Election Prediction

Bob, Becca, Rena, Grace

10/16/2020

1. Introduction

The outcome of the 2016 election not only stunned the nation, but also sent shockwaves through the statistical and polling communities. Over the course of the election year in 2016 up until the week of the election, poll predictions of Hillary Clinton's likelihood of beating Donald Trump ranged from 71%- 99% probability [1]. So when Trump beat these odds, the polling industry lost a lot of trust from the general public [2].

This small, specialized industry that is the political polling business, has a great deal of influence on how the election is portrayed in the news media, voter decisions, and candidate partisan policy initiatives. [1]. Million dollar decisions on advertising and campaign strategy are dictated by polls.

Polls conducted early in the election year are a weak predictor of election outcomes because the general public has paid less attention to the race and is less knowledgeable of the candidates' platforms. Voters that tend to sway between parties often report that they are undecided, however these are arguably the most important opinions in predicting the election outcomes. [3] As the election nears, polls become more accurate, but that is where historical prediction models have been lacking – they fail to take into account the differential accuracy in poll results. Historical models, regression based models that rely on outcomes from past elections and structural factors, predict election outcomes at a single point in time which renders high levels of uncertainty [4].

Drew A. Linzer developed a dynamic bayesian forecasting model to predict the U.S. presidential election at the national and state level that combines these historical models with everchanging poll updates. Linzer's model uses hierarchical specification to handle states being polled on different days and takes into account sampling errors of the polls and national campaign effects [4].

In this report we aim to:

- Predict the outcome of the presidential election and the electoral college vote using the Linzer model to predict swing state outcomes in combination
- Predict whether the US Senate remains in Republican control by using an adaptation of the Linzer model and FiveThirtyEight Senate poll data for each state
- Predict the outcomes of all 13 NC Congressional elections using Linzer model with input from FiveThirtyEight Senate poll data for North Carolina along with our model that predicts who will vote in North Carolina
- Predict the outcome of the NC Senate election and the associated uncertainty using the Linzer model.

Taking into account the anomaly that was the 2016 election, we chose to use the Linzer model to best account for slight and unexpected changes leading up to election day. In Section 2 of this report we will discuss datasets used for all tasks, including a brief exploratory data analysis. In Section 3, we formulate models and methodologies that answer all the research questions. Section 4 will present model diagnostic and sanity check validation of prediction. Then, in section 5 we present the major results of our analysis. Section 6 will be focused on conducting sensitivity analysis to test data imputation hypothesis and prior choices.

2. Data Source and EDA

2.1 Description of Data

In order to answer these questions, we used a total of four datasets: senate polls, house polls, 2020 US presidential election polls, and North Carolina voter registration history snapshot dataset. Both the senate and house polls dataset comes from the fivethirtyeight website [5], while the presidential election polls dataset comes from the Economist website [6].Senate and house polls have 38 variables and 4061 and 2655 observations respectively. The presidential election polls have 1447 observations and 17 variables which are: poll state, pollster, sponsor, start and end date, entry time, number of observations, population, method, Biden, Trump, Biden margin, other, undecided, URL, include, and notes.

The North Carolina voter history dataset contains information on the 2016 elections and about how voters voted (method, election, county, party affiliation, precinct). This will help us model and identify potential voters in 2020 (Interim report). On top of the created model, we used the 2020 voter registration snapshot dataset to identify potential voters, and use this information as additional input to model house polls results.

2.2 Exploratory Data Analysis

To get a basic understanding of the four datasets we were working with, we went through the data to understand what we were working with. In the senate and the house polls, we explored the variables of: methodology of the poll, the state in which the poll was conducted, which party the candidate was, and the percentage of the poll. In the presidential election polls, we explored similar variables of: method of poll and state, in addition to the margin between Biden and Trump. Because all three of these datasets had similar variables, we are able to compare them to each other. In Figure A, we can see how the methods of polling were distributed between the house, senate, and presidential polls. In Figure B, we can see how each poll's distribution of which state was polled, and in Figure C, we can see how the house and senate polls' candidate party are different from each other. Finally, in Figure D, we can see the distribution of the margin between Biden and Trump.

