PREDICTING PRESIDENTIAL ELECTION TURNOUT AMONG NEW VOTERS IN NORTH CAROLINA

A Capstone presented to the Faculty of University of Wisconsin in partial fulfillment of the requirements for the degree of

Master of Science

in

Data Science

by

Peter Torpey

Green Bay, WI

December, 2022

Abstract

This case study uses the North Carolina State Board of Elections data to predict presidential election turnout among new voters. The primary objective is to introduce the reader to the valuable information held within state provided voter files and explore methods that can be utilized to identify these voters in future elections. Of the over 890,000 new registrants in the state of North Carolina during 2020, a sample of 50,000 voters was selected at random and used for this case study. Out of the four models examined, Boosting produced the lowest misclassification error rate of 18.15% and its predictors were further analyzed. This information can be utilized by political campaigns, data companies, and other entities who specialize in political microtargeting to predict voter turnout.

Keywords: voter file, voter turnout, political microtargeting

Contents

List of Tables

List of Figures

Chapter 1: Introduction

Background

Predicting voter turnout in presidential elections has long been an interest in the political world. While the methods to predict turnout and election outcomes have existed for quite some time, only in recent decades has a serious process been developed based on numerous data sources. One of the most crucial elements to this process is the utilization of state provided voter files, which are rosters of important information each state uses to maintain a list of registered voters. These lists are continually getting updated with the introduction of new voters, who often sway the outcome of an election.

Problem statement

State voter files are rich with substantial information pertaining to the eligible voting population. Tapping into this information which is free and accessible to the public in most states, can identify a potential relationship in characteristics among voters in a given state. Modeling this information can be beneficial for political campaigns, voter targeting, and election prediction.

Every election cycle the dynamic of the voting population changes with the introduction of new registrants. Identifying attributes in these new voters and using data science methods can help pundits predict election turnout. The significance of new registrants and their impact on voter turnout and election outcomes will be further explored in Chapter 2.

The state of North Carolina was chosen as a case study for this project for the following reasons:

1. The Political significance of North Carolina in presidential elections.
2. The rich information contained in the state's voter file.
3. The accessibility and cost of the data.

All of these reasons will be expanded upon in Chapters 2 and 3 of this paper.

Inspiration

I have had a deep interest in United States presidential elections for quite some time. In my current position at Data Trust, I oversee the collection, development, and management of voter files from all fifty states, the District of Columbia, and a few US territories. While my position is heavily data warehouse-based, I have always had an interest in conducting analysis on this data to further gain insights into the substance of these files.

My initial awareness of machine learning was in 2017 when I enrolled in an upper division political science data analysis course during my undergraduate education. In this course, we studied how machine learning applications can address common political, social, and economic issues. Over the past few years I have grown a deeper appreciation for machine learning and how its ability to address the aforementioned subjects can be applied. I have read numerous data science case studies, taken courses, and performed projects to further my education on the subject. Using my

institutional knowledge of voter registration data, I believe this project is an excellent opportunity to apply machine learning on state provided voter file information.

Objectives and Purpose

The primary objectives of this project are to:

1. Describe the valuable information held within voter files
2. Review current literature of voter information, the political landscape of North Carolina, and data use in politics.
3. Examine data science methods that can be used for deeper analysis among state voter files.
4. Explore how these findings can be applied in real world applications.

Limitations

While the data in this project contains rich voter material, the data provided on the State Board of Elections website is incomplete. Political and commercial entities who already perform similar analysis on voter turnout typically have access to a wide breadth of information from various sources. Having access to consumer data, social media platforms, and political polling information are just a few examples of how current political data analysts are able to form a complete picture of the eligible voting population. This project lacks data from other sources outside of the North Carolina State Board of Elections website. Combining data from numerous sources would only help create a more honest picture of the voting population.

Organization of the Paper

The structure of this document will follow the specific presentation:

I. Introduction
   a. Background
   b. Problem statement
   c. Inspiration
   d. Objectives & Purpose
   e. Limitations
   f. Organization of Paper

II. Review of Literature
   a. Description of the Electoral College and relevance of battleground states
   b. The political landscape of North Carolina
   c. Voter file description
   d. History of data use in modern politics
   e. Significance of new voters

III. Political Machine Learning Methods
   a. Data Collection
   b. Data Flow Process
   c. Introduction to four machine learning methods used in case study

IV. Findings & Results
   a. Review results from all four models
   b. Explanation of best model

    c. Analysis of best fitting model

V. Summary and Conclusions

    a. Summary and takeaways

    b. Next steps and recommendations

Chapter 2: Review of Literature

The Electoral College and Relevance of Battleground States

The President of the United States is elected indirectly through the electoral college, rather than direct popular vote. The framers of the U.S. Constitution allocated electors to each state based on the number of senators plus representatives (Fon, 2004). The number of representatives and in turn electoral votes are based on a state's population. Electoral votes are awarded utilizing a winner-take-all system in 48 states plus the District of Columbia (Edwards, 2019). There are few exceptions however these are not relevant to this paper and will not be discussed.

The Electoral College has been hotly debated over the past few election cycles, with critics pointing out flaws in the current system. Much of these controversies are highlighted in Bolinger's article where he goes in depth covering flaws such as electoral overrepresentation in smaller states and presidential election victors losing the popular vote (Bolinger, 2007). The common theme expressed in these is the uneven distribution of power voters in specific states have over others.

Perhaps the most noticeable example of this unequal vote power distribution in the electoral college is the significance of battleground states. Commonly referred to as swing or purple states, battleground states are essentially where elections are won or lost. Each election, states can effectively be categorized into three groups: safe, leaning, or highly competitive (Gimpel et al., 2007). There are typically around 8-10 states classified in this highly competitive category. Battleground states consist of undecided voters who can be mobilized and won over by either political party (Johnson,

2005). The phenomenon of battleground states allows presidential candidates to concentrate all their time and resources on a select number of states whose results are projected to be close (Wolak, 2006). Past elections have shown that voter mobilization efforts are more intensive in these states as candidates pay more money, spend more time, and focus on winning over voters in swing states (McDonald, 2008).

The Political Landscape of North Carolina

The state of North Carolina has found itself on the front lines in battleground states over the past two decades. Once considered to be a republican stronghold at the presidential level, recent elections have shown North Carolina to be the textbook definition of a battleground state. Table 1 shows just how close presidential election results have been in the past four cycles, with the party percentage of results within 4% and an average of 1.75%.

Table 1

Vote Share by Party in Four most recent Presidential Elections

Year	Democratic	Republican	Difference
2020	48.59	49.93	1.34
2016	46.17	49.83	3.66
2012	48.35	50.39	2.04
2008	49.7	49.38	0.32

Note. Results are reflective of the North Carolina State Board of Elections website.

North Carolina is one of the most diverse states in the country politically, economically, and geographically. While the state lacks a single massive metropolitan area, it does contain multiple population centers such as Raleigh and Charlotte, creating urban and suburban areas. Between these concentrated population centers lies vast swaths of countryside, farmland, and small towns. The disparate terrain translates to equally diverse political, economic, and social areas of the state (Damore et al., 2021). This creates a truly diverse population with views and voters ranging from across the political spectrum.

Local and state level elections contain somewhat of a balance of power between the two main political parties. Democrats have held a large portion of governorships in the past half century while republicans typically perform well in state house races. Recent presidential elections are where North Carolina has seen a significant evolution. Due to the balance of power and closeness of elections, the authors of the novel, “Presidential Swing states: Why only ten matter” describe North Carolina as “The Bluest Red State in America.” As explained in their book, this nickname was given because “North Carolina is not as red as the Republican legislature would lead you to believe and not as blue as the protestors outside the General Assembly would like it to be (Schultz et al., 2015).” Many elections in the state are close, hard fought, and oftentimes sway from one party to the other between elections.

History of Data Use in Modern Politics

Data use in modern politics as we know it today really grew in the mid 2000’s. Prior to this time frame, voter registration information and census data was available to

the public in most cases, just not utilized. When rationalizing this topic in their article, Nickerson and Rogers attribute the slow embracement of big data in politics to two main reasons:

1. Storage and Computing power
2. Lack of interest in data analytics

Starting with the first reason, campaigns and parties lacked technology, storage, and infrastructure to harness the power of big data. These were large investments that most political entities did not have, and were not were not willing to make. Contributing to this problem was the fact that most states were not required to maintain electronic copies of voting information. While some states did, others did not, creating a lack of centralized system for maintaining voter information. Obtaining electronic records of this information would have been tedious and impractical for campaigns.

The second reason is perhaps more significant. Nickerson and Rogers point out prior to the 2000's data use had not yet made its way into the political world. Political candidates, consultants, and campaign leaders were often not trained in the hard sciences, resulting in strategies developed from non-quantitative fields. There were limited people with the proper technical skills required to make a significant impact in the industry (Nickerson & Rogers, 2013).

