

Election Campaign Projections Using Python

Trace Florio, Yi Luo, Jackson Tabors, Abby Antagnoli

Introduction

As the United States graduates from its recent elections, politics is once again dominating national discourse. In an era where political divisions have intensified, the significance of effective campaign strategies cannot be overstated. While well-funded candidates often have the resources to sustain their campaigns over extended periods, the real challenge lies in targeting the right voters in key battleground states. These states, often referred to as swing states, hold the power to tip the balance in a close election. In recent elections, the margins of victory in these states have been razor-thin, and even a small number of undecided voters can make the difference between success and failure.

In light of this, election campaigns must efficiently allocate resources to areas that are likely to determine the election outcome. However, predicting which states will be decisive in each cycle is a difficult task. The dynamics of each election are unique, influenced by factors such as voter turnout, demographic shifts, and changing political climates. While polling data can offer insights into current voter preferences, it may not always reflect the nuances of voter behavior across diverse regions.

This project seeks to address these challenges by utilizing Python to analyze historical election data to identify states that could be pivotal in future elections. By examining historical voting patterns this analysis aims to predict which states will likely become the most contested in upcoming election cycles. This approach provides a data-driven foundation for campaign teams to allocate resources more effectively and make more informed decisions on where to focus their efforts.

Problem Statement

The U.S. electoral landscape has become increasingly polarized, and election outcomes often hinge on a narrow margin in key states. Identifying and targeting these critical regions is essential for effective campaigning. This project aims to explore how we can use coding to help predict which states are most likely to play a decisive role in an election. Using Python, we will analyze historical election results to develop a model that can predict which states will be crucial in future elections. The model will help guide campaign strategies, ensuring resources are directed toward the most influential regions.

The goal of this project is to identify the states most likely to determine the outcome of the 2024 U.S. presidential election. Using data from the 2020 election as a case study, we will analyze voting patterns from that year to understand trends in voter behavior. By combining this analysis

with current polling data, the project aims to predict how shifts in voter preferences could impact the 2024 election and identify the states that are likely to play a decisive role.

Data Collection

To effectively model and predict which states will be most critical in the next election, we will gather data from a variety of reputable sources. Historical election data from the past two decades will be sourced from USA.gov and the Federal Election Commission (FEC). These datasets provide a detailed view of vote shares, voter turnout, and demographic information, offering critical insights into how different states have voted over time.

Key sources of data include:

1. **USA.gov Election Results:** The dataset provides detailed election results dating back to 1982, broken down by state and candidate vote shares.
 - URL: <https://www.usa.gov/election-results>
2. **Federal Election Commission (FEC):** The FEC offers comprehensive data on election results and voter turnout, dating back to 1789.
 - URL: <https://www.fec.gov/introduction-campaign-finance/election-results-and-voting-in-formation/>

