,252,401, Clinton's votes are 1,161,167. Trump's votes in California according to NY Times are 4,483,810, Clinton's votes are 8,753,788. The numbers are almost the same with the information posted by the NY Times.

Research Question 5: How did democrat party's eletoral support change between 2016 and 2018?

```
# Create a new tibble:
# Keep only the 2018 election year. Keep only "DEMOCRAT" and "REPUBLICAN". Keep only
↪  races for which the stage is "gen" (general election). Create a party variable that
↪  codes an individual as "D" if they are Democrat, and "R" if Republican.

senate_results_2018 <- df_senate_elections %>%
  filter(year == 2018,
         (party_simplified == "DEMOCRAT"| party_simplified =="REPUBLICAN"),
         stage == "gen") %>%
  mutate(party = ifelse(party_simplified == "DEMOCRAT", "D", "R"))
```

```
# Check whether state and party jointly uniquely identify the observations
paste0("Percentage of unique observations: ",
       ((senate_results_2018 %>%
           distinct(state, party) %>%
           nrow())/
          senate_results_2018 %>% nrow())*100,
       "%")
```

```
## [1] "Percentage of unique observations: 92.7536231884058%"
```

State and party cannot jointly uniquely identify the observations.

```
print("States with duplicate values: ")
```

```
## [1] "States with duplicate values: "
```

```r
senate_results_2018 %>%
  group_by(state, party) %>%
  summarize(freq = n()) %>%
  filter(freq > 1) %>%
  select(state, party, freq)
```

```
## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   state [3]
##   state       party  freq
##   <chr>       <chr> <int>
## 1 CALIFORNIA  D         2
## 2 MINNESOTA   D         2
## 3 MINNESOTA   R         2
## 4 MISSISSIPPI D         2
## 5 MISSISSIPPI R         2
```

To analyze "senate_results_2018" and "state_level_context" together, I will merge the two datasets. Before that, a few steps need to be taken before merging:

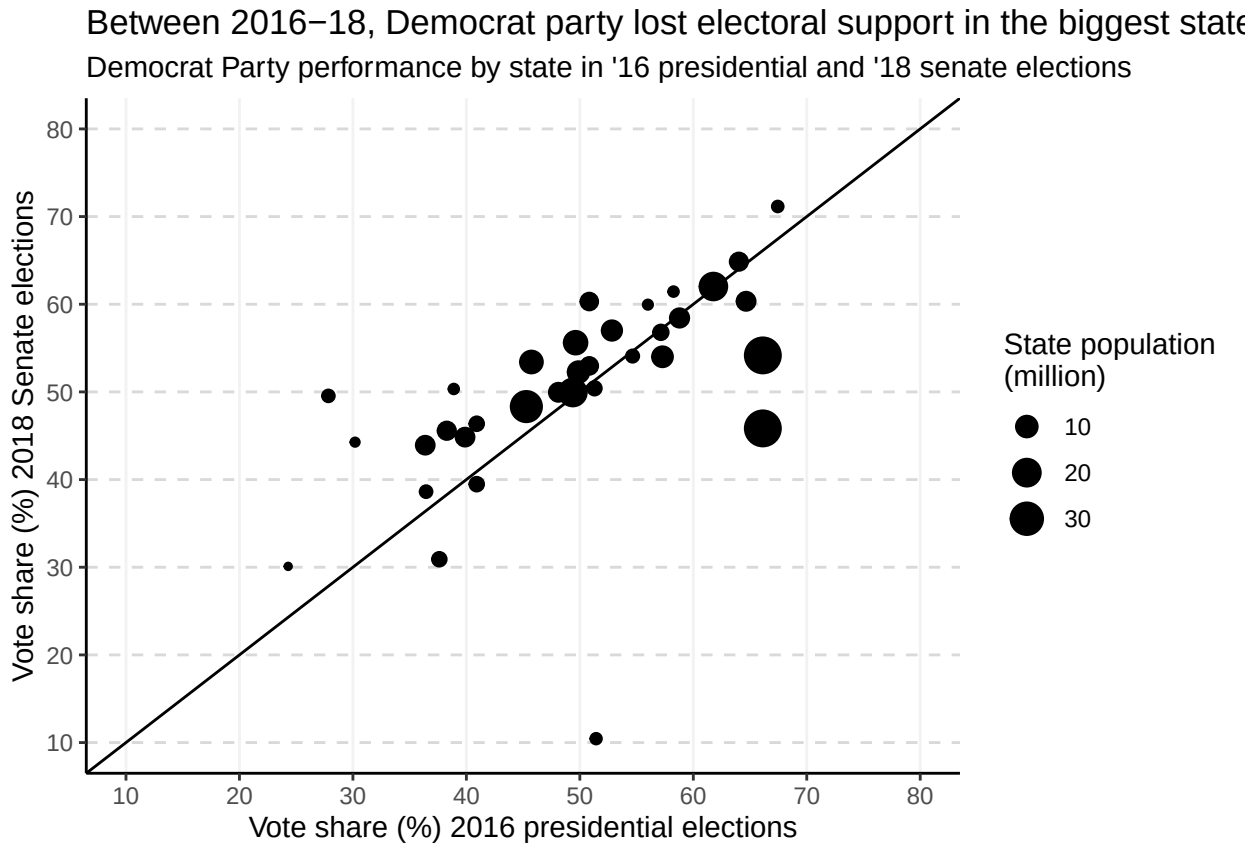
```r
# Change both datasets to lowercase
senate_results_2018$state <- tolower(senate_results_2018$state)
state_level_context$state <- tolower(state_level_context$state)
```

```r
# Implement left_join of "state-level election context" and "senate_results_2018"
senate_results_context <- left_join(senate_results_2018, state_level_context, by =
↪  "state")
```

