Building a Candidate-Based Model of Reelection

Cameron MacDonald

December 16, 2021

Introduction

In 2020, the Democratic incumbent of Florida's 26th congressional district lost her campaign for reelection. The loss was unanticipated; 538's election model estimated that Debbie Mucarsel-Powell had 82 in 100 odds of beating her Republican challenger. In the losing effort, the campaign's committee spent more than $6.5 million. Groups inside the Democratic campaign ecosystem helped finance this spending, Emily's List donated more than $100,000. In the end, the race was not particularly close. Mucarsel-Powell lost by 3.4% of the vote. Recently, election watchers have been concerned about the validity of polling based election models. Currently, a majority of the election models we interact with are derived from public polling of likely voters. There are a number of problems with these kinds of polls and, thus, these kinds of models. First, there is the uptick in selection bias effects when relying on traditional survey methods. In 2019, the issue affecting polling sampling was the inability of pollsters to reach voters over the phone. The pandemic changed things. Covid-conscious individuals, who were at home to take phone calls, were oversampled in the polls, meaning an undersampling of likely Republican voters. Traditional polling also comes with prohibitive costs.

If polls are too expensive, they won't be conducted with the frequency required for use in modeling, especially for smaller races. The final issues, with the models themselves, is also a feature. They take in information up to election day, but because of a lack of polling, are not very good at predicting elections far in advance.

In this project we aim to address these concerns of polling based models by proposing a candidate based model for election outcomes. Specifically, reelection outcomes for House of Representatives members. We aim to provide a prediction based on data that is accessible upon the swearing in of a member to the House. The two-years post prediction is enough to allow for a reassessment of resource allocation by groups in the electoral ecosystem because American electoral strategy is often a cyclical process (i.e. is determined every election cycle).

In the paper, after establishing the scope of the problem, we will consider the process of data collection and the justification of the data included in the model. With this in mind, we will move to the description of the methods of analysis used in the creation and validation of the model. After the model is established, we will summarize the results. Finally, we will discuss the implications of our results and the 'success' of the project based on the definition established in its proposal.

Problem Statement and Background

The candidate-based model aims to predict whether a given member of the house of representatives will be reelected with only data that exists at the time they are initially sworn into office. Of course, it is incredibly difficult to predict an event that will happen two years in the future, especially when considering that a great majority of voters will make their decisions based on information they receive over the course of that period of time. While this information will be unavailable to us, there is certainly information that

is available to us that influences election outcomes. The question at hand is whether that information will be enough for us to make a meaningful prediction. If the model's predictive capacity is not much better than a coin toss, it will be impossible for political decision makers to rely on its results for decision-making, which is the desired outcome.

Fortunately, there are general effects that are well understood to influence election outcomes. These are macro effects that do not rely on the political day-to-day, whereas political polling relies on the 'movement' of voters with political winds. The most famous advantage for reelection is incumbency. Incumbency, or the condition of being the current office holder, is well known to be advantageous in elections. Despite this, the academic literature makes clear that the actual advantage offered by incumbency has fluctuated historically (Carson et al., 2019). Another effect of interest is the midterm loss cycle, "[w]ith only three exceptions, the president's party has lost seats in the House of Representatives in every midterm election since 1870, and on average the president's party loses more than 9% of its House seats." (Cohen, 2020). From the comparative world, we also know that politicians may face higher or lower standards in their public life based on race and gender (Eggers et al., 2018). All of these effects will be considered in the following section when we consider the inclusion of variables and the unit of analysis.

Again, the question underlying this investigation is the relative explanatory power of these variables. Can the incumbency effect, demographic effects, and structural effects explain enough to yield true predictions? It might be that current news and economic conditions during the service of a representative explain too much of the variance for a model to be successful in prediction without them. For this reason, changing the 'cut-off' for existing data to 1 year into the cycle was considered. Because the aim of the analysis is to influence political resource allocation, making this alteration is difficult. By 2019, campaigning for the 2020 election is well-underway (this has not been the case historically) (O'donnell, 2016). Of course, if the current analysis does not meet our expectations, we may consider this alternative formulation.

Data

In an effort to utilize publicly aggregated data, I attemted to scrape all the data used in the analysis from Wikipedia. A complete list of the wikipedia pages which were scraped can be found on the project's github repository in the text file titled "Sources_URLs." While there is some concern about the accuracy of Wikipedia data, articles on American politics tend to be well-moderated. Moreover, a majority of the values are sourced from official government websites and documents. One notable exception is the Presidential Approval Rating. Wikipedia only collects average, high, and low approval ratings for each President. Because the variable aims to describe presidential approval rating for specific points in time, it was necessary to find another data source. The American Presidency Project collects and compiles the Gallup Presidential Poll for each president, going back to Franklin Roosevelt, on a biweekly or monthly basis.