Since the mid 2000's campaigns have effectively been using a practice known as microtargeting to reach out, motivate, and activate voters. Microtargeting is defined as using consumer or demographic data to identify the interests of specific groups or individuals (Crain, & Nadler, 2019). Big data has allowed political entities to compile,

warehouse, and analyze consumer data, voter file information, and social media data, to identify groups of voters who respond to certain messages. For example, a political candidate can focus on a specific issue in campaign ads and speeches resulting in strong support from a specific racial, ethnic, gender, age, or party group. A campaign can then use that information to target that group of voters through messaging, ads, and phone banking. Political campaigns have been using microtargeting through big data as a means to identify voter's interests and demographics (Davis & Taras, 2022).

Voter File Description

Signed into Law by President Clinton on May 20, 1993, The National Voter Registration Act of 1993 (NVRA) created a voter registration system that was designed to make registration easier and more uniform from state to state (Crocker). Prior to this law, states had confusing, unclear registration processes that provided little to no oversight on the process. This system was modernized and further developed with the Help America Vote Act of 2002 (HAVA) which required each state to develop, maintain, and warehouse a computerized statewide registration list of all registered voters (Shanton). This statute allocated funds to all 50 states, along with the District of Columbia and various U.S. territories to improve the voting process and make information easy and accessible to the public. Within this statute, states were required to maintain a list of voters along with crucial information such as names, addresses, and to assign a state voter identification number. Election officials were required to perform regular maintenance to ensure the most accurate and up to date information.

Most states offer more than just the minimum requirements of name, address, and a voter identification number. In North Carolina voters can voluntarily provide gender, age, race, and ethnicity information. Other useful information such as possession of a license, and full voter history are also provided. The State Board of Elections refreshes this data on its website weekly, which allows data users to obtain the most accurate and up to date information of the electorate.

Importance of New Registrants

One of the most crucial groups which can change the political dynamic of a state each election are new voter registrants. New voters are identified as people who register to vote for the first time, and have not been previously involved in the electoral process. Hill Research, a public opinion and market research firm conducted a study on new voters in the state of Texas. In this particular study, voters were closely examined on how a rise in new voters led to a changing party dynamic in the state. While the focus of the analysis was on party realignment, these new voters were shown to have a profound impact on election outcomes in Texas during the 1980's (Dyer, 1988).

One of the most notable trends among new registrants is the changing demographics, specifically the increase of racial minority voters. The current U.S. electorate is primarily composed of non-hisapnic white voters. A study published from the Journal of American Society on Aging is projecting the number of white voters to decrease from 78% to 55% by the year 2060 (Demko & Torres-Gil, 2018). North Carolina's electorate seems to be following this national trend. In 1990 Latinos made up 1.2% of the state's population. This number increased to 8.4% in 2010 creating a 600%

increase in 20 years (Sanchez, 2015). New registrants can change the dynamic of a previous electorate, thus their significance cannot be ignored during election cycles.

Chapter 3: Methodology

The objective of this chapter is to introduce the reader to the data used in this case study, along with examining machine learning (ML) algorithms that can be used to predict voter turnout in presidential elections. A thorough description of the dataset and prediction variables are described in the Data Collection section. The four algorithms used in this case study are described in the Methodology section.

Data Collection

The data used in this project comes from the North Carolina State Board of Elections. Two separate datasets were extracted from their website on August 21, 2022 and later combined into a single table. The two datasets along with a brief description of each are listed below:

1. Statewide Voter Registration - North Carolina's statewide voter file containing basic information such as name, address, birthday, gender, and race information.
2. Statewide Voter History - This includes full vote history for affiliated registered voters.

Both datasets were imported into the programming language R, and later combined by joining on ncid, the state specific identification number assigned to each registered voter. While this data is rich with information, only a specific subset of columns were chosen for analysis. Due to the large number of categorical variables, a series of dummy variables were created to increase model performance. The predictors used, along with the values/encoders used in this analysis are shown below in Table 2.

Table 2

Variable	Value/Encoder	Description
Age	1-100	Age of voter at year's end
Gender	1 0	Male Female
Race	1 0	White Black Asian AmericanIndian Pacific Islander Mixed Other
Ethnicity	1 0	Hispanic Not Hispanic
Party	1 0	Democratic Republican Libertarian
RegistrationMonth	1 0	Months January - October

Registration month was created from the registr_dt column on the state voter file. Rather than leaving this variable in the date format, only month was extracted with the idea that registration month can be used as a predictor. For example, are voters who register closer to election day more likely to vote than those who register earlier in the year? The registration date deadline in North Carolina is in October, so naturally there are no voters in this dataset who registered after this deadline.

The column Voted2020 is the response variable and was created using the vote history file. The data used only contains voters who registered between January 1, 2020 and October 31, 2020. A total of 890,808 million voters registered between these two

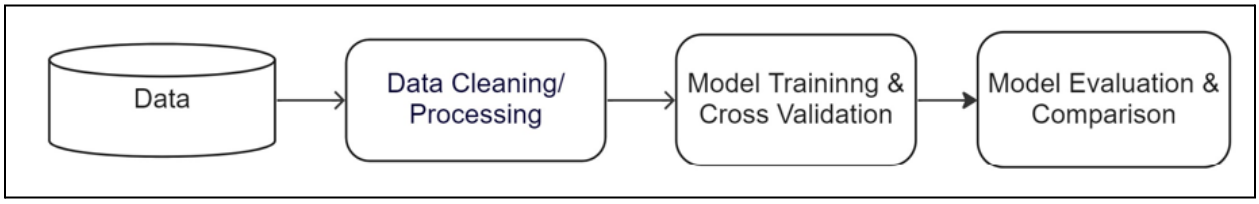
dates meaning 2020 was the first presidential election these registrants were eligible to participate in. Of these 890k, a sample of 50,000 was selected at random. Of the voters pulled in this sample, 40,736 voted in the 2020 election meaning 9,264 registered but did not cast a vote.

Data Flow Process

This case study utilized a specific data flow process for analysis. As mentioned previously, the data was collected from two sources and combined in R. From there, significant data cleaning took place, which involved creating dummy variables for all categorical predictors, and log transforming the quantitative predictor to create a normal distribution. Once a random sample was taken from the primary dataset and modeled on all four ML algorithms, a 5-fold cross validation was performed to train and validate each model. Finally, an in-depth analysis was performed on the variables from the best fitting model. All of the code used in this data flow process can be found in the Appendix. A visual representation of the data flow process is shown below in Figure 1.

Figure 1

Data Flow Process & Pipeline



Note. This diagram was created by the author.

Methods

As stated in the previous two chapters, the primary purpose of this case study is to create a model capable of predicting new voter turnout in presidential elections. Four different supervised machine learning models were chosen for this analysis. A thorough description of the algorithm, along with why it was chosen are discussed in this chapter. The methods used in this case study include:

1. Logistic Regression
2. Linear Discriminant Analysis
3. Quadratic Discriminant Analysis
4. Boosting

These methods were chosen for this case study for specific reasons. Logistic regression and tree based algorithms are often used in political microtargeting (Rusch et al., 2012) and when studying voter turnout among specific groups (Ondercin & Jones-White, 2011). Discriminant classifiers were chosen as they typically perform well on large datasets with a categorical outcome. Using multiple prediction methods with unique sets of strengths and weaknesses assists with model accuracy and validation. A 2001 study analyzing the results of the 2000 Presidential election showed that more models helps reduce bias and forecast error rates in predicting electoral outcomes (Bartels & Zaller, 2001). Each method was evaluated using the lowest misclassification rate based on the confusion matrix in the table below. Each misclassification rate was calculated using the equation directly below Table 3.

Table 3

	Actual Vote: False	Actual Vote: True
Predicted Vote: False	TN (True Negative)	FN (False Negative)
Predicted Vote: True	FP (False Positive)	TP (True Positive)

$$1 - \left(\frac{TP + TN}{TP + TN + FP + FN}\right)$$

Other machine learning methods were considered for this case study but ultimately not chosen. K-Nearest Neighbor (kNN) and Support Vector Machine (SVM) models were briefly explored but not selected for two reasons. While these methods are popular for categorical analysis, they would perform poorly based on the large number of categorical predictor variables in this case study. Both these algorithms rely on euclidean distance, which is commonly used to measure the distance between two points containing numeric, integer, or floating point variables. This data would likely not perform well relative to the four methods selected. KNN and SVM also are not optimal for large datasets. Modeling 50,000 records on multilevel prediction variables is computationally intensive. In conclusion, these methods would be most effective on a smaller sample size containing more quantitative predictors.

Method 1: Logistic Regression

The first method used in this case study is binary logistic regression. While logistic regression can be used to study categorical variables of two or more levels, the

outcome of this case study is simply two levels (Yes or no whether someone voted). Logistic regression is used to predict the probability of an event occurring between the independent and dependent variables in terms of log odds. Prediction is based on the maximum likelihood a linear relationship exists between these variables. The following equation describes logistic regression in more detail:

$$log\left(\frac{\rho\,(\chi)}{1-\rho(\chi)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_k x_k$$

In the equation above, the term $\frac{\rho\,(\chi)}{1-\rho(\chi)}$ is commonly referred to as the odds, or probability the outcome (*y*) will occur given the predictor, (*x*). The term log refers to the natural logarithm of the aforementioned variable. In this case study, the left side of the equation (y) signifies whether or not an individual voted. β_k represents each coefficient and x_k signifies the corresponding predictor variable (Demaris, 1995).

