Predicting a Citizen's Vote in the 2018 Election

Carl Geiger

Github:

Abstract

For various organizations, elections are crucial in pushing preferred policies forward. As such, knowing the outcome of district elections can greatly assist in planning for future congressional terms. The goal for this project was to accurately predict as many state districts as possible for the 2018 election using predictors such as previous election results and demographics. The problem was formulated as a classification problem and solved using multiple different classification methods. The results found that while previous election results can predict the next election with very high accuracy, the precision does not increase when introducing demographics. For swing districts, using demographics rather than election results increases accuracy, but only for specific demographics.

Introduction

Predicting elections has been an important topic ever since the founding of democracy. For American interest groups and businesses, having an accurate estimation of the next election can be crucial in planning for the next fiscal year. While most election predictions utilize citizen polls to predict an election, polls often generate poor predictions if the sample size does not properly reflect the American population. Rather than spend massive amounts of time sending out polls, demographics provide a possible alternative to accurately predict an election. Since gathering demographics is already accomplished every census, the level of work is notably less than administering polls.

The goal of this project was to determine whether demographics truly are an effective way of predicting district elections. Since most districts vote for the same party in consecutive elections and as such are very easy to predict, the project must also specifically test whether demographics accurately predict districts that swing from one party to the other, as these districts are the most difficult for statisticians to predict. To accomplish this, I utilized the 2010 census to gather each district's total population, media age, male percentage and white percentage (which represents the diversity of a district). Next, I gathered every district's House election results for the 2012, 2014, 2016 and 2018 elections. Finally, I used various classification methods to predict the 2018 district elections, with district demographics and previous election results as the attributes and the 2018 election results as the target.

The 2010 Census data was downloaded from the census website, where each state had its own csv file filled with demographics for each district. I then added the demographics I needed for each district into a new csv to make data analysis more straightforward. I next entered the election results for each district into this custom csv by hand using the *New York Times* as a source. When analyzing the data, I separated each district based on whether that district voted for the same party in each House election. This was done to specifically predict swing districts using more consistent districts as the training data. To demonstrate whether demographics actually improve the prediction model, I used a base model where the only attributes are previous election results. To find the optimal attributes for the model, I used multiple combinations of attributes to find the ones that best predicted the 2018 election. My findings found that while demographics do not improve the overall prediction accuracy of district elections, they are substantially better for predicting swing districts. Specifically, using only demographics to

predict the outcome of a swing district's election is better than using previous election results. When using a decision tree model, using demographics resulted in a maximum 69% accuracy.

Data

To obtain each district's demographics I visited the 2010 Census' website[1], which had a csv file to download for each state. This csv file housed demographics for every district of the given state, with columns designating each district and the rows being the specific demographic characteristic. I decided to use total population, median age, the percentage of citizens in the population that are male, and the percentage of citizens in the population that are white. To make the data more convenient for data analysis, I created a new csv file to store all the relevant data. This required me to copy the relevant data from each state csv and enter it into the new csv in a more usable format. The original state csvs also only had male population and white population, so I also needed to calculate the male and white percentages by using Excel's function feature.

For the election results I visited the *New York Times* website. The *Times* still has pages listing the results of the 2012[2], 2014[3], 2016[4], and 2018[5] House elections. However, it does not have this information stored in a csv file. Thus, I had to manually enter the election results into my custom csv by entering whether a district voted Republican or Democrat for every district. The final CSV file was formatted as shown in Figure 1 below.

Columns: [State, District Number, Total Population, Median Age, Male Population, Female Population, Male Percentage, White Percentage, 2012 Outcome, 2014 Outcome, 2016 Outcome, 2018 Outcome]

Figure 1: *List of columns in dataset csv.*

I utilized pandas to read in the dataset csv and construct a DataFrame from it. I next converted the election results from "Republican" and "Democrat" strings into 1s and 0s, where a 0 represented a Republican victory and a 1 represented a Democratic victory. Finally, when analyzing districts with unequal election results, I separated the dataset into two DataFrames: "swing districts" and "stable districts". Districts that voted for the same party in every election were classified as "stable districts", while those that did not fit this condition were classified as "swing districts". Though the combination varied between experiments, the predictor attributes used for data analysis were total population, median age, male percentage, white percentage, and the previous election results. For every experiment, I set the target attribute as the results for the 2018 election.