There were many steps in the process of collecting and wrangling the data. Much of the time, the same data point could be found within multiple Wikipedia articles. The first step was identifying the articles with the most desirable format. In the process of adapting the Wikipedia tables to dataframes, it often became necessary to identify new articles to source information from. Once an article, or series of articles, was selected for data collection. It became necessary to build a web-scraping tool to collect the data from the page. If all the articles were found in one page, using the pandas method 'read_html' was preferable to methods from the BeautifulSoup library. In the case that data had to be collected from multiple articles, such as the vote share of the representative, it was necessary to build a web scraper that collected data from each page through iteration and BeautifulSoup methods. After the data was collected, wrangling was the major challenge. Broadly, the project required two types of wrangling: restructuring and cleaning. In many cases, the tables on Wikipedia articles were poorly structured. This took many forms, sometimes tables were not organized into standard columns and rows, other times the unit of analysis was incorrect. The largest challenge was the restructuring of the complete list of United States Representatives. The list included only one entry per congressperson, unless they changed districts. Since we aimed to predict whether a representative would be reelected, it was necessary to restructure the data so that there was one observation for each congress (e.g. the 117th Congress) the representative served in. Multiple methods were attempted to complete this task. Initially, the Pandas 'iterrows' method was used to loop through each observation

in the data frame, the time complexity of this task meant that it was an infeasible approach for this task. Instead, dummy variables allowed for the creation and subsequent concatenation of filtered data frames on a much shorter time scale.
As described in the paper's background and problem statement sections, the aim was to develop a model which considered individual level characteristics of congresspeople. Finding data at this unit of observation is difficult. Initially, the plan was to use wikipedia 'v-card infobox' to secure a majority of the relevant data including age, previous electoral experience, and education. In early exploratory analysis, it was relatively simple to manipulate this object into a pandas data frame and to 'find' tagged html values. When running the program on the entirety of dataset issues arose. Since wikipedia is open for anyone to edit, there are drastic inconsistencies in formatting and html tagging. After spending a considerable amount of time attempting to pull the information from the infoboxes, it became clear that making adjustments at the level of individual articles was simply infeasible. There are not many sources for information that collect these demographic factors of congresspeople and the amount of time spent on the initial attempt was prohibitive to looking for other sources. Late in the project's progress, I became aware of the'allCongressData' dataset provided by the Harvard dataverse. The data source is by far the most complete set of observations, containing district information, and demographic information on congressional representatives. A left join was performed on some key variables from this dataframe and the existing dataframe based on the wikipedia scraping. This allowed for the completion of the project as described in the problem statement. Upon cleaning the dataframe of variables which did not meet the 1 year cutoff, mostly to demographic factors of the candidates and their districts, a multi-index join allowed the merging of the scraped and pre-constructed data.
The use of the Harvard data meant that observations only went back as far as the 93rd congress. The original data frame went back as far as the 87th. After eliminating missingness, which was not typically an issue, we were left with around 5700 observations.
Below is a table of the final variables and their descriptions:

Analysis

The analysis in this paper relies on machine learning methods, specifically a variety of tools from Scikit-learn. There are three core steps in the pipeline which apply predictive analysis to the data: 1) preparing the data for analysis; 2) establishing the search parameters; 3) training and testing the data. It is important to note that prior to working with this data, it was split into a training and testing data set, 25% of the data went untouched until the final phase of the last step. This measure is put in place to validate outcomes from machine learning methods. Everything in the first two steps are happening within the training data.

##Preparing the data

First, we visualized the distributions of our independent variables. Because of notable skewness, mostly in the percent population statistics, a log function was applied to ensure normality. Following this, dummies were made from the State, District, Gender, and Party Control variables, making sure to drop a single category to avoid perfect multicollinearity. This process allows for the actual modeling to run smoothly.

##Setting the Search

In setting the search, we establish the k-fold cross validation, whereby we allow the models to split the data into 10 sections as they apply different statistical methods This improves the models and allows them to run small tests on withheld sections of the available data. In addition to this numerical variables are scaled, the data 'ingested' at this step also serves to tune the model.

