STAT 184 Final Project

Prajwal Bhandari, Sam Johnson, Jack Mariani

Introduction

For our final project our group decided to explore data relating to previous presidential elections and explore questions relating to it. We explored a couple of topics before landing on this topic. The reason we decided presidential election data is because it is easily accessible; it is public information. We reasoned that we could explore a plethora of different questions. To that end, we came up with these research questions that we then explored in the data analysis.

1. What is the relationship between Pennsylvania, Michigan, and Wisconsin voting for the same candidate, and what can that tell us about the winner of the general election?
2. How does being in a high or low margin affect the 'power of your vote', and if so, is the power statistically significant?

Other items which we explored in the analysis were the historical vote share for each party, along with utilizing a Support Vector Machine classification model on the data to see whether or not the data was sufficient to predict state election winners.

Literature Review

Throughout our exploration of our data we found some things that we were not fully aware of prior to the beginning of our research. For example, during our research we noticed some issues with the amount of votes each candidate got on our table and in the final vote from electors. We found out that the website where we got our data from did not account for faithless voters, which is when an elector goes against the wishes of the popular vote and votes for who they want to win. There are multiple times where electors voted their own way. A recent example was back in 2016, when there were ten deviant electors who voted for other candidates(Oberstaedt, 2024).

Another issue we came across in our research of our data were Nebraska and Maine's system of allocating electoral votes. In most states it is a "winner takes all" system where whomever

receives the most votes in a state receives all of the electoral votes from the state, even in split decision votes where the difference was less than one percent. Nebraska and Maine on the other hand divide up the electoral votes to better represent the wishes of the people who live there. Two electoral votes are allocated to the state winner, and the rest are given to the winner of the congressional districts. This makes it possible for a candidate to win three out of 5 electoral votes while the other candidate wins the other two, in the case of Nebraska(Wikimedia projects, 2024b).

Similarly, Maine also allocates two electoral votes for the state-wide winner and the remaining two for the electoral districts (Wikimedia projects, 2024c).

Following the past election we kept hearing about how the election was coming down to the "Blue Wall States" and how they historically vote together. We decided to learn more about the topic before further exploration of data. Wisconsin, Michigan, and Pennsylvania make up the blue wall states and they are coveted by both parties because of the nature of the people living within them. These states are considered swing states, which means that they are not controlled by either party and could end up voting for either party. The 44 electoral votes that make up the blue wall is the key for both parties to help win an election(Crowley, 2024). Both parties for the 2024 election spent a lot of time campaigning in these states to win the votes. In fact, nearly $16 billion was spent this election cycle, with $62 million going towards MI, WI, and PA during last-minute ad campaigns(Jeffrey-Wilensky, 2024).

Methodology

For data collection, we utilized the University of California Santa Barbara's American Presidency Project(University of California, n.d.), whose purpose is to provide statistics for students in universities to use freely. For this report, we only used data for the past 10 elections, from 1980 to 2020, as 2024 election data was largely incomplete. The data also met CARE and FAIR principles, which was another reason for which we used the project website as a primary data source.

The data meets CARE in the following ways:

C: This data is available for the collective benefit of university students.

A: The host of this project maintains the data ensuring legitimacy. Additionally, users are informed of the data's provenance.

R: The data has citations and sources.

E: The data consists of aggregate elections results and public records, which do not share any sensitive data about individual voters.

The data meets FAIR in the following ways:

F: The data is easy to find. When Google searching 'presidential election data UCSB', the project's website was the first link.

A: The data is free for anyone to look at and use without requiring special permissions. The data is also available in a format which can be easily parsed.

I: The structure of the data reporting allows it to be used in a variety of ways after accessing. For this report, we used this data to create a larger data set to work from.

R: The data is sufficiently detailed for its use in other contexts and contains references to where the data was sourced from.

Due to the nature of the data, we decided that it would be best to import 10 tables to begin with, clean them, and then concatenate them to a single table through R's `rbind` function. Doing this allowed us to customize the attributes that we wished to keep and edit the data tables individually in case of any discrepancies, of which there were many.

A key limitation of the data were missing values, and incorrect values, which had to be corrected. This did present a small problem initially, however, through careful and meticulous cross-validation with other sources, we were able to have data which accurately reflected the true results of the elections. Many of the missing values were simple to solve: we filled them with zero, as there was no data to report.