To explore the North Carolina voter history dataset, we looked at variables such as method, county, and party affiliation of the voter. We can see in Figure E that the majority of North Carolina voters are from two counties, Wake and Mecklenberg. Party affiliation and the method that the voters used are included in our appendix.

3. Model Formulation

3.1 Model for Question 1

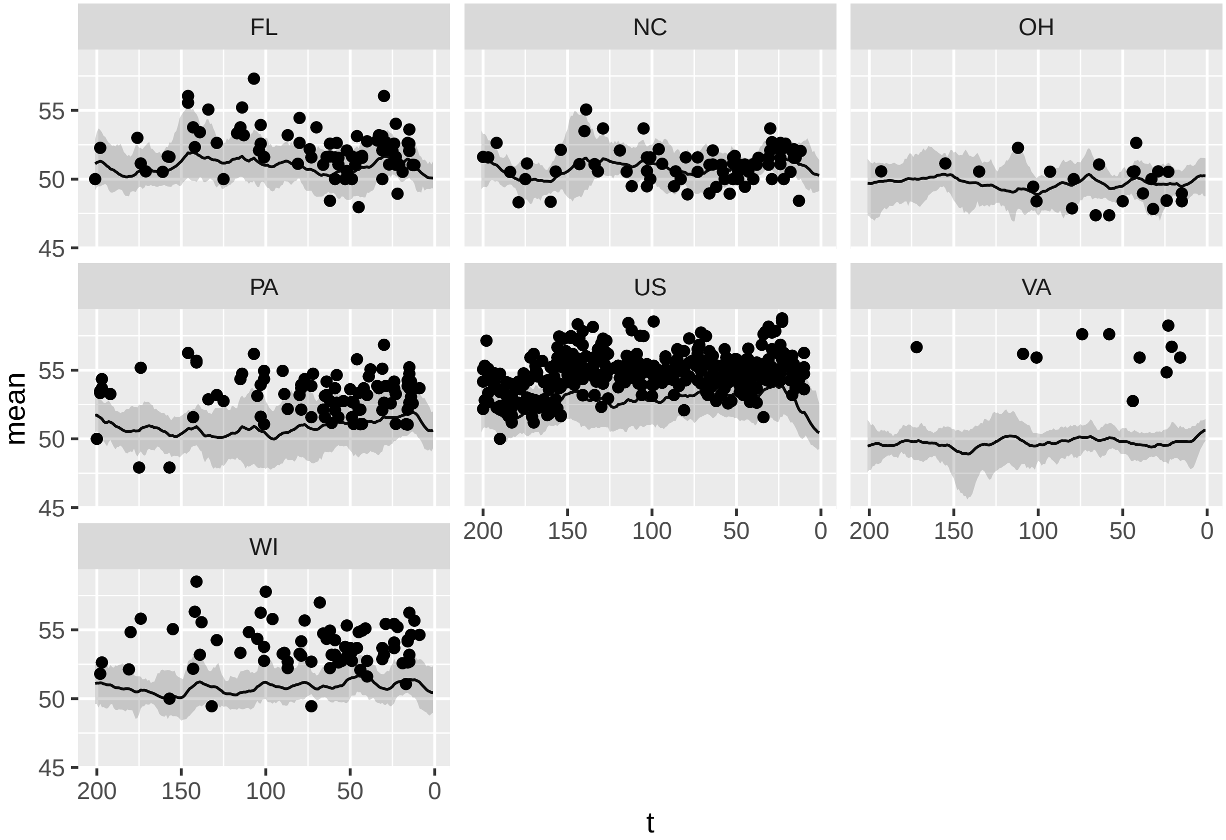
In this section, we build a model to answer our first research question: Predict the outcome of the presidential election and the electoral college vote using the Linzer model to predict swing state outcomes in combination. We follow the route of Linzer and build our model. Specifically, the motivation is that we want to model Joe Biden's percentage of EC vote in the state specific level of detail. We believe that throughout time, each state's preference for Joe Biden is randomly varying and follows a random walk process. Besides, each state's uncertainty of their attitude towards Joe Biden is also state-widely different. We make the assumption that such uncertainty is constant throughout time. Therefore, we create the following extension of Linzer Model.