Create a scatter plot of the share of candidates who voted for Clinton in 2016 against the share of candidates who voted for a democratic candidate in 2018, with the size of the point scaled by total population:

```r
senate_results_context %>%
  filter(party == "D") %>%
  mutate(share_clinton16 = clinton16_total/(clinton16_total+trump16_total),
         share_democrat18 = candidatevotes/totalvotes) %>%
  ggplot(aes(x=share_clinton16, y=share_democrat18)) +
  geom_point(aes(size=population_total)) +
  theme_classic() +
  theme(panel.grid.minor = element_blank(),
        panel.grid.major.y = element_line(color = "grey85", linetype = "dashed"),
        panel.grid.major.x = element_line(color = "grey95")) +
  scale_y_continuous(labels = function(x) format(x*100, decimal.mark = "."),
                     limits = c(0.1, 0.8),
                     n.breaks = 8) +
  scale_x_continuous(labels = function(x) format(x*100, decimal.mark = "."),
                     limits = c(0.1, 0.8),
                     n.breaks = 8) +
  scale_size_continuous(labels = function(x) format(x/1000000, scientific = FALSE)) +
  geom_abline(intercept = 0, slope = 1) +
```

```
  labs(x="Vote share (%) 2016 presidential elections",
       y="Vote share (%) 2018 Senate elections",
       size = "State population\n(million)",
       title="Between 2016-18, Democrat party lost electoral support in the biggest
       ↪  states",
       subtitle = "Democrat Party performance by state in '16 presidential and '18 senate
       ↪  elections") ->plot_rq5

print(plot_rq5)
```