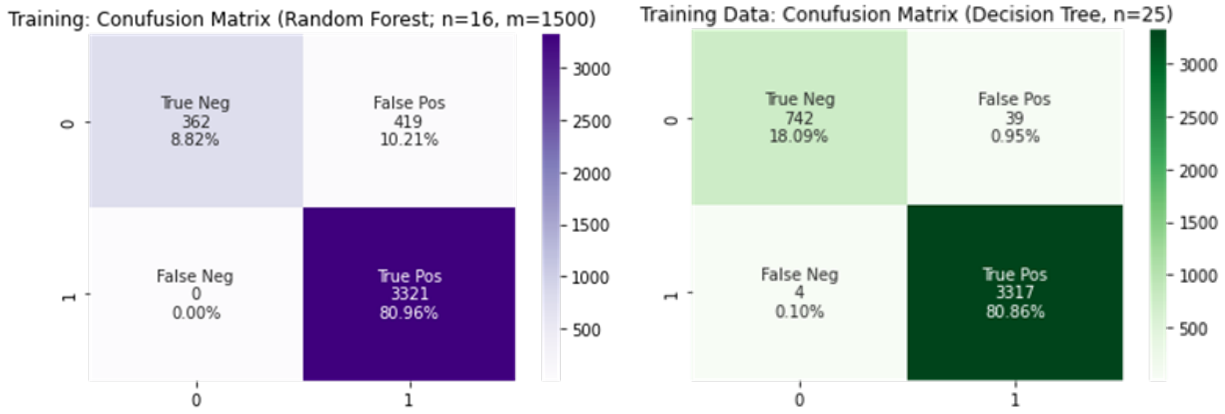
We also establish the models which will be considered. The Naive Bayes Classifier is the first, simplest, model. It assigns probabilities for the outcome of being reelected using the bayes theorem and its underlying assumptions (of which independence of the variables is central). The K-Nearest Neighbors Classifier identifies the closest values to a point of interest (based on all of the independent variables) and assigns a value based on their outcomes. For this model, we consider K specifications between 5 and 150. The

Decision Tree classifier creates split roads for observations to travel down and assigns outcomes at the end of these decisions (for a depth of 1, there is one split. We considered trees between depths of 2 and 50. Finally, we considered the Random Forest Classifier which builds groups of decision trees based on limited samples of the available data and averages the decisions of each tree into a unified model. We consider the max depth of each tree to be between 2 and 20 depth, with forests of between 500 and 1500 trees. In reality, the Random Forest fits the data the best. We have a variety of binary, discrete, and continuous variables that lend themselves to the 'crossroads' approach. The discrete and binary variables will specifically challenge the capacity of k-nearest neighbors to accurately consider distance.
The above methods are then collated into a single 'function' which we can apply to the whole of the training data.

##Training and Testing

This step includes actually 'running' the pipeline we built in the previous step on the training data we prepared in the first step. Here, the computer attempts the model based on our specifications. We are then allowed to consider some of the relative variables of adjustment before making alterations to the tuning parameters. In the case of this project, some tuning parameters were removed after demonstrating that they 'overfit' the data. This was determined by the consideration of a confusion matrix that allows us to understand when the model is making inaccurate predictions. The same principle applies to the underlying scoring of which model is best, we use the "balanced accuracy" measure. According to the documentation, the score returns an average of the accuracy within each potential outcome. In our case, is the model missing more successful outcomes or vice versa. Below is an image of the confusion matrix for a model in one of the search outcomes.

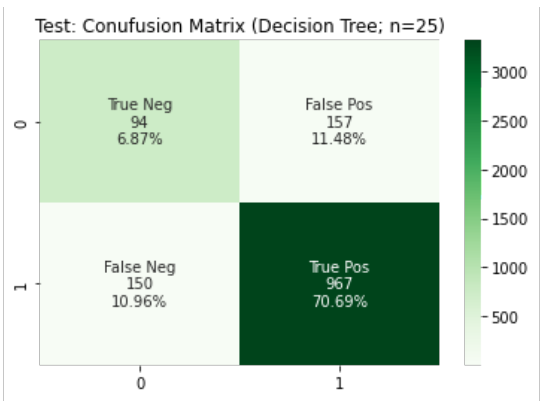
After this process, we selected a final model, a Random Decision Tree with X depth and Y trees. Upon selection of this model, we can apply it to our testing data. Again, this allows the validation of the model on data it has never seen before (i.e. without prior information). This validation tells us how the model might fare against future data. The results from the final test are reported in the following section.