One of the benefits of logistic regression is that very little data manipulation is needed to produce a model, with the only assumption being that the data is normally distributed. There are also no tuning parameters needed to improve model performance making it a simple implementation on the computing side. The Stepwise function in R was used to identify the most useful predictors based on the Akaike's Information Criterion (AIC). This method has been heavily used for prediction selection in previous studies, and evidence suggests that it is fairly accurate at identifying significant categorical turned dummy variable predictors (Cohen, 1991). Once the best model with its predictors was chosen, the base logistic model was fit and used for further analysis.

To evaluate the performance of the base logistic regression model, the coefficients were examined along with their affiliated p-values. Any variable with a p-value deemed statistically insignificant ($p>0.05$) was removed from the equation. The variance inflation factor (VIF) was also used to see which variables resulted in a high variance within the regression outcome. Variables that had a VIF higher than 10 were removed to improve model performance. Once all predictors were tested and evaluated using the parameters above, the misclassification error was calculated to obtain model accuracy.

Method 2: Linear Discriminant Analysis

Similar to logistic regression, linear discriminant analysis (LDA), can also be used to predict a categorical outcome. While LDA shares many similarities with logistic regression, it differs significantly in that it has more assumptions. The primary assumptions are that the predictor variables have a gaussian distribution and equal class covariances (Day & Sandomire, 1942). LDA is derived from Baye's theorem, which states that the conditional probability of an event, based on the occurrence of another event, is equal to the likelihood of the second event given the first event multiplied by the probability of the first event (Berkson, 1930). The main takeaway from discriminant analysis is that it assigns an observation (x) to the class (y) having the largest posterior possibility (Lessmann et al, 2015). It can best be summarized in the equation below.

$$p(A \mid B) = \frac{p(B \mid A) * p(A)}{p(B)}$$

Baye's theorem is best understood in terms of prior probability, and posterior probability. Prior probability is the likelihood of an event occurring before new information is incorporated. In other words, it makes predictions based on current knowledge. Posterior probability is the likelihood of an event occurring after new information has been collected. As shown in the equation above, this is the probability of event A, given that B has already occurred.

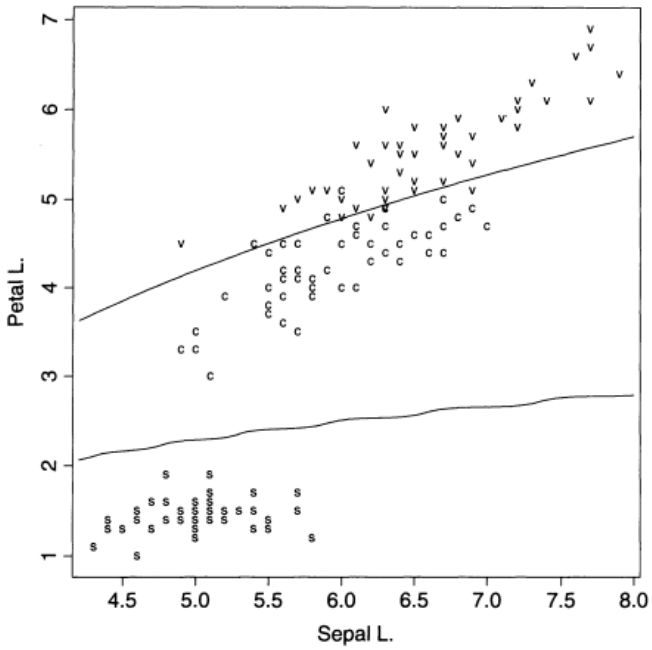
LDA utilizes this framework by essentially saying we can predict the probability of a variable, y, belonging to the class by taking that value of x. Discriminant functions are used to create decision boundaries which can be used to differentiate classes into regions (Tharwat, 2016). As mentioned previously, this case study is predicting whether or not a person voted, so a decision boundary will be based on the Voted2020 column.

Method 3: Quadratic Discriminant Analysis

Similar to LDA, Quadratic Discriminant Analysis (QDA) follows the same logic of Baye's Theorem. The primary difference is that QDA is used to find a non-linear relationship between two variables, whereas LDA works to find a linear relationship. QDA typically improves a model where a linear relationship does not exist, however it will perform poorly where the predictor variables contain high dimensions (Wu, 2019). This same study also cites that QDA is more sensitive to deviations from unstandardized data. This method was chosen in case a linear relationship is not present, but perhaps a quadratic one exists. A visual representation of QDA is shown below in Figure 2.

Figure 2

Visual Representation of Quadratic Discriminant Analysis



Note. Reprinted from “Graphical Tools for Quadratic Discriminant Analysis”, by Pardoe et al, (2007). Retrieved from https://www.jstor.org/stable/25471310

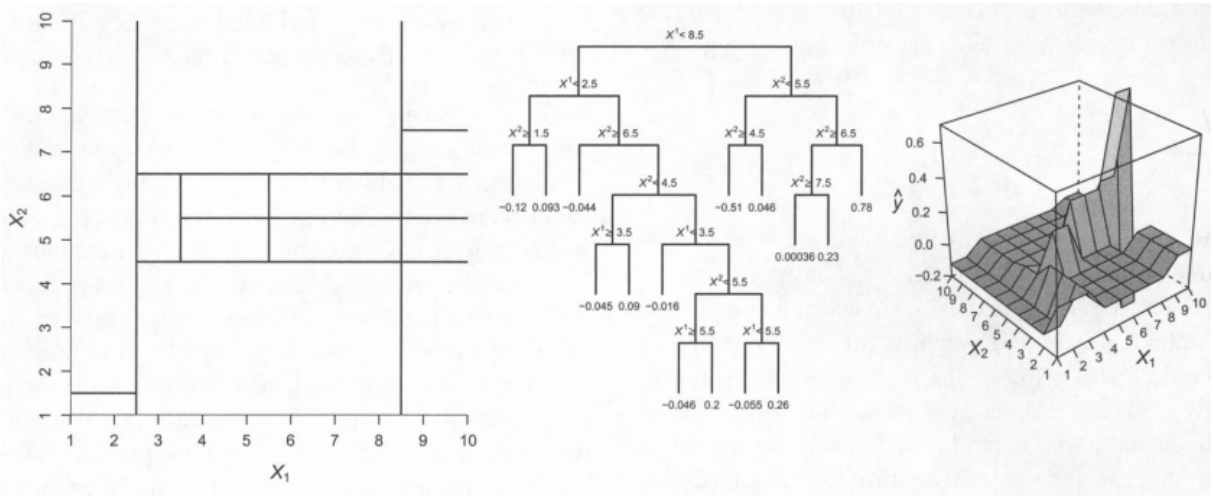
The graph in Figure 2 comes from a study from the American Statistical Association and American Society for Quality. It utilizes the iris dataset, which uses the characteristics of a plant to predict the species of a flower. In this study QDA was used and plotted above. As shown in Figure 2, the three classes of flowers are separated by two boundaries. These curved boundaries mark the quadratic classification regions used. While the example displays three different decision boundaries, this case study will only utilize two.

Method 4: Boosting

Boosting is an ensemble method typically characterized in a group of ML models known as decision trees. Decision trees work by repeatedly partitioning the data into groups based on the best predicting variable at each node. Tree based methods will outperform other ML models, such as various regression techniques, when the predictor variables are nonlinear (Hindman, 2015). These models continue to build a chosen number of trees in parallel, and then identify the best performing tree. An overview of the decision tree process is shown below in Figure 3.

Figure 3

Decision Tree logic



Note. Reprinted from “Tree-Based Models for Political Science Data”, by Montgomery & Santiago Olivella, (2018). Retrieved from https://www.jstor.org/stable/26598778

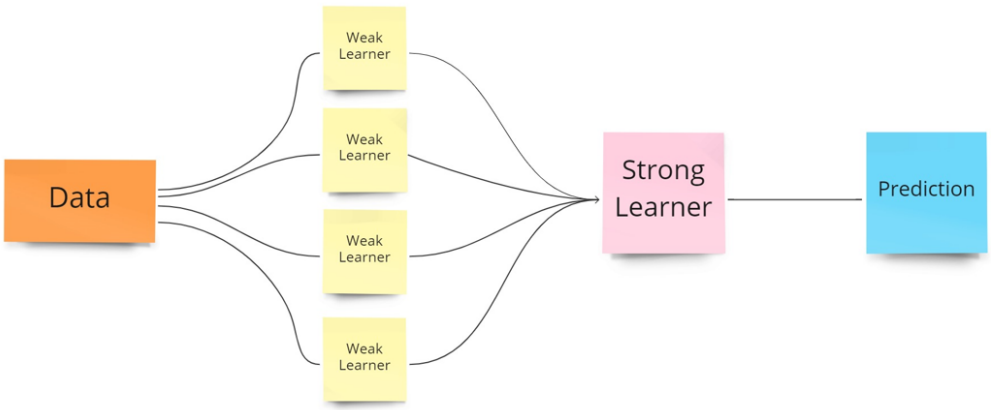
The first panel in Figure 3 displays a two-dimensional space broken up into 14 distinct regions. Each region represents a covariate combination, which can be

replicated in the form of a binary decision tree. The middle panel displays a complete decision tree, along with each terminal node. These trees then replicate and retrain themselves, creating the three dimensional “response surface” displayed in the right most panel.