However, the issue of faithless electors was not so simple. The biggest hurdle we faced before conducting the analysis is to fix the discrepancies of faithless electors in 1988, 2000, and 2016.

Nina Agrawal of the Los Angeles Times mentions all the times throughout American history in which electors have 'defected'. For the timeframe which we worked with, there were three main cases which we mentioned above briefly, but here we will give more specific details about what happened.

In 1988, Democratic elector Margarette Leach from West Virginia voted for vice presidential nominee Lloyd Bentsen as president and Michael Dukakis as Vice President. Leach told the New York Times that she "… wanted to make a statement about the electoral college." (Agrawal, 2016). In 2000, an elector from the state of DC had abstained from voting in protest of DC's lack of congressional representation(Agrawal, 2016). The most interesting case of faithless electors is in 2016, where there were a total of 7 defectors: 1 from Hawaii, 2 from Texas and 4 from Washington state(Wikimedia projects, 2024a).

To account for such discrepancies between the data we collected and the true data, we edited individual cells in our data to more accurately reflect the data.

Results and Discussion

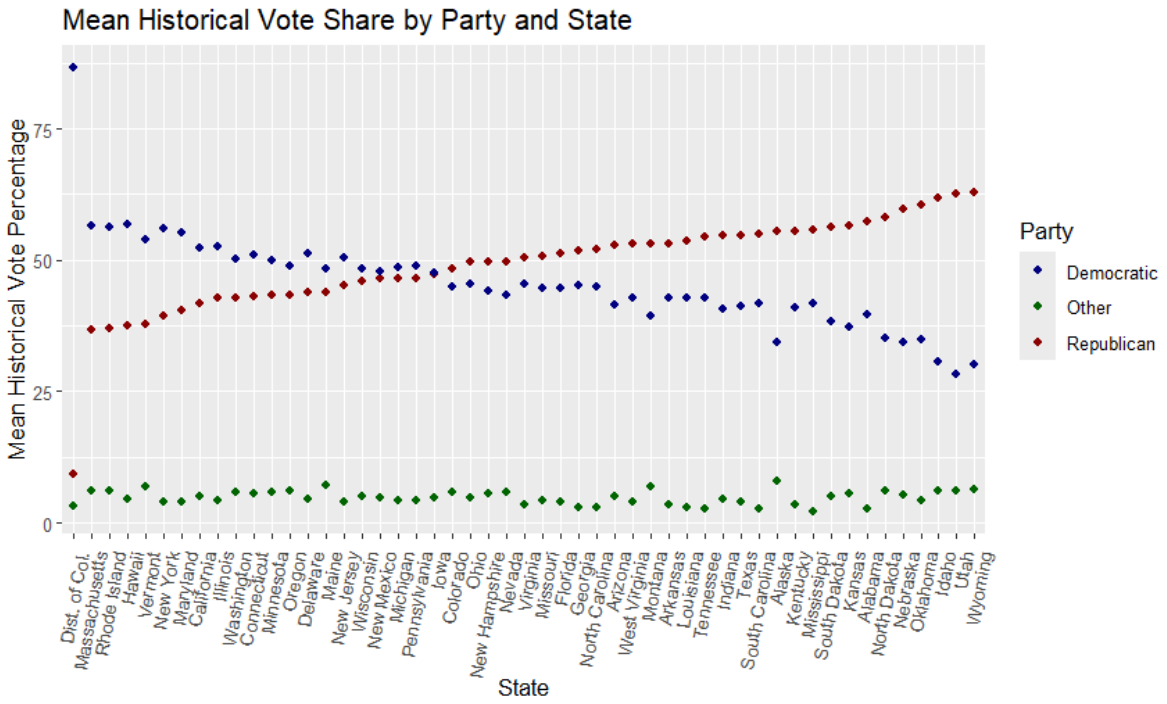

Figure 1: Graph of mean historical vote share of parties in order by state

The first thing that we explored was voting trends throughout the elections. We wanted to see how the party makeup of each state's historical vote was shared. We decided to use the sample mean in this case because we reasoned that states would note vote so differently year to year barring huge demographic changes.

The results from Figure 1 also helped inform our decision to focus on Michigan, Wisconsin, and Pennsylvania for the following table, As their vote share indicates that Republicans and Democrats had, on average, less than a 5% difference in vote share, with Pennsylvania being nearly dead even.

