Title

Building a Deeper Understanding of Voter Turnout Behavior in Los Angeles County in Preparation for the 2020 Election

Author

Beeman, Paul

Publication Date

2019

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Building a Deeper Understanding of Voter Turnout Behavior in Los Angeles County in Preparation for the 2020 Election

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Applied Statistics

by

Paul Beeman

2019

ABSTRACT OF THE THESIS

Building a Deeper Understanding
of Voter Turnout Behavior in Los Angeles County
in Preparation for the 2020 Election

by

Paul Beeman
Master of Applied Statistics in
University of California, Los Angeles, 2019
Professor Frederic R. Paik Schoenberg, Chair

In this study, we set out to better understand the voter behavior of different demographic partitions of Los Angeles County (LAC). We developed a methodology to enrich and analyze LAC voter data by joining it with US census data and modelling vote by mail rates and turnout rates using general additive models. In doing so, we were able to forecast where in LAC we expect to see the highest rates of turnout by vote by mail and turnout in person for the 2020 presidential election. These findings are timely: In 2020, LAC will roll out a new and novel voting system called Voting Solutions for All People. The findings of the current study provide guidance on how to best allocate resources to high-turnout populations and target outreach to low-turnout populations in preparation for the first election under the new system. This paper presents a combination of visualizations, predictive models, and summary statistics that can inform LAC of where to focus outreach to expand the electorate and build a more representative electorate while also improving preparation in areas where turnout has been historically high.

The thesis of Paul Beeman is approved.

Hongquan Xu

Yingnian Wu

Frederic R. Paik Schoenberg, Committee Chair

University of California, Los Angeles

2019

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank the Economic Roundtable and Los Angeles County Registrar-Recorder/County Clerk for giving me the opportunity to expand my skills while working on important projects to the city of Los Angeles. I would especially like to thank Judy Ly who helped me collect my data for this paper and whose input was invaluable to me developing my R programming skills.

CHAPTER 1

Introduction

For the past decade, the Los Angeles County Registrar Recorder/County Clerk (RRCC) office has been working to overhaul the voting system in Los Angeles County. In 2009, when the effort to build a new system called Voting Solutions for all People (VSAP) began, it was impossible to foresee the dramatic shift in the national political landscape and resulting increase in political engagement that would come as a result.

VSAP will shift voting in Los Angeles from a polling place-based system to a voting center-based system. Voters will no longer be bound to their polling place, but can instead vote at any voting center. While polling places are only open for one night, voting centers will be open for multiple days. Beginning eleven days before election day, there will be one open voting center for every 30,000 voters. Beginning three days before election day and on election day, there will be one open voting center for every 7,500 voters. Another important distinction is that voters can register to vote at any voting center and cast provisional ballots through election day.

The convenience of being able to vote and register to vote at any location for over a week leading up to election provides a trade-off with the diminished number of voting locations open on election day. There will be roughly 3,000 less voting locations on election day. This will of course not be an issue if people vote early; however, if a low percentage of voters cast their ballots before election day, there could be lines that stretch for several hours on election day. Los Angeles County is making a significant bet on the likelihood of voters turning out early at voting centers and/or voting by mail.

On top of increasing accessibility by widening the voting window, Los Angeles County has teamed up with the global design firm IDEO to develop a single, user-friendly interface for all voters to use. While the new system is more elegant, more user friendly, and more accessible, voters will be experiencing it for the first time in 2020. The implementation of a new user interface will likely add to the uncertainty in how successful the VSAP system roll-out will be, specifically affecting how long the average voter will remain inside a voting center.

The 2018 midterm election provided a glimpse of what the worst-case scenario could look like. Voters who missed the registration deadline were able to register at the RRCC on election day and cast provisional ballots. The line was hours long and hundreds of people deep; it was an inspiring sight to behold from a civic engagement standpoint, but one couldn't help but wonder how many people saw the line and decided that the wait was not worth it or simply not feasible.

For VSAP to be successful, Los Angeles County voters need to be educated about the new process and made aware of the benefits of turning out early and/or voting by mail. Voters that are not conditioned to the new system will likely default to what they know, which is voting on election day. This will likely result in a similar situation to the 2018 midterm election, which would be an unacceptable outcome for VSAP.

VSAP has dual goals: first, expanding the electorate, and second, diminishing the burden of voting across all socioeconomic groups. These goals can be in conflict, because over-allocating resources to historically low-turnout communities may increase their turnout but simultaneously risks overburdening historically high-turnout communities in the form of long lines.

The worst possible outcome for VSAP would be a dual failure: (1) experiencing low turnout

in historically low-turnout communities and (2) experiencing long lines in historically high-turnout communities; this would be the result of poor allocation of both resources and outreach. This study aims to provide Los Angeles County with guidelines for both allocation of resources and strategic outreach in preparation for the 2020 election. This is achieved building multiple predictive and descriptive models and providing summary statistics using voter and census tract data from midterm and presidential elections going back to 2012.

Two response variables will be explored in informing this study's suggestions for voter outreach in terms of turnout. The first response variable we will look at is the percentage of registered voters that are registered to vote by mail in each census tract. The second response variable we will look at is the percentage of registered voters that actually vote.

The investigation of the first response variable (i.e., the percentage of registered voters that are registered to vote by mail in each census tract) is an important and novel aspect of the study because, to the best of the author's knowledge, there has been no prior extensive research done on what demographic predictors are correlated with vote by mail utilization. Many of the arguments relating to social justice and day-of-voting center on a disproportionate opportunity cost faced by lower income individuals in voting. Vote by mail is, theoretically, a "great equalizer" because it can eliminate lines, travel time, child care, and multiple other barriers to voting. Still, as of 2018, less than 50% of LA County registered voters were registered to vote by mail. Therefore, understanding what demographic predictions are correlated to vote by mail utilization is a key aspect of this study.

Not only does vote by mail theoretically achieve what VSAP hopes to achieve (i.e., decreasing the burden of voting across socioeconomic groups), it has been in existence for decades. Understanding which groups of people have been consistently averse to voting by mail may help Los Angeles County narrow down which groups VSAP could benefit the most, enabling Los Angeles County to target outreach to those groups. For example, if elderly individuals have shown a consistent aversion to vote by mail, Los Angeles County should consider fo-

cusing VSAP outreach towards elderly individuals.

The second response variable (i.e., the percentage of registered voters who actually voted) will also be modeled. By understanding the relationship between the registered voter and voting populations, Los Angeles County can develop greater understanding of which census tracts will produce long lines at voting centers. For example, census tracts that have historically high turnout rates and low rates of vote by mail utilization should trigger concerns for potential long wait times at voting centers.

Finally, a predictive model that is informed by the descriptive models will be built to give Los Angeles County a comprehensive understanding of how many non-vote by mail voters they should expect to turn out from each census tract in the 2020 election.

These predictions will inform Los Angeles County as to where they need to prepare for the largest influx of voters, and also where they need to focus their outreach efforts. The results provided in this paper will highlight areas of Los Angeles County that have both high and low expected turnout. Los Angeles County can then utilize this information to encourage higher turnout in census tracts with low expected turnout and to increase allocation of resources to census tracts with high expected turnout.

CHAPTER 2

Literature Review

A key goal of VSAP is to expand the electorate by making the voting experience more convenient through early voting and election day registration (EDR). However, previous studies on the effects of early voting and EDR have provided mixed levels of support for this hypothesis.

In their paper *Election Day Registration's Effect on U.S. Voter Turnout*, after comparing turnout from several states with EDR laws from before and after the implementation of EDR, Brians and Grofman conclude that EDR "exerts a strong and positive influence on turnout." Furthermore, they conclude that "although turnout inches higher as closing dates shorten, voter turnout still remains higher with the adoption of EDR than with even very short closing dates." They attribute this to the convenience factor of being able to make a single trip to both register to vote and vote. Their findings supports EDR as a tool to level the playing field for people who experience a higher opportunity cost when waiting in line to vote, which is in line with the goals of VSAP in Los Angeles. Brians and Grofman also find that EDR increased turnout by 4 percentage points on average and the group that saw the greatest increase in turnout (i.e., by 5%) was the middle socioeconomic status (SES) group, which they define as people with a high school education and middle income. Both high and low SES groups had an increase of 3% after EDR implementation. Brians and Grofman attribute this to higher barriers to voting for voters with low SES and previously existing high representation of voters with high SES in the voting population [1].

According to a study done by Gronke and Toffee in 2008, "early voters are older, better educated, are more likely to declare a partisan affiliation on the voter registration form, and

tend to be exposed to party mobilization efforts." Gronke and Toffee found these differences to be significant when looking at presidential elections, but insignificant when looking at midterm elections. Gronke and Toffee theorized that this discrepancy was due to the self-selecting nature of midterm voters: midterm voters are already more homogenous in their education and age; as a result, differences between early voters and election day voters are more difficult to discern.

A wide range of studies have found that long lines on election night are disproportionately suffered by urban populations with higher rates of minorities [2]. A study performed in 2011 by Bradly and McNulty found that when a single polling place was moved in Los Angeles County, a subsequent transportation and searching effect resulted in a 3.03% drop in turnout. Some of the drop in turnout was offset by an increase in absentee voting; however, overall, there was a diminished turnout of 1.85% [3].

Under VSAP, with the consolidation of polling places into voting centers, there will also likely be a greater transportation and searching strain on voters as they get to their new voting locations. According to the RRCC, in 2016, 79% of people that utilized one of the limited early voting polling stations dropped ballots off at one of the three closest locations to them. This evidence supports the need to consider transportation limitations in the placement of polling locations, as distance traveled has been shown to have a significant impact on turnout and has a disproportionate strain on lower income individuals with limited access to cars.

CHAPTER 3

Data Collection: Sources and Methodology

In the current study, the census tract level was selected to predict voter turnout levels for the 2020 election. The census tract level was chosen because census data provides a rich, granular description of a county. Using the R package tidycensus [4] we were able to connect directly to the US Census API[1] and load data from either the decennial US Census or the 5-year American Community Survey.

Anonymous voter data was aggregated at the census tract level and provided to us directly from the Los Angeles County RRCC. After considerable data cleaning[2] we were able to join the two data sets using the census tract number variable. Thus, the 2012 5-year American Community Survey was joined with the 2012 presidential election voter data and the 2014 midterm election voter data while the 2017 5-year American Community Survey was joined with the 2016 presidential election voter data and 2018 midterm data.

Data collection methods have been consistently improving in recent years at Los Angeles County over the past four major elections. In 2012, 8.2% of the 4,802,287 registered voters in Los Angeles County were unable to be matched to a census tract due to poor data collection practices. By 2018, only 87 out of 5,325,017 total voters were unsuccessfully mapped to a census tract. Due to the higher quality of data in recent years, we focused mainly on the 2016 presidential and the 2018 midterm election; earlier elections were referenced as needed throughout the study. The data that Los Angeles County provided us included

[1] Application Programming Interface

[2] see R code in appendix for details

voter registration counts, total votes for each election broken down by political party, and whether the voters were registered to vote by mail or vote in person. After each major election, Los Angeles County takes a "snapshot" of registered voters, which allowed us to identify which census tract each voter lived in at their time of voting. For the sake of parsimoniousness, we combined several of the less popular political parties into an "other" bin, resulting in categories of Republican, Democrat, No Party Preference (NPP), and other.

The data available to us from the 5-year American Community Survey is extremely expansive. The 2017 survey provides access to over 25,000 data points for each census tract which was quite intimidating, but the tidycensus package provides a useful set of tools to search through variables. Nonetheless, the process of sifting through variables to find meaningful data was far from trivial. The majority of the variables were overly specific, limiting their usefulness in building models. From the 5-year American Community Survey, we chose to explore variables pertaining to age, race, gender, earnings, employment, poverty, education, transportation, and childcare.

Categories	Variables
Population	• Total Population • Population of Voting Age
Gender	• Count of females • Count of males
Race/Ethnicity	• Count of African Americans • Count of European Americans • Count of Asian Americans • Count of Native Americans • Count of Pacific Islanders • Count of Hispanic or Latino
Earnings	• Average Yearly Earnings

Employment	• Count of unemployed in the workforce • Count of employed in the workforce
Poverty	• Count of people under the poverty level • Count of people between the poverty level and 1.5x the poverty level • Count of people above 1.5x the poverty level
Education	• Count of people that did not graduate high school • Count of people with high school education • Count of people with bachelor's degree • Count of people with graduate degree
Transportation	• Count of people that drive to work • Count of people that bike to work • Count of people that take the bus to work
Childcare	• Count of children under 6 • Count of children between 6 and 17

Table 3.1: Breakdown of variables pulled from the census

CHAPTER 4

Summary Statistics

4.1 Los Angeles County Voting Statistics

Election	Total Registrations	Voters	Turnout Percentage
2004 Presidential	3,972,738	2,710,280	77%
2006 Midterm	3,914,138	2,033,119	52%
2008 Presidential	4,093,488	3,368,057	82%
2010 Midterm	4,403,046	2,377,105	54%
2012 Presidential	4,802,287	3,137,427	65%
2014 Midterm	4,908,969	1,502,482	31%
2016 Presidential	5,250,700	3,445,829	67%
2018 Midterm	5,325,017	2,918,717	55%

Table 4.1: Recent Major Election Turnout in Los Angeles County [4,5,6]

The 2008 presidential election had the highest turnout percentage of registered voters in Los Angeles County going back to 1990. The 2018 midterm election had the second highest turnout for a midterm election going back to 1990. It is important to note, however, that although the turnout percentage for the 2018 midterm was not unprecedented, the raw number of voters was staggering. Turnout was nearly double the unusually low turnout from the previous midterm election and exceeded the turnout from the 2004 presidential election.

We also see that vote by mail voters are making up an increasingly high percentage of registered voters in Los Angeles County. In 2018, the majority of the votes cast were by vote by

Year	Registration Category	Votes	Registered	Percent Voted
2012	NON-VBM	2072216	3423684	0.60525913
2012	VBM	1065211	1378603	0.77267422
2014	NON-VBM	848533	3392698	0.25010567
2014	VBM	653949	1516271	0.43128768
2016	NON-VBM	1841164	3068727	0.59997647
2016	VBM	1604665	2181973	0.73541927
2018	NON-VBM	1299833	2745468	0.47344679
2018	VBM	1618884	2579549	0.62758412

Table 4.2: Turnout and Registration for Vote by Mail vs. In-Person [4]

Year	VBM Percent
2012	28.7
2014	30.9
2016	41.6
2018	48.4

Table 4.3: Percent of voters registered to vote by mail

mail voters, even though vote by mail registrations still made up a minority of total voting registrations. Furthermore, vote by mail voters consistently voted at significantly higher rates than in-person voters. This could be a result of more engaged people self-selecting as vote by mail voters, but convenience may also be an important factor.

Figure 4.2 shows that since 2006, voter registration has been growing at a faster rate than the population of eligible voters. Also, significant jumps are seen leading up to presidential elections in recent years. If recent presidential elections are an indicator of what to expect in 2020, there will likely be another large influx of newly registered voters in 2020.

Figure 4.3 shows that voter registration in recent years has followed a pattern of large in-

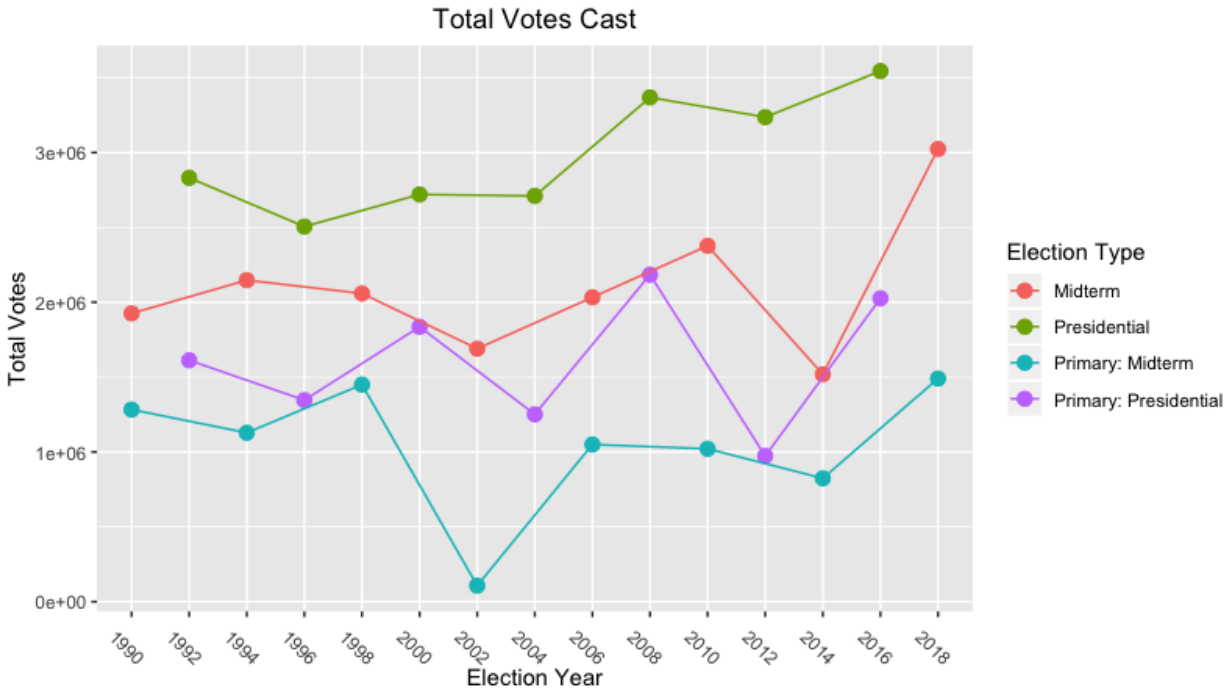

Figure 4.1: Total votes cast for major elections in LA County dating back to 1990

creases leading up to a presidential election followed by small increases in the years leading up to a midterm. The turnout was only 67% in 2016, but thanks to an expanding pool of registered voters, total turnout was the highest on record looking back until 1990.

After President Clinton's first election, we see a two year stagnation of turnout below 70%. This was followed by a large jump in turnout in the second Bush term. A similar pattern of stagnation can be observed following the first Obama term. If recent history is to repeat itself, we can expect to see a large turnout increase in 2020, similar to the increase in turnout percentage in the second Bush term. In these cases, this may be indicative of the predominantly Democratic-leaning voting population in Los Angeles County being lulled by a elected Democratic president and then energized during the first term of a Republican president. This conclusion is of course based on a small sample size, but in preparation for VSAP implementation, it is important to be prepared for the most extreme possibility.

If we were to see 77% turnout, as we did in the second Bush term, and no further registra-

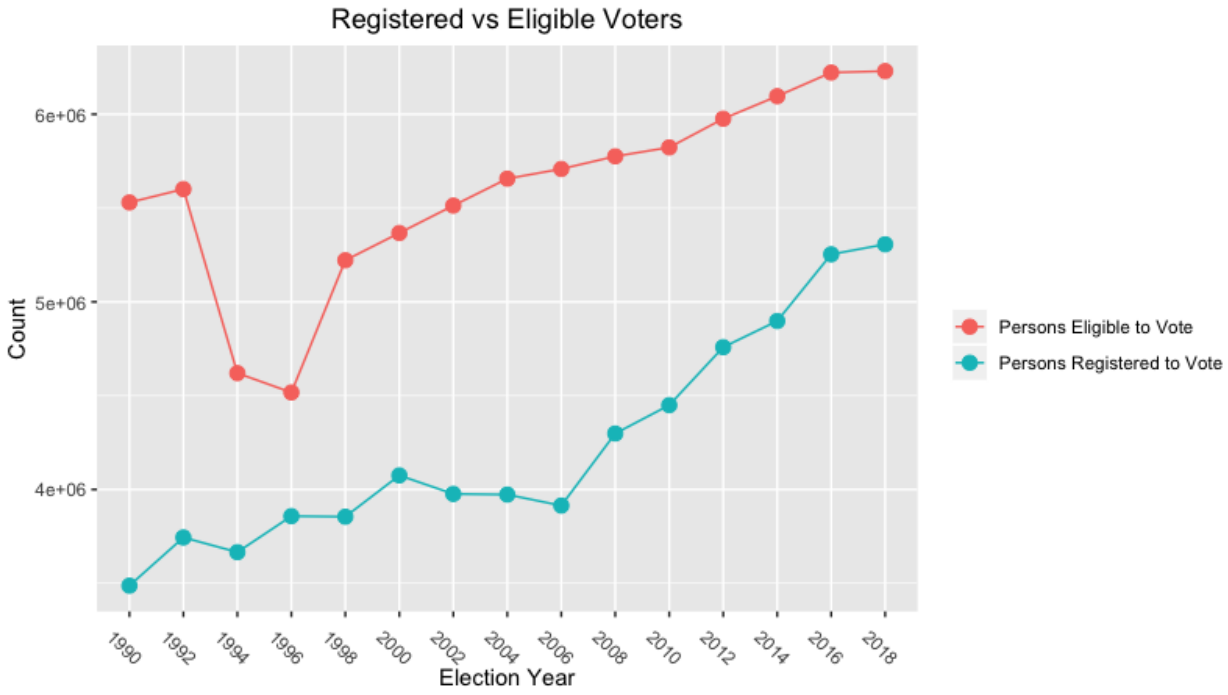

Figure 4.2: Total registered voters vs. total eligible voters in Los Angeles County dating back to 1990, filtered for only general elections

tions[1] from the 2018 registration levels, we would see a turnout of 4,100,263 voters, or over half a million more voters than in the 2016 presidential election. Given the high midterm turnout in 2018 compared to 2014, it seems that political engagement is generally trending up and we should be prepared for a very large turnout in 2020.

[1]The assumption of no further registrations is obviously unrealistic but put in place to demonstrate that even for zero growth of Los Angeles County voting population, we should still expect a historically large turnout

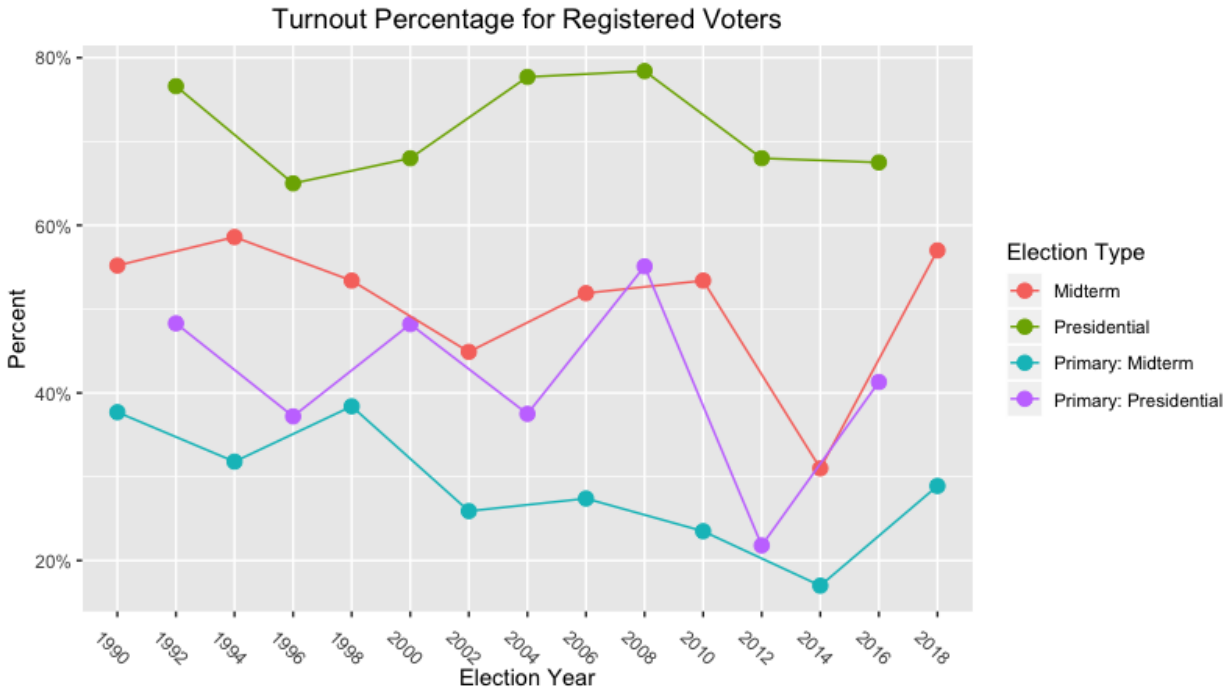

Figure 4.3: Percent turnout of registered voters in Los Angeles County dating back to 1990

4.2 Census Statistics

The R packages tidycensus and ggplot2 were utilized for the purpose of exploring 2017 census tract level data. In performing this preliminary analysis, we begin to see which Angelenos are currently underrepresented in the voting process. Percentiles are mapped for ease of interpretation. In Figures 4.4 through 4.7, we see some visual support for the correlation between poverty, mobility, and turnout.