Boosting is a specific type of decision tree which creates a composite model by assembling multiple models sequentially (Freund & Sapphire, 1996). Unlike other tree based models, which as previously mentioned builds numerous trees in parallel, boosting builds trees sequentially. Each tree constructed improves predictive accuracy, based on the misclassification error from previous inputs. In other words, the algorithm essentially teaches itself and improves performance each time. The boosting process is displayed below in Figure 4.

Figure 4

Boosting Process



Note. This diagram was created by the author.

One of the advantages of using decision trees is that little to no data cleaning or assumptions are needed for the input data. Ensemble methods, which include boosting, also treat continuous and categorical variables the same (Bühlmann & Hothorn, 2007). While there are many advantages to boosting, there are also some disadvantages. Increasing the number of trees in the tuning parameter decreases the error rate, however it may lead to overfitting the model.

Unlike regression models, boosting does not provide a simple, interpretable model containing coefficients and predictor variables. Rather it displays the relative influence a predictor has on the response variable. When analyzing this model in this case study, the relative influence of each predictor variable will be examined. From that, conclusions can be drawn based on which variables highly influence the model.

Summary

Previous voter turnout studies have utilized all of the ML methods mentioned in this chapter. Each method has a unique set of strengths and weaknesses which is expanded upon in Chapter 4. A diverse selection of ML models were chosen to compare and contrast these strengths and weaknesses. The best performing model is identified in Chapter 4, and a thorough analysis of the results were explored.

Chapter 4: Results

The primary objective of this chapter is to examine and compare the results of each ML method on the random sample of 50,000 voters. Once the accuracy of each model was assessed comparisons were drawn and an analysis was conducted on the best performing model. For more information on the coding, parameter tuning, and validation of each model, please see the Appendix.

Results - Logistic Regression

After using the Stepwise function in R to identify the most significant predictor variables, the following variables were deemed to be insignificant by the built in function and were removed from the equation:

- Male
- Race_Asian
- Race_PacificIslander
- Party_Lib

The model chosen had an AIC of 22930. Upon refitting this updated model, every coefficient appeared to be statistically significant. The VIF was then checked on all predictors in this model. Every single registration month had a high variance inflation factor (>100) and was removed from the equation. Finally the model was fit and validated using 5-fold cross validation. Table 4 shows the predicted results of the logistic regression model.

Table 4

Logistic Regression Predicted Results

	Actual Vote: False	Actual Vote: True
Predicted Vote: False	181	239
Predicted Vote: True	9,083	40,497

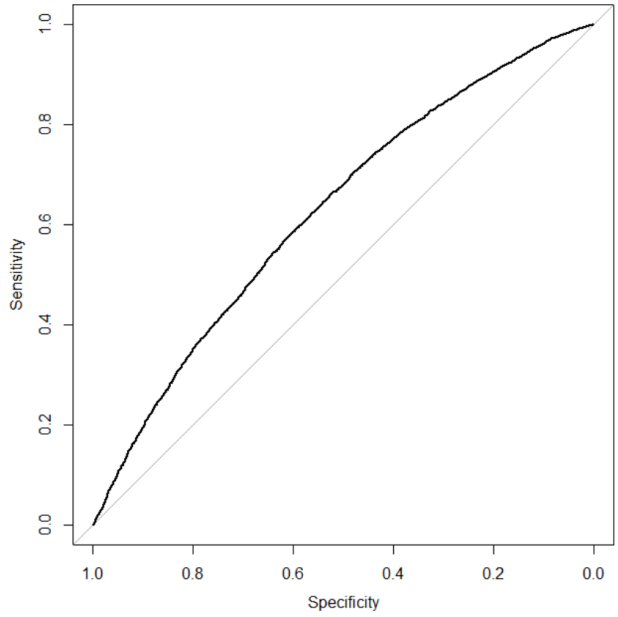
Note. Misclassification Error Rate: 18.6%

As shown above in Table 4, this model correctly predicted 40,497 people as having voted in the 2020 presidential election. Where this model falls short is predicting true negative values, or those who did not actually vote. Of the individuals who did not vote in the 2020 presidential election, only 181 voters were correctly predicted. This model accurately predicted a total of 40,678 voters, resulting in a misclassification rate of 18.6%.

While the model performed well among voters who actually voted, it certainly is lacking in predicting no shows at the voting booth. This is justified looking at the receiver operating characteristic (ROC) curve, as shown below in Figure 5. The ROC curve is commonly used when evaluating logistic regression models, as it visualizes the performance of each classification threshold. Good performing models typically have a curve that hugs the left and upper parts of the graph. A numeric value of space under the curve is provided in the area under the curve (AOC). Good performing models will have an AOC of over 80 and poor performing models will be in the 50’s or 60’s.

Figure 5

ROC Curve for Logistic Regression model



Note. Area under the curve is 62.66

As shown above in Figure 5, this is a rather poor performing logistic regression model. This is likely the result of its inability to accurately predict those who did not actually vote. The AOC is 62.66, which justifies the weak performance of the model. While logistic regression proved to be successful in identifying those who did vote, it falls short in identifying those who did not vote.

Results - Linear Discriminant Analysis

Unlike logistic regression, LDA does not require feature selection to improve model performance. While various methods could be used to see which predictors

would improve the model, this is not necessary for LDA and QDA. All predictors were left in the model and the results are displayed in Table 5.

Table 5

LDA Predicted Results

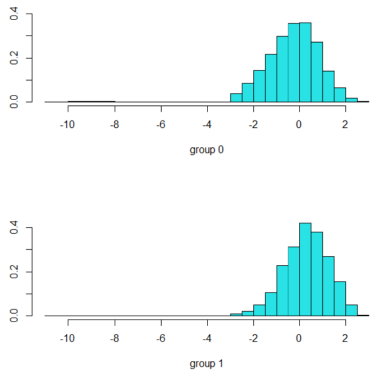
	Actual Vote: False	Actual Vote: True
Predicted Vote: False	161	9,103
Predicted Vote: True	119	40,616

Note. Misclassification error rate: 18.5%

The LDA model appears to be a slight improvement over logistic regression. It accurately predicted 40,777 voters with an error rate of 18.5%. However it appears to suffer from a similar problem to the logistic regression model, in that it poorly predicts voters who did not vote. Evidence of this is shown below in Figure 6, which displays a plot used to show separation of discriminant values.

Figure 6

Discriminant function values



In Figure 6, we see a lot of overlap between those who did not vote (group 0) and those who did vote (group 1). If this model could accurately identify the two predicted outcomes, there would be clear separation between groups 0 and 1. This result implies that a linear relationship does not exist between predictors and those who did not vote.

Results - Quadratic Discriminant Analysis

Similar to LDA, QDA did not require any tuning parameters prior to fitting the base model. Recall from the previous chapter that QDA differs from the other two methods in that it seeks to identify a quadratic relationship between one or more predictor variables, and the response variables. The results of the QDA model are shown below in Table 6.

Table 6

QDA Predicted Results

	Actual Vote: False	Actual Vote: True
Predicted Vote: False	1,823	7,441
Predicted Vote: True	4,633	36,102

Note. Misclassification error rate: 24.15%

The QDA model accurately predicted 37,925 voters for a misclassification rate of 24.15%. Seeing that this is the lowest performing model so far, we can conclude that some sort of quadratic relationship does exist between one or predictors and the response variable. Where QDA appears to have improved over the last two models is in the number of true negative values. Of the three models tested so far, QDA accurately

predicted the most registrants who did not vote in the 2020 election. While the previous two models were only able to identify a few hundred, this model correctly identified 1,823 voters who did not vote. However the much lower true positive rate decreases model accuracy making it the poorest performing model so far.