It's interesting to see that third parties have gotten on average 10% vote for each state. We attribute the high average to Ross Perot's presidential campaign in 1992, where he secured nearly 20% of the national popular vote.

MI, WI, and PA account for 44 electoral votes and are typically considered the most competitive states leading up to an election(Crowley, 2024). As part of our research questions, we wanted to explore the relationship between when they all vote for the same candidate and whether or not that candidate wins that election.

Frequency of National Winner Predictions when MI, PA, WI Vote together

Correct Prediction/Voted Together	FALSE	TRUE
FALSE	1	0
TRUE	7	3

Figure 2: Frequency table of national winner predictions when MI, PA, WI vote together

In Figure 2, the rows indicate whether or not the states predicted the national winner correctly, and the columns indicate whether or not the states voted together during a given election. From this table, it's safe to say that when the three states make a correct prediction they do not end up voting together at a rate of 70 to 30%. It's interesting to see that, when they did not vote together, they did not make any correction predictions for the general election.

Voters, especially those that live in states with competitive elections, are typically inundated with messaging about the 'power of a vote'. It's the message that a small number of votes can have drastic effects on public policy. To an extent this is true, as described by Rebecca Mears and Zachary Geiger in an Article for Center for American Progress. Historically, there have been many times in which a small number of votes have decided outcomes of elections. For example, "From 1976 to 2022, more than 410 U.S. House elections were decided by less than 3 percent of all votes cast."(Mears & Geiger, 2024). In 2020, Joe Biden won Georgia by 11779 votes (0.23%), Arizona by 10457 votes (0.3%), and Wisconsin by 20682 votes (0.63%). Compared to all votes, those three states account for just 0.03% of all votes cast in the 2020 election(Mears & Geiger, 2024).

To explore this further in our data, we used a method called bootstrapping. Bootstrapping is about taking samples from an existing data set repeatedly to estimate the distribution of a statistic in a data set. Here, We wanted to measure the difference in the "power of a vote" between elections with low and high percentage margins. The cutoff we used was 8%. To do so, we created two data sets that met the criteria and sampled 200,000 values from each set.

Our goal was to measure the voting share of each margin voter in the elections. To do so, we divided the number of margin voters in each state and divided it by the number of electoral votes in each state. Then we would compare the distribution of sample means to see if there were any significant differences.