The maps of poverty and public transportation rate percentiles (Figure 4.4) are near inverses of that of voter turnout (Figures 4.5 through 4.7). In terms of turnout rate[2] percentiles, the maps of 2014, 2016, and 2018 look very similar. This suggests that even as absolute turnout goes up or down, the percentile rank does not change by too much.

[2]number of voters who turned out divided by the number of registered voters

In order to dig deeper, the 2017 census tracts were divided into two groups based on whether they were above or below the 50th percentile for voter turnout rates in the 2018 midterm election. The average yearly earnings of a worker in the higher turnout group were $24,167 higher than the average yearly earnings of a worker in the lower turnout group. The higher turnout group reported a 10% poverty rate compared to 24% in the lower turnout group. The population of the higher turnout group was 40.1% minority compared to 55.7% in the lower turnout group. The higher turnout group had an average age of 40.6 compared to 33.1 in the lower turnout group. 13.1% of the voting-age population in the lower turnout group report having a college degree, while 40.4% of the voting-age population in the higher turnout group report having at least a bachelor's degree. 56.6% of voters in the higher turnout group utilized vote by mail compared to 52.9% in the lower turnout group. In short, census tracts that turn out in higher numbers tend to be populated by people that make more money, have higher levels of education, are older, and contain a smaller minority population than people that populate census tracts that turn out in lower numbers. The higher turnout group also utilizes vote by mail at a higher rate than the lower turnout group, which is consistent with previous studies that observed that older, more educated people tend to utilize early voting systems like vote by mail at higher rates.

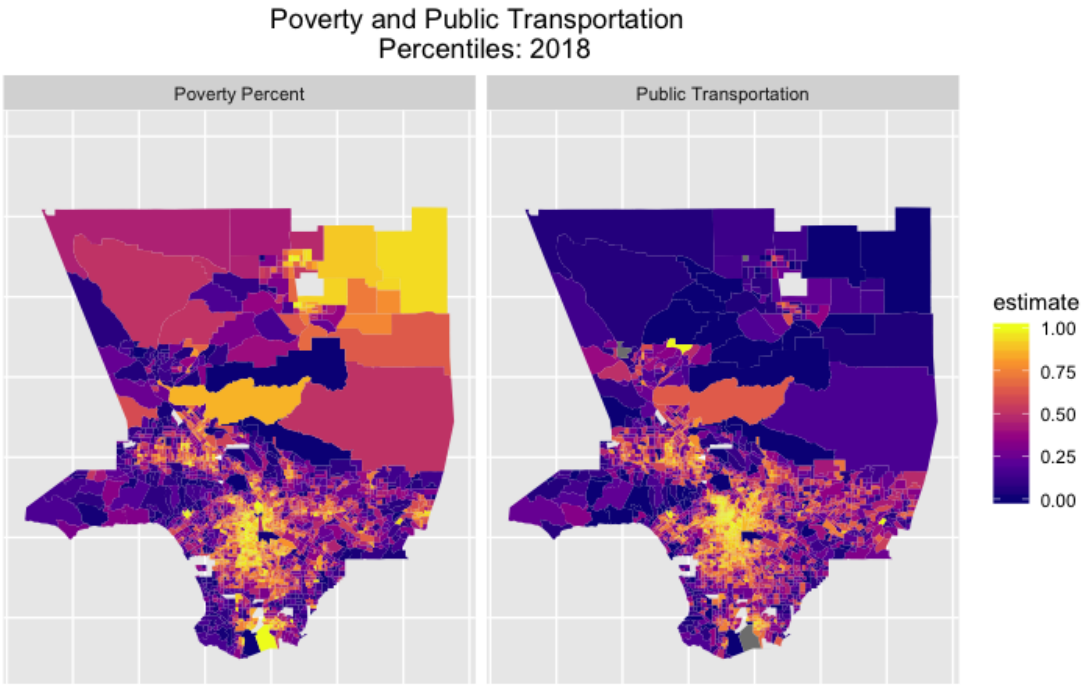

Figure 4.4: Poverty and public transportation percentiles: 2018

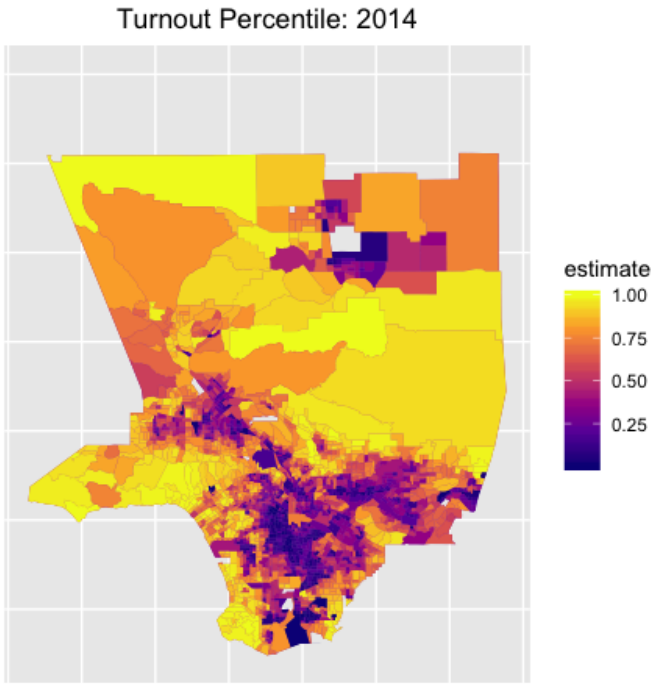

Figure 4.5: Turnout rate percentiles in 2014

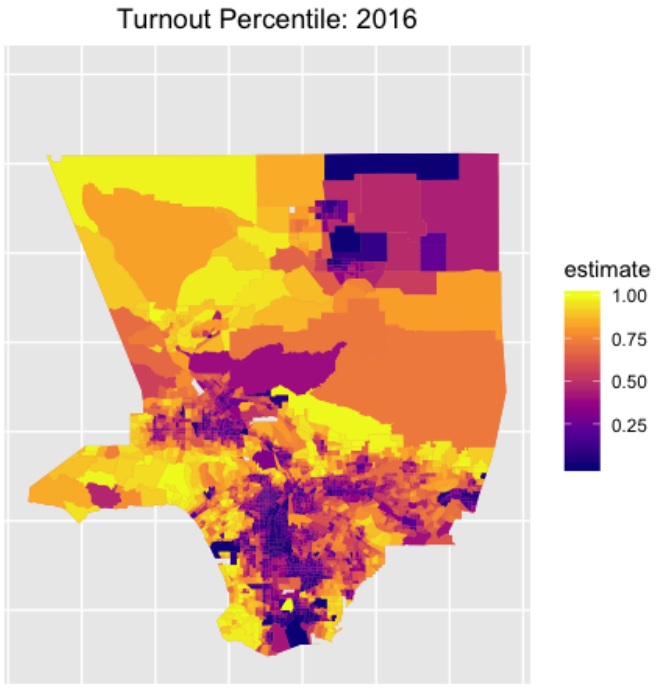

Figure 4.6: Turnout rate percentiles in 2016

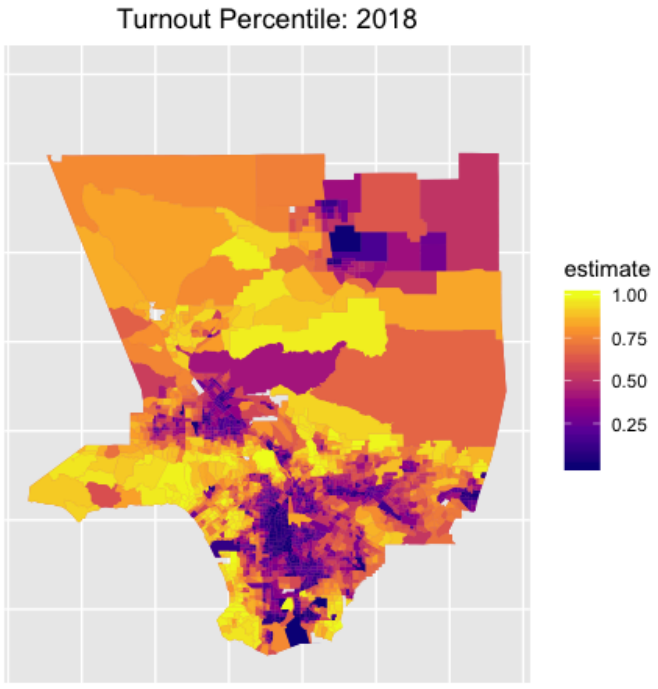

Figure 4.7: Turnout rate percentiles in 2018

CHAPTER 5

General Additive Models (GAMS)

In the current study, we rely on general additive models (GAMs) to model our response variables. GAMs have been chosen in order to take advantage of both their flexible modeling power and simple interpretation. Instead of assuming that the response variables are a linear combination of independent variables, GAMs model the dependent variable as a sum of smoothed functions of the independent variable [7].

$$Y_i = \alpha + \sum_{j=1}^{p} f_j\left(X_{ij}\right) + \epsilon_i$$

The fact that GAMs are additive allows for easy interpretation of the relationships between the predictor variables and the response variable by simply looking at plots of the curved relationships. This is useful for communication to nontechnical audiences. Furthermore, we can adjust the level of smoothness to avoid overly wiggly relationships to increase the interpretability and plausibility of the relationship, assuming that true relationships between variables are relatively smooth [8].

There are a number of methods that can be used to smooth the predictor variables and multiple smoothing methods can be used within the same model. Generally speaking, one can get to very similar results using different methods. In this paper, we use regression splines. Regression splines are built from a linear combination of basis functions that are not reliant on the Y variable. Put simply, the curved approximation of the relationship between our predictor and outcome variable is a weighted sum of k basis functions. The greater k is, the more wiggly the smoother is; therefore, we can use the k parameter to

control for the bias/variance trade-off of the smoother.

$$s(x) = \sum_{l=1}^{K} B_{l,q}(x)\beta_l = B'\beta$$

For our numeric predictors, we predominantly use thin plate splines. Thin plate splines measure a combination of R-squared and penalize for how much the function must bend through the data. This approach is called penalized regression splines and attempts to strike a compromise between fitting the data well without losing extrapolative powers due to overfitting. Thin plate splines penalize the wiggliness of the regression spline by minimizing the following function[1]:

$$E_{\text{tps,smoth}}(f) = \sum_{i=1}^{K} \|y_i - f(x_i)\|^2 + \lambda \iint \left[\left(\frac{\partial^2 f}{\partial x_1^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x_1 \partial x_2} \right)^2 + \left(\frac{\partial^2 f}{\partial x_2^2} \right)^2 \right] \mathrm{d}x_1 \mathrm{d}x_2$$

In the case of proportion predictions, such as vote by mail percentages, we use the beta regression, which is useful when the outcome variable is continuous, but bounded between 0 and 1.

[1]here, x is two dimensional for the sake of simplicity of explanation

CHAPTER 6

Vote By Mail Registration Rates

Vote by mail is of particular interest because it theoretically already provides an option to nearly completely alleviate the strains caused by day-of voting. It should then follow that as the percentage of registered voters that are registered to vote by mail increases, voter turnout percentage would follow a similar increasing trend. While this does occur, a rather large increase in vote by mail registration percentage corresponded to only a modest increase in voter turnout for the past two presidential elections. For the 2012 and 2016 presidential elections, there was a 12.9% increase in percentage of registered voters who were registered to vote by mail, but turnout increased only 2%, from 65% to 67%.

Voter registration has been steadily increasing along with the eligible population of voters in Los Angeles. At the same time, the total number of people actually voting and the percentage of vote by mail voters have steadily increased. However, turnout percentages have not shown the same upward trend. This is consistent with a study performed by fivethirtyeight[9] that showed that early voting reform has historically not resulted in increased turnout percentages, but instead just shifted when people who already voted went to the polls [REF].

It is possible that Los Angeles County is seeing a similar phenomenon, in that people that already vote are simply shifting from voting at the poll to voting by mail. This would account for the higher vote by mail registration and the higher turnout percentage among vote by mail voters, and it would also explain why we are not seeing large increases in overall turnout percentages as a result of increased vote by mail registration.

In other words, if people who are already voting at the polls migrate to vote by mail, we would expect to see higher turnout among vote by mail voters, but not higher overall turnout because it is not expanding the electorate - just redistributing the electorate to a new method of voting. This is still a positive outcome for Los Angeles County in terms of cost-saving and diminishing lines at the polling place, but there is little evidence to show that it is succeeding in expanding the electorate. Diving into the data to deepen our understanding of the relationships between the demographics of neighborhoods and the vote by mail registration percentage of each neighborhood will allow us to understand who is benefiting from vote by mail and who is averse to vote by mail. We can use this knowledge to craft outreach policies and roll-out strategies in regards to VSAP. It is important to note that the dependent variable we are observing is the percentage of those registered to vote who are registered to vote by mail. This dependent variable was chosen to best answer the question: Who are the registered voters that are hesitant to adopt vote by mail?

6.1 Smoothed Univariate Relationships Between Census Data and Vote By Mail Registration Rates

GAMs are well-suited to looking at social science problems when it is more important to understand the relationships between the predictor variables and the outcome variable. In other words, when the interpretation of the model by a non-statistical audience is just as, if not more, important than a powerful prediction, a GAM is a good choice to model the data. In this case, we can look at the smoothed relationships between different demographic characteristics of a census tract and the percentage of voters in that census tract that are registered to vote by mail to immediately gain visual insights as to what groups of individuals are more likely to vote by mail. From a predictive standpoint, we do not improve substantially beyond simply looking at the lag variable of the previous election, but from a descriptive standpoint, we can learn a sizeable amount of information about what popu-

lations are signing up for vote by mail and which populations could benefit from targeted outreach.

Figure 6.1: Univariate relationship plot of average income and VBM registration percentage

We see in figure 6.1 that as average income increases so does vote by mail registration, up to around $80,000 at which point vote by mail registration tapers off.

We also see in Figure 6.2 that as poverty rates increase, vote by mail percentage decreases. This is discouraging, because vote by mail has the most potential to ease the burden of voting for impoverished census tracts. However, it appears that adoption of vote by mail has not caught on in these communities.

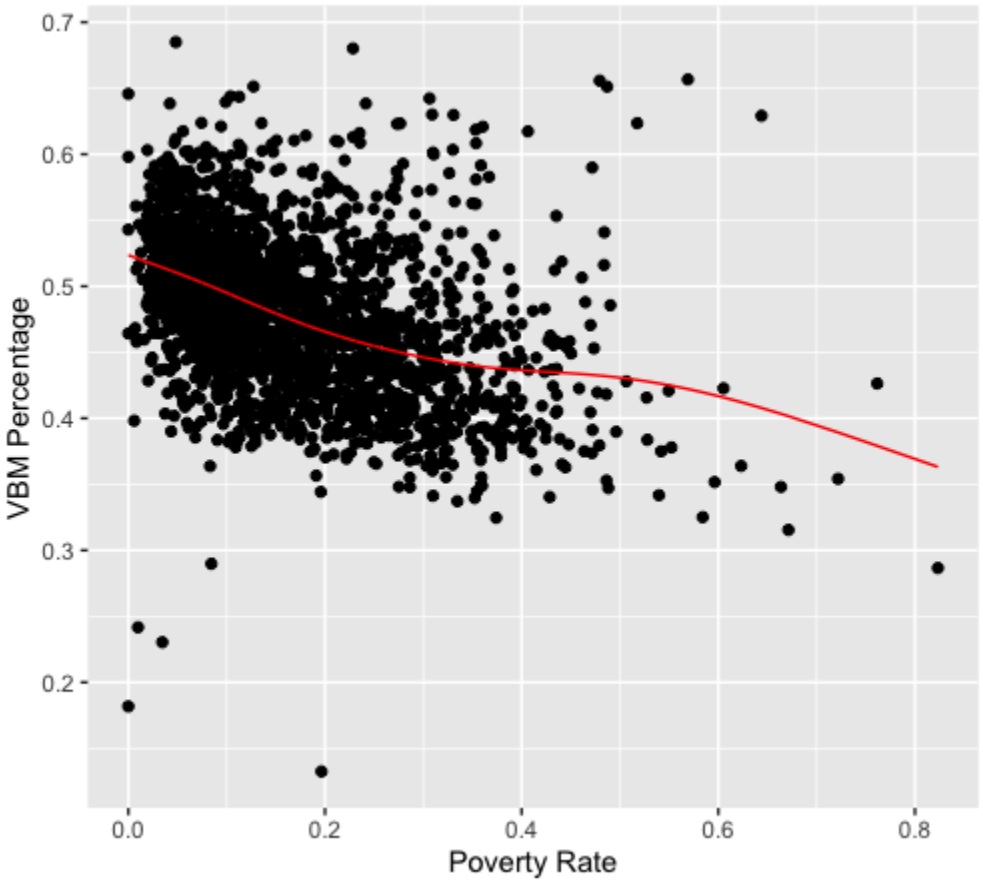

Figure 6.2: Univariate relationship plot of poverty rates and vote by mail registration percentage

Census tracts with an older population also tend to have higher vote by mail registration rates. Figure 6.3 shows a clear linear relationship between older average ages and vote by mail registration percentage.

Figure 6.4 shows that Hispanic communities tend to have lower vote by mail registration rates. The relationship appears to be linear. The relationship between African American population percentage and vote by mail registration does not appear to be strong in any particular direction. This is also the case when looking at Asian population percentages.

Census tracts with a higher rate of people with bachelors' degrees also have higher vote by mail turnout. Generally, census tracts with higher rates of drivers also have higher rates of

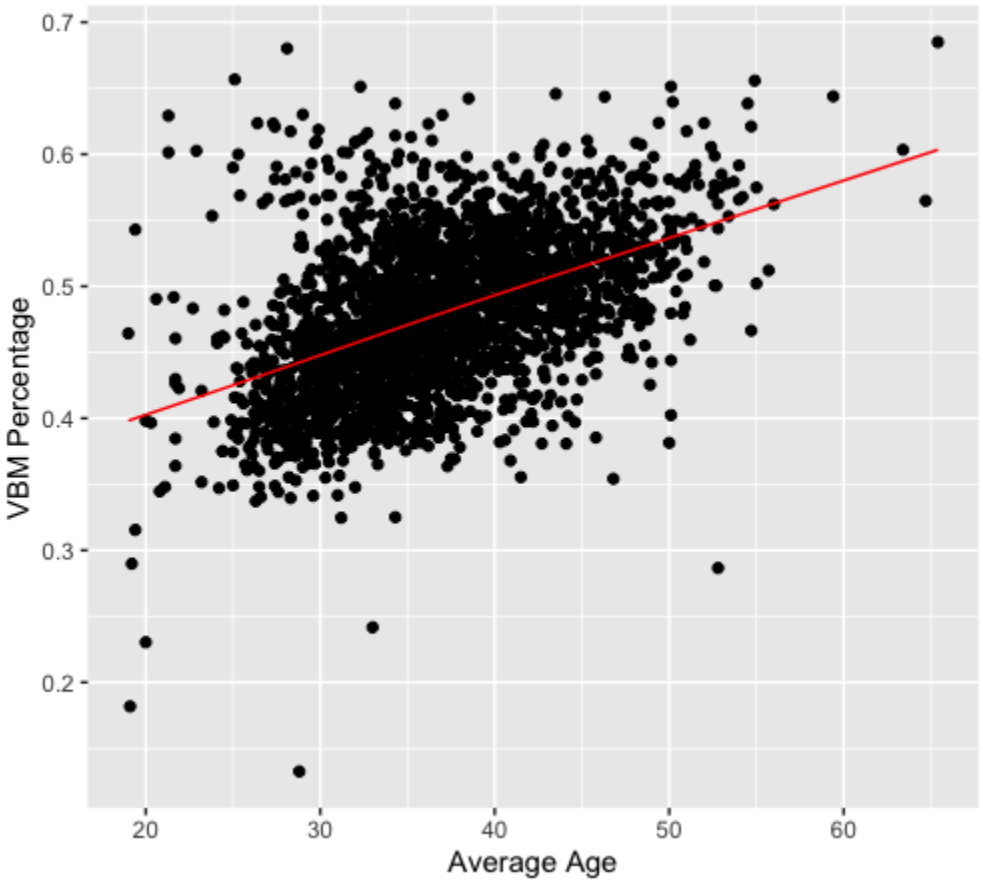

Figure 6.3: Univariate relationship plot of average age and vote by mail registration percentage

vote by mail registration.

This initial exploration supports the theory that vote by mail is predominantly being utilized by more affluent, older populations with higher levels of educational attainment. Census tracts with lower average incomes, lower educational attainment, lower rates of drivers, and younger populations have lower rates of registered voters who are registered to vote by mail.

A possible explanation for this is that people who intend on voting are switching to vote by mail for its convenience, but people who are registered to vote but do not intend to vote do no't see any point in taking the extra step to change their registration status. This would

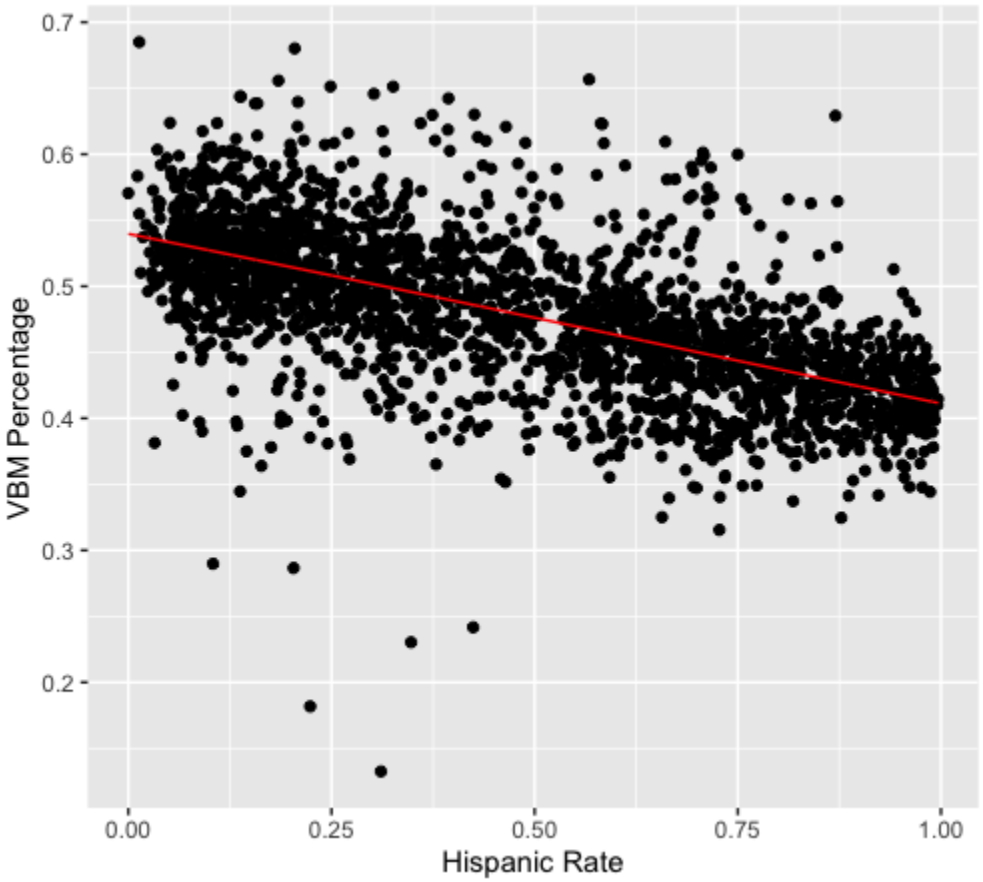

Figure 6.4: Univariate relationship plot of hispanic population rate and Vote By Mail registration percentage

explain why we see similar relationships between our predictor variables and vote by mail rate as we see between our predictor variables and voter turnout rates. Another explanation could be that certain populations are more suspicious of the vote by mail process. Whatever the explanation, if Los Angeles County wants to utilize vote by mail as a tool to expand the voting population, it needs to target outreach to communities that exhibit low turnout rates, which tend to be more impoverished communities.