Results - Method 4 Boosting

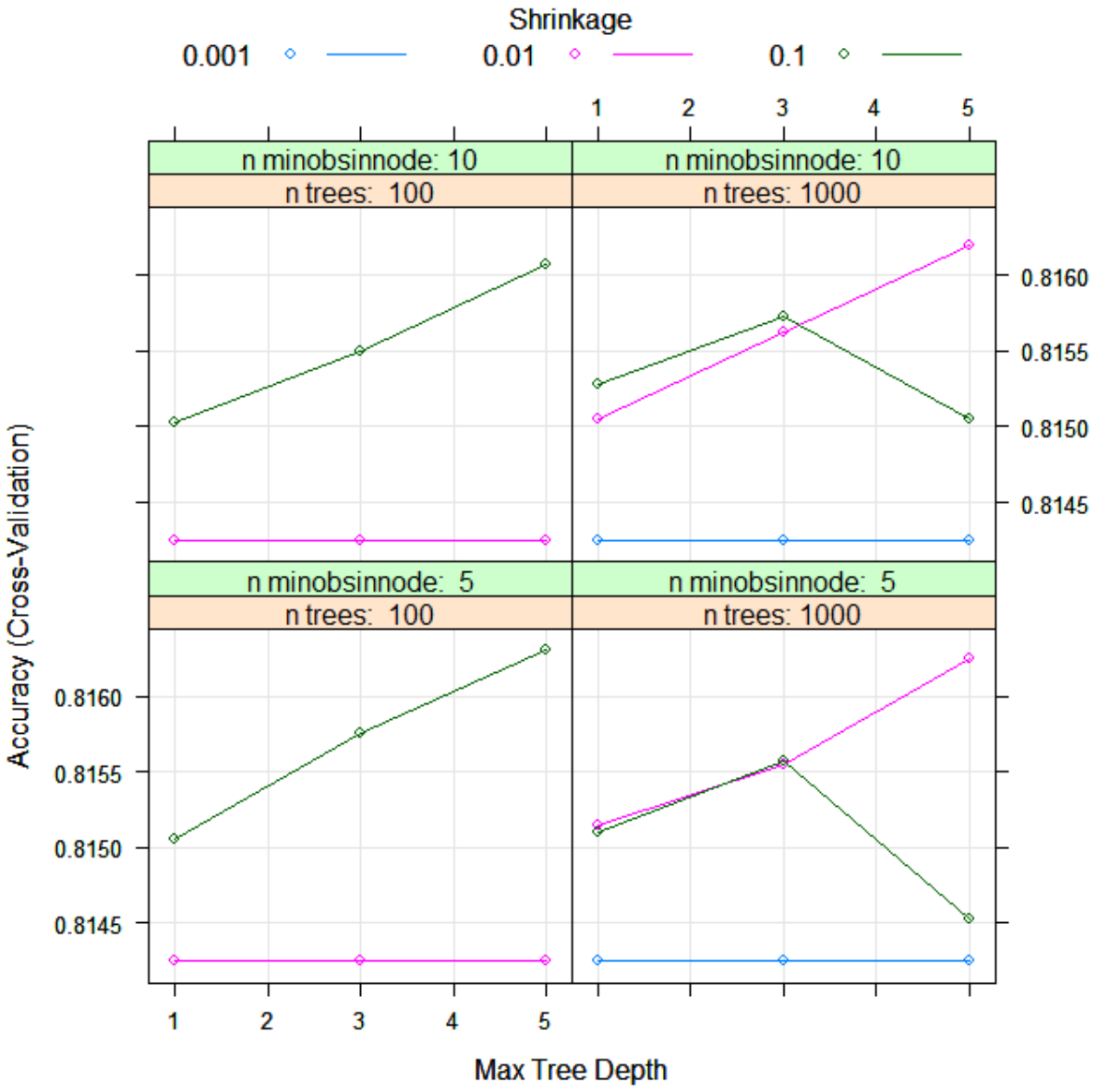
Of the four methods examined in this case study, Boosting utilized the most tuning parameters to enhance the overall model performance. The tuning parameters, along with a brief description of each is listed below.

1. Number of trees
2. Shrinkage
3. Interaction depth

The number of trees is the amount of models the algorithm will create. A small number of trees typically will not provide the most accurate model, however using too many trees can lead to overfitting. Shrinkage is commonly referred to as the learning rate, and essentially tells the tree whether or not to minimize poor performing models. A low shrinkage rate will slow down the model however higher shrinkage rates can also lead to overfitting. The interaction depth signifies the number of splits that should be run on each tree. The more splits there are, the more interactions the model creates. To find the correct parameters, an exhaustive grid search was run testing out various trees, shrinkage values, and interaction depths to see which values produced the best model. The results of the grid search are shown below in Figure 7.

Figure 7

Boosting tuning parameters



As shown above in Figure 7, the most optimal number of trees is 1000, with a shrinkage parameter of 0.01 and an interaction depth of 5. Those parameters were then

re-inserted into the model and five fold cross validation was performed. The results of the cross validation are shown below in Table 7.

Table 7

Boosting predicted results

	Actual Vote: False	Actual Vote: True
Predicted Vote: False	555	368
Predicted Vote: True	8,709	40,368

Note. Misclassification rate: 18.15%

Just like the previous 3 models, boosting proved to poorly predict which voters would not actually vote. However this model does have the highest number of voters correctly classified. Boosting accurately predicted 40,923, just slightly better than the other three models.

Model Comparison

Of the four ML methods examined in this case study, all of them yielded relatively similar results. Logistic regression, LDA, and Boosting all correctly predicted slightly over 40,000 voters. While QDA was slightly lower than the others, it did accurately predict the most voters who did not end up voting in the 2020 presidential election. The common theme among all four models is the inability to predict which voters would not turnout. While all models yielded accuracy of around 80%, the high number of false positives should not be ignored. This issue along with other methods to possibly resolve it are discussed in Chapter 5.

Analysis

The results from the best performing model (Boosting) were analyzed to gain further insights from the data. As mentioned previously in Chapter 3, rather than providing a single linear model, the boosting algorithm instead provides a list of variables that are important, along with their relative influence. Table 8 shows the top four predictor variables, along with their relative influence on the outcome, Voted2020.

Table 8

Relative Influence

Variable	Relative Influence Score
age_at_year_end	25.6203328
Race_White	10.4168510
RegMonth_JAN	7.8639266
RegMonth_OCT	6.9367207

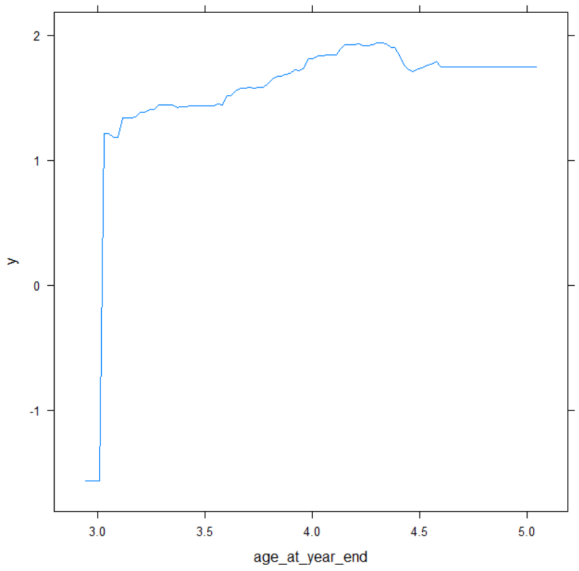
Note. Only the four highest predictor variables are shown in this table.

These variables do not represent a high likelihood that someone will vote, rather they signify that that variable has a greater influence on the response variable than other predictors. These four variables were analyzed on the original sample of 50,000 voters to see what conclusions can be drawn. The marginal effect of each variable will first be examined, and then further conclusions can be made from that.

Variable Importance - Age

Age proved to have the highest relative influence on voter turnout. An interesting observation to note is that age was also the only quantitative predictor in the dataset. In the marginal effect of the variable age below, we see that a higher value of age is associated with a higher likelihood of voting. Because age was log-transformed during pre-processing, the exact age where this increase occurs cannot be easily understood just by looking at this chart. However it signifies that at a certain age, the likelihood of voting will increase.

Figure 8



Many political science studies often group people into an age range, rather than leave them as a quantitative predictor. To dig a little deeper into what these numbers mean, age was broken up into 5 groups. Using the sample of 50,000 voters, these five age groups are shown below, along with whether or not they voted.

Table 9

Voter Turnout by age range

Age Range	Yes	No	Turnout Percent
18-34	18,367	5,260	77.7%
35-49	9,607	2,040	82.5%
50-64	7,644	1,237	86.1%
65+	5,118	727	87.6%

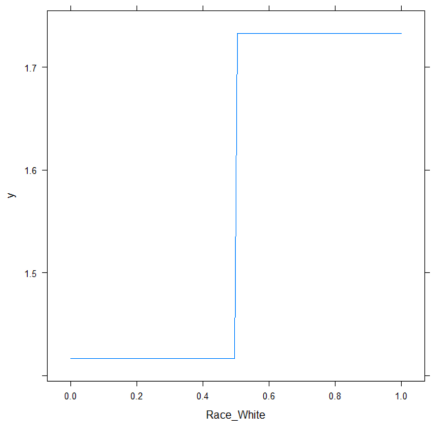
While younger voters make up the majority of new registrants, the results show that they are less likely to turnout rather than older registrants voting for the first time. We can also see the likelihood of voting steadily increases with each age group. The results of this case study aligns with many other studies which back up the claim older voters are more likely to turnout (Poama & Volacu, 2021). This is a promising sign as it shows ML models accurately identify and understand the impact age has on voter turnout.

Variable Importance: Race White

After age, race-white was shown to have the second strongest relative influence on voter turnout. Despite there being seven different race groups used as predictor variables, white was identified to have the greatest impact. This is understandable as 23,722 (47.4%) of the voters in this sample self-identified as white, making up the largest chunk of the state's electorate. This race would have a larger impact on voter

turnout than the remaining race categories due to the sheer number of voters belonging to this group. Looking at the plot below, we see that white voters have a positive influence on election turnout.