federaelections2020												
1	FEC ID	STATE	STATE ABBREVIATION	GENERAL ELECTION DATE	LAST NAME	TOTAL VOTES	PARTY	GENERAL RESULTS	GENERAL %	TOTAL VOTES #	COMBINED GE PARTY TOTALS (NY)	COMBINE
2												
3	P80001571	Alabama	AL	11/3/20	Trump		R	1,441,170	62.03%			
4	P80000722	Alabama	AL	11/3/20	Biden		D	849,624	36.57%			
5	P00013524	Alabama	AL	11/3/20	Jorgensen		IND	25,176	1.08%			
6	n/a	Alabama	AL	11/3/20	Scattered		W	7,312	0.31%			
7	n/a	Alabama	AL	11/3/20		Total State Votes:				2,323,282		
8	n/a			11/3/20								
9	P80001571	Alaska	AK	11/3/20	Trump		R	189,951	52.83%			
10	P80000722	Alaska	AK	11/3/20	Biden		D	153,778	42.77%			
11	P00013524	Alaska	AK	11/3/20	Jorgensen		LIB	8,897	2.47%			
12	P00003418	Alaska	AK	11/3/20	Janos		GRE	2,873	0.74%			
13	n/a	Alaska	AK	11/3/20	Scattered		W	1,961	0.55%			
14	P00014209	Alaska	AK	11/3/20	Blankenship		CON	1,127	0.31%			
15	P00016560	Alaska	AK	11/3/20	Pierce		PET	825	0.23%			
16	P80016342	Alaska	AK	11/3/20	De La Fuente		ALL	318	0.09%			
17	n/a	Alaska	AK	11/3/20		Total State Votes:				359,530		
18	n/a			11/3/20								
19	P80000722	Arizona	AZ	11/3/20	Biden		D	1,672,143	49.36%			
20	P80001571	Arizona	AZ	11/3/20	Trump		R	1,661,686	49.06%			
21	P00013524	Arizona	AZ	11/3/20	Jorgensen		LIB	51,465	1.52%			
22	P00012211	Arizona	AZ	11/3/20	Hawkins		W	1,557	0.05%			
23	P00015537	Arizona	AZ	11/3/20	Simmons		W	236	0.01%			
24	P80005572	Arizona	AZ	11/3/20	La Riva		W	190	0.01%			
25	n/a	Arizona	AZ	11/3/20	Cummings		W	36	0.00%			
26	P00017202	Arizona	AZ	11/3/20	Boddie		W	13	0.00%			
27	n/a	Arizona	AZ	11/3/20		Total State Votes:				3,387,326		

Above is a small sample of the data we use in our Python analysis. The data provides a detailed overview of the results from the 2020 federal election in the United States. It includes state-by-state results, highlighting the number of votes each candidate receives, their party affiliations, and the percentage of total votes earned in each state. Additionally, the total number of votes cast in each state is included to provide context for the individual candidate results. The data also notes the number of electoral votes awarded to each candidate from each state, though this aspect is not directly referenced in our outcomes.

Methodology

The methodology for this project involves merging 2020 election data into a cohesive dataset, cleaning the data by removing irrelevant variables, addressing missing values, and focusing on the margin of victory rather than the electoral college outcome to avoid state-specific biases. We will use Python libraries like Matplotlib and Pandas to create visualizations, such as heatmaps and bar charts, that highlight trends in voter preferences and identify key battleground states for future elections.

Step-by-Step Process of Data Cleaning

1. **Remove Non-Impactful Candidates:** Initially, we will filter out third-party candidates whose votes did not influence the overall election outcome, focusing solely on the Republican and Democrat parties.
2. **Eliminate Irrelevant Data:** We will remove columns and rows that do not pertain to our analysis, ensuring that the dataset is concise and relevant to the project's objectives.
3. **Index States by Abbreviation:** To maintain data accuracy and ensure consistency, states will be indexed by their official abbreviations, which will also allow for easier referencing when analyzing or visualizing the data.

```
# Identifying relevant columns for analysis
# From the data sample, the most relevant columns are state name, state abbreviation, total votes, and party affiliation.
relevant_columns = ['STATE', 'STATE ABBREVIATION', 'TOTAL VOTES', 'PARTY']

# Filtering out columns not in the relevant list and dropping rows with missing essential data
cleaned_data = data[relevant_columns].dropna(subset=relevant_columns)

# Converting TOTAL VOTES to numeric type for analysis
cleaned_data['TOTAL VOTES'] = pd.to_numeric(cleaned_data['TOTAL VOTES'], errors='coerce')

# Removing rows where TOTAL VOTES could not be converted
cleaned_data = cleaned_data.dropna(subset=['TOTAL VOTES'])

# Summarizing votes by state and party
state_party_votes = (
    cleaned_data.groupby(['STATE', 'PARTY'])['TOTAL VOTES']
    .sum()
    .reset_index()
    .pivot(index='STATE', columns='PARTY', values='TOTAL VOTES')
    .fillna(0) # Fill NaN with 0 for parties with no votes in a state
)

print("Cleaned State and Party Votes Data:")
print(state_party_votes)

# Check the columns that might contain vote totals or similar metrics
possible_vote_columns = ['TOTAL VOTES', 'GENERAL RESULTS', 'TOTAL VOTES #']
data[possible_vote_columns].info(), data[possible_vote_columns].head()
```