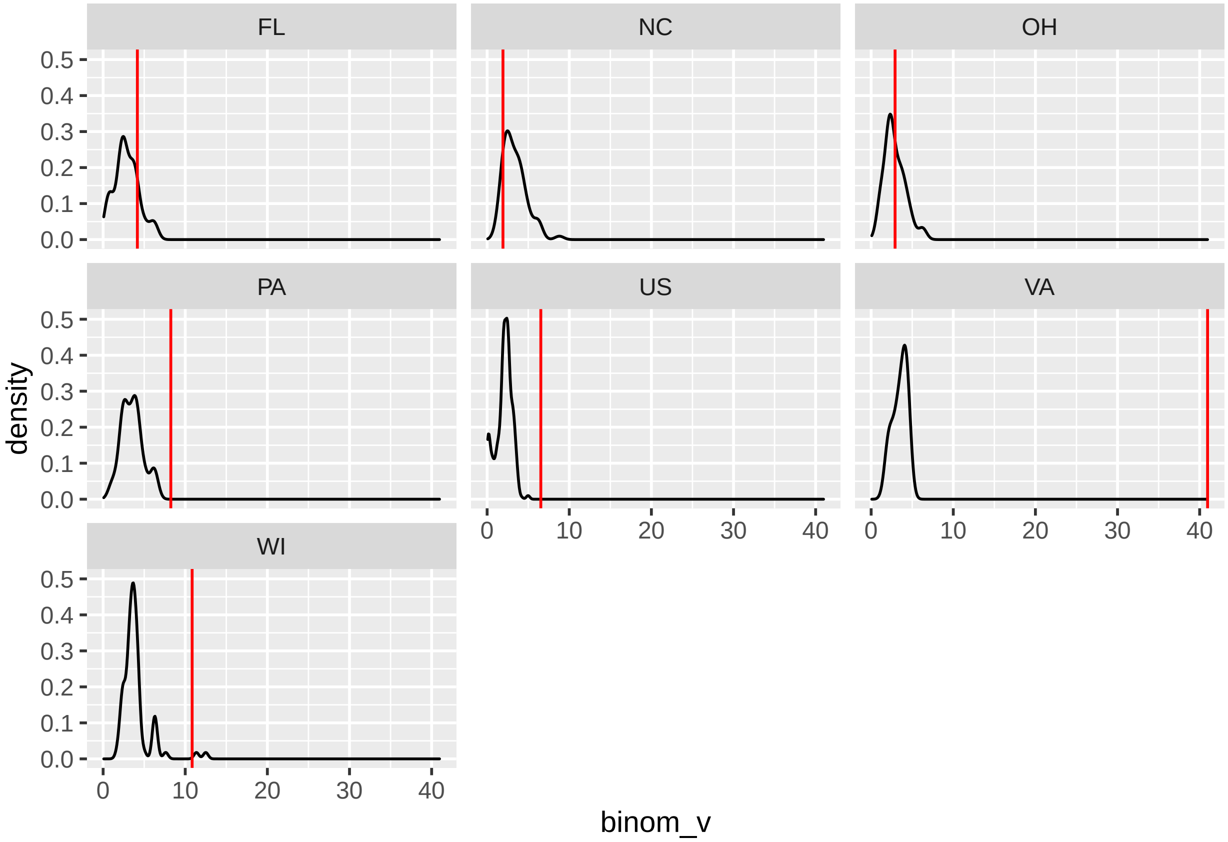
$$
\begin{aligned}
y_k &\sim N\left(\beta_{i[k]t[k]}, (\sigma_y^2)_{i[k]}\right) \\
\text{for } t > 1 : \beta_{it} &\sim N\left(\beta_{i,t-1}, (\sigma_\beta^2)\right) \\
\text{for } t = 1 : \beta_{i1} &\sim N\left(\mu_0, \sigma_0^2\right) \\
(\sigma_y^2)_{i[k]} &\sim \text{InvGamma}(\nu_y, \tau_y) \\
(\sigma_\beta^2) &\sim \text{InvGamma}(\nu_\beta, \tau_\beta)
\end{aligned}
$$