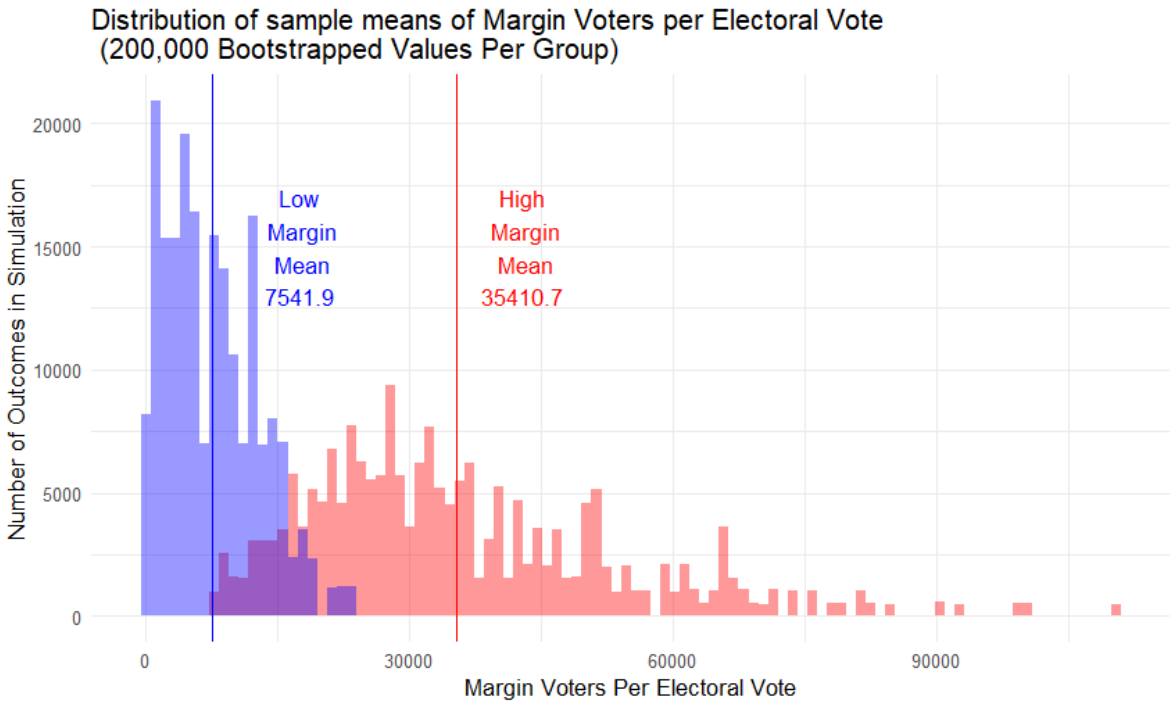

Figure 3: Histogram of Sample Mean Distribution

Numerical Results for Bootstrap Simulations

High or Low Margin	Mean Vote per EV	SD Vote per EV	SE Vote Per EV	Lower 95% CI	Upper 95% CI
High	35410.691	17678.757	39.53090	35333.212	35488.171
Low	7541.914	5252.531	11.74502	7518.894	7564.934

Figure 4: A table displaying means, standard deviations and errors, and upper and lower thresholds for 95% CI.

From Figure 3, we see that the mean voters per electoral vote was nearly 5 times smaller in low margin states than in high margin states (7541.9 compared with 35410.7). This makes sense heuristically, and in order to verify those findings, we constructed a table of means, standard deviations and errors and confidence intervals for the two groups. We conducted a hypothesis test with

$$H_0 : \mu_H = \mu_L \quad \text{and} \quad H_A : \mu_H \neq \mu_L$$

We see from Figure 4 that the estimates of population means are clearly outside of the other groups' confidence intervals, so we reject the null hypothesis.

For an added bonus, we thought it would be interesting to see whether our data could be useful for a Support Vector Machine (SVM) model to predict the winners of each state election. We used a 70% split to the data and ran the model. One challenge with the SVM model was that all predictor variables would have to be numerical values. To work this challenge, we had to come up with a clever way to encode the state values as numerical, since they are character values in the original data set. Once we figured that out, we ran the model and produced these results.

We are quite surprised with the results of this model (seen in Figure 5) as typical models do not always achieve 100% accuracy. We believe that there were some variables in the data which gave away the winning party of each state, although further investigation and testing would be required.

Confusion Matrix for Testing Results from 70% Data Split

Predicted	Actual	Freq
Democrat	Democrat	66
Republican	Democrat	0
Democrat	Republican	0
Republican	Republican	103

Figure 5: A table displaying the results of applying the SVM model to the data with a 70% training/testing split