Figure 9

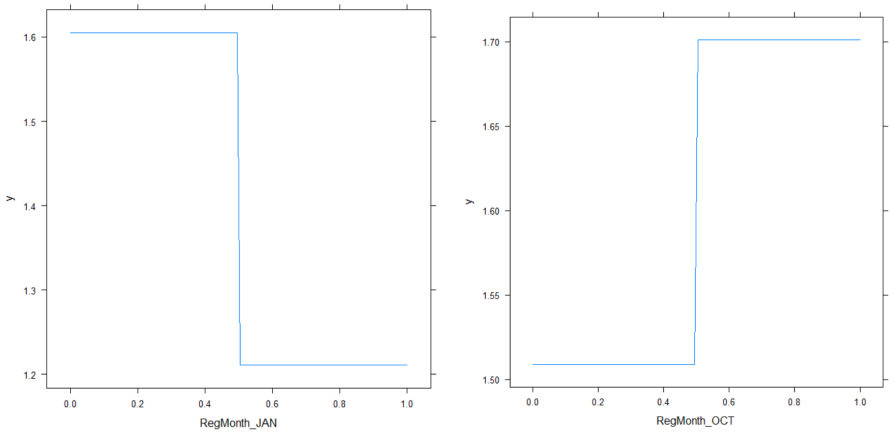


To expand upon this, of the white voters in this case study 20,204 (85.2%) turned out to vote in the 2020 presidential election. This is the highest percentage of every race group in the dataset (see code in appendix for other turnout numbers). Based on these factors, it's easy to see how white voters would have a strong influence on presidential election turnout in North Carolina.

Variable Importance: Registration Month

The next two predictor variables identified in the boosting model that had a strong influence on outcome were the registration months of January and October. Since both originate from the same category, these two were grouped together and compared for this section of the paper. Each month's relative influence is shown below in Figure 10.

Figure 10



As shown above, January appears to have a low relative influence on voter turnout, while October appears to have a stronger influence. Recall this does not necessarily mean that people who register in January are less likely to vote (although that does appear to be true). Instead October registrations have a strong influence on the model outcome. Recall from Chapter 3 that the purpose of extracting months from registration date was to see if specific months had an impact, or correlation on voter turnout. This turned out to be the case as October had the highest number of new registrants, and October registrants had the highest turnout percentage of any month. From this we can see that October registrants are more likely to vote than those in January.

Results and Analysis Conclusion

The results from the four models used in this case study were presented in this chapter. All appeared to accurately predict which voters would turnout in the 2020 presidential election. There was clearly an issue with predicting which voters would not

turnout in all four models, which is something that should be explored in future research. Boosting proved to be the most accurate model with a voter misclassification rate of 18.15%. Due to the limited time of this project, only the four variables with the highest relative influence were further analyzed. However gender, political party, and ethnicity all had influence on the outcome, albeit much lower than the four aforementioned variables. A deeper analysis into the other variables could prove to be valuable to those using this information for election prediction.

Chapter 5: Summary and Conclusion

This chapter summarizes the key findings from this case study, including a discussion on model performance, significant predictors, and possible improvements. Additionally, recommendations for future research conducting similar studies are presented. Lastly, the overall conclusion of this case study is summarized.

Summary of Findings

This case study examined four ML methods that all performed relatively well in predicting voter turnout. The best fitting model, boosting, had a voter misclassification rate of 18.15% which was slightly better than the other three models. The results from boosting showed that age, race, and registration month play a significant role in predicting election turnout. After examining these variables in more depth, we saw that:

- Older voters tend to vote at a higher rate, despite younger voters making up the majority of new registrants.
- White voters have the largest impact of any race on voter turnout
- Registration month plays a crucial role in predicting turnout. Voters who registered closer to election day were more likely to vote than those who registered earlier in the year.

While other predictors had some influence on election turnout, their scores were significantly lower than the aforementioned variables. Using state provided voter files to predict presidential election turnout proved to be a relatively effective form of voter microtargeting among new registrants in North Carolina.

Suggestions for Future Research

Predicting new registrants who will not vote was a weakness in all four methods examined. This is significant because political campaigns, companies, and others do not want to waste time, money, and resources targeting voters who likely will not vote. This issue does not appear to be unique to this case study, as previous political studies tend to overestimate voter turnout (Ansolabehere & Hersh, 2012). This signifies the challenge of predicting voter turnout among new registrants. Registering to vote typically shows an intent to be involved in the electoral process, however this is not a guarantee that a person will actually cast a ballot. The large number of false positives signifies that the methods chosen may not be the most optimal models for this analysis. One method that was briefly considered for this case study was Artificial Neural Networks (ANN). Similar to boosting, ANN may be a good next step as the algorithm uses weights to enhance model performance and continually learns from past mistakes to improve. Moving away from linear and quadratic relationships between predictor and response variables may be a more effective method in future research.

The discriminant methods of LDA and QDA were also used for prediction. However these methods are typically meant for data that utilizes all, or at least a large number of quantitative predictors. The challenge of finding more quantitative variables may have contributed to these methods not performing to their optimal potential. This is where utilizing other sources to improve the model would be beneficial, as the North Carolina voter file did not contain many quantitative variables. Organizations that specialize in microtargeting often have access to a wide range of datasets, beyond state

voter files. Credit scores, salary, and other numeric predictors could be used to improve these models in a similar study.

Conclusion

The purpose of this case study was to introduce the reader to the valuable information held within state provided voter files. North Carolina was chosen due to its rich voter file information, accessibility, and political relevance in recent presidential elections. This case study was undertaken to see if predictions can be made strictly based on registrant information provided on state voter files. Specifically we:

1. Reviewed current literature of data use in modern politics
2. Discussed the political importance of North Carolina in presidential elections
3. Discussed the importance of new voters in presidential elections
4. Examined four machine learning methods that were used on voter file information to predict voter turnout
5. Analyzed the results of the most accurate model

With political microtargeting on the rise, the use of ML methods for voter turnout prediction will only continue to grow. This study proved to be accurate in predicting voter turnout, however the low number of true negatives should require further research. As the utilization of big data continues to grow in the political realm, the improvements made to voter microtargeting will be key in future presidential elections.

References

Ansolabehere, S., & Hersh, E. (2012). Validation: What Big Data reveal about survey misreporting and the real electorate. *Political Analysis*, *20*(4), 437–459. https://doi.org/10.1093/pan/mps023

Bartels, L. M., & Zaller, J. (2001). Presidential vote models: A recount. *Political Science & Politics*, *34*(01), 9–20. https://doi.org/10.1017/s1049096501000026

Berkson, J. (1930). Bayes' theorem. *The Annals of Mathematical Statistics*, *1*(1), 42–56. https://doi.org/10.1214/aoms/1177733259

Bühlmann, P., & Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, *22*(4). https://doi.org/10.1214/07-sts242

Cohen, A. (1991). Dummy variables in stepwise regression. *The American Statistician*, *45*(3), 226. https://doi.org/10.2307/2684296

Congressional Research Committee, & Shanton, K. L., The Help America Vote Act of 2002 (HAVA): Overview and ongoing role in election administration policy (n.d.).

Crain, & Nadler. (2019). Political Manipulation and Internet Advertising Infrastructure. *Journal of Information Policy*, *9*, pp. 370-410. https://doi.org/10.5325/jinfopoli.9.2019.0370

Hawley, G. (2021). Blue Metros, red states: The shifting urban-rural divide in America's swing states by David F.Damore, Robert E.Lang, and Karen A.Danielsen.

Washington, DC, Brookings Institution Press, 2020. *Political Science Quarterly*, *136*(3), 566–568. https://doi.org/10.1002/polq.13215

Davis, R. & Taras, D. (2022). Conclusion. In *Electoral Campaigns, Media, and the New World of Digital Politics.* University of Michigan Press., pp. 307-314 https://www.jstor.org/stable/10.3998/mpub.12013603.18

Day, Besse B., and Marion M. Sandomire (1942). Use of the Discriminant Function for More than Two Groups. *Journal of the American Statistical Association*, vol. 37, no. 220, 1942, pp. 461–472. https://doi.org/10.1080/01621459.1942.10500647

DeMaris, A. (1995). A tutorial in logistic regression. *Journal of Marriage and the Family*, *57*(4), 956. https://doi.org/10.2307/353415

Dyer, J. A., Vedlitz, A., & Hill, D. B. (1988). New voters, switchers, and political party realignment in Texas. *Western Political Quarterly*, *41*(1), 155–167. https://doi.org/10.1177/106591298804100111

Edwards, G. C. (2019). Why the Electoral College is bad for America. https://doi.org/10.12987/yale/9780300243888.001.0001

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*(2), 179–188. https://doi.org/10.1111/j.1469-1809.1936.tb02137.x

Fon, V. (2004). Electoral College Alternatives and US presidential elections. *Supreme Court Economic Review*, *12*, 41–73. https://doi.org/10.1086/scer.12.3655317

Gimpel, J. G., Kaufmann, K. M., & Pearson-Merkowitz, S. (2007). Battleground states versus blackout states: The behavioral implications of modern presidential campaigns. *The Journal of Politics*, *69*(3), 786–797. https://doi.org/10.1111/j.1468-2508.2007.00575.x

Hindman, M. (2015). Building better models. *The ANNALS of the American Academy of Political and Social Science*, *659*(1), 48–62. https://doi.org/10.1177/0002716215570279

Johnson, B. J. (2005). Identities of competitive states in U.S. presidential elections: Electoral College bias or candidate-centered politics? *Publius: The Journal of Federalism*, *35*(2), 337–355. https://doi.org/10.1093/publius/pji017

Lessmann, Stefan, et al (2015). "Benchmarking State-of-the-Art Classification Algorithms for Credit Scoring: An Update of Research." *European Journal of Operational Research*: Vol. 247(1), pp. 124–136., https://doi.org/10.1016/j.ejor.2015.05.030.