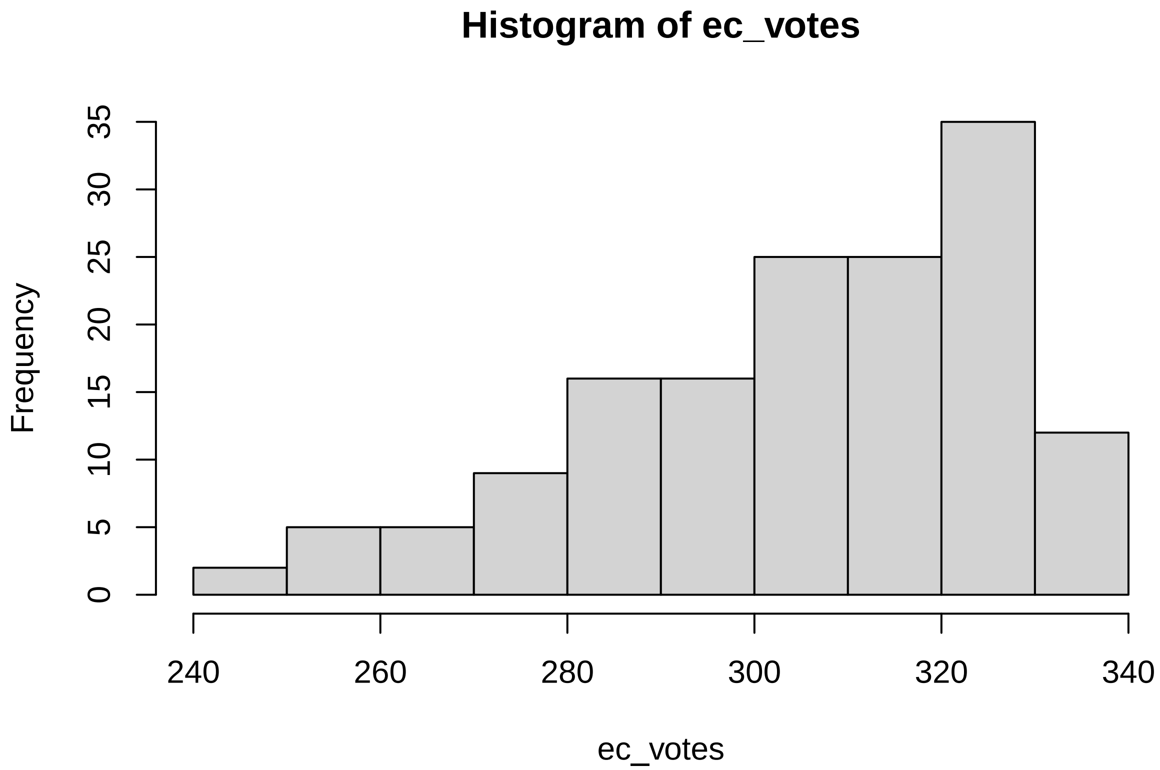
$$
\begin{aligned}
\mu_0 &\sim N(50, 17) \\
\sigma_0^2 &\sim \text{InvGamma}(\frac{1}{2}, \frac{1}{2}) \\
\nu_y &\sim \text{Uni}(0, 100) \\
\tau_y &\sim \text{Uni}(0, 100) \\
\nu_\beta &\sim \text{Uni}(0, 100) \\
\tau_\beta &\sim \text{Uni}(0, 100)
\end{aligned}
$$

