Report

Overview

The goal of this project was to assign probabilities to the upcoming Georgia Senate Election Runnoff between David Perdue and Jon Ossof. I approached this project with a combination of prior political science understadning as well as an impentatsion of simulation techniques. The end result of this project is a graphic that shows the user the projected probabilities for each candidate. Additionally this simulation can be updated as new polls come in.

Implenatation

Every file within this repo is self contained and will run independently. Within the repo is a package called `georgia` when holds a few of the key functions used in the project. The three functions it contains are `probability_winning_plot`, `probability_dataset`, and `initial_poll_cleaning`. The goal of these three functions is to take dataset from `fivethirtyeight.com` and allow them to be easily implemented in the probablity forcast model. `initial_poll_cleaning` takes in a downlaoded `.cvs` file that is saved as a variable and does the data manulation necessary to proceed with the forcasting. The function `probability_dataset` works on this by doing some of the more complicated data manulation, involving logial statement, and make it ready for the last function, `probability_winning_plot`. `probability_winning_plot` then plots the probabilities in a way that allows for dinamitc poll values.

Bootstrapping and Mapping

In addtion to the functions decribed above, there are the actual siminaltoin techniques used to come up the probabilities. At the time of writing there are 114 polls for this particular Senate race. Althogh this is actually a decent amount of polling compared to most senate races, in my opinion it was not enough to make an accurate prediction. My first attempt to model the uncertianty in the election can be seen in my file `01_Johnson_Disribution.Rmd`. Although this did not pass the mathimiatical muster required of an election simulation it was an important part of the process towards finding a good way to go about the project. It lead to the implemenation of bootstraping, which can found in `02_Bootstrap.Rmd`. After cleaning the polling dataset, resulting in the named dataset `boot_data`, I preformed the following bootstrap:

```
map(1:10000, ~sample(boot_data$actual_spread, size = length(boot_data), replace = TRUE))
%>% map_dbl(mean)
```

This line of code takes samples from the `boot_data` with replacement and then maps over them ten thousand times and lastly extracting the mean. This allow met to achieve normailty and make some preductions.

Assigning Probability Values