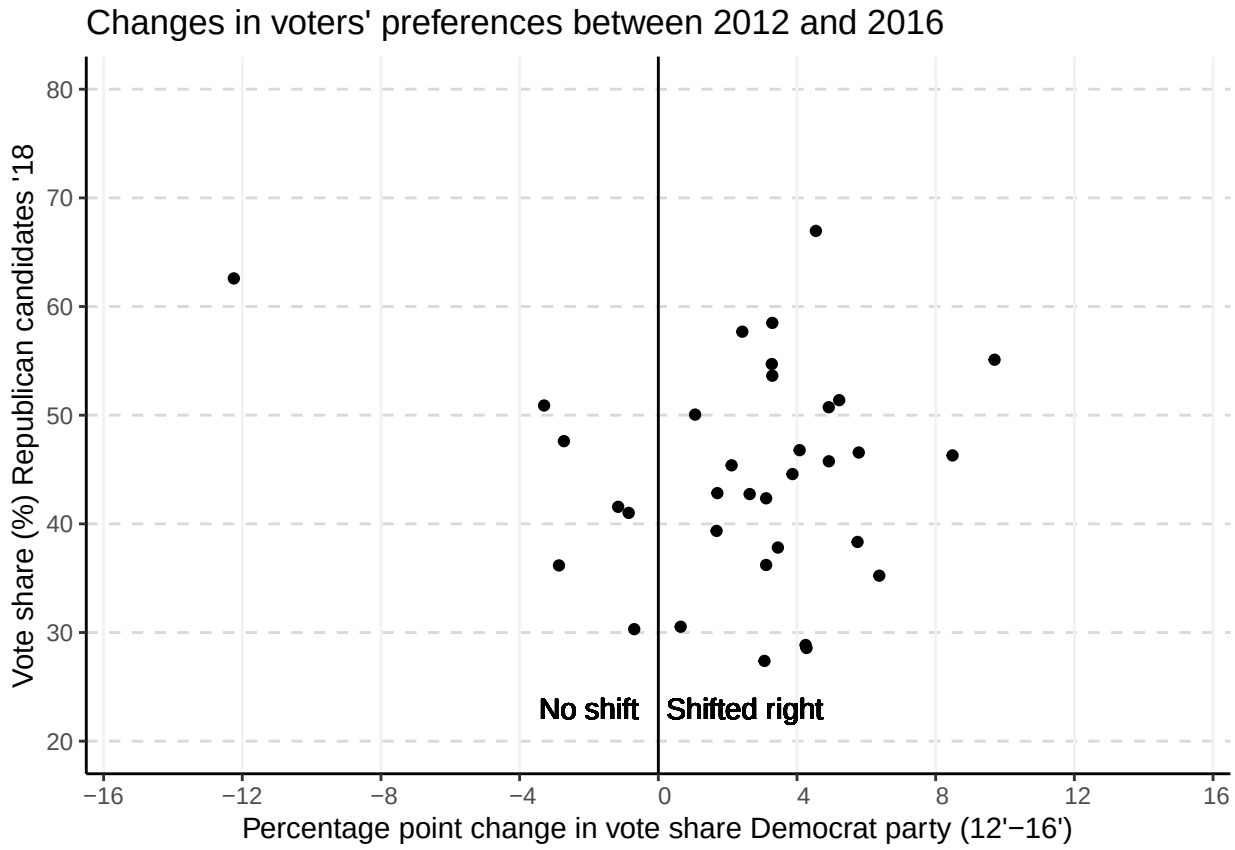



```
ggsave("scatter_voteshare_2018senate_2016president.png", plot_rq5, width = 6, height = 4,
↪  dpi = 300)
```

Research Question 6: What is the association of (i) the "shift right" from 2012 to 2016 against (ii) the share voting for a republican candidate in 2018?

```
senate_results_context %>%
  filter(party == "R") %>%
  mutate(share_clinton16 = clinton16_total/(clinton16_total+trump16_total),
         share_obama12 = obama12_total/(obama12_total+romney12_total),
         share_democrats_change = (share_clinton16 - share_obama12)* - 1,
         shift_right = ifelse(share_democrats_change > 0, "Shifted right",
```

```
                              "No"),
         share_republican18 = candidatevotes/totalvotes) %>%
  ggplot(aes(x=share_democrats_change, y=share_republican18)) +
  geom_point() +
  theme_classic() +
  theme(panel.grid.minor = element_blank(),
        panel.grid.major.y = element_line(color = "grey85", linetype = "dashed"),
        panel.grid.major.x = element_line(color = "grey95")) +
  scale_y_continuous(labels = function(x) format(x*100, decimal.mark = "."),
                     limits = c(0.2, 0.8), n.breaks = 8) +
  scale_x_continuous(labels = function(x) format(x*100, decimal.mark = "."),
                     limits = c(-0.15, 0.15), n.breaks = 10) +
  geom_vline(xintercept = 0) +
  geom_text(aes(x=-0.02, y=0.23,label="No shift"), size=4) +
geom_text(aes(x=0.025, y=0.23,label="Shifted right"), size=4) +
  labs(x="Percentage point change in vote share Democrat party (12'-16')",
     y="Vote share (%) Republican candidates '18",
     title = "Changes in voters' preferences between 2012 and 2016") ->plot_rq6

print(plot_rq6)
```




```
ggsave("scatter_votesharechange_rep_dem_2012_2016.png", plot_rq6, width = 6, height = 4,
↪  dpi = 300)
```

There does not seem to be a strong pattern based on whether a state "shifted" towards right between 2012

and 2016. To test that, I designed a regression:

$$\text{ShareR}_{2018,s} = \beta_0 + \beta_1 \,\text{RightShift}_s + \gamma^\top C_s + \varepsilon_s$$

Y: Republican vote share in state S in 2018 general election.