Above is a small example of the data cleaning we implement in our project. During the data cleaning process, we take several steps to ensure the dataset is properly aligned and ready for analysis. We use conversions to standardize the format of the data and maintain consistency across all variables. The drop function removes unessential rows and columns that do not contribute to the focus of the analysis. Pivot tables align states and political parties with their corresponding vote counts, creating a more structured and analyzable format. Finally, we apply the sum function to calculate summary statistics, providing an overview of the dataset's scope and enabling deeper insights into voting patterns. These processes ensure that the dataset is both comprehensive and efficiently organized for analysis.

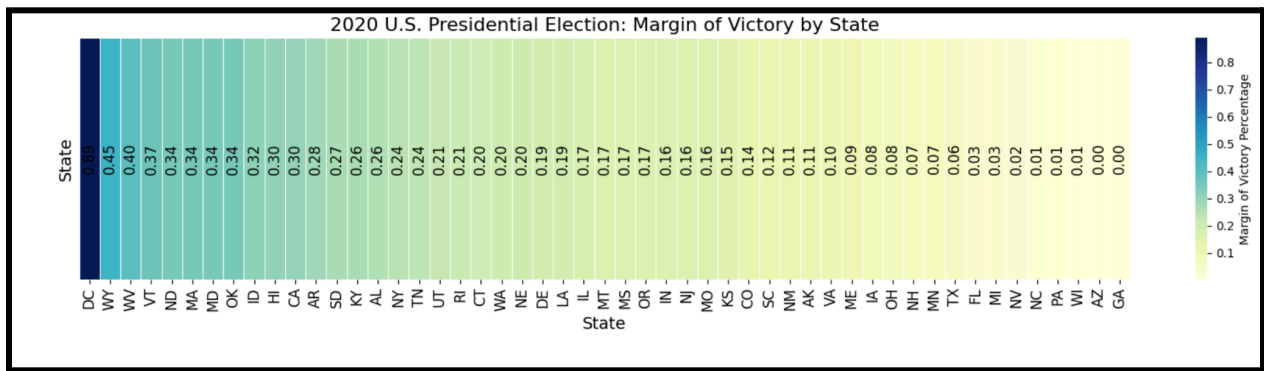
Heat Map Analysis

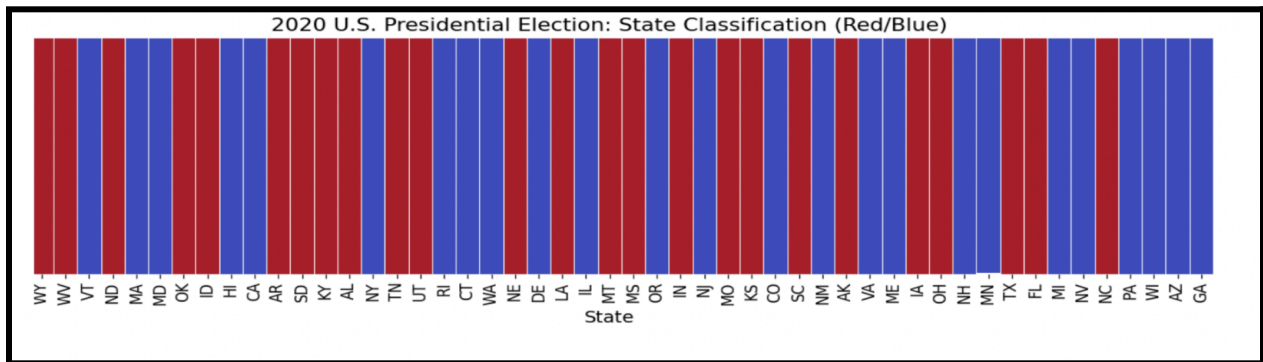
Once the data is cleaned and prepared, we use a heatmap to visualize the margins of victory across the United States in the 2020 election. This process begins by loading and cleaning the election data, standardizing vote totals, and addressing any missing values. We then analyze the state results, calculating the margin of victory for each state and classifying them as either "Red" (Republican) or "Blue" (Democrat) based on the winning party. The data is then prepared for visualization by creating a table that includes states, their margins of victory, and classifications, sorted by margin percentage.