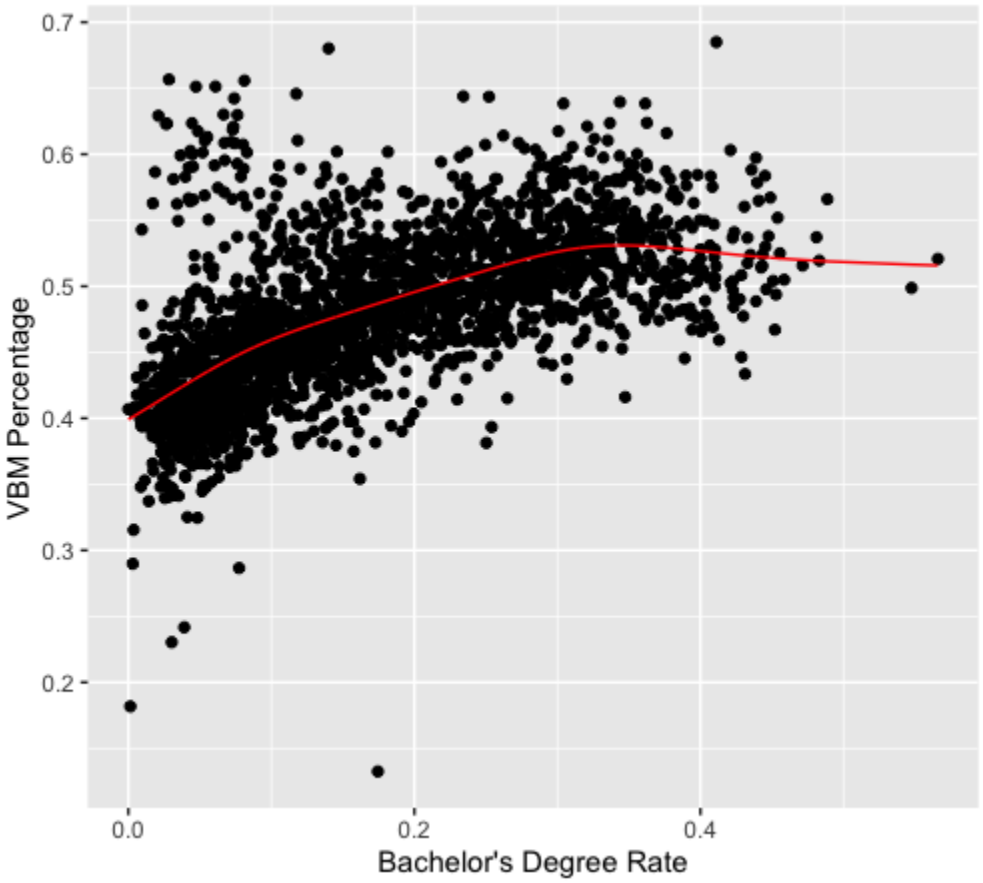

Figure 6.5: Univariate relationship plot of bachelors' degree attainment rate and vote by mail registration percentage

6.2 Forcasting Vote By Mail Registration Percentiles With GAMs

While the original goal of modeling this data was to predict vote by mail registration rates at the census tract level, out of training predictions proved to systematically under or over predict the vote by mail registration rate depending on the year.

Ideally, a multivariate time series approach could have been used to capture the trend of vote by mail registration but since reliable census tract level data only goes back four elections that approach was not a viable option. With this in mind, our first attempt was to build a model to predict vote by mail registration rates. The model was built using 2014 as the training data and a combination of demographic data points and a single lag variable (the

vote by mail registration rate from 2012). Then, the model was used to attempt to predict the 2018 vote by mail registration rates using 2016 as the lag variable. The results were disheartening, but not at all surprising; the model failed to capture the effect of the political shocks of 2016 and the increased voter activity that came after. This approach to modeling the data was taken because 2018 and 2014 were both midterm elections, so it seemed possible that the previous presidential election turnout combined with demographic data points would create a sufficiently good model to predict future voting activity.

In retrospect, the shortcomings of this approach may have been foreseeable; simply looking at the increase in vote by mail registration from 2012 to 2014 compared to the increase in vote by mail registration from 2016 to 2018, while being aware of the evolving political environment over those 6 years, one could assume that a model with only demographic and lag data would not likely capture the political shock of the election of Donald Trump. While the model did not do a good job of predicting absolute vote by mail registration rates, it did perform well when predicting relative vote by mail registration rates, answering the question: Where should we expect the highest and lowest rates of vote by mail registration?

6.3 Model Evaluation and Selection

As previously mentioned, the 2014 data was used with lags from 2012 to model vote by mail registration. The 2014 census tracts were divided into 75% training and 25% test data. Stepwise selection was used, in which a full model was fitted and insignificant variables were removed, comparing root mean squared error (RMSE), Akaike information criterion (AIC), and Bayesian information criterion (BIC) to select the model. Initially, the percentage of voters registered as vote by mail was used as the outcome variable.

The full model, using median age, median earnings, percent Hispanic, percent African American, percent white, percent Asian, percent living in poverty, percent with a bachelors' degree,

percent with a graduate degree, percent with a high school degree, percent of population that are children, and vote by mail percent from the preceding federal election as predictor variables, performed well in modeling the vote by mail percent at the census tract level in 2014. The RMSE for the test data was 0.02302, meaning that on average the predicted vote by mail percent for a census tract was only in error by 2.302%. When insignificant variables (percent white, percent Asian, and percent with a bachelors' degree) were stripped away, the resulting RMSE decreased to 2.3%, indicating that the full model was overfitting noise in the data and doing a worse job of generalizing outside of the training data. After multiple cycles of stripping away variables with the highest p-values, we obtained a model that simply used the single lag variable. This lag-only model achieved a RMSE of 2.4%. The lag only model has an R-squared of 0.816 compared to the full model, which has an R-squared of 0.836. When looking at BIC and AIC, the two criteria disagreed on which model was the best. The lag only model had the lowest BIC while the full model had the lowest AIC.

This leads us to the somewhat unsatisfying conclusion that a model using just the information from the previous election had nearly the same predictive power as a model that included the lag variable along with an array of demographic data points from the US Census. However, the true usefulness of the census data is not in its added predictive power, but instead in its descriptive power. Census data helps us answer the question: what are the demographics of census tracts that are adopting vote by mail and what are the demographics of the census tracts that are not adopting vote by mail? When the lag variable is removed, all of the demographic data points from the full model are significant other than the percent of population with a bachelors' degree, and the R-squared is 0.527. Adding the lag variable makes the demographic data points redundant in the model, but the lag variable alone does not provide us with much insight into what type of person is likely to vote by mail. It is important to look at both the univariate relationship plots and the final model to get a fuller picture of the breakdown of vote by mail voters and how they are distributed throughout LAC.

The major failing of this model is that it did not translate when looking at 2018. The relationship between the lag variable of 2014 and the VBM registration percent of 2014 was not consistent with that of the lag variable of 2018 and the VBM registration percent of 2018. Put simply, this model built from 2014 data systematically underpredicted vote by mail registration in 2018. The RMSE increased to 4.6% and nearly every prediction was an underestimate.

This clearly shows that demographic data alone fails to capture the current political environment, which is obviously a driving force for registration and turnout but is not easily measured by population parameters. The 2018 midterm election saw a 24% increase in turnout rate over the 2014 midterm, which is most likely a reflection of the response to the 2016 presidential election. This begs the question: if so much of registration and turnout hinges on variables that are not currently observable and have no historical reference points for the 2020 election (i.e., who the Democratic candidate will be or what the results of the impeachment investigation will be), what use can these models provide? The answer is that while these models do a poor job of predicting absolute values of vote by mail registration, they do a fairly good job of predicting ordinal values.

In other words, the model does a good job of answering the question: where do we expect to see the highest rates of vote by mail registration? even though it does a bad job of answering the question: what percent of registered voters in each census tract do we expect to be registered to vote by mail? This is a key outcome: understanding where the highest rates of vote by mail registration exist will provide a roadmap of how to best allocate resources for Los Angeles County.

When we changed the response variable to reflect the percentile of vote by mail registration, RMSE was only 0.046, which is to say that the previous vote by mail registration rate is a good indicator of what we will see in the next election. It is worth giving an example at this point to help clarify interpretation of what this model is predicting: If a given census tract

had the highest vote by mail registration percentage, it would be given a value of 1 because it is in the 100th percentile. If the model predicts 0.96 for that census tract the residual is 0.04.

The residuals are also i.i.d. normal with mean close to 0, so there is no systematic under- or over-prediction. The recent past is a good indicator of the future from a relative ranking standpoint, but not from an absolute vote by mail rate standpoint. We should look to the past election to understand where there will be larger vote by mail populations and thus smaller in-person turnout and we can look at census data to understand what these populations look like and what demographic groups and neighborhoods should be the focus of outreach for vote by mail registration.

CHAPTER 7

Voter Turnout Rates

When modelling voter turnout rates, we took the same approach as was used in modelling vote by mail registration percentage. However, for voter turnout rates, adding the census data into our model resulted in a significant improvement over simply using the lag of the previous election.

7.1 Smoothed Univariate Relationships Between Census Data and Voter Turnout Rates

We see in Figure 7.1 that as average yearly earnings increase, voter turnout percent also increases sharply up to an average yearly income of approximately $75,000; above this income, the increase in voter turnout percent tapers off. This is a similar relationship as was observed in vote by mail rates, but this relationship is more pronounced.

Unsurprisingly, Figure 7.2 shows that voter turnout rates sharply decline as poverty rates of a community increase. This is consistent with previous research that shows a positive correlation between one's financial security and one's probability of voting.

Figure 7.3 shows that older populations have higher turnout rates of registered voters than census tracts with younger average ages.

Similar to vote by mail registration, we see in Figure 7.4 a negative relationship between the

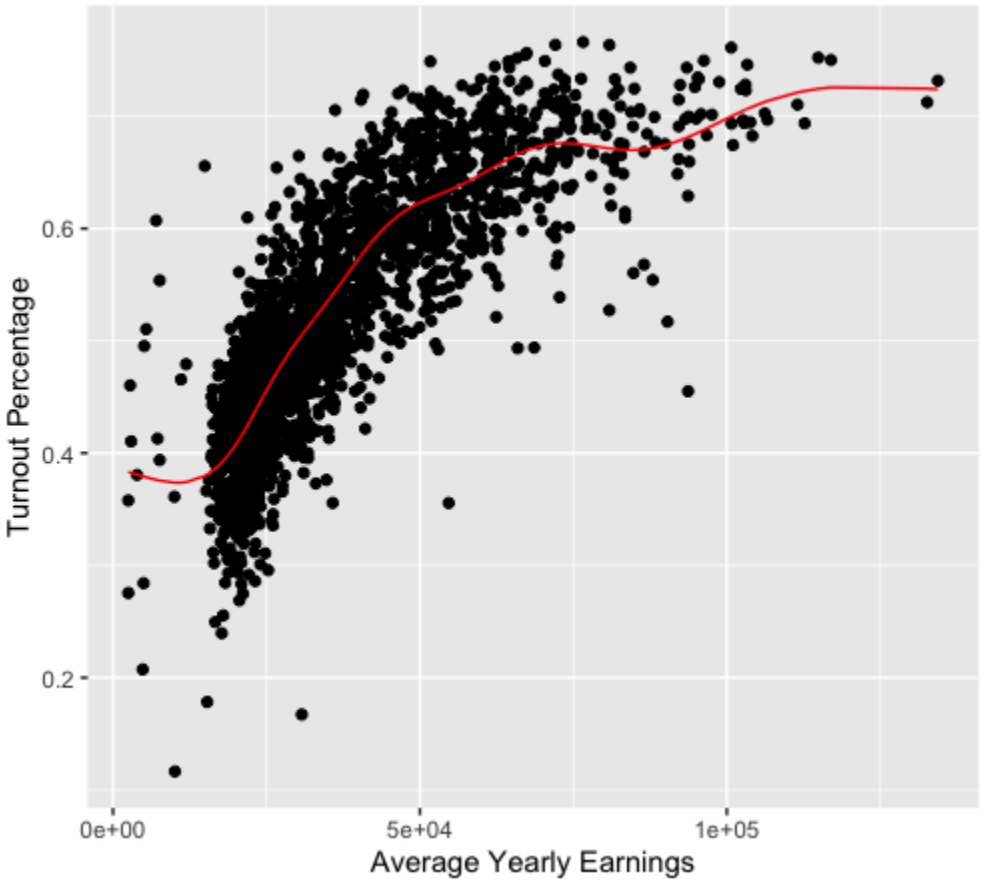

Figure 7.1: Univariate relationship plot of average income and voter turnout rate

density of Hispanic people in a census tract and the voter turnout rate in that census tract.

In Figures 7.5 and 7.6, we see that higher levels of educational attainment correspond to higher voter turnout rates.

Higher rates of children appear to correspond to higher turnout rates up to 10%; above this point, increased rates of children corresponds to lower turnout rates. This is consistent with the theory that finding childcare is a detriment to turning out to vote.

The univariate relationships between census tract data points and voter turnout rates are consistent with previous research in which older, more financially secure people who have attained higher levels of education turn out at higher rates than younger, less financially

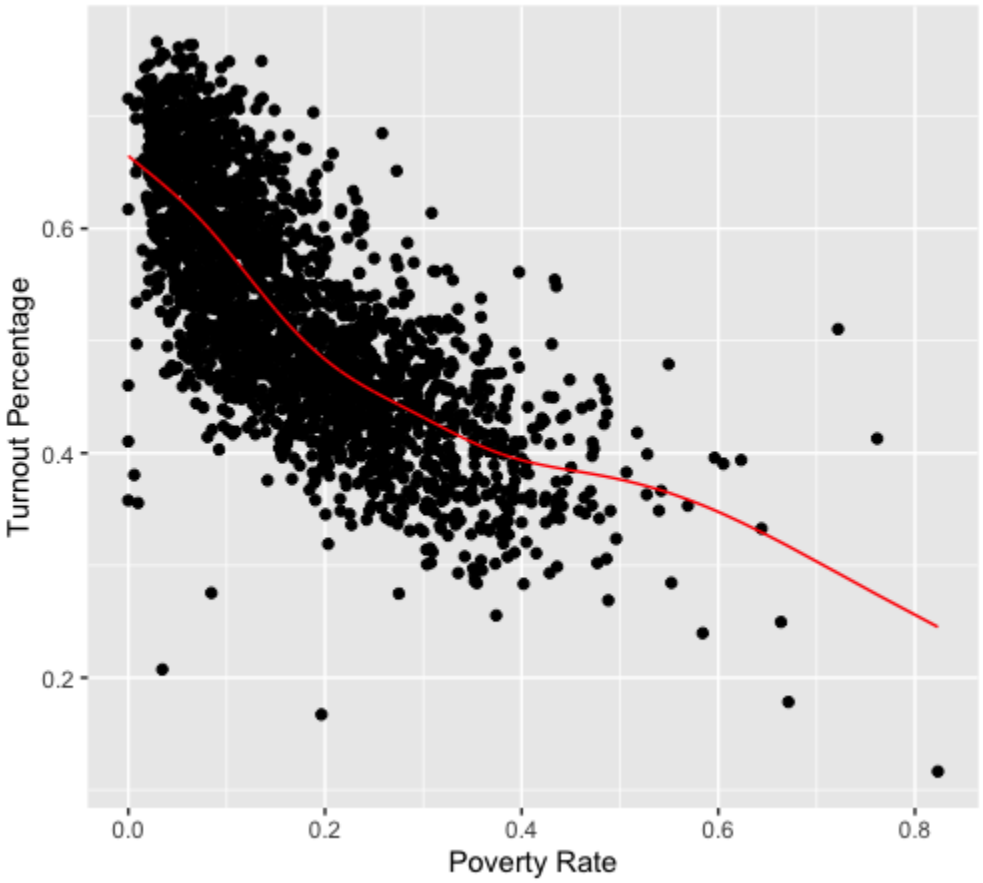

Figure 7.2: Univariate relationship plot of poverty rates and voter turnout rate

secure people who have not attained higher levels of education. It is worth noting that if we look at the correlation matrix between covariates, income is strongly positively correlated to both higher rates of bachelors' degree achievement and higher average ages. At this point a clear picture of the average Los Angeles voter begins to emerge: this voter is older and more financially secure than the average Angeleno, and as a result of their financial security they likely have access to better transportation and are less burdened by the one day voting sytem. These voters are also more likely to have college degrees and more likely to utilize vote by mail.

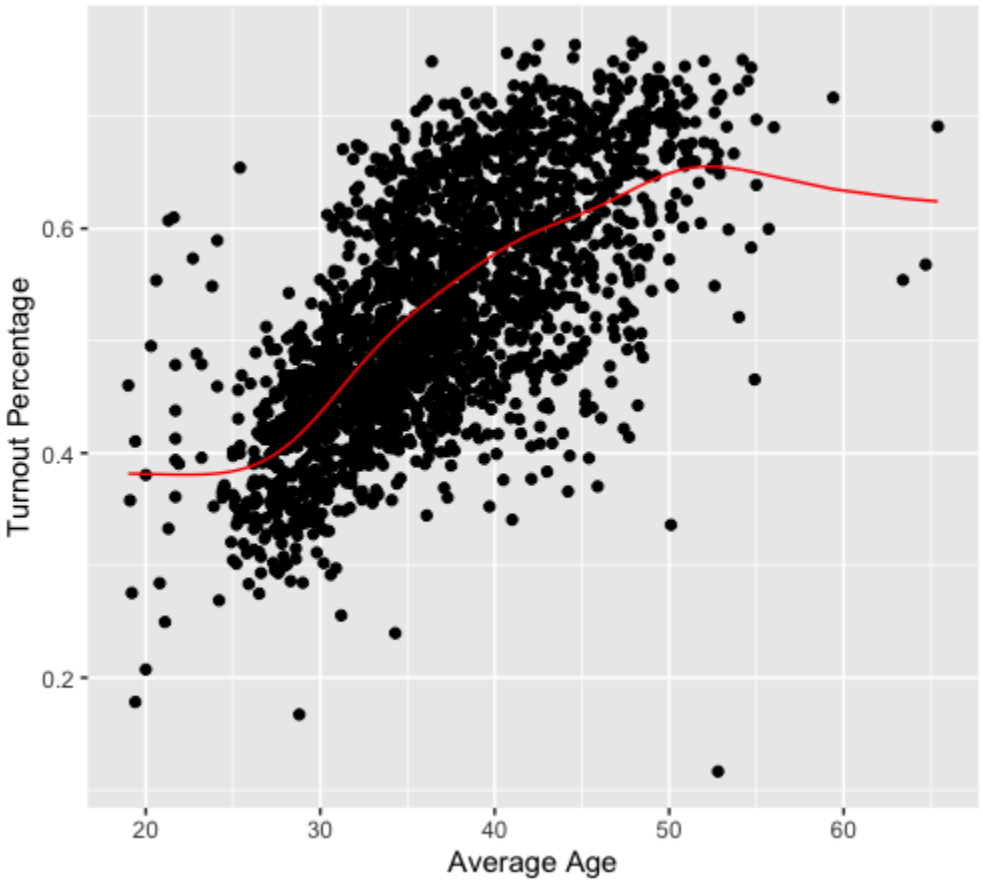

Figure 7.3: Univariate relationship plot of average age and voter turnout rate

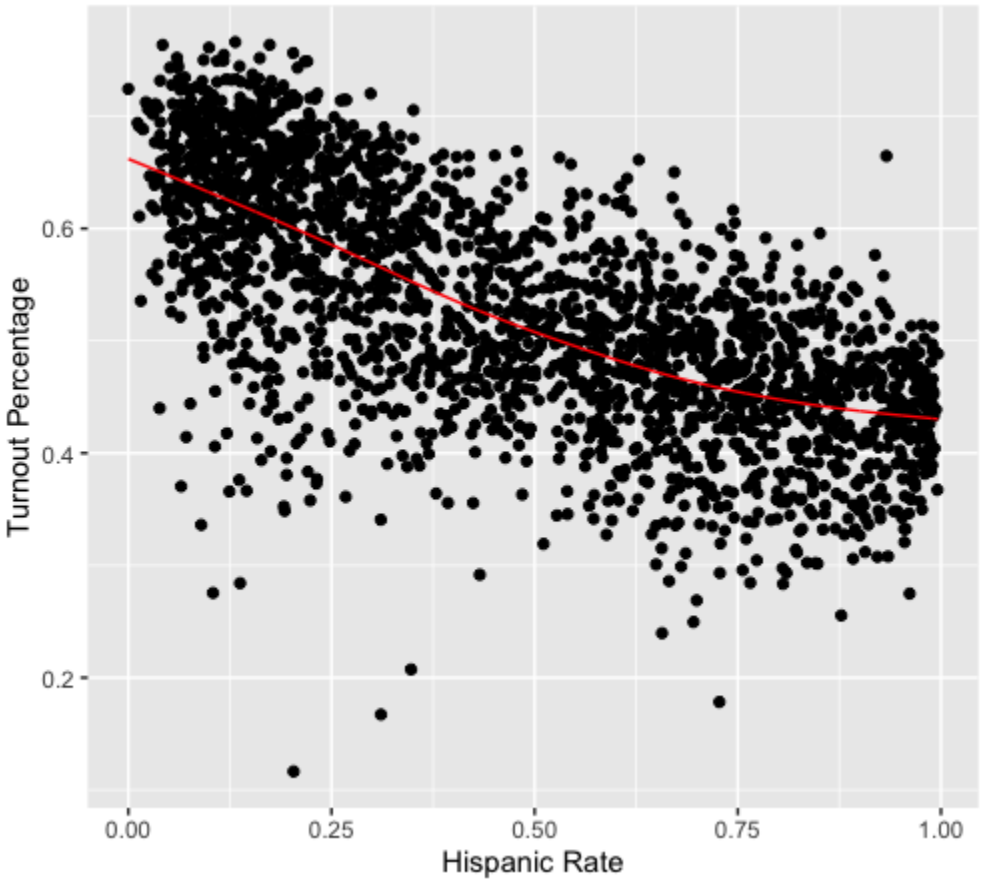

Figure 7.4: Univariate relationship plot of Hispanic population rate and voter turnout rate

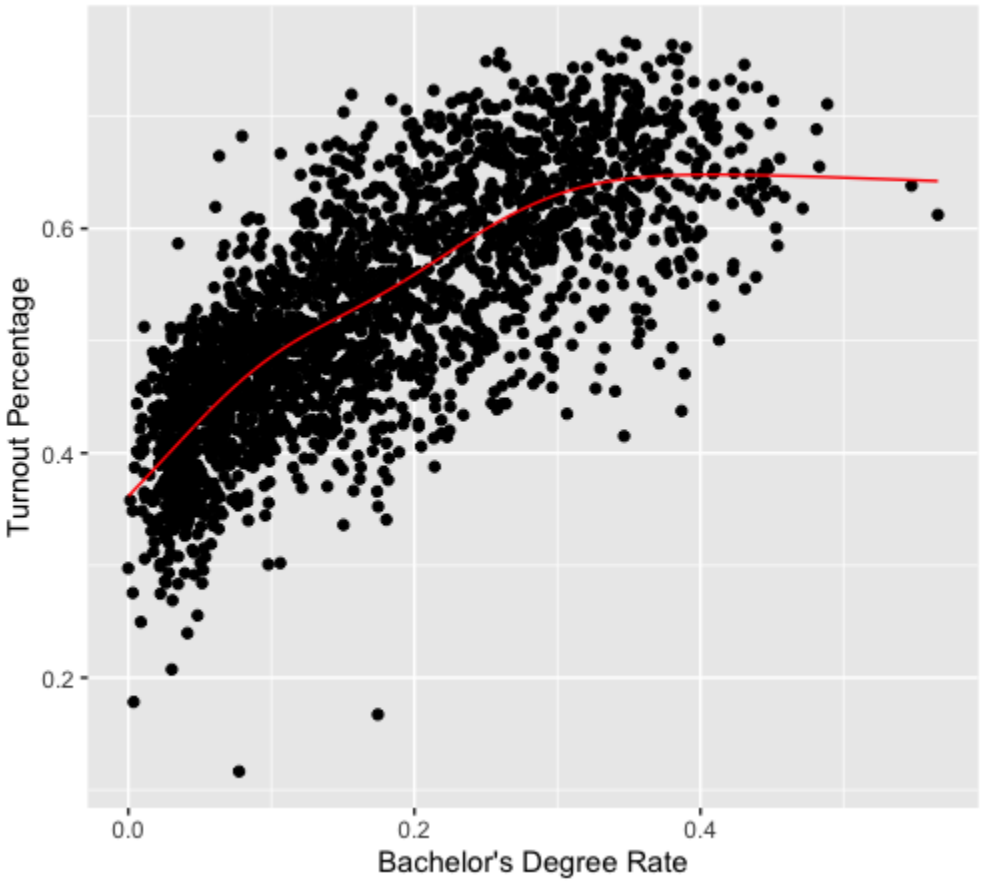

Figure 7.5: Univariate relationship plot of bachelors' degree attainment rate and voter turnout rate