X: Change in Democratic vote share in state S from 2012 to 2016.

Controls (Cs): % White, % Black, % Hispanic, log(Population).

```
# Create a tibble with X, Y, and controls
df_reg <- senate_results_context %>%
  filter(party == "R") %>%
  mutate(
    demshare_2016 = clinton16_total / (clinton16_total + trump16_total),
    demshare_2012 = obama12_total   / (obama12_total   + romney12_total),
    right_shift   = -(demshare_2016 - demshare_2012),
    share_r_2018  = candidatevotes / totalvotes,
    ln_pop        = log(population_total)
  ) %>%
    tidyr::drop_na(share_r_2018, right_shift, white_percent, black_percent,
↪  hispanic_percent, ln_pop)

# Baseline regression without controls
m1 <- lm(share_r_2018 ~ right_shift, data = df_reg, weights = population_total)

# Full regression with controls
m2 <- lm(share_r_2018 ~ right_shift + white_percent + black_percent + hispanic_percent +
↪  ln_pop,
         data = df_reg, weights = population_total)

coeftest(m1, vcov = vcovHC(m1, type = "HC2"))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.452016   0.018642 24.2473   <2e-16 ***
## right_shift -0.570433   0.514873 -1.1079   0.2762
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

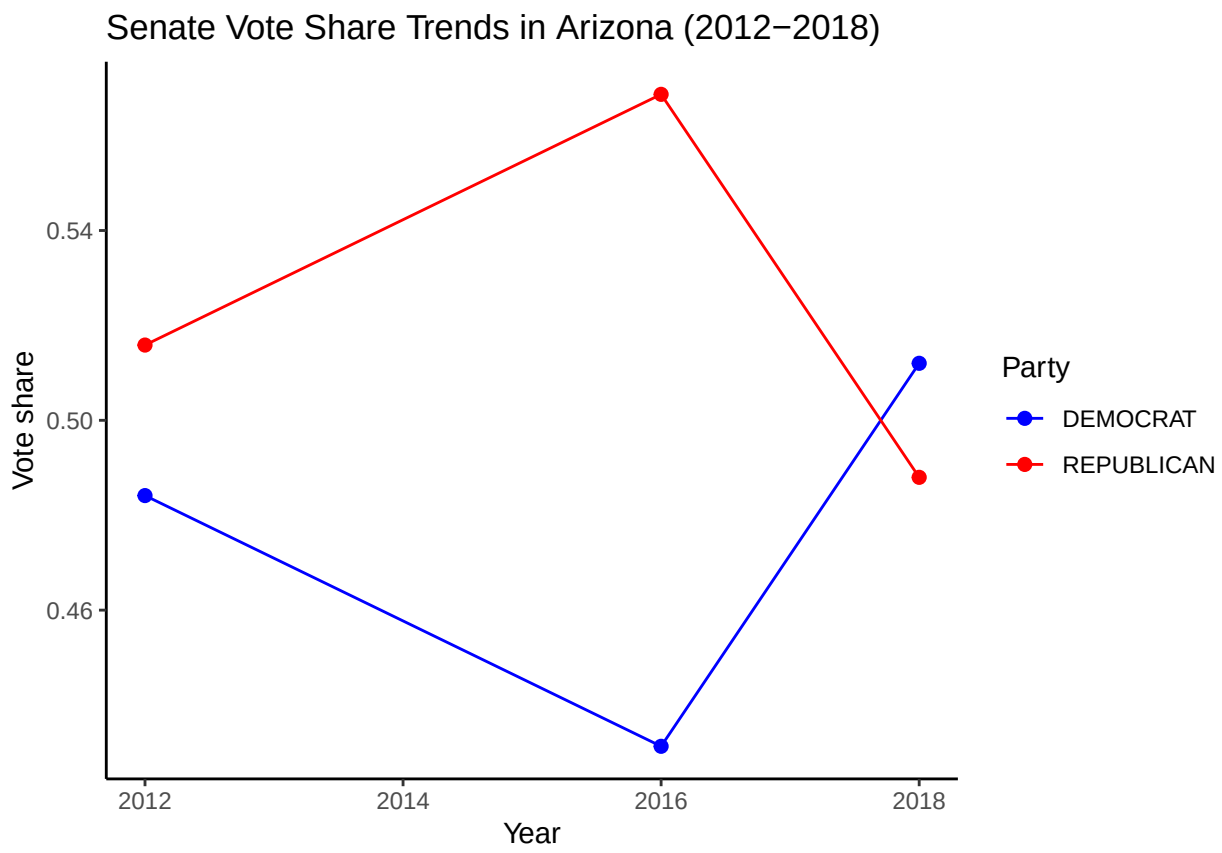
```
coeftest(m2, vcov = vcovHC(m2, type = "HC2"))
```