The heatmap provides a clear depiction of state-by-state voting dynamics. On the right side of the graph, we see states with the closest margins between Democrats and Republicans, such as Georgia, Arizona, and Wisconsin. These swing states had highly competitive races where the outcome could have shifted with a small change in voter preferences. On the far left of the heatmap, we observe states with much larger margins of victory, indicating a more decisive win for one party. These states show a significant gap in voter support and were not as contested.

This heatmap is a crucial tool in identifying key regions with varying levels of voter competition. It offers valuable insights into where voter preferences were most tightly contested and highlights potential battlegrounds for future elections.

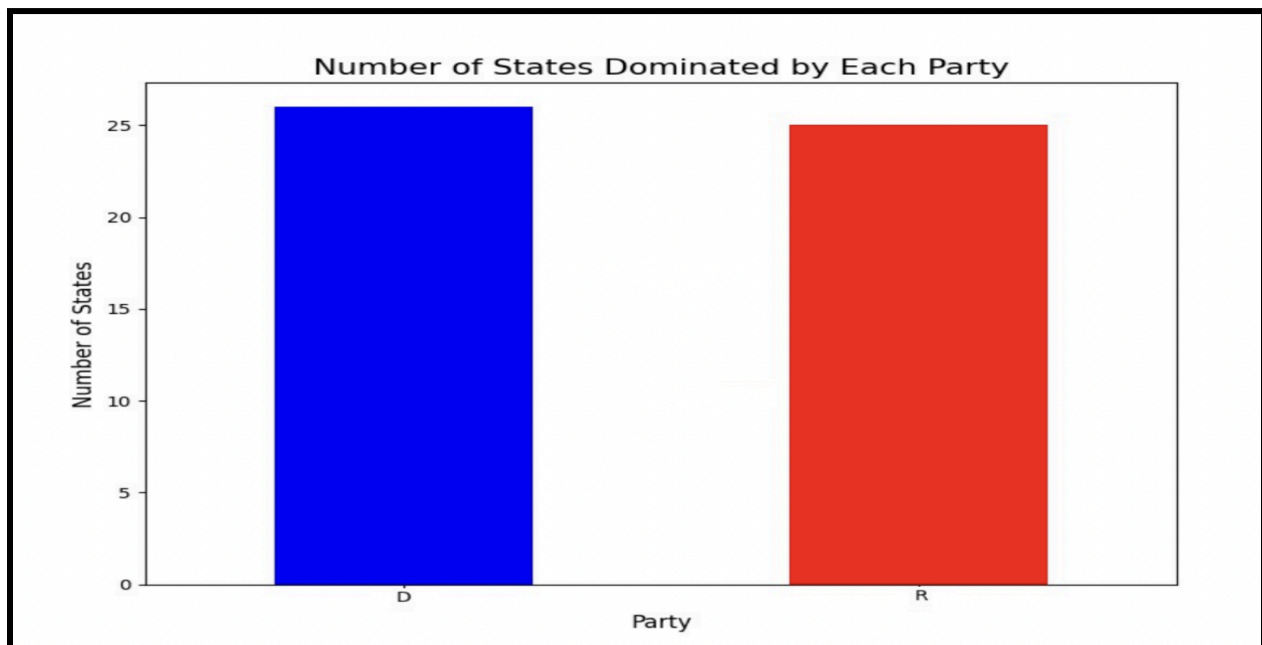
1





State Results Overview

In addition to the heatmap analysis, we also display the number of states won by each party in the 2020 election. This provides a clear and simple breakdown of which states were won by the Democrats and which were won by the Republicans. By categorizing states based on the winning party, we create a straightforward visualization that helps to identify the political landscape of the 2020 election. This overview not only clarifies the election outcomes at a glance but also sets the foundation for our analysis of the 2024 election. By understanding the distribution of victories in 2020, we can more effectively analyze and predict key battleground states for the upcoming election.



Results

Our analysis revealed several key findings, particularly with regard to the most visited states during the 2020 election campaign. Based on our model, the projected swing states were Florida (FL), Michigan (MI), Nevada (NV), North Carolina (NC), Pennsylvania (PA), Wisconsin (WI), Arizona (AZ), and Georgia (GA).

Interestingly, the states that candidates visited most frequently in 2020 were:

- **Pennsylvania:** 17 visits
- **Michigan:** 10 visits
- **Wisconsin:** 8 visits
- **Georgia:** 6 visits
- **North Carolina:** 6 visits
- **Arizona:** 4 visits
- **Nevada:** 4 visits
- **New York:** 2 visits (fundraisers)
- **California:** 2 visits (fundraisers)
- **Texas:** 1 visit (rally)

These findings align with the projected swing states in our analysis, validating the predictive power of our model. However, the frequent visits to New York, which was not classified as a battleground state in our model, suggest that other factors, such as fundraising and media attention can influence campaign strategies.