Methodology

The general procedure I followed for data analysis is shown in Figure 2. To best analyze how each classifier performs given the predictor attributes, I used Decision Tree, Logistic Regression, and SVM and graphed the accuracies for all three models to compare the three options. This is first performed on the baseline, and then using all the attributes I picked for the experiment. The data analysis itself is split into two halves: the overall accuracy of the models and the accuracy of the model for swing districts. For the overall accuracy, the dataset is split

into 70% training and 30% testing. The first experiment for overall accuracy used both previous election results as well as total population, median age, male percentage, and white population as the attributes. Next, I did one experiment for each demographic characteristic where only that one characteristic and the previous election results are the attributes to pinpoint which attributes best improve accuracy, if any. When analyzing swing district accuracy, I used stable districts as the training set, and swing districts as the test set. The first experiment for overall accuracy used both previous election results as well as total population, median age, male percentage, and white population as the attributes. I next tested whether this accuracy increased if only demographics were used as predictor attributes. I next did one experiment for each demographic characteristic where only that one characteristic and the previous election results are the attributes to pinpoint which attributes best improve accuracy, if any. I next repeat this process for swing districts, but also analyze whether accuracy improves when election results are not included in the list of attributes. I finally pick the model and data attribute with the best accuracy and use it as the final accuracy result.

My sole python code, ElectionDataAnalysis.ipynb handles loading the data into DataFrames, selecting the proper attributes and finally outputting and plotting the results of each experiment. When designing decision trees, the code also saves the decision tree graph as a pdf.