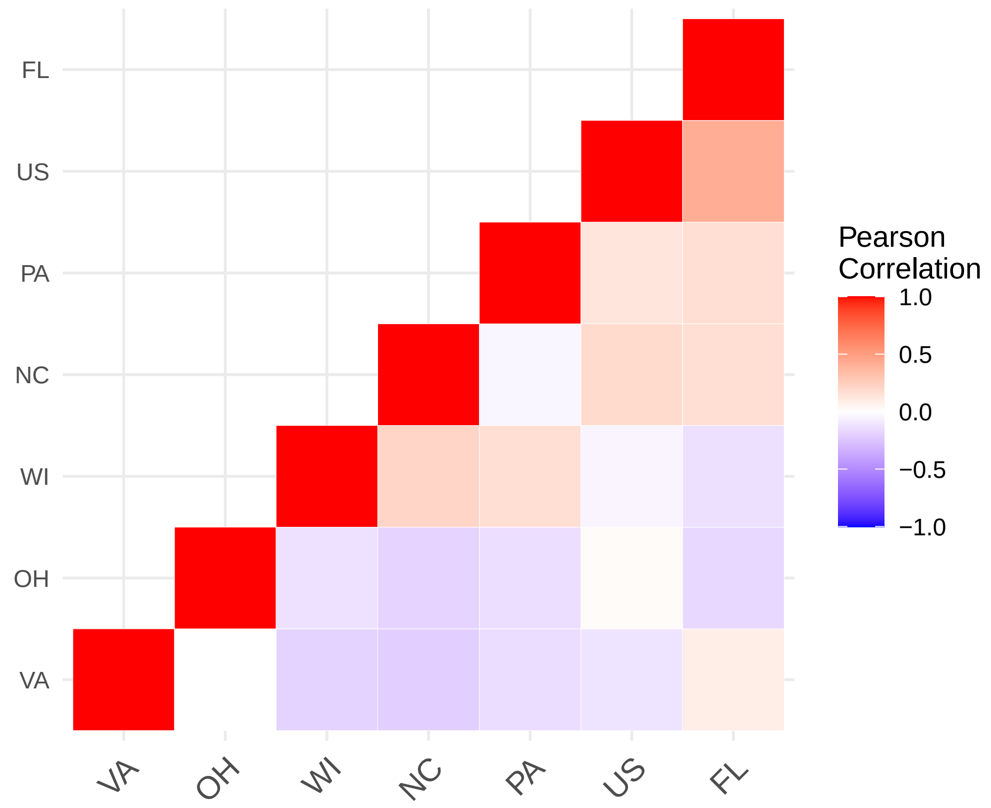
Where k index the polls, i index the states, and t index the date. $i[k]$ represents the k^{th} poll's corresponding state index, and $t[k]$ represents the k^{th} poll's corresponding date index. We model samples from the economist poll data of 2020 and therefore are able to sample the posterior distribution of 1) Joe Biden's chance of winning, and 2) total electoral vote. The sampled posterior distribution y (percentage of supportance within each state) will be calculated by comparing to rounding by 50% to simulate the "winner takes all" procedure. After such adjustment, we can simply multiply the number of electoral college votes of each state to obtain a posterior distribution of electoral college votes of the entire nation. Thus, 2) can be obtained. By comparing to 270 again provides the probability of Joe Biden being elected.

```
## Warning: Removed 985 rows containing missing values (geom_point).
```


FL
NC
OH
PA
US
VA
WI
mean
t
45
50
55
200
150
100
50
0


FL
NC
OH
PA
US
VA
WI
density
binom_v
0.0
0.1
0.2
0.3
0.4
0.5
0
10
20
30
40


Histogram of ec_votes
Frequency
0
5
10
15
20
25
30
35
240
260
280
300
320
340
ec_votes



3.2 Model for Question 2, 4

In this section, we're answering question 2 and 4 jointly: Predict whether the US Senate remains in republican control and predict the outcome of the NC Senate election. Similarly, using the above model with exactly the same notation, and by switching Economist dataset to FiveThirtyEight senate polls data, the exact inference procedure can be produced. After modeling and predicting the Republican's support percentage, we can predict whether the Republican or Democrat party wins the state majority. This procedure includes the state of North Carolina. Furthermore, by summing the posterior samples of Republican senators in each state on election day, we can obtain the posterior distribution of US Senate remains in Republican Party's control.

```
sort(states)
```

```
##  [1] "Alabama"        "Alaska"         "Arizona"        "Arkansas"
##  [5] "California"     "Colorado"       "Connecticut"    "Delaware"
##  [9] "Florida"        "Georgia"        "Idaho"          "Indiana"
## [13] "Iowa"           "Kansas"         "Kentucky"       "Louisiana"
## [17] "Maine"          "Maryland"       "Massachusetts"  "Michigan"
## [21] "Minnesota"      "Mississippi"    "Missouri"       "Montana"
## [25] "Nebraska"       "Nevada"         "New Hampshire"  "New Jersey"
## [29] "New Mexico"     "New York"       "North Carolina" "North Dakota"
## [33] "Ohio"           "Oklahoma"       "Oregon"         "Pennsylvania"
## [37] "Rhode Island"   "South Carolina" "Tennessee"      "Texas"
## [41] "Utah"           "Vermont"        "Virginia"       "Washington"
## [45] "West Virginia"  "Wisconsin"      "Wyoming"
```

