Using Neural Networks to Predict the 2018 Midterm Election

Sean Swayze Follow
Oct 10, 2018 · 6 min read

By Sean Swayze

October 10th, 2018

ABSTRACT

Using python to scrape historical data from online sources and a neural network to analyze this data, a model was created to predict the outcome of the 2018 midterm elections for the House of Representatives on a district by district basis. The information used for this model primarily consists of census data from each district as well as the historical results of elections and publicly available finance data. Two models were created, both predicting a Democratic victory, but with different margins depending on whether the results from the previous congressional elections were included. When this data was included, the Democrats held a 17 seat advantage and when it was not, a 3 seat advantage.

NEURAL NETWORK

The neural network was constructed as a feed forward neural network that used 14 sets of input data:

Table 1 — Sources of Data used for the Prediction:

DATASET USED	SOURCE FOR DATA
PRESIDENTIAL APPROVAL (2002 – 2008)	https://news.gallup.com/poll/116500/presidential-approval-ratings-george-bush.aspx
PRESIDENTIAL APPROVAL (2010 – 2016)	https://news.gallup.com/poll/116479/barack-obama-presidential-job-approval.aspx
PRESIDENTIAL APPROVAL (2018)	https://news.gallup.com/poll/203198/presidential-approval-ratings-donald-trump.aspx

	approval-ratings-donald-trump.aspx
DISTRICT DEMOGRAPHIC: **POPULATION** **MEDIAN AGE** **NUMBER OF RESIDENTS OVER 18** **NUMBER OF RESIDENTS OVER 65** **NUMBER OF WHITE RESIDENTS** **MEDIAN INCOME** **GRADUATION RATE** **PERCENT OF RESIDENTS WITH A BACHELOR'S DEGREE OR HIGHER**	https://factfinder.census.gov/faces/nav/jsf/pages/index.xhtml
FUNDRAISING **ELECTION RESULTS (2002 – 2014)** **INCUMBENCY**	https://www.opensecrets.org/races
UNEMPLOYMENT BY STATE	https://www.bls.gov/lau/
GENERIC CONGRESSIONAL BALLOT	https://realclearpolitics.com
ELECTION RESULTS 2016	https://www.fec.gov/introduction-campaign-finance/election-and-voting-information/#election-results

The sets of data in Table 1 were chosen as a representative cross section of the various factors influential in the outcomes of elections. The data from the census bureau was retrieved manually. The data on campaign finance, election results from the years 2002 to 2014, and the data on incumbency were all retrieved via a python web scraping program using the *Beautiful Soup* package in September 2018. This data was accumulated and stored in an Excel spreadsheet (see Appendix).

The end state of the template neural network had 14 input nodes, two sets hidden layers, each with five neurons, and a single output node that gave the projected Republican and Democratic share of the vote.

Figure 1 –Neural Network Design:

The neural network evolved over the course of the project by minimizing the loss function when the network was asked to predict the results of the 2016 elections using only the 2000–2014 data to train. In order to minimize the loss function for the neural network, it was instructed to stop after the loss function began to increase due to over-fitting. After much trial and error, using the 2016 and 2014 election results as trial sets, it was found that including the 2000 elections decreased the accuracy of the model. Therefore, only the data from 2002 to 2016 was used to predict the results for 2018.

A second model was also created which included the election results from the previous congressional elections. While it did not significantly alter the loss on the validation set, it did result in a wider margin of victory for Democrats. As testing revealed no meaningful difference in the loss over the training set, results for both are included below, with the original model labeled “Model A” and the updated model, which includes the results of the previous election labeled “Model B.”

One concept that has been used in this model is for the neural network to be reinitialized and run many times in order to create statistics for each individual district. For the final prediction, the network was reinitialized and run 10,000 times. Because the weights are initialized randomly for each run of the network, the results are different for each generated neural network. In this fashion, a sample mean and standard deviation were found for each district.

While this process was successful for finding unbiased sample means, the variance caused by the weight initialization was lower than the real-life variances. A possible explanation for the low predicted standard deviation could be insufficient randomness

in the model from the original generation of weights to account for the real-life variability in election results. In order to account for this, the standard deviation of each district, as calculated by the model, was multiplied by a factor of 2 to bring it in line with the observed variability of districts.

Table 2 — Results of both Models:

**These numbers are derived from the probability distribution of the results, which are calculated with the assumption that races are independent events. Due to elections being correlated, the probablity density distribution is wider than predicted and these numbers are likely much closer to 50%.*

Before interpreting the results, it should be noted that the variability of the distribution of seats at a national level as predicted by the model is potentially flawed. The model calculates the results for each district independently and then treats each district as an independent variable when calculating the distribution. In reality, there are multiple variables outside of the scope of the model that are difficult to incorporate into a neural network, but that significantly widen the probability distribution of seats. To correct this inaccurate assumption, one would need to incorporate a factor which correlates similar districts. The nature of this effect is impossible to determine using a neural network without adding possible bias to the process. While the amount of spread in the model in therefore flawed, the mean result is not affected by this flaw, and significant predictions can still be drawn.

Further analysis of this data, especially the reason the model predicted any given result at the district level, is made difficult by the nature of the model. The primary obscuring factor is that neural networks are simply a vast array of weights and biases, so trying to understand what each iteration of the neural network does with a set of data is not feasible. However, the far greater factor that obscures the process is that the model

consists of not one, but thousands of smaller models who have their results averaged for each district. Thus, to properly analyze the results for any given district, one would need to dissect not one, but thousands of neural networks. However, this does not mean general abstractions about the results of the model can not be made. The two models give a different prediction for the outcomes of the House come November. While both predict a Democratic lead over Republicans, the magnitude of that advantage is dissimilar. Model A predicts a small Democratic lead while Model B predicts a larger one. Given that Model B considers the results of past elections, this could demonstrate that Model B has analyzed the pattern incumbent parties doing worse during the midterms. This discrepancy could also indicate that other non-demographic factors are conducive to a larger Democratic lead.

CONCLUSION

My hypothesis is that Model B is more accurate because it includes more relevant data. The additional data in Model B is the results of the last election in the district which could help the model predict the partisanship of a district beyond what demographics can describe. This should lead to more accurate results, especially in districts where the relationship between demographics and political leanings is asymmetrical to the national average.

Table 3 — Top Ten most competitive districts according to Model B:

There are advantages and disadvantages to using the neural network method described above. One of the greatest strengths of using a neural network is that it shields researchers it from a certain amount of the bias usually inherent in creating any sort of prediction. Once given a set of data, the neural network blindly optimizes the fit to the historical data, and entirely removes human bias from the process. However, this does not mean the process is without potential for bias. Bias can still be introduced by the selection of which datasets are used to train the neural network. The primary weakness of this model is also one of its greatest strengths. The opaque nature of the model eliminates much of the bias usually present in making predictions, but also makes it very difficult to analyze in depth. The results of the upcoming midterm House election will be interesting to compare to the model predictions.

APPENDIX

Spreadsheet with Data:
https://docs.google.com/spreadsheets/d/1oqODh1eXi80oltRlFolRgdx5nkdhsf1fFGONOLHpFPM/edit?usp=sharing

I would like to acknowledge the Schilling School for Gifted Children for providing me the time and space for this research and Dr. Frank, who served as my academic advisor during the course of this project.

Machine Learning Elections Neural Networks Midterms Predictions