McDonald, M. P. (2009). The return of the voter: Voter turnout in the 2008 presidential election. *The Forum*, *6*(4). https://doi.org/10.2202/1540-8884.1278

Montgomery, J. M., & Olivella, S. (2018). Tree-based models for Political Science Data. *American Journal of Political Science*, *62*(3), 729–744. https://doi.org/10.1111/ajps.12361

Nickerson, D., & Rogers, T. (2013). Political campaigns and Big Data. *Journal of Economic Perspectives*, *28*(2), 51–74. https://doi.org/10.2139/ssrn.2354474

Ondercin, H. L., & Jones-White, D. (2011). Gender jeopardy: What is the impact of gender differences in political knowledge on political participation?*. *Social Science Quarterly*, *92*(3), 675–694. https://doi.org/10.1111/j.1540-6237.2011.00787.x

Pardoe, I., Yin, X., & Cook, R. D. (2007). Graphical tools for quadratic discriminant analysis. *Technometrics*, *49*(2), 172–183. https://doi.org/10.1198/004017007000000074

Poama, A., & Volacu, A. (2021). Too old to vote? A Democratic analysis of age-weighted voting. *European Journal of Political Theory*, 147488512110626. https://doi.org/10.1177/14748851211062604

Rusch, T., Lee, I., Hornik, K., Jank, W., & Zeileis, A. (2012). Influencing elections with statistics: Targeting voters with logistic regression trees. *The Annals of Applied Statistics*, *7*(3), 1612–1639. https://doi.org/10.2139/ssrn.2016956

Sanchez, G. R. (2015). *Latinos and the 2012 election: The new face of the American voter*. Michigan State University Press.

Schultz, D. A., & Hecht, S. H. (2015). *Presidential Swing states: Why only ten matter*. Lexington Books.

Tharwat, A. (2016). Linear vs. quadratic discriminant analysis classifier: A tutorial. *International Journal of Applied Pattern Recognition*, *3*(2), 145. https://doi.org/10.1504/ijapr.2016.079050

Wolak, Jennifer. The Consequences of Presidential Battleground Strategies for Citizen Engagement. *Political Research Quarterly*, vol. 59, no. 3, 2006, pp. 353–361., https://doi.org/10.1177/106591290605900303.

Wu, Yilei, et al (2019). Quadratic Discriminant Analysis for High-Dimensional Data. *Statistica Sinica*, https://doi.org/10.5705/ss.202016.0034.