Code appendix and References

```
## Prajwal Bhandari, Jack Mariani, Sam Johnson
# data analysis script


# import required libraries and the data
library(tidyverse)
library(rvest)
library(ggplot2)
library(tidyr)
library(dplyr)
library(rvest)
library(readr)
library(googlesheets4)
library(knitr)
library(janitor)
library(esquisse)
library(infer)

# getting the data
gs4_deauth()
data <- read_sheet(
  ss = "https://docs.google.com/spreadsheets/d/1GT8UOmD4l2j88wCLfCAlh_R7r0jWCtsmoL84kVvDOlc/e
)

# viewing the data on the web
# esquisser(data = data, viewer = 'browser')


## visualize historical vote share by state and party ----
rep_dem_other <- data %>%
  group_by(state) %>%
  summarize(
    mean_r = mean(r_percent),
    mean_d = mean(d_percent),
    mean_o = mean(other_percent)
  ) %>%
  mutate(state = fct_reorder(state, mean_r))

# visualization of historical vote share for each state and DC
  ggplot() +
```

```
    geom_point(data = rep_dem_other, aes(x = state, y = mean_r, color = 'red')) +
    geom_point(data = rep_dem_other, aes(x = state, y = mean_d, color = 'blue')) +
    geom_point(data = rep_dem_other, aes(x = state, y = mean_o, color = 'green')) +
    theme(axis.text.x = element_text(angle = 80, vjust = 0.95, hjust = 1)) +
    scale_color_manual(
       name = 'Party',
       values = c("blue" = "navy", "green" = "darkgreen", "red" = "darkred"),
       labels = c("blue" = "Democratic", "green" = "Other", "red" = "Republican")) +
    labs(
      x = 'State',
      y = 'Mean Historical Vote Percentage',
      title = "Mean Historical Vote Share by Party and State"
      )



## table on three key blue wall states ----

# States: MI, PA, WI.
# We wish to see what party usually wins when they vote in a block
# Create two new columns, all_together and national_winner for this table.

blue_wall <-  data %>%
  group_by(year) %>% # have an election winner for every year
  mutate(
    election_winner = as.factor(case_when(
      sum(r_ev) > sum(d_ev) + sum(other_ev) ~ "Republican",
      sum(d_ev) > sum(r_ev) + sum(other_ev) ~ 'Democratic',
      sum(other_ev) > (r_ev) + (d_ev) ~ 'Other'
        )
      )
    ) %>%
  ungroup() %>%
  filter(
    state %in% c('Michigan', 'Pennsylvania', 'Wisconsin')
    # filter for three main blue wall states
  ) %>%
  group_by(year) %>%
  mutate(
    all_together = if_else(n_distinct(winning_party) == 1, TRUE, FALSE)
    # if they all voted for the same party
    ) %>%
```

```
  ungroup() %>%
  group_by(
    year
    ) %>%
  summarize(
  all_together = first(all_together),
  state_winner = first(winning_party),
  national_winner = first(election_winner)
  ) %>%
  mutate(
    correct_prediction = if_else(
      all_together & state_winner == national_winner,
      TRUE,
      FALSE)
    )
# the table itself
initial_table <- tabyl(blue_wall, all_together , correct_prediction) %>%
  # make it look nice with adorn
  adorn_title(
    placement = 'combined',
    row_name = "Correct Prediction",
    col_name = 'Voted Together'
  )
# nice formatting with kable
kable(
 initial_table,
 caption = 'Frequency of National Winner Predictions when MI,
 PA, WI Vote together',
 align = 'c'
) %>%
  kableExtra::kable_classic(
    full_width = FALSE
  )


## bootstrapping and numerical results ----

# use a sample size of 100 and 2000 repetitions for each

set.seed(123) # set seed for reproduibility
bstrap_high_margin <- data.frame(
  data %>%
```

```
    filter(win_margin_percent > 8) %>%
    rep_sample_n(size = 100, reps = 2000, replace = TRUE)
  )

bstrap_low_margin <- data.frame(
  data %>%
    filter(win_margin_percent < 8) %>%
    rep_sample_n(size = 100, reps = 2000, replace = TRUE)
 )
# we now plot a histogram  and also review the numerical results
ggplot() +
  geom_histogram(aes((win_margin_votes / total_ev)),
                 data = bstrap_high_margin, bins = 100,
                 alpha = 0.4, fill = 'red'
                 ) +
  geom_histogram(aes((win_margin_votes / total_ev)),
                 data = bstrap_low_margin, bins = 100,
                 alpha = 0.4, fill = 'blue'
                 ) +
  geom_vline(xintercept = mean(bstrap_high_margin$win_margin_votes /
                                 bstrap_high_margin$total_ev), color = 'red') +
  geom_vline(xintercept = mean(bstrap_low_margin$win_margin_votes /
                                 bstrap_low_margin$total_ev), color = 'blue') +
  labs(
    x = 'Margin Voters Per Electoral Vote',
    y = 'Number of Outcomes in Simulation',
    title = 'Distribution of sample means of Margin Voters per Electoral Vote\n
    (200,000 Bootstrapped Values Per Group)'
    ) +
  annotate(
    'text',
    x = mean(bstrap_high_margin$win_margin_votes / bstrap_high_margin$total_ev)
    + 7500,
    y = 15000,
    label = paste0('High\n Margin\n Mean\n',
                   round(mean(bstrap_high_margin$win_margin_votes /  bstrap_high_margin$total
    color = 'red',
    angle = 0
    ) +
  annotate(
    'text',
    x = mean(bstrap_low_margin$win_margin_votes / bstrap_low_margin$total_ev)
```

```
    + 10000,
    y = 15000,
    label = paste0('Low\n Margin\n Mean\n',
                   round(mean(bstrap_low_margin$win_margin_votes / bstrap_low_margin$total_ev
    color = 'blue',
    angle = 0,
    vjust = 0.5
    ) +
  theme_minimal()

# review of numerical results

init_table_bstrap_num_results <- rbind(bstrap_high_margin, bstrap_low_margin) %>%
  mutate(
    high_or_low = if_else(win_margin_percent > 8, 'High', 'Low')
    ) %>%
  group_by(high_or_low) %>%
  summarize(
    mean_vote_per_ev = mean(win_margin_votes / total_ev),
    sd_vote_per_ev = sd(win_margin_votes / total_ev),
    se_vote_per_ev = sd(win_margin_votes / total_ev) / sqrt(n()), # standard error
    # confidence intervals
    lower_ci = mean_vote_per_ev - qt(0.975, df = n() - 1) * se_vote_per_ev,
    upper_ci = mean_vote_per_ev + qt(0.975, df = n() - 1) * se_vote_per_ev
  )

# bstrap_res_table <-; TODO: polish the table
polished_bstrap_res <- tibble(init_table_bstrap_num_results) %>%
  kable(
  caption = '\t\t\t\t\t\t\tNumerical Results for Bootstrap Simulations',
  col.names = c("High or Low Margin",
                "Mean Vote per EV",
                "SD Vote per EV",
                'SE Vote Per EV',
                "Lower 95% CI",
                "Upper 95% CI"
                ),
  align = 'c'
  ) %>%
  kableExtra::kable_classic(full_width = FALSE)
```

```r
## incorporating a SVM model ----

# import required library
library(e1071)
set.seed(123) # seed for reproducibility

# create numeric data frame by mutating original data frame

svm_data <- data %>%
  mutate(
    state = as.numeric(row_number() %% 51),
    third_party_swing_potential = if_else(
      third_party_swing_potential == TRUE,1 , 0
      ),
    winning_party = as.factor(winning_party)
  )

# split training data
n <- nrow(svm_data)
train_set <- sample(1:n, size = 0.7*n) #train on 70% of the data
training_data <- svm_data[train_set, ]
testing_data <- svm_data[-train_set, ]
# run model and predict
svm_model <- svm(winning_party ~ ., data = training_data, kernel = 'linear')
# linear kernel splits the data linearly into groups
#
true_results <- testing_data$winning_party
prediction <- predict(svm_model, newdata = testing_data)
# check results
confusion_matrix <- table(Predicted = prediction, Actual = true_results)
print(confusion_matrix)

accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
print(accuracy)

# polish confusion matrix table
data.frame(confusion_matrix) %>%
  kable(
  caption = 'Confusion Matrix for Testing Results from 70% Data Split'
  ) %>%
  kableExtra::kable_classic(
  full_width = FALSE
```

```
)
```