```
##
## t test of coefficients:
##
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.0727062  0.7211468  0.1008   0.9204
## right_shift      -0.6017449  0.6689622 -0.8995   0.3760
## white_percent     0.0073861  0.0072456  1.0194   0.3167
## black_percent     0.0072391  0.0070526  1.0264   0.3135
## hispanic_percent  0.0080523  0.0076490  1.0527   0.3015
## ln_pop           -0.0194782  0.0247703 -0.7864   0.4383
```

The results confirmed that the association between states shifting right in 2016 senate election and the vote share of republican in 2018 presidential election is not significant, which means states shifting right in 2016 do not necessarily elect more Republican Senators in 2018.

Research Question 7: Design a function to analyze the trend of democratic and republican vote share across states from 2012 to 2018 (visualize Arizona as an example)

```
# a. Define a function to subset one year's vote share
get_senate_results <- function(df, year_select) {
  df %>%
    filter(year == year_select,
           stage == "gen",
           party_simplified %in% c("DEMOCRAT","REPUBLICAN")) %>%
    group_by(state) %>%
    mutate(total_state_votes = sum(candidatevotes, na.rm=TRUE)) %>%
    group_by(state, party_simplified) %>%
    summarise(
      total_votes = sum(candidatevotes, na.rm=TRUE),
      total_state_votes = first(total_state_votes),
      .groups = "drop"
    ) %>%
    mutate(
      year = year_select,
      share = total_votes / total_state_votes
    )
}

# b. Apply function to multiple years
years_to_check <- c(2012, 2016, 2018)
results_list <- purrr::map(years_to_check,
                           ~ get_senate_results(df_senate_elections, .x))

# c. Combine results into one tibble
results_all <- dplyr::bind_rows(results_list)

# d. Create a line chart: party share trends by year for Arizona
results_all %>%
  filter(state == "ARIZONA") %>%
  ggplot(aes(x = year, y = share, color = party_simplified)) +
  geom_line() + geom_point(size=2) +
  theme_classic() +
  labs(title = "Senate Vote Share Trends in Arizona (2012-2018)",
       y = "Vote share", x = "Year", color = "Party"
       ) +
  scale_color_manual(
    values = c("DEMOCRAT" = "blue",
               "REPUBLICAN" = "red")) -> plot_rq7

print(plot_rq7)
```




```
ggsave("line_voteshare_rep_dem_arizona_2012_2018.png", plot_rq7, width = 6, height = 4,
↪  dpi = 300)
```

```
df_senate_elections %>% filter(year == "2014", state == "ARIZONA")
```

```
## # A tibble: 0 x 19
## # ... with 19 variables: year <int>, state <chr>, state_po <chr>,
## #   state_fips <int>, state_cen <int>, state_ic <int>, office <chr>,
## #   district <chr>, stage <chr>, special <lgl>, candidate <chr>,
## #   party_detailed <chr>, writein <lgl>, mode <chr>, candidatevotes <int>,
## #   totalvotes <int>, unofficial <lgl>, version <int>, party_simplified <chr>
```