Hawaii

Illinois

South Dakota

```
senator_poll_plot_data <- tibble(y=senator_jags_data$y,
                                 t=senator_jags_data$t %>% as.integer(),
                                 state = states[senator_jags_data$r])

senator_beta_plot_data <- lapply(1:N_states,function(st){
  sims <- jags_sims_senator$BUGSoutput$sims.list$theta[,st,]
  data.frame(state = states[st],
             t=1:N_days,mean=colMeans(sims),
             lb=apply(sims,2,quantile,probs=.025),
             ub=apply(sims,2,quantile,probs=.975))}) %>%
  bind_rows()

left_join(senator_beta_plot_data,senator_poll_plot_data) %>%
  ggplot(aes(x=t)) +
  geom_line(aes(y=mean)) +
  geom_ribbon(aes(ymin = lb,ymax = ub),alpha = .2) +
  geom_point(aes(y=y)) +
  scale_x_reverse() +
  facet_wrap(~ state)
```

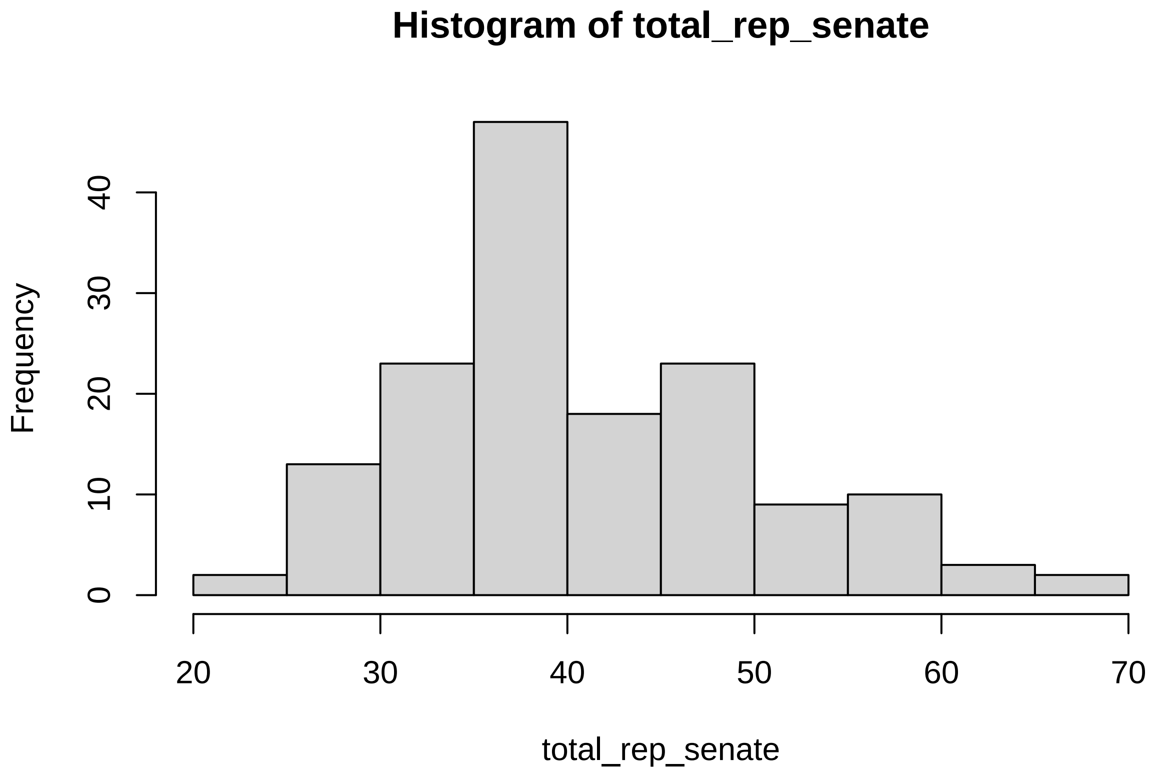
```
senator_poll_plot_data$p <- jags_sims_senator$BUGSoutput$mean$p
senator_poll_plot_data$sigma2y <- jags_sims_senator$BUGSoutput$mean$sigma2_y[senator_jags_data$r]
senator_poll_plot_data$binom_v <- (senator_poll_plot_data$p)*(100-senator_poll_plot_data$p)/ senator$sa


senator_poll_plot_data %>%
  ggplot(aes(x=binom_v)) +
  geom_density() +
  geom_vline(aes(xintercept = sigma2y),color = "red") +
  facet_wrap(~ state)
```

```
temp <- as.data.frame( t(senator_sims %>% {.>50} %>% colMeans()) )
temp$‘North Carolina‘
```

```
## [1] 0.2
```

```
# names(senator_sims)
```

```
#need to change!!
#in other states: Biden: 232-13, Trump: 306-29-18-15-10-20
# c_ex_votes <- 232-13; t_ex_votes <- 306-29-18-15-10-20
total_rep_senate <-  ( { senator_sims>50 } * 2) %>% rowSums() + 2
hist(total_rep_senate)
```



```
sum( rowMeans(senator_sims* 48/50 + 2)>50   ) / dim(senator_sims)[1]
```

```
## [1] 0.1933333
```