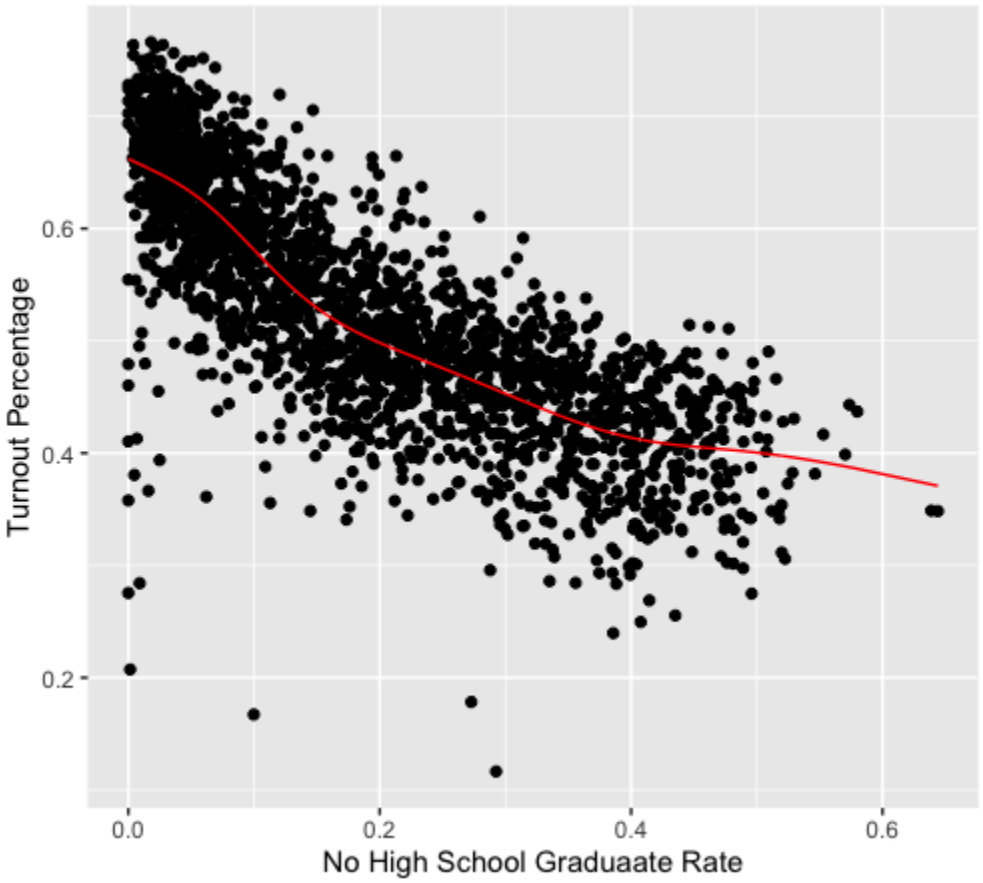

Figure 7.6: Univariate relationship plot of non-graduation from high school rate and voter turnout rate

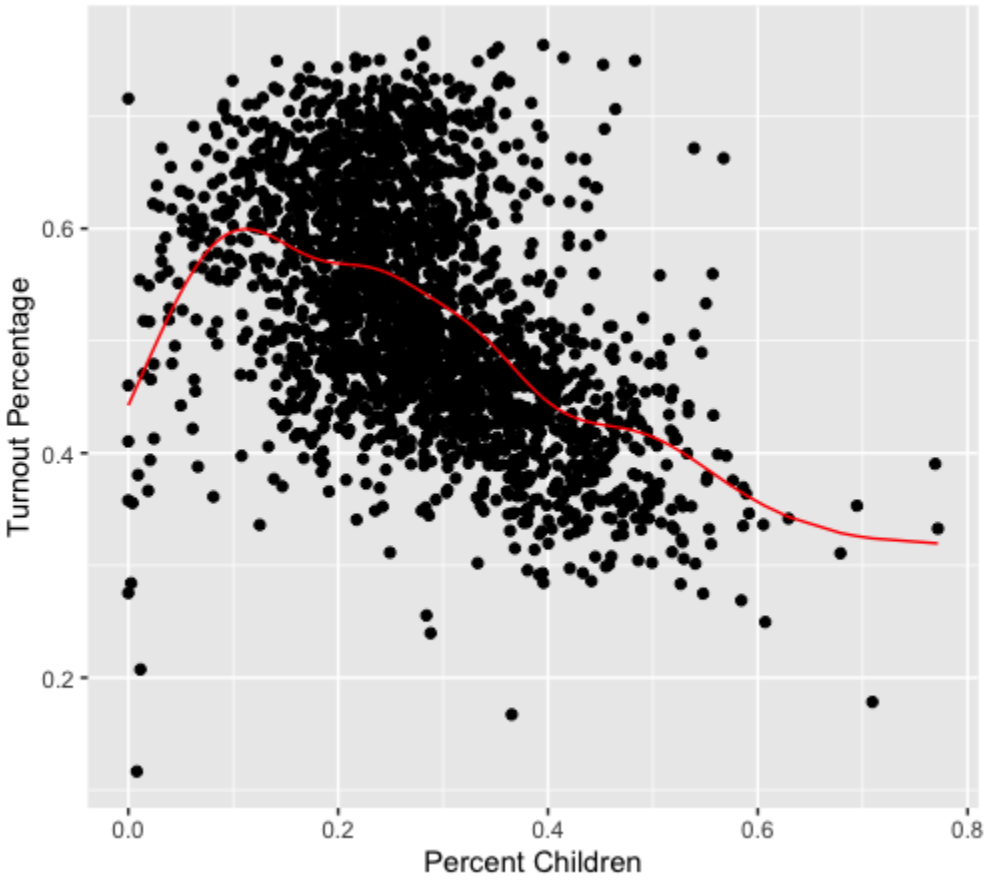

Figure 7.7: Univariate relationship plot of percentage of population that are children and voter turnout rate

7.2 Forcasting Voter Turnout Percentiles With GAMs

When building a model for turnout percent we once again took the approach of modelling 2014 turnout rates using age, earnings, Hispanic percent, African American percent, white percent, poverty percent, Asian percent, bachelors' degree percent, graduate school percent, high school graduation percent, no high school graduation percent, child percent, and turnout percent from the previous federal election. The RMSE of this full model was only 0.02517. We removed insignificant variables in a stepwise fashion and compared AIC, BIC, and RMSE. The result was that many redundant variables were removed (i.e., there were several variables that reflected educational attainment). The final model chosen utilized age, Hispanic percent, African American percent, poverty percent, Asian percent, bachelors' degree percent, graduate degree percent, child percent, and 2012 turnout percent. The final RMSE for the test data was .02519.

However, we once again see that while this approach does a good job of describing the past, it catastrophically fails when faced with shocks that cannot be accounted for in census and voter data. When we use the same model to predict 2018 turnout, the RMSE skyrockets to 0.2353 and systematically underpredicts the turnout rate.

This outcome led us back to taking an ordinal approach to modelling the data. We transformed the response variable from turnout percentage to turnout percentage percentile and we achieved results with normally distributed residuals and respectable RMSE of 0.09. The difference between modelling turnout percentage percentiles and vote by mail percentage percentiles is that in the case of turnout, census data had a meaningful impact on predicting future turnout. When we used the lag only model to predict turnout percentiles the RMSE went up to 0.098.

This result is encouraging because it shows that there are both descriptive and predictive benefits in joining census tract data to voter data. Demographic data, to an extent, not only

captures the relationship between demographics and turnout rates at a static point in time, but also captures information about changes in turnout rates from one election to the next.

CHAPTER 8

Projection Maps

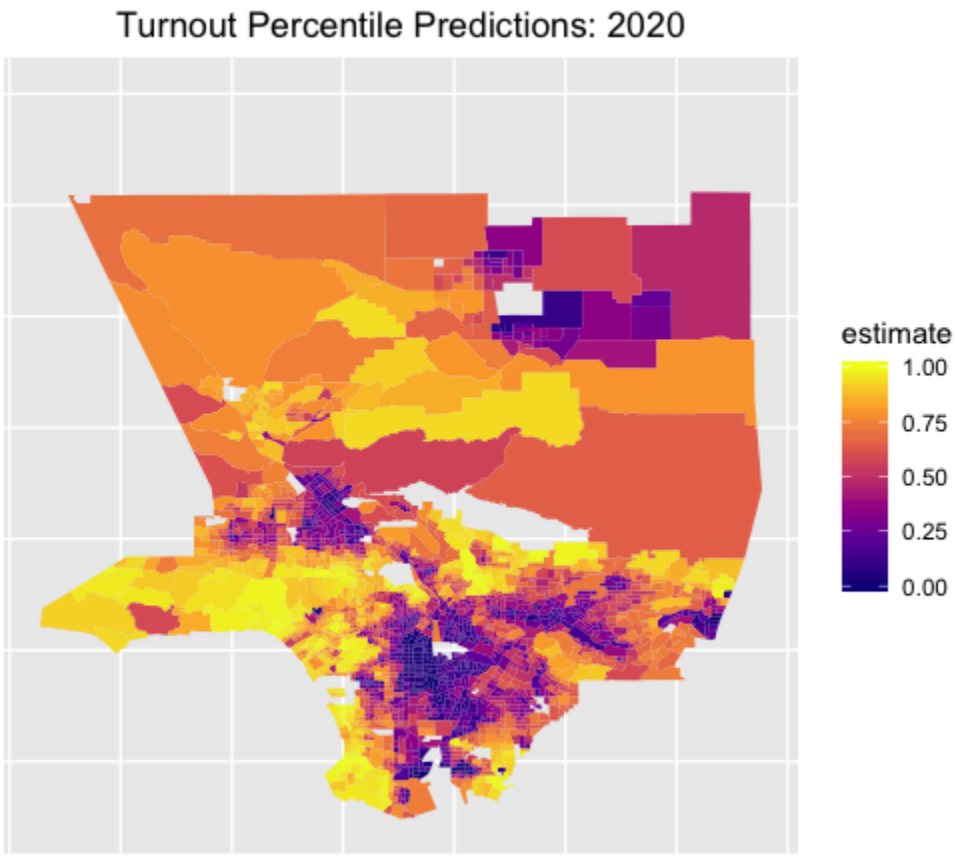

Figure 8.1: Projected turnout rate percentiles for 2020

When looking at our projected percentiles for vote by mail registration rate in Figure 8.2 and voter turnout rate in Figure 8.1 we see two maps that look fairly similar. This indicates that there is a positive correlation between being registered to vote by mail and one's probability of voting.

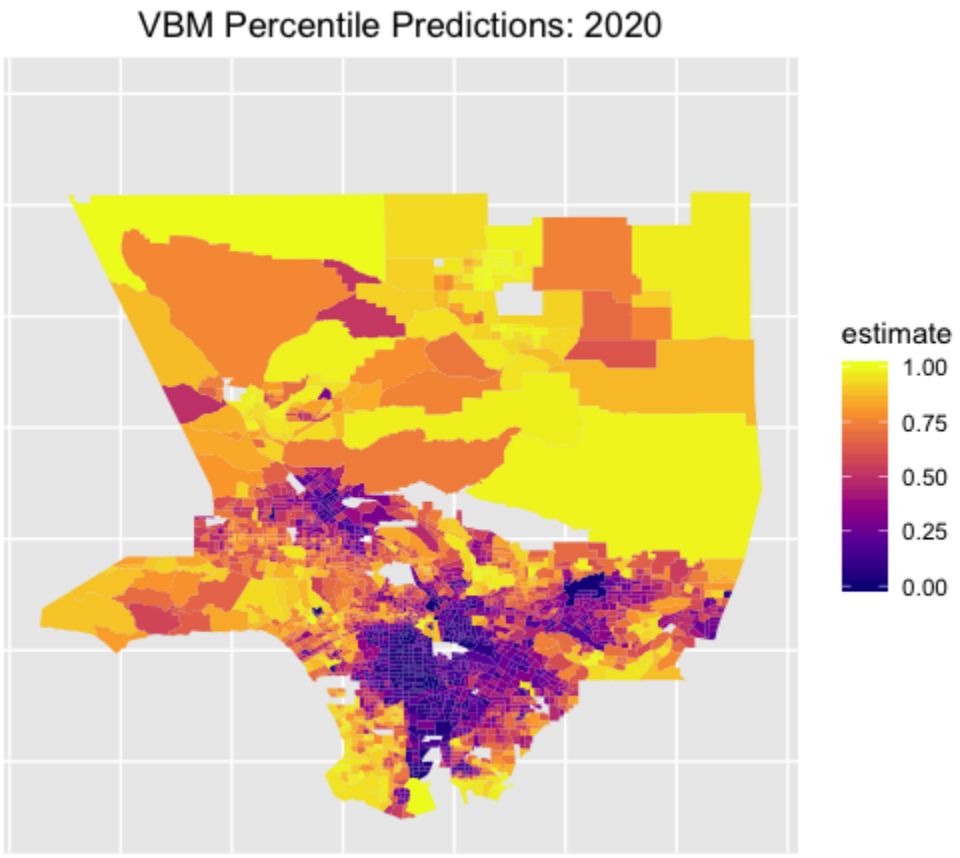

Figure 8.2: Projected vote by mail registration rate percentiles for 2020

When simply plotting the percentage of voters registered to vote by mail in a given census tract against the turnout percent for that census tract in Figure 8.3, we see that as vote by mail percentage increases so does the expected turnout percentage for that census tract. There are two possible interpretations of this: the first being that vote by mail is making it easier for people to vote and thus resulting in higher turnout, or, alternatively, people who are already likely to vote seek out easier ways to vote. Based on our interpretation of the univariate smoothed relationships presented in this paper it seems that the latter explanation is more likely.

Consistent with the earlier studies, we found that census tracts made up of financially secure, older people with higher levels of educational attainment were more likely to have

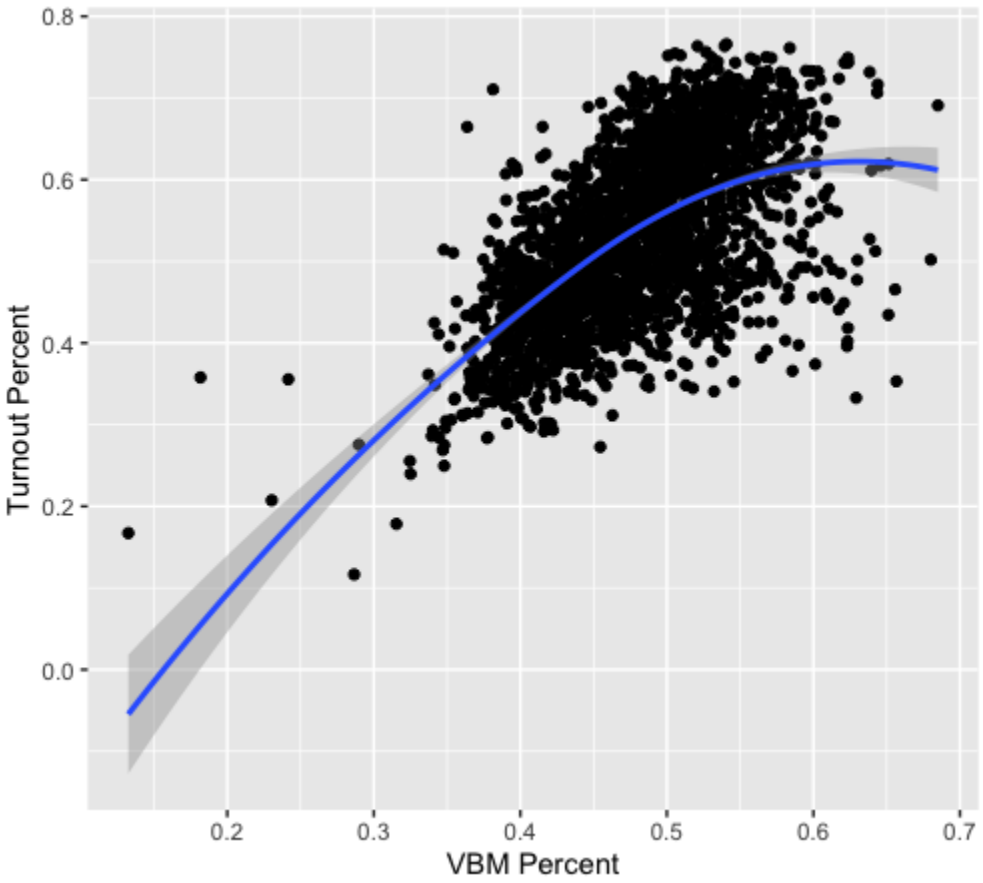

Figure 8.3: Smoothed relationship between vote by mail registration rates and voter turnout rate

higher turnout and vote by mail rates. Looking at our forecast maps for 2020 we see densely populated areas such as south and east Los Angeles that have both low vote by mail rates and low turnout rates. These are the neighborhoods that Los Angeles County should focus outreach on as potential areas to expand voter participation. These neighborhoods tend to be younger, more densely populated, and lower income neighborhoods compared to the rest of Los Angeles County. Also worth noting is that neighborhoods with higher percentages of Hispanic individuals have lower vote by mail registration and voter turnout.

From a preparation and outreach standpoint, Los Angeles County should be concerned about these more densely populated, poorer areas of South LA and East LA. These areas represent

the largest potential in expanding the voting population both in number and in diversity. Reaching out to the young, diverse, less financially secure Angelenos represents an exciting opportunity to have a more representative electorate in Los Angeles County, but it also could spell trouble from a logistical standpoint on election night. This is because studies have shown that people that are already likely to vote are the ones that turn out early. This paper has shown that people that are already likely to vote are more likely to be registered to vote by mail. In other words, getting more people to vote in south and east Los Angeles would also be expanding the population that is likely to show up in person on election night. Los Angeles County needs to make a concerted outreach effort to make these neighborhoods aware of the benefits of turning out to vote early. If this is not done, it is easy to imagine that these voters show up in unprecedented numbers based on the massive turnout rates seen in 2018 across all of Los Angeles County. If this occurs, there will be bottlenecks that disproportionately impact low income neighborhoods, which would be seen as an unequivocal failure in VSAP's rollout.

CHAPTER 9

Conclusion

In this paper, we have shown the power of joining census data to aggregated voter data in helping us understand the driving forces behind Los Angeles County voting behavior. We were able to visualize and understand the relationships between the demographics of a census tract and the voter turnout for that census tract.

We were able to build descriptive models that had low RMSE, but the models failed to predict the actual voter turnout percentage for years that they were not trained on. Put simply, voter data and census tract data did a good job of predicting which census tract would have high or low turnout compared to other census tracts, but failed to predict the magnitude of the turnout due to their inability to capture the effects of political shocks. Yhis paper has successfully laid out a new methodology for building descriptive models of voter behavior using census data and GAMs; still, the failure of current modeling approaches to capture political shocks is significant. In future work, the effects of political shocks must be factored in to truly build accurate models of absolute voter turnout moving forward.

While we were only able to attain aggregate data for use in this study, Los Angeles County has individual level data that can be used to model the likelihood of an individual turning out to vote. As of now, Los Angeles County does not keep demographic data on individual voters, but much can be discerned from just one's name. The R package "gender" uses a large historical dataset of state recorded names and genders to allow the user to infer the gender of an individual based purely from their name. The R package "wru" takes a similar approach to ethnicity inference. Using these methods, along with joining in census data

using the R package "tidycensus", would give Los Angeles County a rich understanding of an individual's demographic makeup. Combining these factors with the individual's voting history would allow Los Angeles County to build models to accurately predict the likelihood of an individual turning out to vote.

The most difficult part of the problem of predicting voter turnout is understanding and quantifying which candidates will generate higher turnout and by how much. In Los Angeles County - and in California at large - a Democrat will, with near certainty, win the state in the 2020 presidential election, but it is easy to imagine a large difference in voter turnout rates depending on the winner of the Democratic primary. It is also easy to imagine large numbers of voters showing up regardless of who the winner of the Democratic primary is simply because of the extremely polar views of the current president. In order to set bounds of expected turnout, polling should be performed to estimate the lower and upper bounds of expected turnout for different Democratic candidates. This information, combined with a rich data set of individual and census level demographic data will give Los Angeles County the best chance to accurately forecast absolute turnout in 2020 and best prepare themselves for the strains that will be placed on VSAP in its inaugural election.