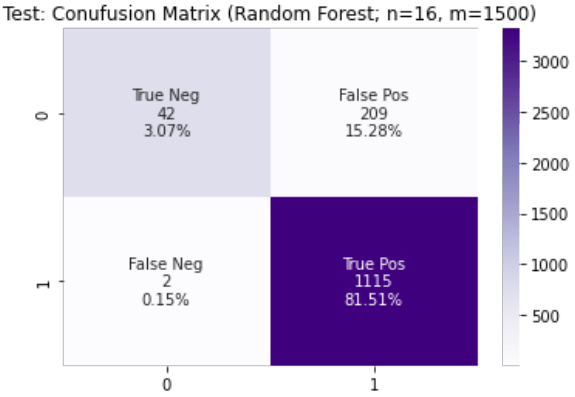


Results

Most importantly is the success of our model on new data. Below is the confusion matrix that results from the testing data, we considered the models which performed best on the Balanced and ROC scores respectively. The decision tree wrongly categorizes 157 of the negative values as positives, while the random forest classifies 209 negative values incorrectly. We see that 12-15% is a relatively high false positivity rate. Because the goal of the model is the allocation of resources, it would be concerning to give funding to dozens of House members who could go on to lose their reelection. The random forest is able to do a much

better job of predicting positive values, hitting 81.5%. These combined results indicate that the models are successful because they are overpredicting successful reelections (which make up the bulk of the data). The Decision Tree, which was measured with the balanced accuracy metric, does a better job of avoiding this issue.





Also of note is the relative importance of each independent variable. This gives us insight into the importance of certain variables within the general prediction problem. While these relationships do not imply causality, it is a good picture in showing how a model works and makes a decision tree. Below is a plot of the Election Year Presidential Approval variable measured against other variables in a partial dependence plot. The graph visualizes the change in predicted value as we move within the value field of both variables. We can see that if there is not high presidential approval and there are relatively fewer households, reelections are less frequent.
[Figure 5](Figure5.png){width=100%}

Discussion

It is unfortunate for me to say that I feel that I did not meet my own standards for success. While I was particularly careful about the structuring of files (whereby individual notebooks were used to complete individual taks and folders were constructed to organize project components). I did not use version control as much as I would have liked. In large part, I think it speaks to bad tendencies in my actual programming process. I do think I did successful work with the data visualization. As far as process for data wrangling, I consider myself to have learned a lot about web scraping.

Word Count: 3000

Works Cited

A 501tax-exempt, OpenSecrets, et al. "House Majority PAC PAC Contributions to Federal Candidates." OpenSecrets, https://www.opensecrets.org/political-action-committees-pacs/house-majority-pac/C00495028/candidate-recipients/2020. Accessed 17 Dec. 2021.

Are Members of Congress Becoming Telemarketers? https://www.cbsnews.com/news/60-minutes-are-members-of-congress-becoming-telemarketers/. Accessed 17 Dec. 2021.

Carson, Jamie L., et al. "Nationalization and the Incumbency Advantage." Political Research Quarterly, vol. 73, no. 1, Mar. 2020, pp. 156–68. DOI.org (Crossref), https://doi.org/10.1177/1065912919883696.

Cohen, Cameron Chase. Electoral Composition and the Midterm Loss Cycle. June 2020. dash.harvard.edu, https://dash.harvard.edu/handle/1/37364762.

Cohn, Nate. “No One Picks Up the Phone, but Which Online Polls Are the Answer?” The New York Times, 2 July 2019. NYTimes.com, https://www.nytimes.com/2019/07/02/upshot/online-polls-analyzing-reliability.html.

Eggers, Andrew C., et al. “Corruption, Accountability, and Gender: Do Female Politicians Face Higher Standards in Public Life?” The Journal of Politics, vol. 80, no. 1, Jan. 2018, pp. 321–26. DOI.org (Crossref), https://doi.org/10.1086/694649.

Foster-Molina, Ella. Historical Congressional Legislation and District Demographics 1972-2014. Harvard Dataverse, 2017. DOI.org (Datacite), https://doi.org/10.7910/DVN/CI2EPI.

“List of Members of the United States House of Representatives Who Served a Single Term.” Wikipedia, 20 Nov. 2021. Wikipedia, https://en.wikipedia.org/w/index.php?title=List_of_members_of_the_United_States_House_of_Representatives_who_served_a_single_term&oldid=1056256206.

Russonello, Giovanni, and Sarah Lyall. “Surprising Poll Results: People Are Now Happy to Pick Up the Phone.” The New York Times, 17 Apr. 2020. NYTimes.com, https://www.nytimes.com/2020/04/17/us/politics/polling-coronavirus.html.

Silver, Nate. “2020 House Forecast.” FiveThirtyEight, 12 Aug. 2020, https://projects.fivethirtyeight.com/2020-election-forecast/house/.