3.3 Model for Question 3

In this section, we answer research question 3: Predict the outcomes of all 13 NC Congressional elections. The dataset we've used is the FiveThirtyEight house poll data. However, the dataset's sample sizes on each NC congressional district are small. This is due to the fact house polls happen less frequently compared to either senate polls or presidential polls. Besides, not all congressional districts are competitive because there is no term limit for representatives. FiveThirtyEight has only recorded poll history data for competitive districts including districts 2,3,7,8,9,11,13. Thus, correlation between non-competitive districts cannot be captured without inputting extra datasets. Thus, we concatenated this section's modeling procedure with the output of the interim reports' "who vote" results.

Using the model to predict who will vote in 2020, we ran this binary output model on all the registered voters in NC to predict whether they'll vote or not. In the NC voter registration profile 2020 snapshot dataset, there is a column indicating party affiliation of each potential voter. Thus, we can predict the percentage of Republican voters for each congressional district and use this percentage as an imputed value for polls percentage. In this way, we're essentially imputing polling results for non-competitive districts, which can be also fed into the Linzer Model defined below. Such imputation is reasonable but bold. Thus, sensitivity analysis of imputed percentage will be further explored later in section ********

Below is the model we're using for this task:

$$\begin{aligned} y_k &\sim N\left(\beta_{i[k]t[k]}, (\sigma_y^2)_{i[k]}\right)\\ \text{for } t > 1 : \beta_{it} &\sim N\left(\beta_{i,t-1}, (\sigma_\beta^2)\right)\\ \text{for } t = 1 : \beta_{i1} &\sim N\left(\mu_0, \sigma_0^2\right)\\ (\sigma_y^2)_{i[k]} &\sim \text{InvGamma}(\nu_y, \tau_y)\\ (\sigma_\beta^2) &\sim \text{InvGamma}(\nu_\beta, \tau_\beta) \end{aligned}$$

$$\begin{aligned} \mu_0 &\sim N(50, 17)\\ \sigma_0^2 &\sim \text{InvGamma}(\frac{1}{2}, \frac{1}{2})\\ \nu_y &\sim \text{Uni}(0, 100)\\ \tau_y &\sim \text{Uni}(0, 100)\\ \nu_\beta &\sim \text{Uni}(0, 100)\\ \tau_\beta &\sim \text{Uni}(0, 100) \end{aligned}$$

Where k index the polls or the imputed polls, i index the congressional district, and t index the polls date. $i[k]$ represents the k^{th} poll's corresponding congressional district index, and $t[k]$ represents the k^{th} poll's corresponding date index.

```r
house_sims <- jags_sims_house$BUGSoutput$sims.list$theta[,,1]
colnames(house_sims) <- cong_dist
house_sims %>% {.>50} %>% colMeans()
```

```
## District 11  District 8  District 9  District 3 District 13  District 2
##   0.2400000   0.6266667   0.1800000   0.3400000   0.4400000   0.6066667
##  District 7  district 1  district 4  district 5  district 6 district 10
##   0.3600000   0.5000000   0.2600000   0.8066667   0.0400000   0.4866667
## district 12
##   0.2733333
```