Appendix A

R Code

A.1 Clean and Merge Data

```
##########################################################
#This script is for collecting, cleaning, and joining data
##########################################################



##########################################################
#Load libraries
##########################################################

library(dplyr)
library(tidyr)
library(stringr)
library(readr)
library(readxl)
library(tidycensus)
library(data.table)
library(ggplot2)

################################################
#Load Data
################################################
#Voting stats at a census tract level
setwd("~/Google Drive/masters thesis copy")
CT_votes <- read_csv("Paul_thesis.csv")
```

```
26
27 #Recode election ID
28 CT_votes <- CT_votes %>%
29   mutate(election_type = recode(as.character(election_id), "791" = "
     Presidential",
30                                 "905" = "Midterm",
31                                 "3496" = "Presidential",
32                                 "3861" = "Midterm")) %>%
33   mutate(year = ifelse(election_id == 791, 2012,
34                        ifelse(election_id == 905, 2014,
35                               ifelse(election_id == 3496, 2016,
36                                      ifelse(election_id == 3861, 2018, NA)
     )))) %>%
37   #recode CT_votes party to DEM, REP, NPP, OTHER
38   mutate(party = ifelse(party == "DEM" | party == "REP" | party == "NPP",
     party, "Other")) %>%
39   group_by(election_id, ctract, perm_category, party, election_type, year)
       %>%
40   dplyr::summarise(total.reg = sum(total.reg, na.rm = T), voted = sum(
     voted, na.rm = T))
41
42
43 ################################################
44 #Explore NAs:Some precinct ids could not map to census tracts
45 ################################################
46
47 CT_NAs <- CT_votes %>%
48   filter(is.na(ctract))
49 #we see that data collection has improved over time and most every voter
     can now be mapped to a
50 #ct
51
52 ########################################
53 #Report summary stats
54 ########################################
```

```
#Voting Percentage by party
View(CT_votes %>%
       mutate(party_recode = ifelse(party == "DEM" | party == "REP" |
   party == "NPP", party, "other")) %>%
       group_by(year, party_recode) %>%
       dplyr::summarise(votes = sum(voted), registered = sum(total.reg),
   percent_voted = votes/registered))
#There was a huge leap in voter turnout for a midterm for 2018 with
   turnout being more similar to a presidential
#election than a midterm

#Voting percentage by PERM vs non-PERM
View(CT_votes %>%
       group_by(year,perm_category) %>%
       dplyr::summarise(votes = sum(voted), registered = sum(total.reg),
   percent_voted = votes/registered))
#PERM voters vote turnout rate is about 15% higher across election when
   compared to non-perm voters

#######################################
#Spread data
#######################################
#2018
CT_votes_2018 <- CT_votes %>%
  filter(year == 2018) %>%
  unite("party_count", c(perm_category, party), sep = "-") %>%
  select(ctract, party_count, total.reg, voted)

CT_votes_2018 <- dcast(setDT(CT_votes_2018),  ctract ~ paste0(party_count)
   , value.var = c("total.reg", "voted"), sep = "-") %>%
  replace(is.na(.), 0)

#########################################
#2016
CT_votes_2016 <- CT_votes %>%
```

```
  filter(year == 2016) %>%
  unite("party_count", c(perm_category, party), sep = "-") %>%
  select(ctract, party_count, total.reg, voted)

CT_votes_2016 <- dcast(setDT(CT_votes_2016),  ctract ~ paste0(party_count)
    , value.var = c("total.reg", "voted"), sep = "-") %>%
  replace(is.na(.), 0)


################################################
#2014
CT_votes_2014 <- CT_votes %>%
  filter(year == 2014) %>%
  unite("party_count", c(perm_category, party), sep = "-") %>%
  select(ctract, party_count, total.reg, voted)

CT_votes_2014 <- dcast(setDT(CT_votes_2014),  ctract ~ paste0(party_count)
    , value.var = c("total.reg", "voted"), sep = "-") %>%
  replace(is.na(.), 0)



################################################
#2012
CT_votes_2012 <- CT_votes %>%
  filter(year == 2012) %>%
  unite("party_count", c(perm_category, party), sep = "-") %>%
  select(ctract, party_count, total.reg, voted)

CT_votes_2012 <- dcast(setDT(CT_votes_2012),  ctract ~ paste0(party_count)
    , value.var = c("total.reg", "voted"), sep = "-") %>%
  replace(is.na(.), 0)

#do this after joining census data
CT_votes_2012 <- CT_votes_2012 %>%
```

```
  mutate(total_regs = rowSums(CT_votes_2012[,2:9]))  %>%
  mutate(total_votes = rowSums(CT_votes_2012[,10:17]))


##############################################
#Read in census data
##############################################

#API access
#http://api.census.gov/data/key_signup.html
census_api_key("b28ca17c1a5ffac1d0c7464d4df0ffa6a912ea7e", install = TRUE,
    overwrite = T)
#Variable lookup 2017
v17 <- load_variables(2017, "acs5", cache = TRUE)
View(v17)
#variable lookup 2012
v12 <- load_variables(2012, "acs5", cache = TRUE)
View(v17)

#2017 variables of interest

ct_data <- get_acs(geography = "tract", state = "CA", county = "Los
   Angeles County",
                   variables = c(population = "B01003_001",
                                 #transportation
                                 #how many people take public
   transportation
                                 public_transportation = "B08006_008",
                                 #how many people drive to work
                                 drivers = "B08006_002",
                                 #how many people bike to work
                                 bike = "B08006_014",
                                 #poverty
                                 #below poverty
                                 poverty = "B06012_002",
```

```
                                #1-1.5x poverty
                                poverty_1.5x = "B06012_003",
                                #1.5x or above poverty
                                poverty_2x = "B06012_004",
                                #education
                                #no highschool diploma
                                edu_no_highschool = "B06009_002",
                                #highschool grad
                                edu_highschool = "B06009_003",
                                #bachelors degree
                                edu_bachelors = "B06009_005",
                                #grad school
                                edu_graduate = "B06009_006",
                                #employed 16 and older and in work force
                                employed = "B23025_004",
                                unemployed = "B23025_005",
                                total_labor_force = "B23025_002",
                                #race or ethnicity
                                african_american = "B02009_001",
                                european_american = "C02003_003",
                                asian_american = "C02003_006",
                                native_american = "C02003_005",
                                pacific_islander = "C02003_007",
                                hispanic_latino = "B03002_012",
                                #gender
                                total_male = "B01001_002",
                                total_female = "B01001_026",
                                #age
                                age_median = "B01002_001",
                                #earnings
                                earnings_median = "B24081_001",
                                #childcare
                                under_6 = "B05009_002",
                                over_6_under_18 = "B05009_020",
                                male_under_5 = "B01001_003",
```

```
                                male_5_9 = "B01001_004",
                                male_10_14 = "B01001_005",
                                male_15_17 = "B01001_006",
                                female_under_5 = "B01001_027",
                                female_5_9 = "B01001_028",
                                female_10_14 = "B01001_029",
                                female_15_17 = "B01001_030"),
                  year = 2017,
                  geometry = TRUE,
                  cb = F)

#2012 variables of interest
ct_data_2012 <- get_acs(geography = "tract", state = "CA", county = "Los
    Angeles County",
                  variables = c(population = "B01003_001",
                                #transportation
                                #how many people take public
    transportation
                                public_transportation = "B08006_008",
                                #how many people drive to work
                                drivers = "B08006_002",
                                #how many people bike to work
                                bike = "B08006_014",
                                #poverty
                                #below poverty
                                poverty = "B06012_002",
                                #1-1.5x poverty
                                poverty_1.5x = "B06012_003",
                                #1.5x or above poverty
                                poverty_2x = "B06012_004",
                                #education
                                #no highschool diploma
                                edu_no_highschool = "B06009_002",
                                #highschool grad
                                edu_highschool = "B06009_003",
```

```
                                #bachelors degree
                                edu_bachelors = "B06009_005",
                                #grad school
                                edu_graduate = "B06009_006",
                                #employed 16 and older and in work force
                                employed = "B23025_004",
                                unemployed = "B23025_005",
                                total_labor_force = "B23025_002",
                                #race or ethnicity
                                african_american = "B02009_001",
                                european_american = "C02003_003",
                                asian_american = "C02003_006",
                                native_american = "C02003_005",
                                pacific_islander = "C02003_007",
                                hispanic_latino = "B03002_012",
                                #gender
                                total_male = "B01001_002",
                                total_female = "B01001_026",
                                #age
                                age_median = "B01002_001",
                                #earnings
                                earnings_median = "B24081_001",
                                #childcare
                                under_6 = "B05009_002",
                                over_6_under_18 = "B05009_020",
                                male_under_5 = "B01001_003",
                                male_5_9 = "B01001_004",
                                male_10_14 = "B01001_005",
                                male_15_17 = "B01001_006",
                                female_under_5 = "B01001_027",
                                female_5_9 = "B01001_028",
                                female_10_14 = "B01001_029",
                                female_15_17 = "B01001_030"),
                    year = 2012,
                    geometry = TRUE)
```

```
251
252
253
254
255 ##############################################
256 #clean census data to join to voter data
257 ##############################################
258
259
260 ct_data_clean <- ct_data %>%
261   select(-moe) %>%
262   spread(key = variable, value = estimate) %>%
263   #strip text from census number
264   mutate(ct_number = gsub("[^0-9]","",NAME)) %>%
265   #add 00 to end of 4 digit numbers to match with county data
266   mutate(ct_number = ifelse(nchar(ct_number) == 4, paste0(ct_number,"00"),
      ct_number)) %>%
267   #change to numeric
268   mutate(ct_number = as.numeric(ct_number)) %>%
269   #reorder columns
270   #change these numbers when you add variable from census data
271   select(37,1,3:36,38)
272
273 ct_data_2012_clean <- ct_data_2012 %>%
274   select(-moe) %>%
275   spread(key = variable, value = estimate) %>%
276   #strip text from census number
277   mutate(ct_number = gsub("[^0-9]","",NAME)) %>%
278   #add 00 to end of 4 digit numbers to match with county data
279   mutate(ct_number = ifelse(nchar(ct_number) == 4, paste0(ct_number,"00"),
      ct_number)) %>%
280   #change to numeric
281   mutate(ct_number = as.numeric(ct_number)) %>%
282   #reorder columns
283   #change these numbers when you add variable from census data
```

```
  select(37,1,3:36,38)
################################################
#model data
#Join census data to 2018 voter data
all_data_2018 <- CT_votes_2018 %>%
  left_join(ct_data_clean, by = c("ctract" = "ct_number")) %>%
  mutate(ctract = ifelse(ctract == 0, NA, ctract)) %>%
  mutate(ctract = as.factor(ctract)) %>%
  mutate(total.reg_PERM = rowSums(CT_votes_2018[,6:9])) %>%
  mutate(total.reg_NON_PERM = rowSums(CT_votes_2018[,2:6])) %>%
  mutate(total_regs = rowSums(CT_votes_2018[,2:9]))  %>%
  mutate(total.votes_PERM = rowSums(CT_votes_2018[,14:17])) %>%
  mutate(total.votes_NON_PERM = rowSums(CT_votes_2018[,10:13])) %>%
  mutate(total_votes = rowSums(CT_votes_2018[,10:17])) %>%
  mutate(turnout_percent = total_votes/total_regs) %>%
  mutate(of_voting_age = population -  (female_10_14 + female_15_17 +
   female_under_5 + female_5_9 +
                                          male_10_14 + male_15_17 + male_
   under_5 + male_5_9)) %>%
  mutate(reg_percent = total_regs/of_voting_age)

#Join census data to 2016 voter data
all_data_2016 <- CT_votes_2016 %>%
  left_join(ct_data_clean, by = c("ctract" = "ct_number")) %>%
  mutate(ctract = ifelse(ctract == 0, NA, ctract)) %>%
  mutate(ctract = as.factor(ctract)) %>%
  mutate(total.reg_PERM = rowSums(CT_votes_2016[,6:9])) %>%
  mutate(total.reg_NON_PERM = rowSums(CT_votes_2016[,2:6])) %>%
  mutate(total_regs = rowSums(CT_votes_2016[,2:9]))  %>%
  mutate(total.votes_PERM = rowSums(CT_votes_2016[,14:17])) %>%
  mutate(total.votes_NON_PERM = rowSums(CT_votes_2016[,10:13])) %>%
  mutate(total_votes = rowSums(CT_votes_2016[,10:17])) %>%
  mutate(turnout_percent = total_votes/total_regs) %>%
  mutate(of_voting_age = population -  (female_10_14 + female_15_17 +
   female_under_5 + female_5_9 +
```

```
                                          male_10_14 + male_15_17 + male_
    under_5 + male_5_9)) %>%
  mutate(reg_percent = total_regs/of_voting_age)


#Join census data to 2014 voter data
all_data_2014 <- CT_votes_2014 %>%
  left_join(ct_data_2012_clean, by = c("ctract" = "ct_number")) %>%
  mutate(ctract = ifelse(ctract == 0, NA, ctract)) %>%
  mutate(ctract = as.factor(ctract)) %>%
  mutate(total.reg_PERM = rowSums(CT_votes_2014[,6:9])) %>%
  mutate(total.reg_NON_PERM = rowSums(CT_votes_2014[,2:6])) %>%
  mutate(total_regs = rowSums(CT_votes_2014[,2:9]))  %>%
  mutate(total.votes_PERM = rowSums(CT_votes_2014[,14:17])) %>%
  mutate(total.votes_NON_PERM = rowSums(CT_votes_2014[,10:13])) %>%
  mutate(total_votes = rowSums(CT_votes_2014[,10:17])) %>%
  mutate(turnout_percent = total_votes/total_regs) %>%
  mutate(of_voting_age = population -  (female_10_14 + female_15_17 +
    female_under_5 + female_5_9 +
                                          male_10_14 + male_15_17 + male_
    under_5 + male_5_9)) %>%
  mutate(reg_percent = total_regs/of_voting_age)

#Join census data to 2014 voter data
all_data_2012 <- CT_votes_2012 %>%
  left_join(ct_data_2012_clean, by = c("ctract" = "ct_number")) %>%
  mutate(ctract = ifelse(ctract == 0, NA, ctract)) %>%
  mutate(ctract = as.factor(ctract)) %>%
  mutate(total.reg_PERM = rowSums(CT_votes_2012[,6:9])) %>%
  mutate(total.reg_NON_PERM = rowSums(CT_votes_2012[,2:6])) %>%
  mutate(total_regs = rowSums(CT_votes_2012[,2:9]))  %>%
  mutate(total.votes_PERM = rowSums(CT_votes_2012[,14:17])) %>%
  mutate(total.votes_NON_PERM = rowSums(CT_votes_2012[,10:13])) %>%
  mutate(total_votes = rowSums(CT_votes_2012[,10:17])) %>%
  mutate(turnout_percent = total_votes/total_regs) %>%
```

```
  mutate(of_voting_age = population -  (female_10_14 + female_15_17 +
   female_under_5 + female_5_9 +
                                          male_10_14 + male_15_17 + male_
   under_5 + male_5_9)) %>%
  mutate(reg_percent = total_regs/of_voting_age)


################################################
#LA almanac turnout data
################################################
turnoutdata <- read_excel("turnoutdata.xlsx") %>%
  #get order right by replacing Primary with "B"
  mutate(order_reference = gsub("Primary", "B", Election)) %>%
  mutate(election_year = gsub("[^0-9]+", "", Election)) %>%
  arrange(order_reference) %>%
  mutate(election_type = c(rep(c("Primary: Midterm", "Midterm", "Primary:
   Presidential", "Presidential"), 7),
                           c("Primary: Midterm", "Midterm"))) %>%
  mutate(`Total Votes Cast` = as.numeric(gsub(",","",`Total Votes Cast`)))

################################################
#Data formatted for mapping
################################################
#map friendly data
vote_data_2018 <- all_data_2018 %>%
  select(c(1:17,54:62)) %>%
  mutate(ctract = as.numeric(as.character(ctract)))

vote_data_2016 <- all_data_2016 %>%
  select(c(1:17,54:62)) %>%
  mutate(ctract = as.numeric(as.character(ctract)))

vote_data_2014 <- all_data_2014 %>%
  select(c(1:17,54:62)) %>%
  mutate(ctract = as.numeric(as.character(ctract)))
```

```
380
381 vote_data_2012 <- all_data_2012 %>%
382   select(c(1:17,54:62)) %>%
383   mutate(ctract = as.numeric(as.character(ctract)))
384
385 map_2018 <- ct_data_clean %>%
386   left_join(vote_data_2018, by = c("ct_number" = "ctract"))
387
388 map_2016 <- ct_data_clean %>%
389   left_join(vote_data_2016, by = c("ct_number" = "ctract"))
390
391 map_2014 <- ct_data_2012_clean %>%
392   left_join(vote_data_2014, by = c("ct_number" = "ctract"))
393
394 map_2012 <- ct_data_2012_clean %>%
395   left_join(vote_data_2012, by = c("ct_number" = "ctract"))
396
397
398 ##################################################
399 #year over year per %
400 ##################################################
401
402 perm_2012 <- all_data_2012 %>%
403   mutate(year = rep("2012"), nrow(all_data_2012)) %>%
404   dplyr::select(year, total_regs, total.reg_PERM)
405
406 perm_2014 <- all_data_2014 %>%
407   mutate(year = rep("2014"), nrow(all_data_2014)) %>%
408   dplyr::select(year, total_regs, total.reg_PERM)
409
410 perm_2016 <- all_data_2016 %>%
411   mutate(year = rep("2016"), nrow(all_data_2016)) %>%
412   dplyr::select(year, total_regs, total.reg_PERM)
413
414 perm_2018 <- all_data_2018 %>%
```

```
  mutate(year = rep("2018"), nrow(all_data_2018)) %>%
  dplyr::select(year, total_regs, total.reg_PERM)

perm_perc <- rbind(perm_2012, perm_2014, perm_2016, perm_2018)

perm_perc <- perm_perc %>%
  group_by(year) %>%
  summarise(percent_perm = sum(total.reg_PERM)/sum(total_regs))
```

A.2 Summary Statistics

```
#run thesis.r before running this script

library(leaflet)
library(sf)
library(tidyverse)


#######################################################
#summary stats
#######################################################

#overall turnnout

#2012
sum(all_data_2012$total_votes)/sum(all_data_2012$total_regs)

#2014
sum(all_data_2014$total_votes)/sum(all_data_2014$total_regs)

#2016
sum(all_data_2016$total_votes)/sum(all_data_2016$total_regs)

#2018
sum(all_data_2018$total_votes)/sum(all_data_2018$total_regs)
```

```
25
26 ##########################################################
27 #timeseries plots of registration, voters, turnout%
28 ##########################################################
29 turnoutdata %>%
30   ggplot(aes(x = election_year, y= `Turnout of Registered Voters`, group =
       election_type)) +
31   geom_point(aes(color = election_type), size = 3) +
32   geom_line(aes(color = election_type)) +
33   labs(x = "Election Year", y= "Percent", title = "Turnout Percentage for
      Registered Voters", color = "Election Type") +
34   theme(plot.title = element_text(hjust = .5),
35         axis.text.x = element_text(angle = -45)) +
36   scale_y_continuous(labels = scales::percent_format(accuracy = 1)) ->
      turnout_percent
37
38 turnoutdata %>%
39   ggplot(aes(x = election_year, y= `Total Votes Cast`, group = election_
      type)) +
40   geom_point(aes(color = election_type), size = 3) +
41   geom_line(aes(color = election_type)) +
42   labs(x = "Election Year", y= "Total Votes", title = "Total Votes Cast",
      color = "Election Type") +
43   theme(plot.title = element_text(hjust = .5),
44         axis.text.x = element_text(angle = -45))  -> total_votes
45
46 turnoutdata %>%
47   filter(!grepl("Primary", election_type)) %>%
48   select(election_year, `Persons Registered to Vote`, `Persons Eligible to
       Vote`) %>%
49   gather(key = "count_type", value = "count", -election_year) %>%
50   arrange(election_year) %>%
51   ggplot() +
52   geom_point(aes(x = election_year, y= count, color = count_type), size =
      3) +
```

```
  geom_line(aes(x = election_year, y= count, color = count_type, group =
   count_type)) +
  labs(x = "Election Year", y= "Count", title = "Registered vs Eligible
   Voters", color = "") +
  theme(plot.title = element_text(hjust = .5),
        axis.text.x = element_text(angle = -45))  -> eligible_registered

##############################################
#Census Level stats: Use leaflet/ggplot to look at turnout stats and
   demographic stats
#we need to look at this as a feature collection instead of a data frame
   for leaflet
##############################################

##############################################
#ggplot of some basic demographic data

map_2018 %>%
  select(geometry,
         african_american,
         european_american,
         hispanic_latino,
         asian_american) %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  facet_wrap(~variable) +
  geom_sf(color = NA) +
  coord_sf(crs = 26911) +
  scale_fill_viridis_c(option = "magma") +
  theme(axis.text.x = element_blank(),
        #axis.text.y = element_blank(),
        axis.ticks = element_blank())  -> ethnicity_plot

map_2018 %>%
  mutate(`Poverty Percent` = poverty/population,
```

```
         `Public Transportation` = public_transportation/total_labor_force
  ) %>%
  filter(population > 0) %>%
  select(geometry,
         `Poverty Percent`,
         `Public Transportation`) %>%
  #standardize variables to a percent of the max
  mutate(`Poverty Percent` = percent_rank(`Poverty Percent`),
         `Public Transportation` = percent_rank(`Public Transportation`))
   %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  facet_wrap(~variable) +
  geom_sf(color = NA) +
  coord_sf(ylim = c(33.7,35)) +
  scale_fill_viridis_c(option = "plasma") +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = .5)) +
  labs(title = "Poverty and Public Transportation \n Percentiles: 2018")
   -> poverty_transportation


map_2018 %>%
  select(geometry,
         turnout_percent) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
  #filter out wilderness
  filter(turnout_percent > 0) %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  coord_sf(ylim = c(33.7,35)) +
  scale_fill_viridis_c(option = "plasma") +
```

```
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = .5)) +
  labs(title = "Turnout Percentile: 2018") -> turnout_map_2018


map_2016 %>%
  select(geometry,
         turnout_percent) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
  #filter out wilderness
  filter(turnout_percent > 0) %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  coord_sf(ylim = c(33.7,35)) +
  scale_fill_viridis_c(option = "plasma") +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = .5)) +
  labs(title = "Turnout Percentile: 2016") -> turnout_map_2016

map_2014 %>%
  select(geometry,
         turnout_percent) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
  #filter out wilderness
  filter(turnout_percent > 0) %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  coord_sf(ylim = c(33.7,35)) +
  scale_fill_viridis_c(option = "plasma") +
```

```
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = .5)) +
  labs(title = "Turnout Percentile: 2014") -> turnout_map_2014


################################################
#summary stats for percentile turnout groups
#earnings
#age
#race
#public transport
#poverty
#PERM %
################################################

all_data_2018 %>%
  filter(population > 100) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
  mutate(turnout_categorical = ifelse(turnout_percentile >=.5, "top_50", "
    bottom_50")) %>%
  group_by(turnout_categorical) %>%
  dplyr::summarise(avg_earnings = mean(earnings_median, na.rm = T),
                   poverty_rate = mean(poverty/population),
                   avg_age = mean(age_median, na.rm = T),
                   minority_rate = 1 - mean(european_american/population),
                   college = mean((edu_bachelors + edu_graduate)/ of_
    voting_age),
                   perm_vote = mean(total.votes_PERM/total_votes))


all_data_2016 %>%
  filter(population > 100) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
```

```
  mutate(turnout_categorical = ifelse(turnout_percentile >=.5, "top_50", "
   bottom_50")) %>%
  group_by(turnout_categorical) %>%
  dplyr::summarise(avg_earnings = mean(earnings_median, na.rm = T),
                   poverty_rate = mean(poverty/population),
                   avg_age = mean(age_median, na.rm = T),
                   minority_rate = 1 - mean(european_american/population))
all_data_2012 %>%
  filter(population > 100) %>%
  mutate(turnout_percentile = percent_rank(turnout_percent)) %>%
  mutate(turnout_categorical = ifelse(turnout_percentile >=.5, "top_50", "
   bottom_50")) %>%
  group_by(turnout_categorical) %>%
  dplyr::summarise(avg_earnings = mean(earnings_median, na.rm = T),
                   poverty_rate = mean(poverty/population),
                   avg_age = mean(age_median, na.rm = T),
                   minority_rate = 1 - mean(european_american/population))


#perm vs non perm

perm_votes <- CT_votes %>%
  group_by(year,perm_category) %>%
  dplyr::summarise(votes = sum(voted),
                   registered = sum(total.reg),
                   percent_voted = votes/registered)

#PERM registration percentage
perm_votes %>%
  group_by(year) %>%
  summarise(percent_perm = registered[perm_category == "PERM"]/(registered
   [perm_category == "NON-PERM"] +registered[perm_category == "PERM"]))
```

A.3 VBM GAMs

```
#######################################################################
#The goal of this script is to use GAMs to model
#the percentage of voter age population that are registered to
#vote by mail
#We will explore basic assumptions about barriers people face in voting
#and see if those are good predictors of if people are using VBM or not.
#We will attempt to use both beta regression to model the proportions
    (0,1)
#as well as modeling the absolute number.
#We will evaluate the stregth and usefulness (interpretability) of each
    model.
#We will compare mse and aic and gcv
#cross validate
#######################################################################
#libraries
library(mgcv)
library(caret)
library(tigris)
library(spdep)
library(tidyverse)
library(PerformanceAnalytics)
library(gamclass)
library(caTools)

#load environment
load("/Users/pbeeman/Google Drive/masters thesis copy/masterenvironment.
    RData")

#2014 data manipulation
PERM_data <- all_data_2014 %>%
  mutate(PERM_percent = total.reg_PERM/total_regs,
         children = (over_6_under_18 + under_6)) %>%
  filter(total_regs > 100) %>%
  dplyr::select(african_american,
                age_median,
```

```
                asian_american,
                earnings_median,
                edu_bachelors,
                edu_graduate,
                edu_highschool,
                edu_no_highschool,
                employed,
                european_american,
                hispanic_latino,
                native_american,
                pacific_islander,
                population,
                poverty,
                total_female,
                total_male,
                unemployed,
                children,
                PERM_percent,
                turnout_percent,
                GEOID,
                geometry
  )

#add it lag variables

lags_2012 <- all_data_2012 %>%
  filter(total_regs > 100) %>%
  mutate(lags = total.reg_PERM/total_regs) %>%
  dplyr::select(lags, GEOID)


PERM_data <- PERM_data %>%
  left_join(lags_2012, "GEOID")