Do Our 2020 Findings Validate These Campaign Decisions?

Based on the data, the swing states we identified FL, MI, NV, NC, PA, WI, AZ, and GA were indeed the focus of extensive campaign visits. However, the inclusion of New York in the campaign trail, despite not being a projected swing state, suggests that factors like media coverage, donor activity, and personal political ties can also impact campaign decisions.

Conclusion

This project demonstrates the power of Python in analyzing historical election data to predict key battleground states in U.S. elections. By combining data from multiple sources and utilizing statistical and machine learning techniques, we can help campaigns make more informed decisions about where to focus their resources. As future election cycles approach, our model can serve as a valuable tool for identifying the most critical regions, ensuring that candidates can target the right voters and allocate resources effectively.

Figures

Figure 1: The heatmap visualized the 2020 election's margin of votes, highlighting swing states with close races (e.g., GA, AZ, WI) and states with decisive outcomes. It helped identify regions with varying levels of voter competition.

Figure 2: A map of voter decisions and outcomes based on state. Used to compare voter outcomes to margins of victory.

Figure 3: Figure 3: The bar graph shows the number of states won by each party in the 2020 election. It highlights the distribution of victories between the Republican and Democratic parties.

References:

Fox News. (2020, October 30). Trump, Harris have visited these states the most heading into Election Day. Fox News.

<https://www.foxnews.com/politics/trump-harris-have-visited-states-most-heading-election-day>

Federal Election Commission. (n.d.). Election results and voting information. U.S. Federal Election Commission.

<https://www.fec.gov/introduction-campaign-finance/election-results-and-voting-information/>

Python Tutor. (n.d.). Visualize Python code execution. Python Tutor.

<https://pythontutor.com/visualize.html#mode=edit>

Pink, J. (2020). Election analysis. GitHub. <https://github.com/JovaniPink/election-analysis>

Cay, G. (2020, October 22). Predicting presidential election by bootstrapping in Python. Medium.

<https://medium.com/@geencay/predicting-president-election-by-bootstrapping-in-python-a1bd8fa122cc>

CNN. (2020, November 3). 2020 presidential election results. CNN.

<https://www.cnn.com/election/2020/results/president>