Agrawal, N. (2016). All the times in u.s. History that members of the electoral college voted their own way. *Los Angeles Times*. https://www.latimes.com/nation/la-na-faithless-electors-2016-story.html

Crowley, K. (2024). What is the "blue wall"? Latest polls from key states of michigan, pennsylvania, wisconsin. *USA TODAY*. https://www.usatoday.com/story/news/politics/elections/2024/10/29/blue-wall-states/75912615007/

Jeffrey-Wilensky, J. (2024). $16 billion will be spent in the 2024 election. Where's it all going? *U.S. News & World Report*. https://www.usnews.com/news/national-news/articles/2024-11-01/16-billion-will-be-spent-in-the-2024-election-wheres-it-all-going

Mears, R., & Geiger, Z. (2024). The power of one vote. *Center for American Progress*. https://www.americanprogress.org/article/the-power-of-one-vote/

Oberstaedt, M. (2024). Do faithless electors change presidential election results? *FairVote*. https://fairvote.org/do-faithless-electors-change-presidential-election-results/

University of California, S. B. (n.d.). Election listing. In *The American Presidency Project*. https://www.presidency.ucsb.edu/statistics/elections/

Wikimedia projects, C. to. (2024a). *2016 united states presidential election*. https://en.wikipedia.org/wiki/2016_United_States_presidential_election

Wikimedia projects, C. to. (2024b). *2016 united states presidential election in nebraska*. https://en.wikipedia.org/wiki/2016_United_States_presidential_election_in_Nebraska

Wikimedia projects, C. to. (2024c). *United states presidential elections in maine*. https://en.wikipedia.org/wiki/United_States_presidential_elections_in_Maine