```

```r
#quick exploration of the linear relationship between PERM registration and turnout
#we see that perm registration is positively coorelated to voter turnout

summary(lm(PERM_data$turnout_percent ~ PERM_data$PERM_percent))
hist(PERM_data$PERM_percent)
hist(PERM_data$turnout_percent)
plot(PERM_data$PERM_percent, PERM_data$turnout_percent)
abline(.02374, 1.05352, col = "red")

#latino
plot(PERM_data$hispanic_latino/PERM_data$population, PERM_data$PERM_percent)

#start modeling
PERM_data <- PERM_data %>%
  dplyr::select(-turnout_percent) %>%
  mutate(hispanic_percent = hispanic_latino/population,
         black_percent = african_american/population,
         white_percent = european_american/population,
         pov_percent = poverty/population,
         asian_percent = asian_american/population,
         bachelor_percent = edu_bachelors/(population-children),
         grad_percent = edu_graduate/(population - children),
         hs_percent = edu_highschool/(population - children),
         nohs_percent = edu_no_highschool/(population - children),
         employed_percent = employed/(population - children),
         unemployed_percent = unemployed/(population - children),
         child_percent = children/(population - children)
  ) %>%
  na.omit()

#split into training/test data
sample_size <- floor(0.75 * nrow(PERM_data))
set.seed(123)
```

```r
train_ind <- sample(seq_len(nrow(PERM_data)), size = sample_size)

PERM_data <- PERM_data[train_ind, ]
PERM_data_test <- PERM_data[-train_ind, ]


########################################
#Markov Random Fields Geo approach
########################################

#read in shap files for CTs
LA_shapes <- tracts("CA", "Los Angeles", cb = TRUE)


#check for empty geometries
which(is.na(st_dimension(PERM_data$geometry)))


#convert to neighborhood
nb <- poly2nb(PERM_data$geometry, row.names = PERM_data$GEOID)
names(nb) <- attr(nb, "region.id")
str(nb[1:6])

#fit GAM for PERM using MRF with a beta response
#https://stat.ethz.ch/R-manual/R-patched/library/mgcv/html/Beta.html

ctrl <- gam.control(nthreads = 6) # use 6 parallel threads, reduce if
   fewer physical CPU cores
m1 <- gam(PERM_percent ~ s(as.factor(GEOID), bs = 'mrf', k = 20, xt = list
   (nb = nb)), # define MRF smooth
          data = PERM_data,
          method = 'REML',
          # fast version of REML smoothness selection
          family = betar()) # fit a beta regression

```

```r
viz_data <- PERM_data %>%
  mutate(new_vals = predict(m1, type = 'response'))


summary(m1)
gam.check(m1)

#visualize
#cross validation
#https://rdrr.io/cran/gamclass/man/CVgam.html

################################################
#Let's start looking at univariate relationships and plotting them out
################################################
##############################
#avg earnings
##############################
earnings <- gam(PERM_percent ~ s(earnings_median) , data = PERM_data,
    family = betar())
summary(earnings)
plot(earnings, residuals = T)

#partial effects plot
p <- predict(earnings, type="lpmatrix")
beta <- coef(earnings)[grepl("median", names(coef(earnings)))]
s <- p[,grepl("median", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$earnings_median), aes(x=PERM_
    data$earnings_median, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(earnings$fitted.values, earnings$y, PERM_data
    $earnings_median))
names(fit_data) <- c("yhat", "y", "earnings")

```

```r
fit_data %>%
  ggplot() +
  geom_point(aes(x = earnings, y = y)) +
  geom_line(aes(x = earnings, y = yhat), color= "red")

#########################################################
#poverty percent
#########################################################


pov <- gam(PERM_percent ~ s(pov_percent) , data = PERM_data, family =
    betar())
summary(pov)
plot(pov)

#partial effects plot
p <- predict(pov, type="lpmatrix")
beta <- coef(pov)[grepl("pov", names(coef(pov)))]
s <- p[,grepl("pov", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$pov_percent), aes(x=PERM_data$
    pov_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(pov$fitted.values, pov$y, PERM_data$pov_
    percent))
names(fit_data) <- c("yhat", "y", "pov")

fit_data %>%
  ggplot() +
  geom_point(aes(x = pov, y = y)) +
  geom_line(aes(x = pov, y = yhat), color= "red")

#########################################################
#unemployed percent
```

```r
#########################################################


unemployed <- gam(PERM_percent ~ s(unemployed_percent) , data = PERM_data,
    family = betar())
summary(unemployed)
plot(unemployed)

#partial effects plot
p <- predict(unemployed, type="lpmatrix")
beta <- coef(unemployed)[grepl("unemployed", names(coef(unemployed)))]
s <- p[,grepl("unemployed", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$unemployed_percent), aes(x=PERM_
    data$unemployed_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(unemployed$fitted.values, unemployed$y, PERM_
    data$unemployed_percent))
names(fit_data) <- c("yhat", "y", "unemployed")

fit_data %>%
  ggplot() +
  geom_point(aes(x = unemployed, y = y)) +
  geom_line(aes(x = unemployed, y = yhat), color= "red")

###################################
#age
###################################

age <- gam(PERM_percent ~ s(age_median) , data = PERM_data, family = betar
    ())
summary(age)
plot(earnings)

```

```
#partial effects plot
p <- predict(age, type="lpmatrix")
beta <- coef(age)[grepl("median", names(coef(age)))]
s <- p[,grepl("median", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$age_median), aes(x=PERM_data$age
    _median, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(age$fitted.values, age$y, PERM_data$age_
    median))
names(fit_data) <- c("yhat", "y", "age")

fit_data %>%
  ggplot() +
  geom_point(aes(x = age, y = y)) +
  geom_line(aes(x = age, y = yhat), color= "red")

#we see a linear relationship between age and PERM %

##########################################################
#hispanic percent
##########################################################

hispanic <- gam(PERM_percent ~ s(hispanic_percent) , data = PERM_data,
    family = betar())
summary(hispanic)
plot(hispanic)

#partial effects plot
p <- predict(hispanic, type="lpmatrix")
beta <- coef(hispanic)[grepl("hispanic", names(coef(hispanic)))]
s <- p[,grepl("hispanic", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$hispanic_percent), aes(x=PERM_
    data$hispanic_percent, y=s)) + geom_line()
```

```
260
261 #fitted vs predictors
262
263 fit_data <- data.frame(cbind(hispanic$fitted.values, hispanic$y, PERM_data
      $hispanic_percent))
264 names(fit_data) <- c("yhat", "y", "hispanic")
265
266 fit_data %>%
267   ggplot() +
268   geom_point(aes(x = hispanic, y = y)) +
269   geom_line(aes(x = hispanic, y = yhat), color= "red")
270
271 #we see a linear relationship between age and PERM %
272
273 ##########################################################
274 #black percent
275 ##########################################################
276
277
278 black <- gam(PERM_percent ~ s(black_percent) , data = PERM_data, family =
      betar())
279 summary(black)
280 plot(black)
281
282 #partial effects plot
283 p <- predict(black, type="lpmatrix")
284 beta <- coef(black)[grepl("black", names(coef(black)))]
285 s <- p[,grepl("black", colnames(p))] %*% beta
286 ggplot(data=cbind.data.frame(s, PERM_data$black_percent), aes(x=PERM_data$
      black_percent, y=s)) + geom_line()
287
288 #fitted vs predictors
289
290 fit_data <- data.frame(cbind(black$fitted.values, black$y, PERM_data$black
      _percent))
```

```r
names(fit_data) <- c("yhat", "y", "black")

fit_data %>%
  ggplot() +
  geom_point(aes(x = black, y = y)) +
  geom_line(aes(x = black, y = yhat), color= "red")


#########################################################
#asian percent
#########################################################


asian <- gam(PERM_percent ~ s(asian_percent) , data = PERM_data, family =
    betar())
summary(asian)
plot(asian)

#partial effects plot
p <- predict(asian, type="lpmatrix")
beta <- coef(asian)[grepl("asian", names(coef(asian)))]
s <- p[,grepl("asian", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$asian_percent), aes(x=PERM_data$
    asian_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(asian$fitted.values, asian$y, PERM_data$asian
    _percent))
names(fit_data) <- c("yhat", "y", "asian")

fit_data %>%
  ggplot() +
  geom_point(aes(x = asian, y = y)) +
  geom_line(aes(x = asian, y = yhat), color= "red")
```

```
323
324
325 ########################################################
326 #hs percent
327 ########################################################
328
329
330 hs <- gam(PERM_percent ~ s(hs_percent) , data = PERM_data, family = betar
    ())
331 summary(hs)
332 plot(hs)
333
334 #partial effects plot
335 p <- predict(hs, type="lpmatrix")
336 beta <- coef(hs)[grepl("hs", names(coef(hs)))]
337 s <- p[,grepl("hs", colnames(p))] %*% beta
338 ggplot(data=cbind.data.frame(s, PERM_data$hs_percent), aes(x=PERM_data$hs_
    percent, y=s)) + geom_line()
339
340 #fitted vs predictors
341
342 fit_data <- data.frame(cbind(hs$fitted.values, hs$y, PERM_data$hs_percent)
    )
343 names(fit_data) <- c("yhat", "y", "hs")
344
345 fit_data %>%
346   ggplot() +
347   geom_point(aes(x = hs, y = y)) +
348   geom_line(aes(x = hs, y = yhat), color= "red")
349
350 #doesn't appear too meaningful
351
352 ########################################################
353 #bachelor percent
354 ########################################################
```

```
355
356
357 bachelor <- gam(PERM_percent ~ s(bachelor_percent) , data = PERM_data,
        family = betar())
358 summary(bachelor)
359 plot(bachelor)
360
361 #partial effects plot
362 p <- predict(bachelor, type="lpmatrix")
363 beta <- coef(bachelor)[grepl("bachelor", names(coef(bachelor)))]
364 s <- p[,grepl("bachelor", colnames(p))] %*% beta
365 ggplot(data=cbind.data.frame(s, PERM_data$bachelor_percent), aes(x=PERM_
        data$bachelor_percent, y=s)) + geom_line()
366
367 #fitted vs predictors
368
369 fit_data <- data.frame(cbind(bachelor$fitted.values, bachelor$y, PERM_data
        $bachelor_percent))
370 names(fit_data) <- c("yhat", "y", "bachelor")
371
372 fit_data %>%
373   ggplot() +
374   geom_point(aes(x = bachelor, y = y)) +
375   geom_line(aes(x = bachelor, y = yhat), color= "red")
376
377 ########################################################
378 #nohs percent
379 ########################################################
380
381
382 nohs <- gam(PERM_percent ~ s(nohs_percent) , data = PERM_data, family =
        betar())
383 summary(nohs)
384 plot(nohs)
385
```

```
#partial effects plot
p <- predict(nohs, type="lpmatrix")
beta <- coef(nohs)[grepl("nohs", names(coef(nohs)))]
s <- p[,grepl("nohs", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$nohs_percent), aes(x=PERM_data$
    nohs_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(nohs$fitted.values, nohs$y, PERM_data$nohs_
    percent))
names(fit_data) <- c("yhat", "y", "nohs")

fit_data %>%
  ggplot() +
  geom_point(aes(x = nohs, y = y)) +
  geom_line(aes(x = nohs, y = yhat), color= "red")



#########################################################
#child percent
#########################################################


child <- gam(PERM_percent ~ s(child_percent) , data = PERM_data, family =
    betar())
summary(child)
plot(child)

#partial effects plot
p <- predict(child, type="lpmatrix")
beta <- coef(child)[grepl("child", names(coef(child)))]
s <- p[,grepl("child", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data$child_percent), aes(x=PERM_data$
```

```
    child_percent, y=s)) + geom_line()
418
419 #fitted vs predictors
420
421 fit_data <- data.frame(cbind(child$fitted.values, child$y, PERM_data$child
    _percent))
422 names(fit_data) <- c("yhat", "y", "child")
423
424 fit_data %>%
425   ggplot() +
426   geom_point(aes(x = child, y = y)) +
427   geom_line(aes(x = child, y = yhat), color= "red")
428
429 #doesn't look like a strong predictor
430
431
432
433 ###################################################################
434 #correlation matrix
435 ###################################################################
436
437 cor_data <- PERM_data %>%
438   select(20,2,4, 23:31)
439
440 chart.Correlation(cor_data, histogram=TRUE, pch=19)
441
442
443 ###################################################
444 #
445 #
446 #Start modelling
447 #
448 ###################################################
449
450
```

```
451
452 ##########################################################
453 #model with just lags
454 ##########################################################
455 lag_mod <- gam(PERM_percent ~ s(lags)  , data = PERM_data, family = betar
    ())
456 summary(lag_mod)
457 plot(lag_mod)
458
459 #try to better understand relationship with lags by fitting simple linear
    model
460 lag_line <- lm(PERM_percent ~ lags, data = PERM_data)
461 summary(lag_line)
462
463 ################################################
464 #build a model with lags and predictor variables
465 ################################################
466 full_gam <- gam(PERM_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
467                 s(black_percent) + s(white_percent) + s(pov_percent) + s
    (asian_percent) +
468                 s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent),
469               data = PERM_data,
470               family = betar(),
471               select = T)
472
473 summary(full_gam)
474 gam.check(full_gam)
475
476 #lets ad geo data into this
477
478 full_gam_geo <- gam(PERM_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
479                 s(black_percent) + s(white_percent) + s(pov_percent) + s
```

```
    (asian_percent) +
480                 s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent) +
481                 s(child_percent) +
482                 s(as.factor(GEOID), bs = 'mrf', k = 20, xt = list(nb =
    nb)),
483               data = PERM_data,
484               family = betar(),
485               select = T)
486
487 summary(full_gam)
488 gam.check(full_gam)
489 plot(full_gam)
490
491 #lets ad lags into this
492 full_gam <- gam(PERM_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
493                 s(black_percent) + s(white_percent) + s(pov_percent) + s
    (asian_percent) +
494                 s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent) +
495                 s(child_percent) +
496                 s(as.factor(GEOID), bs = 'mrf', k = 20, xt = list(nb =
    nb)) +
497                 s(lags),
498               data = PERM_data,
499               family = betar(),
500               select = T)
501
502 summary(full_gam)
503 gam.check(full_gam)
504 plot(full_gam)
505
506
507 #remove geo
```

```
full_gam <- gam(PERM_percent ~ s(age_median) + s(earnings_median) + s(hispanic_percent) +
                  s(black_percent) + s(white_percent) + s(pov_percent) + s(asian_percent) +
                  s(bachelor_percent) + s(grad_percent) + s(hs_percent) + s(nohs_percent) +
                  s(child_percent) +
                  s(lags),
                data = PERM_data,
                family = betar(),
                select = T)

summary(full_gam)
gam.check(full_gam)
plot(full_gam)

################################################
#performance

predict_perm_test <- predict.gam(full_gam, PERM_data_test, se.fit = T, type = "response")
resids_test <- cbind(predict_perm_test$fit, PERM_data_test$PERM_percent)
resids_test <- as.data.frame(resids_test)
names(resids_test) <- c("pred", "obs")
resids_test <- resids_test %>%
  mutate(resids = obs - pred)

rmse_full <- RMSE(obs = resids_test$obs, pred = resids_test$pred)
#just for fun I want to double check
sqrt(sum(resids_test$resids^2)/nrow(resids_test))
#more or less iid normally distributed residual
plot(resids_test$obs, resids_test$resids)
hist(resids_test$resids)
################################################

```

```
539 ################################################
540 #strip variables that do not appear to be useful
541 diminished_gam <- gam(PERM_percent ~ s(age_median) + s(earnings_median) +
    s(hispanic_percent) +
542                   s(black_percent)  + s(pov_percent) +
543                   s(bachelor_percent)  + s(nohs_percent) +
544                   s(child_percent) +
545                   s(lags),
546                 data = PERM_data,
547                 family = betar(),
548                 select = T)
549
550 summary(diminished_gam)
551 gam.check(diminished_gam)
552
553 #performance for gam_diminished
554 predict_perm_test2 <- predict.gam(diminished_gam, PERM_data_test, se.fit =
      T, type = "response")
555 resids_test2 <- cbind(predict_perm_test2$fit, PERM_data_test$PERM_percent)
556 resids_test2 <- as.data.frame(resids_test2)
557 names(resids_test2) <- c("pred", "obs")
558 resids_test2 <- resids_test2 %>%
559   mutate(resids = obs - pred)
560
561 rmse_dim <- RMSE(obs = resids_test2$obs, pred = resids_test2$pred)
562 #more or less iid normally distributed residual
563 plot(resids_test2$obs, resids_test2$resids)
564 hist(resids_test2$resids)
565
566 ##############################################
567 #lastly, just check rmse for lag model
568
569 predict_perm_test3 <- predict.gam(lag_mod, PERM_data_test, se.fit = T,
    type = "response")
570 resids_test3 <- cbind(predict_perm_test3$fit, PERM_data_test$PERM_percent)
```

```r
resids_test3 <- as.data.frame(resids_test3)
names(resids_test3) <- c("pred", "obs")
resids_test3 <- resids_test3%>%
  mutate(resids = obs - pred)

rmse_lag <- RMSE(obs = resids_test3$obs, pred = resids_test3$pred)
#more or less iid normally distributed residual
plot(resids_test3$obs, resids_test3$resids)
hist(resids_test3$resids)
################################################
#
#fit on 2018 data and see how it performs out of box
################################################
#2018 data manipulation
PERM_data_2018 <- all_data_2018 %>%
  mutate(PERM_percent = total.reg_PERM/total_regs,
         children = (over_6_under_18 + under_6)) %>%
  filter(population > 100) %>%
  dplyr::select(african_american,
                age_median,
                asian_american,
                earnings_median,
                edu_bachelors,
                edu_graduate,
                edu_highschool,
                edu_no_highschool,
                employed,
                european_american,
                hispanic_latino,
                native_american,
                pacific_islander,
                population,
                poverty,
                total_female,
                total_male,
```

```
                unemployed,
                children,
                PERM_percent,
                turnout_percent,
                GEOID,
                geometry
  )

#add in lag variables

lags_2016 <- all_data_2016 %>%
  mutate(lags = total.reg_PERM/total_regs) %>%
  select(lags, GEOID)


PERM_data_2018 <- PERM_data_2018 %>%
  left_join(lags_2016, "GEOID")

#start modeling
PERM_data_2018 <- PERM_data_2018 %>%
  dplyr::select(-turnout_percent) %>%
  mutate(hispanic_percent = hispanic_latino/population,
         black_percent = african_american/population,
         white_percent = european_american/population,
         pov_percent = poverty/population,
         asian_percent = asian_american/population,
         bachelor_percent = edu_bachelors/(population-children),
         grad_percent = edu_graduate/(population - children),
         hs_percent = edu_highschool/(population - children),
         nohs_percent = edu_no_highschool/(population - children),
         employed_percent = employed/(population - children),
         unemployed_percent = unemployed/(population - children),
         child_percent = children/(population - children)
  ) %>%
  na.omit()
```

```
641
642 predict_perm_2018 <- predict.gam(diminished_gam, PERM_data_2018, se.fit =
      T, type = "response")
643 resids_2018 <- cbind(predict_perm_2018$fit, PERM_data_2018$PERM_percent,
      as.numeric(PERM_data_2018$GEOID))
644 resids_2018 <- as.data.frame(resids_2018)
645 names(resids_2018) <- c("pred", "obs", "geo")
646 resids_2018 <- resids_2018 %>%
647   mutate(resids = pred - obs)
648
649
650 #systematically underpredicts and RMSE doubles
651 RMSE(obs = resids_2018$obs, pred = resids_2018$pred)
652 #just for fun I want to double check
653 sqrt(sum(resids_2018$resids^2)/nrow(resids_2018))
654 #more or less iid normally distributed residual
655 plot(resids_2018$obs, resids_2018$resids)
656 hist(resids_2018$resids)
657
658 #now let's look at it using the lag model.
659 predict_perm_20182 <- predict.gam(lag_mod, PERM_data_2018, se.fit = T,
      type = "response")
660 resids_20182 <- cbind(predict_perm_20182$fit, PERM_data_2018$PERM_percent,
       as.numeric(PERM_data_2018$GEOID))
661 resids_20182 <- as.data.frame(resids_20182)
662 names(resids_20182) <- c("obs", "pred", "geo")
663 resids_20182 <- resids_20182 %>%
664   mutate(resids = pred - obs)
665
666
667 #systematically underpredicts and RMSE doubles
668 RMSE(obs = resids_20182$obs, pred = resids_20182$pred)
669 #more or less iid normally distributed residual
670 plot(resids_2018$obs, resids_2018$resids)
671 hist(resids_2018$resids)
```

```r
#here we see that the lag model predicts worse, but not by that much and still systematically under predicts

#########################################################
#lets look at 2016 and 2018 as the outcome again, but let's build the model with our outcome variable
#standardized
#
#
#########################################################
#we now have stnadardized variables that we can use are our outcome variable
#tow different approaches to standardizing
#attempt 1
PERM_data_standard1 <- PERM_data %>%
  mutate(PERM_percent2 = PERM_percent/max(PERM_percent)) %>%
  mutate(lags2 = lags/max(lags, na.rm = T)) %>%
  dplyr::select(PERM_percent, lags, PERM_percent2, lags2)

#attempt 2
PERM_data_standard2 <- PERM_data %>%
  mutate(PERM_percent2 = scale(PERM_percent)) %>%
  mutate(lags2 = scale(lags)) %>%
  dplyr::select(PERM_percent, lags, PERM_percent2, lags2)

#this approach ended up not being terribly convincing.

#############################################################
#Let's look at how the model performed if we interpret the out put as an ordinal output
#we don't need to rerun the model , we just need to order the predictions and the actual numbers and see how we did
#############################################################


```

```r
#let's look at resids_2018 but change obs, pred to ordinal and see what the resids are

ordinal_resids_2018 <- resids_2018 %>%
  mutate(obs2 = percent_rank(obs)) %>%
  mutate(pred2 = percent_rank(pred)) %>%
  mutate(resids2 = obs2-pred2)

#just lag
ordinal_resids_20182 <- resids_20182 %>%
  mutate(obs2 = percent_rank(obs)) %>%
  mutate(pred2 = percent_rank(pred)) %>%
  mutate(resids2 = obs2-pred2)

#just lag does a decent job of predicting the percentile of the next election.
RMSE(pred = ordinal_resids_20182$pred, obs = ordinal_resids_20182$obs)
#when estimating the percentile rank its RMSE is .045
#residuals are roughly normally distributed
plot(ordinal_resids_20182$resids2)
hist(ordinal_resids_20182$resids2)


############################################################
#winning model diagnostics
############################################################
#original
#partial effects interpretation: https://www.researchgate.net/figure/Generalized-additive-model-GAM-plots-showing-the-partial-effects-of-selected_fig4_324422761
gam.check(lag_mod)
plot(lag_mod)
#ordinal
plot(ordinal_resids_20182$resids2)
hist(ordinal_resids_20182$resids2)
```

```
733
734 ##############################################################
735 #univariate relationships for 2018
736 ##############################################################
737 ##############################
738 #avg earnings
739 ##############################
740 earnings <- gam(PERM_percent ~ s(earnings_median) , data = PERM_data_2018,
        family = betar())
741 summary(earnings)
742 plot(earnings, residuals = T)
743
744 #partial effects plot
745 p <- predict(earnings, type="lpmatrix")
746 beta <- coef(earnings)[grepl("median", names(coef(earnings)))]
747 s <- p[,grepl("median", colnames(p))] %*% beta
748 ggplot(data=cbind.data.frame(s, PERM_data_2018$earnings_median), aes(x=
        PERM_data_2018$earnings_median, y=s)) + geom_line()
749
750 #fitted vs predictors
751
752 fit_data <- data.frame(cbind(earnings$fitted.values, earnings$y, PERM_data
        _2018$earnings_median))
753 names(fit_data) <- c("yhat", "y", "earnings")
754
755 fit_data %>%
756   ggplot() +
757   geom_point(aes(x = earnings, y = y)) +
758   geom_line(aes(x = earnings, y = yhat), color= "red") +
759   ylab("VBM Percentage") +
760   xlab("Average Yearly Earnings")
761
762 ##############################################################
763 #poverty percent
764 ##############################################################
```

```
765
766
767 pov <- gam(PERM_percent ~ s(pov_percent) , data = PERM_data_2018, family =
        betar())
768 summary(pov)
769 plot(pov)
770
771 #partial effects plot
772 p <- predict(pov, type="lpmatrix")
773 beta <- coef(pov)[grepl("pov", names(coef(pov)))]
774 s <- p[,grepl("pov", colnames(p))] %*% beta
775 ggplot(data=cbind.data.frame(s, PERM_data_2018$pov_percent), aes(x=PERM_
       data_2018$pov_percent, y=s)) + geom_line()
776
777 #fitted vs predictors
778
779 fit_data <- data.frame(cbind(pov$fitted.values, pov$y, PERM_data_2018$pov_
       percent))
780 names(fit_data) <- c("yhat", "y", "pov")
781
782 fit_data %>%
783   ggplot() +
784   geom_point(aes(x = pov, y = y)) +
785   geom_line(aes(x = pov, y = yhat), color= "red") +
786   ylab("VBM Percentage") +
787   xlab("Poverty Rate")
788
789
790 ##########################################################
791 #unemployed percent
792 ##########################################################
793
794
795 unemployed <- gam(PERM_percent ~ s(unemployed_percent) , data = PERM_data_
       2018, family = betar())
```

```r
summary(unemployed)
plot(unemployed)

#partial effects plot
p <- predict(unemployed, type="lpmatrix")
beta <- coef(unemployed)[grepl("unemployed", names(coef(unemployed)))]
s <- p[,grepl("unemployed", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$unemployed_percent), aes(x=
    PERM_data_2018$unemployed_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(unemployed$fitted.values, unemployed$y, PERM_
    data_2018$unemployed_percent))
names(fit_data) <- c("yhat", "y", "unemployed")

fit_data %>%
  ggplot() +
  geom_point(aes(x = unemployed, y = y)) +
  geom_line(aes(x = unemployed, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("Unemployment Rate")


##################################
#age
##################################

age <- gam(PERM_percent ~ s(age_median) , data = PERM_data_2018, family =
    betar())
summary(age)
plot(earnings)

#partial effects plot
p <- predict(age, type="lpmatrix")
```

```
beta <- coef(age)[grepl("median", names(coef(age)))]
s <- p[,grepl("median", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$age_median), aes(x=PERM_
    data_2018$age_median, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(age$fitted.values, age$y, PERM_data_2018$age_
    median))
names(fit_data) <- c("yhat", "y", "age")

fit_data %>%
  ggplot() +
  geom_point(aes(x = age, y = y)) +
  geom_line(aes(x = age, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("Average Age")


#we see a linear relationship between age and PERM %

#########################################################
#hispanic percent
#########################################################

hispanic <- gam(PERM_percent ~ s(hispanic_percent) , data = PERM_data_
    2018, family = betar())
summary(hispanic)
plot(hispanic)

#partial effects plot
p <- predict(hispanic, type="lpmatrix")
beta <- coef(hispanic)[grepl("hispanic", names(coef(hispanic)))]
s <- p[,grepl("hispanic", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$hispanic_percent), aes(x=
```

```
   PERM_data_2018$hispanic_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(hispanic$fitted.values, hispanic$y, PERM_data
   _2018$hispanic_percent))
names(fit_data) <- c("yhat", "y", "hispanic")

fit_data %>%
  ggplot() +
  geom_point(aes(x = hispanic, y = y)) +
  geom_line(aes(x = hispanic, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("Hispanic Rate")


#we see a linear relationship between age and PERM %

#########################################################
#black percent
#########################################################


black <- gam(PERM_percent ~ s(black_percent) , data = PERM_data_2018,
   family = betar())
summary(black)
plot(black)

#partial effects plot
p <- predict(black, type="lpmatrix")
beta <- coef(black)[grepl("black", names(coef(black)))]
s <- p[,grepl("black", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$black_percent), aes(x=PERM_
   data_2018$black_percent, y=s)) + geom_line()