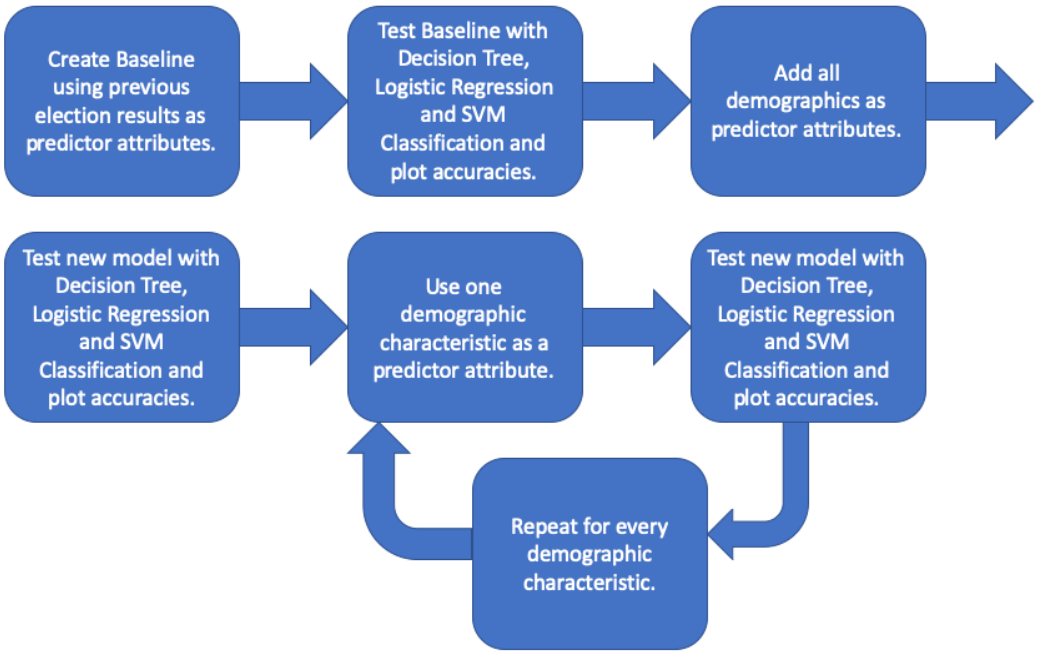

***Figure 2**: Flowchart for Election Prediction*

Experimental Evaluation

Setup

The code was run on macOS 10.14 on a mid-2012 MacBook Pro. Since it runs on Jupyter at low cost, other computers only need Jupyter or similar software to run the code. To determine the success of my models, I compared them to a baseline model that solely uses previous election data as the predictor attributes. The final accuracy results were generated using sklearn metrics' accuracy function.

Results

Experiment	Accuracy using Decision Tree Classifier
Overall Baseline accuracy (2012-2016 election outcomes)	0.8854961832061069
Overall Demographic Accuracy (total pop, median age, male pct, white pct, 2012-2016 election outcomes)	0.8320610687022901
Swing District Baseline accuracy (2012-2016 election outcomes)	0.38461538461538464
Swing District Demographic Accuracy (total pop, median age, male pct, white pct, 2012-2016 election outcomes)	0.4230769230769231
Swing District Demographic Accuracy (total pop, median age, male pct, white pct)	0.6153846153846154
Swing District Demographic Accuracy (Median Age)	0.6923076923076923

Figure 3: *Accuracies of different predictors*

When looking at overall prediction accuracy, using only previous election results does result in a higher accuracy than also using demographics as predictors. Even when using different classifiers such as logistic regression, demographics will at most have the same accuracy as only having previous election results. However, this is mainly due to most districts voting for the same party in every House election. When analyzing districts that do not vote consistently in every election ("swing districts"), I find that using previous election results actually worsens the prediction accuracy rather than just using demographics. Even more interesting is which demographic generates the highest accuracy. To achieve maximum accuracy when predicting swing districts, the best predictor is median age. However, when using cross validation the hyperparameter does not produce the highest accuracy, possibly due to the limitations of only using one attribute. When analyzing swing districts in particular, the logistic regression's accuracy increases when demographic attributes other than total population are used while at the same time the decision tree's accuracy drops. Also interesting is how all of the tested models (decision tree, logistic regression and support vector machine) suffered accuracy drops when solely using male percentage and white percentage. At least for these four

elections, male population and white population did not improve predictions in swing district results. More results can be found in the iPython notebook, including more information on how different demographics affected accuracy.

Conclusions

Overall, while adding demographics to the prediction model did lower the accuracy when analyzing all districts, it dramatically increased accuracy when it was the sole predictor for swing districts. However, the final accuracy for swing districts rests at 0.69, which is still far too low to confidently rely on. As such, while demographics can slightly assist in predicting an election outcome, predicting how swing districts will vote remains a tough challenge. Further work should expand the year range to also include the 2000 census and elections that occurred during that era. With this addition the model could then use changes in demographics from one census to the next as a predictor for an election. Overall, demographics provide an interesting approach to analyzing how districts vote.

References

[1] Center for New Media & Promotion, and US Census Bureau. "My Congressional District." Census.gov. January 25, 2017. Accessed April 26, 2019. https://www.census.gov/mycd/.

[2] "House - Live Election Results." The New York Times. Accessed April 26, 2019. https://www.nytimes.com/elections/2012/results/house.html.

[3] "House Election Results." The New York Times. Accessed April 26, 2019. https://www.nytimes.com/elections/2014/results/house?utm_source=top_nav&utm_medium=web&utm_campaign=election-2014.

[4] "House Election Results: G.O.P. Keeps Control." The New York Times. Accessed April 26, 2019. https://www.nytimes.com/elections/2016/results/house.

[5] "U.S. House Election Results 2018." The New York Times. November 06, 2018. Accessed April 26, 2019. https://www.nytimes.com/interactive/2018/11/06/us/elections/results-house-elections.html.