Appendices

Appendix A: R Code

```
# Capstone R Code


# load libraries

library(dplyr)

library(anytime)

library(readr)

library(data.table)

library(ISLR)

library(MASS)

library(car)

library(pROC)

library(gbm)

library(GGally)


# Load 2 datasets

# Voter File

vf_data<-fread("C:/Users/peter/Downloads/DS 785/ncvoter_Statewide/ncvoter_Statewide.txt", select =
        c("ncid","status_cd","voter_status_desc","voter_status_reason_desc",
        "state_cd","registr_dt","race_code","ethnic_code","party_cd","gender_code","birth_year","age_at_year_end",
        "drivers_lic"))

# Vote History

vh_data<-fread("C:/Users/peter/Downloads/DS 785/ncvhis_Statewide/ncvhis_Statewide.txt", select =
        c("ncid","election_lbl","voting_method","voted_county_id"))

# 33,831,528


# Keep only 2020 voters

vh_data<-vh_data[vh_data$election_lbl=="11/03/2020"]

# 5,542,959
```

```
### Data Cleaning ###


# Combine datasets into one

raw_data <- merge(vf_data,vh_data, by='ncid', all.x=TRUE)

raw_data <- unique(raw_data)


# Convert registr_dt to Date

raw_data$registr_dt <- as.Date(raw_data$registr_dt, "%m/%d/%Y")


# Only keep new voters

raw_data<- raw_data %>%

  filter(registr_dt >= anydate("01/01/2020") & registr_dt <= anydate("11/1/2020"))

# 890,757


# Remove inactive/dead voters

raw_data<-raw_data[raw_data$voter_status_reason_desc!=c("DENIED","REMOVED","DECEASED")]

# 1,335,081




# Take a random sample of 50k

set.seed(123, sample.kind = "Rounding")

subset_data<- raw_data[sample(nrow(raw_data),50000),]


# Create new df using only columns needed

subset_data<-subset_data[,c("ncid","status_cd","registr_dt","race_code","ethnic_code","party_cd","gender_code","ag
            e_at_year_end","drivers_lic","election_lbl")]


# Create dependent variable, identifying whether or not someone voted

subset_data <- subset_data %>%

  mutate(Voted2020 = case_when(
```

```
    endsWith(election_lbl,"0") ~ 1,
    is.na(election_lbl) ~ 0
  ))

# Data types of each variable
sapply(raw_data,class)

# Create new variable extracting only the month from registration date
subset_data$registration_month<- strftime(subset_data$registr_dt,"%m")

# Create new df of cleaned columns
subset_data2<-subset_data[,c("ncid","Voted2020","age_at_year_end")]

# Make dummy variables for all categorical predictors
subset_data2$Male <- ifelse(subset_data$gender_code == 'M', 1, 0)
subset_data2$Female <- ifelse(subset_data$gender_code == 'F', 1, 0)

subset_data2$Race_White <- ifelse(subset_data$race_code == 'W', 1, 0)
subset_data2$Race_Black <- ifelse(subset_data$race_code == 'B', 1, 0)
subset_data2$Race_Asian <- ifelse(subset_data$race_code == 'A', 1, 0)
subset_data2$Race_AmericanIndian <- ifelse(subset_data$race_code == 'I', 1, 0)
subset_data2$Race_PacificIslander <- ifelse(subset_data$race_code == 'P', 1, 0)
subset_data2$Race_Mixed <- ifelse(subset_data$race_code == 'M', 1, 0)
subset_data2$Race_Other <- ifelse(subset_data$race_code == 'O', 1, 0)

subset_data2$Ethnicity_Hispanic <- ifelse(subset_data$ethnic_code == 'HL', 1, 0)
subset_data2$Ethnicity_NotHispanic <- ifelse(subset_data$ethnic_code == 'NL', 1, 0)

subset_data2$Party_DEM <- ifelse(subset_data$party_cd == 'DEM', 1, 0)
```

```
subset_data2$Party_REP <- ifelse(subset_data$party_cd == 'REP', 1, 0)
subset_data2$Party_LIB <- ifelse(subset_data$party_cd == 'LIB', 1, 0)

subset_data2$RegMonth_JAN <- ifelse(subset_data$registration_month == '01', 1, 0)
subset_data2$RegMonth_FEB <- ifelse(subset_data$registration_month == '02', 1, 0)
subset_data2$RegMonth_MAR <- ifelse(subset_data$registration_month == '03', 1, 0)
subset_data2$RegMonth_APR <- ifelse(subset_data$registration_month == '04', 1, 0)
subset_data2$RegMonth_MAY <- ifelse(subset_data$registration_month == '05', 1, 0)
subset_data2$RegMonth_JUN <- ifelse(subset_data$registration_month == '06', 1, 0)
subset_data2$RegMonth_JUL <- ifelse(subset_data$registration_month == '07', 1, 0)
subset_data2$RegMonth_AUG <- ifelse(subset_data$registration_month == '08', 1, 0)
subset_data2$RegMonth_SEP <- ifelse(subset_data$registration_month == '09', 1, 0)
subset_data2$RegMonth_OCT <- ifelse(subset_data$registration_month == '10', 1, 0)
subset_data2$RegMonth_NOV <- ifelse(subset_data$registration_month == '11', 1, 0)

# Create final cleaned dataset to be used
data <- subset_data2[,-c("ncid")]

# Check for normality in quantitative variable
hist(data$age_at_year_end, main="Age Distribution in Sample",col="light blue",xlab="Age",ylab="Count")
# Left skewed

# Log transform to create a normal distribution
data$age_at_year_end<-log(data$age_at_year_end+1)

# Should be more normally distributed now
hist(data$age_at_year_end, main="Age Distribution in Sample",col="light blue",xlab="Age",ylab="Count")
# Left skewed
```

```
################################################################################

# Machine Learning Method 1: Logistic Regression


# Base model

glm.model = glm(Voted2020 ~., data=data,family = "binomial")

summary(glm.model)



# Find optimum predictors

library(MASS)

step.model <- glm.model %>% stepAIC(trace = FALSE)

coef(step.model)



# New base model with only optimum predictors

glm.model <- glm(Voted2020 ~. -RegMonth_NOV-Party_LIB, data=data, family = "binomial")

summary(glm.model)


# Check for VIF

library(car)

vif(glm.model)



# Run new model with predictors less than a VIF of 10

glm.model <- glm(Voted2020 ~ age_at_year_end + Female + Race_White + Race_Black  + Race_AmericanIndian +
          Race_Mixed + Race_Other + Ethnicity_Hispanic + Ethnicity_NotHispanic + Party_DEM + Party_REP,
          data=data, family = "binomial")

summary(glm.model)
```

```
# Cross Validation for Logistic Regression

n=dim(data)[1]

k=5

groups=c(rep(1:k,floor(n/k)),1:(n-floor(n/k)*k))


set.seed(123, sample.kind = "Rounding")

cvgroups=sample(groups,n)


predictvals=rep(-1,n)


for(i in 1:k){

  groupi=(cvgroups==i)

  fit=glm(Voted2020 ~ age_at_year_end + Female + Race_White + Race_Black  + Race_AmericanIndian +
            Race_Mixed + Race_Other + Ethnicity_Hispanic + Ethnicity_NotHispanic + Party_DEM + Party_REP,
            data=data[!groupi,],family="binomial")

  predictvals[groupi]=predict(fit,data[groupi,],type="response")

}


# ROC Curve

myroc=roc(response=data$Voted2020,predictor=predictvals)

plot.roc(myroc)

auc(myroc)


# Table

table(predictvals>0.6,data$Voted2020)

predicted.classes <- ifelse(predictvals > .6, "pos", "neg")

# Correctly predicted: 40,678 81.36%

# Misclassification Rate: 9,322
```

```
################################################################################

# Machine Learning Method 2: Linear Discriminant Analysis

# Base LDA Model
n = dim(data)[1]

model.lda = lda(Voted2020 ~., data=data)

par(mfrow=c(2,2))
plot(model.lda)

plot(model.lda, data$Voted2020)

fittedclass2 = predict(model.lda,data=data)$class
table(data$Voted2020,fittedclass2)
Error = sum(data$Voted2020 != fittedclass2)/n; Error

# Check class distributions for both outcomes
plot(model.lda, type="both",main="Discriminant Function Values")

# Cross Validation
n=dim(data)[1]
k=5
groups=c(rep(1:k,floor(n/k)),1:(n-floor(n/k)*k))

set.seed(123, sample.kind = "Rounding")
cvgroups=sample(groups,n)
```

```
predictvals2=rep(-1,n)


for (i in 1:k)  {
  groupi=(cvgroups==i)
  fit2=lda(Voted2020 ~., data=data[!groupi,])
  newdata = data.frame(data[cvgroups==i,])
  predictvals2[cvgroups==i] = as.character(predict(fit2,newdata)$class)
}



# Model Summary
model.lda



# Table
CVmodel = sum(predictvals2!=data$Voted2020)/n; CVmodel
table(data$Voted2020,predictvals2)
# Correctly predicted: 40,777 81.5%
# Misclassifcation: 4,649 18.6%

################################################################################


# Machine Learning Method 3: Quadratic Discriminant Analysis


# QDA Base model
qda.model = qda(Voted2020 ~ .-RegMonth_NOV, data=data)


# Check class distributions for both outcomes
plot(qda.model, type="both")
```

```
# Cross Validation
library(MASS)
n=dim(data)[1]
k=5
groups=c(rep(1:k,floor(n/k)),1:(n-floor(n/k)*k))

set.seed(123, sample.kind = "Rounding")
cvgroups=sample(groups,n)

predictvals3=rep(-1,n)

for (i in 1:k)  {
  fit3=qda(Voted2020 ~ .-RegMonth_NOV, data=data[!groupi,])
  newdata = data.frame(data[cvgroups==i,])
  predictvals3[cvgroups==i] = as.character(predict(fit3,newdata)$class)
}

# Table
Error3 = sum(subset_data$Voted2020 != fittedclass3)/n; Error3
table(data$Voted2020,predictvals3)
# Correctly predicted: 37,925 75.85%
# Incorreclty: 12,075: 24.15

# Model Summary
qda.model
```

```
################################################################################

# Method 4: Boosting

library(gbm)
library(caret)
library(rpart.plot)
library(tidyverse)

data$age_at_year_end<-subset_data2$age_at_year_end

# Base model
set.seed(123,sample.kind = "Rounding")
boost = gbm(Voted2020 ~ .,data=data,distribution="bernoulli",n.trees=1000,
            shrinkage=0.01,interaction.depth=5)
summary(boost)


boost_pred=predict(boost,newdata=data,n.trees=1000,type="response")
# Table
table(boost_pre>.5,data$Voted2020)
a = table(boost_pred>.5,data$Voted2020)
(a[1,2] + a[2,1])/n


# Important predictors
p1<-plot(boost,i="age_at_year_end")
p2<-plot(boost,i="RegMonth_JAN")
p3<-plot(boost,i="Race_White")
```

```
p4<-plot(boost,i="RegMonth_OCT")

p1

p2

p3

p4

print(model_result)


# Best model ntrees(200), max_depth(3), Accuracy (81.5)

###############

# Grid search

###############


## WARNING: This runs for almost an hour :(

# Finding optimum number of trees

data_train=data[1:40000,]

data_test <- data[40001:50000,]


optimal_trees <- list()

library(caret)


trainControl <- trainControl(method = "cv",

                             number = 5,

                             returnResamp="all", ### use "all" to return all cross-validated metrics

                             search = "grid")


tuneGrid <- expand.grid(

  n.trees = c(100, 1000),

  interaction.depth = c( 1,3,5),

  shrinkage = c(0.1,0.01, 0.001),
```

```
  n.minobsinnode=c(5,10)
)

gbm_op <- train(Voted2020 ~.,
                data = data_train,
                method = "gbm",
                tuneGrid = tuneGrid,
                trControl = trainControl,
                verbose=FALSE)



plot(gbm_op)
############################################################
trellis.par.set(caretTheme())
plot(gbm_op, metric = "Kappa")

# Model from Grid Search
set.seed(123,sample.kind = "Rounding")
boost = gbm(Voted2020 ~ .,data=data,distribution="bernoulli",n.trees=1000,
            shrinkage=0.01,interaction.depth=5)
summary(boost)

# Cross Validation for boosting
n=dim(data)[1]
k=5
groups=c(rep(1:k,floor(n/k)),1:(n-floor(n/k)*k))
set.seed(123, sample.kind = "Rounding")
cvgroups=sample(groups,n)
```

```
boost.pred=rep(-1,n)


for(i in 1:k){

  groupi=(cvgroups==i)

  fit4 = gbm(Voted2020 ~ .-RegMonth_NOV, data=data[!groupi,], distribution="bernoulli", n.trees=200, shrinkage=0.3,
             interaction.depth=3)

  boost.pred[groupi]=predict(fit4,newdata=data[groupi],n.trees=200,type="response")


}

# Table

table(boost_pre>.5,data$Voted2020)

a = table(boost.pred>.5,data$Voted2020)

(a[1,2] + a[2,1])/n


################################################################################


# Analysis


# Important predictors

p1<-plot(boost,i="age_at_year_end")

p2<-plot(boost,i="RegMonth_JAN")

p3<-plot(boost,i="Race_White")

p4<-plot(boost,i="RegMonth_OCT")




# Age


p1<-plot(boost,i="age_at_year_end")
```

```
p1


# Create a new df grouping age into categories
age_df <-
subset_data %>%
  mutate(age_range = case_when(age_at_year_end >= 18 &  age_at_year_end <= 34 ~ '18-34',
                               age_at_year_end >= 35 &  age_at_year_end <= 49 ~ '35-49',
                               age_at_year_end >= 50 &  age_at_year_end <= 64 ~ '50-64',
                               age_at_year_end >= 65 ~ '65+'))


# Look at tunrout by age group
table(age_df$Voted2020, age_df$age_range)
# Race White


p2<-plot(boost,i="Race_White")
p2
table(subset_data2$Voted2020, subset_data2$Race_White)
table(subset_data2$Race_White)
# Voted on left, white on top


# Grouped Bar Plot
counts <- table(subset_data2$Voted2020, subset_data2$Race_White)
age_plot<-barplot(counts, main="Voter Turnout by Race: White",
                  xlab="Race White", col=c("seashell2","pale green"),
                  legend = rownames(counts), beside=TRUE)
text(age_plot, font=2, col=2:8)
```

```
# Registration Month


p3<-plot(boost,i="RegMonth_JAN")

p4<-plot(boost,i="RegMonth_OCT")

p3;p4


table(subset_data2$Voted2020, subset_data2$RegMonth_JAN) #5,741 72%

table(subset_data2$Voted2020, subset_data2$RegMonth_OCT) #14,311 84.6%
```