```

```r
#fitted vs predictors

fit_data <- data.frame(cbind(black$fitted.values, black$y, PERM_data_2018$
    black_percent))
names(fit_data) <- c("yhat", "y", "black")

fit_data %>%
  ggplot() +
  geom_point(aes(x = black, y = y)) +
  geom_line(aes(x = black, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("African American Rate")



##########################################################
#asian percent
##########################################################


asian <- gam(PERM_percent ~ s(asian_percent) , data = PERM_data_2018,
    family = betar())
summary(asian)
plot(asian)

#partial effects plot
p <- predict(asian, type="lpmatrix")
beta <- coef(asian)[grepl("asian", names(coef(asian)))]
s <- p[,grepl("asian", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$asian_percent), aes(x=PERM_
    data_2018$asian_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(asian$fitted.values, asian$y, PERM_data_2018$
```

```
    asian_percent))
names(fit_data) <- c("yhat", "y", "asian")

fit_data %>%
  ggplot() +
  geom_point(aes(x = asian, y = y)) +
  geom_line(aes(x = asian, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("Asian Rate")



##########################################################
#hs percent
##########################################################


hs <- gam(PERM_percent ~ s(hs_percent) , data = PERM_data_2018, family =
    betar())
summary(hs)
plot(hs)

#partial effects plot
p <- predict(hs, type="lpmatrix")
beta <- coef(hs)[grepl("hs", names(coef(hs)))]
s <- p[,grepl("hs", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$hs_percent), aes(x=PERM_
    data_2018$hs_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(hs$fitted.values, hs$y, PERM_data_2018$hs_
    percent))
names(fit_data) <- c("yhat", "y", "hs")

```

```
fit_data %>%
  ggplot() +
  geom_point(aes(x = hs, y = y)) +
  geom_line(aes(x = hs, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("High School Graduate Rate")


#doesn't appear too meaningful

#########################################################
#bachelor percent
#########################################################


bachelor <- gam(PERM_percent ~ s(bachelor_percent) , data = PERM_data_
    2018, family = betar())
summary(bachelor)
plot(bachelor)

#partial effects plot
p <- predict(bachelor, type="lpmatrix")
beta <- coef(bachelor)[grepl("bachelor", names(coef(bachelor)))]
s <- p[,grepl("bachelor", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$bachelor_percent), aes(x=
    PERM_data_2018$bachelor_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(bachelor$fitted.values, bachelor$y, PERM_data
    _2018$bachelor_percent))
names(fit_data) <- c("yhat", "y", "bachelor")

fit_data %>%
  ggplot() +
```

```
  geom_point(aes(x = bachelor, y = y)) +
  geom_line(aes(x = bachelor, y = yhat), color= "red") +
  ylab("VBM Percentage") +
  xlab("Bachelor's Degree Rate")


#########################################################
#nohs percent
#########################################################


nohs <- gam(PERM_percent ~ s(nohs_percent) , data = PERM_data_2018, family
    = betar())
summary(nohs)
plot(nohs)

#partial effects plot
p <- predict(nohs, type="lpmatrix")
beta <- coef(nohs)[grepl("nohs", names(coef(nohs)))]
s <- p[,grepl("nohs", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, PERM_data_2018$nohs_percent), aes(x=PERM_
   data_2018$nohs_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(nohs$fitted.values, nohs$y, PERM_data_2018$
   nohs_percent))
names(fit_data) <- c("yhat", "y", "nohs")

fit_data %>%
  ggplot() +
  geom_point(aes(x = nohs, y = y)) +
  geom_line(aes(x = nohs, y = yhat), color= "red")


```

```
1018
1019 ########################################################
1020 #child percent
1021 ########################################################
1022
1023
1024 child <- gam(PERM_percent ~ s(child_percent) , data = PERM_data_2018,
        family = betar())
1025 summary(child)
1026 plot(child)
1027
1028 #partial effects plot
1029 p <- predict(child, type="lpmatrix")
1030 beta <- coef(child)[grepl("child", names(coef(child)))]
1031 s <- p[,grepl("child", colnames(p))] %*% beta
1032 ggplot(data=cbind.data.frame(s, PERM_data_2018$child_percent), aes(x=PERM_
        data_2018$child_percent, y=s)) + geom_line()
1033
1034 #fitted vs predictors
1035
1036 fit_data <- data.frame(cbind(child$fitted.values, child$y, PERM_data_2018$
        child_percent))
1037 names(fit_data) <- c("yhat", "y", "child")
1038
1039 fit_data %>%
1040   ggplot() +
1041   geom_point(aes(x = child, y = y)) +
1042   geom_line(aes(x = child, y = yhat), color= "red")
1043
1044
1045
1046
1047 ########################################################
1048 #prediction map for 2020
1049 ########################################################
```

```
1050
1051 PERM_data_2020 <- PERM_data_2018 %>%
1052   mutate(lags = PERM_percent)
1053
1054 #it seems reasonable to rerun the model built on 2016 and 2018 to try and
       campture the most recent political
1055 #atmosphere
1056
1057 lag_gam_2018 <- gam(PERM_percent ~ s(lags),
1058                             data = PERM_data_2018,
1059                             family = betar(),
1060                             select = T)
1061
1062 summary(diminished_gam)
1063 gam.check(diminished_gam)
1064
1065 #predict
1066 predict_perm_2020 <- predict.gam(lag_gam_2018, PERM_data_2020, se.fit = T,
       type = "response")
1067
1068 PERM_data_2020_full <- PERM_data_2020 %>%
1069   mutate(predicts_percentiles = percent_rank(predict_perm_2020$fit))
1070
1071 #map
1072 PERM_data_2020_full %>%
1073   select(geometry,
1074          predicts_percentiles) %>%
1075   gather(key = "variable", value = "estimate", -geometry) %>%
1076   ggplot(aes(fill = estimate)) +
1077   geom_sf(aes(geometry = geometry),color = NA) +
1078   coord_sf(ylim = c(33.7,35)) +
1079   scale_fill_viridis_c(option = "plasma") +
1080   theme(axis.text.x = element_blank(),
1081         axis.text.y = element_blank(),
1082         axis.ticks = element_blank(),
```

```
        plot.title = element_text(hjust = .5)) +
  labs(title = "VBM Percentile Predictions: 2020") -> perm_map_2020

#END SCRIPT

plot_data <- all_data_2018 %>%
  select(total.reg_PERM, total_regs, turnout_percent, population) %>%
  mutate(perm_percent = total.reg_PERM/total_regs) %>%
  filter(population >100)

plot_data %>%
  ggplot(aes(x= perm_percent, y = turnout_percent)) +
  geom_point() +
  geom_smooth(method = "loess", span = 2) +
  xlab("VBM Percent") +
  ylab("Turnout Percent")
```

A.4 Voter Turnout GAMs

```
#####################################################################
#The goal of this script is to use GAMs to model
#the percentage of registered voters that turnout to vote
#We will explore basic assumptions about barriers people face in voting
#and see if those are good predictors of if people are voting.
#We will attempt to use both beta regression to model the proportions
   (0,1)
#as well as modeling the absolute number.
#We will evaluate the stregth and usefulness (interpretability) of each
   model.
#We will compare mse and aic and gcv
#cross validate
#####################################################################
#libraries
library(mgcv)
library(caret)
```

```
library(tigris)
library(spdep)
library(tidyverse)
library(PerformanceAnalytics)
library(gamclass)
library(caTools)

#load environment
load("/Users/pbeeman/Google Drive/masters thesis copy/masterenvironment.
    RData")

#2014 data manipulation
Votes_data <- all_data_2014 %>%
  mutate(turnout_percent = total_votes/total_regs,
         children = (over_6_under_18 + under_6)) %>%
  filter(total_regs > 100) %>%
  dplyr::select(african_american,
                age_median,
                asian_american,
                earnings_median,
                edu_bachelors,
                edu_graduate,
                edu_highschool,
                edu_no_highschool,
                employed,
                european_american,
                hispanic_latino,
                native_american,
                pacific_islander,
                population,
                poverty,
                total_female,
                total_male,
                unemployed,
                children,
```

```
                turnout_percent,
                GEOID,
                geometry
  )

#add it lag variables

lags_2012 <- all_data_2012 %>%
  filter(total_regs > 100) %>%
  mutate(lags = total_votes/total_regs) %>%
  dplyr::select(lags, GEOID)


Votes_data <- Votes_data %>%
  left_join(lags_2012, "GEOID")


#latino
plot(Votes_data$hispanic_latino/Votes_data$population, Votes_data$turnout_
    percent)

#start modeling
Votes_data <- Votes_data %>%
  mutate(hispanic_percent = hispanic_latino/population,
         black_percent = african_american/population,
         white_percent = european_american/population,
         pov_percent = poverty/population,
         asian_percent = asian_american/population,
         bachelor_percent = edu_bachelors/(population-children),
         grad_percent = edu_graduate/(population - children),
         hs_percent = edu_highschool/(population - children),
         nohs_percent = edu_no_highschool/(population - children),
         employed_percent = employed/(population - children),
         unemployed_percent = unemployed/(population - children),
         child_percent = children/(population - children)
```

```
  ) %>%
  na.omit()

#split into training/test data
sample_size <- floor(0.75 * nrow(Votes_data))
set.seed(123)
train_ind <- sample(seq_len(nrow(Votes_data)), size = sample_size)

Votes_data <- Votes_data[train_ind, ]
Votes_data_test <- Votes_data[-train_ind, ]


################################################
#Markov Random Fields Geo approach
################################################

#read in shap files for CTs
LA_shapes <- tracts("CA", "Los Angeles", cb = TRUE)


#check for empty geometries
which(is.na(st_dimension(Votes_data$geometry)))


#convert to neighborhood
nb <- poly2nb(Votes_data$geometry, row.names = Votes_data$GEOID)
names(nb) <- attr(nb, "region.id")
str(nb[1:6])

#fit GAM for PERM using MRF with a beta response
#https://stat.ethz.ch/R-manual/R-patched/library/mgcv/html/Beta.html

ctrl <- gam.control(nthreads = 6) # use 6 parallel threads, reduce if
   fewer physical CPU cores
m1 <- gam(turnout_percent ~ s(as.factor(GEOID), bs = 'mrf', k = 20, xt =
```

```
   list(nb = nb)), # define MRF smooth
117          data = Votes_data,
118          method = 'REML',
119          # fast version of REML smoothness selection
120          family = betar()) # fit a beta regression
121
122 viz_data <- Votes_data %>%
123   mutate(new_vals = predict(m1, type = 'response'))
124
125
126 summary(m1)
127 gam.check(m1)
128
129 #visualize
130 #cross validation
131 #https://rdrr.io/cran/gamclass/man/CVgam.html
132
133 ##############################################################
134 #Let's start looking at univariate relationships and plotting them out
135 ##############################################################
136 ##############################
137 #avg earnings
138 ##############################
139 earnings <- gam(turnout_percent ~ s(earnings_median) , data = Votes_data,
    family = betar())
140 summary(earnings)
141 plot(earnings, residuals = T)
142
143 #partial effects plot
144 p <- predict(earnings, type="lpmatrix")
145 beta <- coef(earnings)[grepl("median", names(coef(earnings)))]
146 s <- p[,grepl("median", colnames(p))] %*% beta
147 ggplot(data=cbind.data.frame(s, Votes_data$earnings_median), aes(x=Votes_
    data$earnings_median, y=s)) + geom_line()
148
```

```
#fitted vs predictors

fit_data <- data.frame(cbind(earnings$fitted.values, earnings$y, Votes_
    data$earnings_median))
names(fit_data) <- c("yhat", "y", "earnings")

fit_data %>%
  ggplot() +
  geom_point(aes(x = earnings, y = y)) +
  geom_line(aes(x = earnings, y = yhat), color= "red")

#########################################################
#poverty percent
#########################################################


pov <- gam(turnout_percent ~ s(pov_percent) , data = Votes_data, family =
    betar())
summary(pov)
plot(pov)

#partial effects plot
p <- predict(pov, type="lpmatrix")
beta <- coef(pov)[grepl("pov", names(coef(pov)))]
s <- p[,grepl("pov", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$pov_percent), aes(x=Votes_data$
    pov_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(pov$fitted.values, pov$y, Votes_data$pov_
    percent))
names(fit_data) <- c("yhat", "y", "pov")

fit_data %>%
```

```
  ggplot() +
  geom_point(aes(x = pov, y = y)) +
  geom_line(aes(x = pov, y = yhat), color= "red")

#########################################################
#unemployed percent
#########################################################


unemployed <- gam(turnout_percent ~ s(unemployed_percent) , data = Votes_
    data, family = betar())
summary(unemployed)
plot(unemployed)

#partial effects plot
p <- predict(unemployed, type="lpmatrix")
beta <- coef(unemployed)[grepl("unemployed", names(coef(unemployed)))]
s <- p[,grepl("unemployed", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$unemployed_percent), aes(x=
    Votes_data$unemployed_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(unemployed$fitted.values, unemployed$y, Votes
    _data$unemployed_percent))
names(fit_data) <- c("yhat", "y", "unemployed")

fit_data %>%
  ggplot() +
  geom_point(aes(x = unemployed, y = y)) +
  geom_line(aes(x = unemployed, y = yhat), color= "red")

###################################
#age
###################################
```

```
212
213 age <- gam(turnout_percent ~ s(age_median) , data = Votes_data, family =
        betar())
214 summary(age)
215 plot(earnings)
216
217 #partial effects plot
218 p <- predict(age, type="lpmatrix")
219 beta <- coef(age)[grepl("median", names(coef(age)))]
220 s <- p[,grepl("median", colnames(p))] %*% beta
221 ggplot(data=cbind.data.frame(s, Votes_data$age_median), aes(x=Votes_data$
        age_median, y=s)) + geom_line()
222
223 #fitted vs predictors
224
225 fit_data <- data.frame(cbind(age$fitted.values, age$y, Votes_data$age_
        median))
226 names(fit_data) <- c("yhat", "y", "age")
227
228 fit_data %>%
229   ggplot() +
230   geom_point(aes(x = age, y = y)) +
231   geom_line(aes(x = age, y = yhat), color= "red")
232
233 #we see a linear relationship between age and PERM %
234
235 ##########################################################
236 #hispanic percent
237 ##########################################################
238
239 hispanic <- gam(turnout_percent ~ s(hispanic_percent) , data = Votes_data,
        family = betar())
240 summary(hispanic)
241 plot(hispanic)
242
```

```
#partial effects plot
p <- predict(hispanic, type="lpmatrix")
beta <- coef(hispanic)[grepl("hispanic", names(coef(hispanic)))]
s <- p[,grepl("hispanic", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$hispanic_percent), aes(x=Votes_
    data$hispanic_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(hispanic$fitted.values, hispanic$y, Votes_
    data$hispanic_percent))
names(fit_data) <- c("yhat", "y", "hispanic")

fit_data %>%
  ggplot() +
  geom_point(aes(x = hispanic, y = y)) +
  geom_line(aes(x = hispanic, y = yhat), color= "red")

#we see a linear relationship between age and PERM %

##########################################################
#black percent
##########################################################


black <- gam(turnout_percent ~ s(black_percent) , data = Votes_data,
    family = betar())
summary(black)
plot(black)

#partial effects plot
p <- predict(black, type="lpmatrix")
beta <- coef(black)[grepl("black", names(coef(black)))]
s <- p[,grepl("black", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$black_percent), aes(x=Votes_
```

```
    data$black_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(black$fitted.values, black$y, Votes_data$
    black_percent))
names(fit_data) <- c("yhat", "y", "black")

fit_data %>%
  ggplot() +
  geom_point(aes(x = black, y = y)) +
  geom_line(aes(x = black, y = yhat), color= "red")


#########################################################
#asian percent
#########################################################


asian <- gam(turnout_percent ~ s(asian_percent) , data = Votes_data,
    family = betar())
summary(asian)
plot(asian)

#partial effects plot
p <- predict(asian, type="lpmatrix")
beta <- coef(asian)[grepl("asian", names(coef(asian)))]
s <- p[,grepl("asian", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$asian_percent), aes(x=Votes_
    data$asian_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(asian$fitted.values, asian$y, Votes_data$
    asian_percent))
```

```r
names(fit_data) <- c("yhat", "y", "asian")

fit_data %>%
  ggplot() +
  geom_point(aes(x = asian, y = y)) +
  geom_line(aes(x = asian, y = yhat), color= "red")


#########################################################
#hs percent
#########################################################


hs <- gam(turnout_percent ~ s(hs_percent) , data = Votes_data, family =
    betar())
summary(hs)
plot(hs)

#partial effects plot
p <- predict(hs, type="lpmatrix")
beta <- coef(hs)[grepl("hs", names(coef(hs)))]
s <- p[,grepl("hs", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data$hs_percent), aes(x=Votes_data$
    hs_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(hs$fitted.values, hs$y, Votes_data$hs_percent
    ))
names(fit_data) <- c("yhat", "y", "hs")

fit_data %>%
  ggplot() +
  geom_point(aes(x = hs, y = y)) +
  geom_line(aes(x = hs, y = yhat), color= "red")
```

```
337
338 #doesn't appear too meaningful
339
340 ##########################################################
341 #bachelor percent
342 ##########################################################
343
344
345 bachelor <- gam(turnout_percent ~ s(bachelor_percent) , data = Votes_data,
        family = betar())
346 summary(bachelor)
347 plot(bachelor)
348
349 #partial effects plot
350 p <- predict(bachelor, type="lpmatrix")
351 beta <- coef(bachelor)[grepl("bachelor", names(coef(bachelor)))]
352 s <- p[,grepl("bachelor", colnames(p))] %*% beta
353 ggplot(data=cbind.data.frame(s, Votes_data$bachelor_percent), aes(x=Votes_
       data$bachelor_percent, y=s)) + geom_line()
354
355 #fitted vs predictors
356
357 fit_data <- data.frame(cbind(bachelor$fitted.values, bachelor$y, Votes_
       data$bachelor_percent))
358 names(fit_data) <- c("yhat", "y", "bachelor")
359
360 fit_data %>%
361   ggplot() +
362   geom_point(aes(x = bachelor, y = y)) +
363   geom_line(aes(x = bachelor, y = yhat), color= "red")
364
365 ##########################################################
366 #nohs percent
367 ##########################################################
368
```

```
369
370 nohs <- gam(turnout_percent ~ s(nohs_percent) , data = Votes_data, family
    = betar())
371 summary(nohs)
372 plot(nohs)
373
374 #partial effects plot
375 p <- predict(nohs, type="lpmatrix")
376 beta <- coef(nohs)[grepl("nohs", names(coef(nohs)))]
377 s <- p[,grepl("nohs", colnames(p))] %*% beta
378 ggplot(data=cbind.data.frame(s, Votes_data$nohs_percent), aes(x=Votes_data
    $nohs_percent, y=s)) + geom_line()
379
380 #fitted vs predictors
381
382 fit_data <- data.frame(cbind(nohs$fitted.values, nohs$y, Votes_data$nohs_
    percent))
383 names(fit_data) <- c("yhat", "y", "nohs")
384
385 fit_data %>%
386   ggplot() +
387   geom_point(aes(x = nohs, y = y)) +
388   geom_line(aes(x = nohs, y = yhat), color= "red")
389
390
391
392 ##########################################################
393 #child percent
394 ##########################################################
395
396
397 child <- gam(turnout_percent ~ s(child_percent) , data = Votes_data,
    family = betar())
398 summary(child)
399 plot(child)
```

```
400
401 #partial effects plot
402 p <- predict(child, type="lpmatrix")
403 beta <- coef(child)[grepl("child", names(coef(child)))]
404 s <- p[,grepl("child", colnames(p))] %*% beta
405 ggplot(data=cbind.data.frame(s, Votes_data$child_percent), aes(x=Votes_
    data$child_percent, y=s)) + geom_line()
406
407 #fitted vs predictors
408
409 fit_data <- data.frame(cbind(child$fitted.values, child$y, Votes_data$
    child_percent))
410 names(fit_data) <- c("yhat", "y", "child")
411
412 fit_data %>%
413   ggplot() +
414   geom_point(aes(x = child, y = y)) +
415   geom_line(aes(x = child, y = yhat), color= "red")
416
417 #doesn't look like a strong predictor
418
419
420
421 ##################################################################
422 #correlation matrix
423 ##################################################################
424
425 cor_data <- Votes_data %>%
426   select(20,2,4, 23:31)
427
428 chart.Correlation(cor_data, histogram=TRUE, pch=19)
429
430
431 ##################################################################
432 #
```

```
#
#Start modelling
#
#################################################################

#####################################################################
#model with just lags
#####################################################################
lag_mod <- gam(turnout_percent ~ s(lags)  , data = Votes_data, family =
    betar())
summary(lag_mod)
plot(lag_mod)

#try to better understand relationship with lags by fitting simple linear
    model
lag_line <- lm(turnout_percent ~ lags, data = Votes_data)
summary(lag_line)

#####################################################################
#build a model with lags and predictor variables
#####################################################################
full_gam <- gam(turnout_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
                  s(black_percent) + s(white_percent) + s(pov_percent) + s
    (asian_percent) +
                  s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent),
                data = Votes_data,
                family = betar(),
                select = T)

summary(full_gam)
gam.check(full_gam)

#lets ad geo data into this
```

```
463
464 full_gam_geo <- gam(turnout_percent ~ s(age_median) + s(earnings_median) +
        s(hispanic_percent) +
465                     s(black_percent) + s(white_percent) + s(pov_percent)
        + s(asian_percent) +
466                     s(bachelor_percent) + s(grad_percent) + s(hs_percent
    ) + s(nohs_percent) +
467                     s(child_percent) +
468                     s(as.factor(GEOID), bs = 'mrf', k = 20, xt = list(nb
        = nb)),
469                   data = Votes_data,
470                   family = betar(),
471                   select = T)
472
473 summary(full_gam)
474 gam.check(full_gam)
475 plot(full_gam)
476
477 #lets ad lags into this
478 full_gam <- gam(turnout_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
479                   s(black_percent) + s(white_percent) + s(pov_percent) + s
    (asian_percent) +
480                   s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent) +
481                   s(child_percent) +
482                   s(as.factor(GEOID), bs = 'mrf', k = 20, xt = list(nb =
    nb)) +
483                   s(lags),
484                 data = Votes_data,
485                 family = betar(),
486                 select = T)
487
488 summary(full_gam)
489 gam.check(full_gam)
```

```
plot(full_gam)


#remove geo
full_gam <- gam(turnout_percent ~ s(age_median) + s(earnings_median) + s(
    hispanic_percent) +
                  s(black_percent) + s(white_percent) + s(pov_percent) + s
    (asian_percent) +
                  s(bachelor_percent) + s(grad_percent) + s(hs_percent) +
    s(nohs_percent) +
                  s(child_percent) +
                  s(lags),
                data = Votes_data,
                family = betar(),
                select = T)

summary(full_gam)
gam.check(full_gam)
plot(full_gam)

#######################################################
#performance

predict_perm_test <- predict.gam(full_gam, Votes_data_test, se.fit = T,
    type = "response")
resids_test <- cbind(predict_perm_test$fit, Votes_data_test$turnout_
    percent)
resids_test <- as.data.frame(resids_test)
names(resids_test) <- c("pred", "obs")
resids_test <- resids_test %>%
  mutate(resids = obs - pred)

rmse_full <- RMSE(obs = resids_test$obs, pred = resids_test$pred)
#just for fun I want to double check
sqrt(sum(resids_test$resids^2)/nrow(resids_test))
```

```
#more or less iid normally distributed residual
plot(resids_test$obs, resids_test$resids)
hist(resids_test$resids)
############################################################

############################################################
#strip variables that do not appear to be useful
diminished_gam <- gam(turnout_percent ~ s(age_median)  + s(hispanic_
    percent) +
                         s(black_percent) + s(pov_percent) + s(asian_
    percent) +
                         s(bachelor_percent) + s(grad_percent) +
                         s(child_percent) +
                         s(lags),
                       data = Votes_data,
                       family = betar(),
                       select = T)

summary(diminished_gam)
gam.check(diminished_gam)

#performance for gam_diminished
predict_perm_test2 <- predict.gam(diminished_gam, Votes_data_test, se.fit
    = T, type = "response")
resids_test2 <- cbind(predict_perm_test2$fit, Votes_data_test$turnout_
    percent)
resids_test2 <- as.data.frame(resids_test2)
names(resids_test2) <- c("pred", "obs")
resids_test2 <- resids_test2 %>%
  mutate(resids = obs - pred)

rmse_dim <- RMSE(obs = resids_test2$obs, pred = resids_test2$pred)
#more or less iid normally distributed residual
plot(resids_test2$obs, resids_test2$resids)
hist(resids_test2$resids)
```

```
551
552 ##################################################
553 #lastly, just check rmse for lag model
554
555 predict_perm_test3 <- predict.gam(lag_mod, Votes_data_test, se.fit = T,
    type = "response")
556 resids_test3 <- cbind(predict_perm_test3$fit, Votes_data_test$turnout_
    percent)
557 resids_test3 <- as.data.frame(resids_test3)
558 names(resids_test3) <- c("pred", "obs")
559 resids_test3 <- resids_test3%>%
560   mutate(resids = obs - pred)
561
562 rmse_lag <- RMSE(obs = resids_test3$obs, pred = resids_test3$pred)
563 #more or less iid normally distributed residual
564 plot(resids_test3$obs, resids_test3$resids)
565 hist(resids_test3$resids)
566 ##########################################################
567 #
568 #fit on 2018 data and see how it performs out of box
569 ##########################################################
570 #2018 data manipulation
571 Votes_data_2018 <- all_data_2018 %>%
572   mutate(turnout_percent = total_votes/total_regs,
573          children = (over_6_under_18 + under_6)) %>%
574   filter(population > 100) %>%
575   dplyr::select(african_american,
576                 age_median,
577                 asian_american,
578                 earnings_median,
579                 edu_bachelors,
580                 edu_graduate,
581                 edu_highschool,
582                 edu_no_highschool,
583                 employed,
```

```
                european_american,
                hispanic_latino,
                native_american,
                pacific_islander,
                population,
                poverty,
                total_female,
                total_male,
                unemployed,
                children,
                turnout_percent,
                GEOID,
                geometry
  )


#add in lag variables

lags_2016 <- all_data_2016 %>%
  mutate(lags = total_votes/total_regs) %>%
  select(lags, GEOID)


Votes_data_2018 <- Votes_data_2018 %>%
  left_join(lags_2016, "GEOID")

#start modeling
Votes_data_2018 <- Votes_data_2018 %>%
  mutate(hispanic_percent = hispanic_latino/population,
         black_percent = african_american/population,
         white_percent = european_american/population,
         pov_percent = poverty/population,
         asian_percent = asian_american/population,
         bachelor_percent = edu_bachelors/(population-children),
         grad_percent = edu_graduate/(population - children),
```

```
        hs_percent = edu_highschool/(population - children),
        nohs_percent = edu_no_highschool/(population - children),
        employed_percent = employed/(population - children),
        unemployed_percent = unemployed/(population - children),
        child_percent = children/(population - children)
  ) %>%
  na.omit()

predict_perm_2018 <- predict.gam(diminished_gam, Votes_data_2018, se.fit =
    T, type = "response")
resids_2018 <- cbind(predict_perm_2018$fit, Votes_data_2018$turnout_
   percent, as.numeric(Votes_data_2018$GEOID))
resids_2018 <- as.data.frame(resids_2018)
names(resids_2018) <- c("pred", "obs", "geo")
resids_2018 <- resids_2018 %>%
  mutate(resids = pred - obs)


#systematically underpredicts and RMSE doubles
RMSE(obs = resids_2018$obs, pred = resids_2018$pred)
#just for fun I want to double check
sqrt(sum(resids_2018$resids^2)/nrow(resids_2018))
#more or less iid normally distributed residual
plot(resids_2018$obs, resids_2018$resids)
hist(resids_2018$resids)

#now let's look at it using the lag model.
predict_perm_20182 <- predict.gam(lag_mod, Votes_data_2018, se.fit = T,
   type = "response")
resids_20182 <- cbind(predict_perm_20182$fit, Votes_data_2018$turnout_
   percent, as.numeric(Votes_data_2018$GEOID))
resids_20182 <- as.data.frame(resids_20182)
names(resids_20182) <- c("obs", "pred", "geo")
resids_20182 <- resids_20182 %>%
  mutate(resids = pred - obs)
```

```
650
651
652 #systematically underpredicts and RMSE doubles
653 RMSE(obs = resids_20182$obs, pred = resids_20182$pred)
654 #more or less iid normally distributed residual
655 plot(resids_2018$obs, resids_2018$resids)
656 hist(resids_2018$resids)
657 #here we see that the lag model predicts worse, but not by that much and
       still systematically under predicts
658
659 ################################################################
660 #lets look at 2016 and 2018 as the outcome again, but let's build the
       model with our outcome variable
661 #standardized
662 ################################################################
663 #Let's look at how the model performed if we interpret the out put as an
       ordinal output
664 #we don't need to rerun the model , we just need to order the predictions
       and the actual numbers and see how we did
665 ################################################################
666
667
668 #let's look at resids_2018 but change obs, pred to ordinal and see what
       the resids are
669
670 ordinal_resids_2018 <- resids_2018 %>%
671   mutate(obs2 = percent_rank(obs)) %>%
672   mutate(pred2 = percent_rank(pred)) %>%
673   mutate(resids2 = obs2-pred2)
674
675
676 RMSE(pred = ordinal_resids_2018$pred2, obs = ordinal_resids_2018$obs2)
677 #when estimating the percentile rank its RMSE is .09
678 #residuals are roughly normally distributed
679 plot(ordinal_resids_2018$resids2)
```

```r
hist(ordinal_resids_2018$resids2)

#the diminished model does a decent job of predicting the percentiles for voter turnout for the next election.
#let's see how the lag model did
ordinal_resids_20182 <- resids_20182 %>%
  mutate(obs2 = percent_rank(obs)) %>%
  mutate(pred2 = percent_rank(pred)) %>%
  mutate(resids2 = obs2-pred2)


RMSE(pred = ordinal_resids_20182$pred2, obs = ordinal_resids_20182$obs2)
#when estimating the percentile rank its RMSE is .09
#residuals are roughly normally distributed
plot(ordinal_resids_2018$resids2)
hist(ordinal_resids_2018$resids2)


#the full model does a better job of predicting the percentiles for voter turnout for the next election
#than just using the lag.
#We should be able to use this to get a reasonable estimate for 2020 voter turnout percentage percentiles

##################################################
#winning model diagnostics
##################################################
#original
#partial effects interpretation: https://www.researchgate.net/figure/Generalized-additive-model-GAM-plots-showing-the-partial-effects-of-selected_fig4_324422761
gam.check(full_gam)
plot(full_gam)
#ordinal
plot(ordinal_resids_2018$resids2)
```

```r
hist(ordinal_resids_2018$resids2)

#What happens if we build a model where percentiles are the outcome variable?
Votes_data_percentile <- Votes_data %>%
  mutate(turnout_percent = percent_rank(turnout_percent))

Votes_data_2018_percentile <- Votes_data_2018 %>%
  mutate(turnout_percent = percent_rank(turnout_percent))

full_gam_percentils <- gam(turnout_percent ~ s(age_median) + s(earnings_median) + s(hispanic_percent) +
                  s(black_percent) + s(white_percent) + s(pov_percent) + s(asian_percent) +
                  s(bachelor_percent) + s(grad_percent) + s(hs_percent) + s(nohs_percent) +
                  s(child_percent) +
                  s(lags),
                data = Votes_data_percentile,
                family = betar(),
                select = T)

summary(full_gam)
gam.check(full_gam)
plot(full_gam)

#predict 2018
predict_perm_2018_percentile <- predict.gam(full_gam_percentils, Votes_data_2018_percentile, se.fit = T, type = "response")
resids_2018 <- cbind(predict_perm_2018$fit, Votes_data_2018$turnout_percent, as.numeric(Votes_data_2018$GEOID))
resids_2018 <- as.data.frame(resids_2018)
names(resids_2018) <- c("pred", "obs", "geo")
resids_2018 <- resids_2018 %>%
  mutate(resids = pred - obs)
```

```r
RMSE(obs = resids_2018$obs, pred = resids_2018$pred)

#####################################################
#univariate relationships for 2018
#####################################################
##############################
#avg earnings
##############################
earnings <- gam(turnout_percent ~ s(earnings_median) , data = Votes_data_
    2018, family = betar())
summary(earnings)
plot(earnings, residuals = T)

#partial effects plot
p <- predict(earnings, type="lpmatrix")
beta <- coef(earnings)[grepl("median", names(coef(earnings)))]
s <- p[,grepl("median", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$earnings_median), aes(x=
    Votes_data_2018$earnings_median, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(earnings$fitted.values, earnings$y, Votes_
    data_2018$earnings_median))
names(fit_data) <- c("yhat", "y", "earnings")

fit_data %>%
  ggplot() +
  geom_point(aes(x = earnings, y = y)) +
  geom_line(aes(x = earnings, y = yhat), color= "red") +
  ylab("Turnout Percentage") +
  xlab("Average Yearly Earnings")

#####################################################
#poverty percent
```

```
771 ########################################################
772
773
774 pov <- gam(turnout_percent ~ s(pov_percent) , data = Votes_data_2018,
        family = betar())
775 summary(pov)
776 plot(pov)
777
778 #partial effects plot
779 p <- predict(pov, type="lpmatrix")
780 beta <- coef(pov)[grepl("pov", names(coef(pov)))]
781 s <- p[,grepl("pov", colnames(p))] %*% beta
782 ggplot(data=cbind.data.frame(s, Votes_data_2018$pov_percent), aes(x=Votes_
        data_2018$pov_percent, y=s)) + geom_line()
783
784 #fitted vs predictors
785
786 fit_data <- data.frame(cbind(pov$fitted.values, pov$y, Votes_data_2018$pov
        _percent))
787 names(fit_data) <- c("yhat", "y", "pov")
788
789 fit_data %>%
790   ggplot() +
791   geom_point(aes(x = pov, y = y)) +
792   geom_line(aes(x = pov, y = yhat), color= "red") +
793   ylab("Turnout Percentage") +
794   xlab("Poverty Rate")
795
796
797 ########################################################
798 #unemployed percent
799 ########################################################
800
801
802 unemployed <- gam(turnout_percent ~ s(unemployed_percent) , data = Votes_
```

```
    data_2018, family = betar())
summary(unemployed)
plot(unemployed)

#partial effects plot
p <- predict(unemployed, type="lpmatrix")
beta <- coef(unemployed)[grepl("unemployed", names(coef(unemployed)))]
s <- p[,grepl("unemployed", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$unemployed_percent), aes(x
    =Votes_data_2018$unemployed_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(unemployed$fitted.values, unemployed$y, Votes
    _data_2018$unemployed_percent))
names(fit_data) <- c("yhat", "y", "unemployed")

fit_data %>%
  ggplot() +
  geom_point(aes(x = unemployed, y = y)) +
  geom_line(aes(x = unemployed, y = yhat), color= "red")

###################################
#age
###################################

age <- gam(turnout_percent ~ s(age_median) , data = Votes_data_2018,
    family = betar())
summary(age)
plot(earnings)

#partial effects plot
p <- predict(age, type="lpmatrix")
beta <- coef(age)[grepl("median", names(coef(age)))]
s <- p[,grepl("median", colnames(p))] %*% beta
```

```
ggplot(data=cbind.data.frame(s, Votes_data_2018$age_median), aes(x=Votes_
    data_2018$age_median, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(age$fitted.values, age$y, Votes_data_2018$age
    _median))
names(fit_data) <- c("yhat", "y", "age")

fit_data %>%
  ggplot() +
  geom_point(aes(x = age, y = y)) +
  geom_line(aes(x = age, y = yhat), color= "red") +
  ylab("Turnout Percentage") +
  xlab("Average Age")

#we see a linear relationship between age and PERM %

#########################################################
#hispanic percent
#########################################################

hispanic <- gam(turnout_percent ~ s(hispanic_percent) , data = Votes_data_
    2018, family = betar())
summary(hispanic)
plot(hispanic)

#partial effects plot
p <- predict(hispanic, type="lpmatrix")
beta <- coef(hispanic)[grepl("hispanic", names(coef(hispanic)))]
s <- p[,grepl("hispanic", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$hispanic_percent), aes(x=
    Votes_data_2018$hispanic_percent, y=s)) + geom_line()

#fitted vs predictors
```

```
865
866 fit_data <- data.frame(cbind(hispanic$fitted.values, hispanic$y, Votes_
    data_2018$hispanic_percent))
867 names(fit_data) <- c("yhat", "y", "hispanic")
868
869 fit_data %>%
870   ggplot() +
871   geom_point(aes(x = hispanic, y = y)) +
872   geom_line(aes(x = hispanic, y = yhat), color= "red") +
873   ylab("Turnout Percentage") +
874   xlab("Hispanic Rate")
875
876 #we see a linear relationship between age and PERM %
877
878 ##########################################################
879 #black percent
880 ##########################################################
881
882
883 black <- gam(turnout_percent ~ s(black_percent) , data = Votes_data_2018,
    family = betar())
884 summary(black)
885 plot(black)
886
887 #partial effects plot
888 p <- predict(black, type="lpmatrix")
889 beta <- coef(black)[grepl("black", names(coef(black)))]
890 s <- p[,grepl("black", colnames(p))] %*% beta
891 ggplot(data=cbind.data.frame(s, Votes_data_2018$black_percent), aes(x=
    Votes_data_2018$black_percent, y=s)) + geom_line()
892
893 #fitted vs predictors
894
895 fit_data <- data.frame(cbind(black$fitted.values, black$y, Votes_data_2018
    $black_percent))
```

```r
names(fit_data) <- c("yhat", "y", "black")

fit_data %>%
  ggplot() +
  geom_point(aes(x = black, y = y)) +
  geom_line(aes(x = black, y = yhat), color= "red")


#########################################################
#asian percent
#########################################################


asian <- gam(turnout_percent ~ s(asian_percent) , data = Votes_data_2018,
    family = betar())
summary(asian)
plot(asian)

#partial effects plot
p <- predict(asian, type="lpmatrix")
beta <- coef(asian)[grepl("asian", names(coef(asian)))]
s <- p[,grepl("asian", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$asian_percent), aes(x=
    Votes_data_2018$asian_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(asian$fitted.values, asian$y, Votes_data_2018
    $asian_percent))
names(fit_data) <- c("yhat", "y", "asian")

fit_data %>%
  ggplot() +
  geom_point(aes(x = asian, y = y)) +
  geom_line(aes(x = asian, y = yhat), color= "red")
```

```
928
929
930 ########################################################
931 #hs percent
932 ########################################################
933
934
935 hs <- gam(turnout_percent ~ s(hs_percent) , data = Votes_data_2018, family
        = betar())
936 summary(hs)
937 plot(hs)
938
939 #partial effects plot
940 p <- predict(hs, type="lpmatrix")
941 beta <- coef(hs)[grepl("hs", names(coef(hs)))]
942 s <- p[,grepl("hs", colnames(p))] %*% beta
943 ggplot(data=cbind.data.frame(s, Votes_data_2018$hs_percent), aes(x=Votes_
       data_2018$hs_percent, y=s)) + geom_line()
944
945 #fitted vs predictors
946
947 fit_data <- data.frame(cbind(hs$fitted.values, hs$y, Votes_data_2018$hs_
       percent))
948 names(fit_data) <- c("yhat", "y", "hs")
949
950 fit_data %>%
951   ggplot() +
952   geom_point(aes(x = hs, y = y)) +
953   geom_line(aes(x = hs, y = yhat), color= "red") +
954   ylab("Turnout Percentage") +
955   xlab("High School Graduate Rate")
956
957 #doesn't appear too meaningful
958
959 ########################################################
```

```r
#bachelor percent
#########################################################


bachelor <- gam(turnout_percent ~ s(bachelor_percent) , data = Votes_data_
    2018, family = betar())
summary(bachelor)
plot(bachelor)

#partial effects plot
p <- predict(bachelor, type="lpmatrix")
beta <- coef(bachelor)[grepl("bachelor", names(coef(bachelor)))]
s <- p[,grepl("bachelor", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$bachelor_percent), aes(x=
    Votes_data_2018$bachelor_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(bachelor$fitted.values, bachelor$y, Votes_
    data_2018$bachelor_percent))
names(fit_data) <- c("yhat", "y", "bachelor")

fit_data %>%
  ggplot() +
  geom_point(aes(x = bachelor, y = y)) +
  geom_line(aes(x = bachelor, y = yhat), color= "red") +
  ylab("Turnout Percentage") +
  xlab("Bachelor's Degree Rate")

#########################################################
#nohs percent
#########################################################


nohs <- gam(turnout_percent ~ s(nohs_percent) , data = Votes_data_2018,
```

```
    family = betar())
summary(nohs)
plot(nohs)

#partial effects plot
p <- predict(nohs, type="lpmatrix")
beta <- coef(nohs)[grepl("nohs", names(coef(nohs)))]
s <- p[,grepl("nohs", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$nohs_percent), aes(x=Votes
    _data_2018$nohs_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(nohs$fitted.values, nohs$y, Votes_data_2018$
    nohs_percent))
names(fit_data) <- c("yhat", "y", "nohs")

fit_data %>%
  ggplot() +
  geom_point(aes(x = nohs, y = y)) +
  geom_line(aes(x = nohs, y = yhat), color= "red") +
  ylab("Turnout Percentage") +
  xlab("No High School Graduaate Rate")

#########################################################
#child percent
#########################################################


child <- gam(turnout_percent ~ s(child_percent) , data = Votes_data_2018,
    family = betar())
summary(child)
plot(child)

#partial effects plot
```

```
p <- predict(child, type="lpmatrix")
beta <- coef(child)[grepl("child", names(coef(child)))]
s <- p[,grepl("child", colnames(p))] %*% beta
ggplot(data=cbind.data.frame(s, Votes_data_2018$child_percent), aes(x=
    Votes_data_2018$child_percent, y=s)) + geom_line()

#fitted vs predictors

fit_data <- data.frame(cbind(child$fitted.values, child$y, Votes_data_2018
    $child_percent))
names(fit_data) <- c("yhat", "y", "child")

fit_data %>%
  ggplot() +
  geom_point(aes(x = child, y = y)) +
  geom_line(aes(x = child, y = yhat), color= "red") +
  ylab("Turnout Percentage") +
  xlab("Percent Children")

#########################################################
#prediction map for 2020
#########################################################

Votes_data_2020 <- Votes_data_2018 %>%
  mutate(lags = turnout_percent)

#it seems reasonable to rerun the model built on 2016 and 2018 to try and
    campture the most recent political
#atmosphere

diminished_gam_2018 <- gam(turnout_percent ~ s(age_median)  + s(hispanic_
    percent) +
                             s(black_percent) + s(pov_percent) + s(asian_
    percent) +
                             s(bachelor_percent) + s(grad_percent) +
```

```
                              s(child_percent) +
                              s(lags),
                            data = Votes_data_2018,
                            family = betar(),
                            select = T)

summary(diminished_gam)
gam.check(diminished_gam)

#predict
predict_perm_2020 <- predict.gam(diminished_gam_2018, Votes_data_2020, se.
    fit = T, type = "response")

Votes_data_2020_full <- Votes_data_2020 %>%
  mutate(predicts_percentiles = percent_rank(predict_perm_2020$fit))

#map
Votes_data_2020_full %>%
  select(geometry,
         predicts_percentiles) %>%
  gather(key = "variable", value = "estimate", -geometry) %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(aes(geometry = geometry),color = NA) +
  coord_sf(ylim = c(33.7,35)) +
  scale_fill_viridis_c(option = "plasma") +
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = .5)) +
  labs(title = "Turnout Percentile Predictions: 2020") -> turnout_map_2020

#END SCRIPT
```

References

[1] Brians, Craig Leonard, and Bernard Grofman. "Election Day Registration's Effect on U.S. Voter Turnout." *Social Science Quarterly*, vol. 82, no. 1, 2001, pp. 170–183. JSTOR, www.jstor.org/stable/42955710.

[2] Famighetti, Christopher. "Long Voting Lines: Explained." *Brenon Center for Justice*. 4 November, 2016. https://www.brennancenter.org/our-work/research-reports/long-voting-lines-explained

[3] BRADY, HENRY E., and JOHN E. MCNULTY. "Turning Out to Vote: The Costs of Finding and Getting to the Polling Place." *The American Political Science Review*, vol. 105, no. 1, 2011, pp. 115–134. JSTOR, www.jstor.org/stable/41480830.

[4] Judy Ly, 2019. "Anonymous Voter Data: Los Angeles County 2012-2018." Los Angeles County Registrar Recorder/County Clerk.

[5] Los Angles Almanac 2019. "Eligible & Registered Voters, 1990-2018 Including Turnout & Votes by Mail." http://www.laalmanac.com/election/el02.php

[6] "Vote By Mail." Alex Padilla California Secretary of State. https://www.sos.ca.gov/elections/voter-registration/vote-mail/#hist

[7] Marx, Brian D., and Paul HC Eilers. "Direct generalized additive modeling with penalized likelihood." Computational Statistics & Data Analysis 28.2 (1998): 193-209.

[8] Larsen, Kim. "GAM : The Predictive Modeling Silver Bullet." (2015). https://pdfs.semanticscholar.org/dea4/adaaf06e6fc99179a2620b7a031188c6e532.pdf

[9] Rakich, Nathaniel. "Early-Voting Laws Probably Don't Boost Turnout." Fivethirtyeight. 30 January, 2019. https://fivethirtyeight.com/features/early-voting-laws-probably-dont-boost-turnout/