Statistically "Learning" the 2016 Election

Taking the class for 231 credit

Zhuowei Cheng
Perm: 4917159

Jing Xu
Perm: 3970654

In this paper, the 2016 election is analyzed and predicted with both election and census data. Exploratory data analysis, clustering, and various classification methods are conducted to investigate and predict the election. Different classifiers, including decision tree, logistic regression, LASSO regression, KNN, LDA, QDA, SVM, random forest and boosting, are compared. Counties with a higher percentage of white people are more likely to support Trump and counties with bigger population are more likely to support Clinton. Also, `Transit`, `Minority`, `Income`, `Professional` are found as important predictors from models. Linear regression is utilized to predict total votes of each candidate but found to have lower accuracy than classification when predicting county winner.

I. INTRODUCTION

Election prediction have been a hot topic, especially when a statistician, Nate Silver, predicts every state correctly in 2012.

Voter behavior prediction is a hard problem for many reasons. An unobserved variable, the intended voting behaviour in each state and nationally, is to model for election result prediction. There are much uncertainty in this process. First, when conducting polls, there can be sampling errors. For example, pollsters might have selection bias when choosing samples or people do not provide their true intention due to different reasons. Second, the data collected only represents how people think they will do at the time they are asked but their minds can change over time, which is not that easy to model accurately. Last, the voting is a quite subjective behavior, affected by many unobserved factors such as emotion, political socialization, peer effects from family and friends and so on. These factors are hard to be well integrated to the analysis and the behavior itself might not have good predictability.

So what about Nate Silver's approach in 2012 allowing him to achieve good predictions? Nate Silver built a hierarchical model to predict the states and national results. He integrated polling data by weights to reduce the error and combined them with other kinds of empirically informative indicators. Bayesian approach was used to calculate the probability of support. His model accounted for the uncertainty in the forecast and simulated the election thousands of times. Numerous uncertainty considered in Nate Silver's approach, including demographics, temporal change of voting intention, sampling error, house effect and others, helped to achieve such a high accuracy.

However, in 2016, the prediction went off. On the last day before the 2016 election, Silver's model showed Hillary Clinton winning 71% of simulations. Although in Silver's defense, he allowed Donald Trump a 29% chance of winning. One reason about the off probability might be the systematic undetected polling error. Every poll has its error and poll based forecast reduces the total error by aggregating them. However, when the polls all missed in the same direction, aggregated results will not cancel out the systematic error. Another potential reason is that there is not enough empirical data to validate probability models[1]. The mechanism of the prediction is based on that all assumptions in the model are true. To get the margin of the error, the model can be ran numerous time with other assumptions. But the absence of empirical data makes the likelihood of alternate assumption still arbitrary. With hundreds of presidential election data, the best we can do is improving survey quality, discarding outliers and aggregating existing models to improve future prediction.

In this paper, the 2016 election is analyzed and predicted with both election and county census data. Exploratory data analysis, clustering, and various classification are conducted to investigate and predict the election. Important predictors are studied and good election prediction models are obtained after model comparison to this end.

II. DATA

The election data used in our study include two datasets. Dataset `election.raw` contains information about the votes each candidate received on county, state and also national level.

The `census` data contains various information on higher than county resolution including population, gender, race, income, employment, occupation, transportation modes, etc. Description of the variables are shown

TABLE I. Variables in `election.raw`

Variable	Explanation
county	The name of the county
fips	Federal information processing standard
candidate	The name of candidate
state	The name of state
votes	The number of votes one candidate received

in Supplement. Collectively, the two dataset brings possibility to identify the important election predictors and also construct prediction models.

A. Data prepocessing

Six rows with `fips = 2000` in `election.raw` are removed because they replicate the 6 rows with `fips = "AK"`. After the removal, the dimension of `election.raw` becomes 18345 rows and 5 columns. Then `election.raw` is separated into federal-level dataset `election_federal`, state-level dataset `election_state` and county-level dataset `election`.

Based on `election` and `election_state`, two dataset, `county_winner` and `state_winner`, are created by taking the candidate with the highest proportion of votes for each county and state, respectively.

TABLE II. First three observations in `census.ct`

State	County	Men	Citizen	Income	IncomeErr
Alabama	Autauga	48.43266	73.74912	51696.29	7771.009
Alabama	Baldwin	48.84866	75.69406	51074.36	8745.050
Alabama	Barbour	53.82816	76.91222	32959.30	6031.065
IncomePerCap	IncomePerCapErr	Poverty	ChildPoverty	Professional	Service
24974.50	3433.674	12.91231	18.70758	32.79097	17.17044
27316.84	3803.718	13.42423	19.48431	32.72994	17.95092
16824.22	2430.189	26.50563	43.55962	26.12404	16.46343
Office	Production	Drive	Carpool	Transit	OtherTransp
24.28243	17.15713	87.50624	8.781235	0.09525905	1.3059687
27.10439	11.32186	84.59861	8.959078	0.12662092	1.4438000
23.27878	23.31741	83.33021	11.056609	0.49540324	1.6217251
WorkAtHome	MeanCommute	Employed	PrivateWork	SelfEmployed	FamilyWork
1.8356531	26.50016	43.43637	73.73649	5.433254	0.00000000
3.8504774	26.32218	44.05113	81.28266	5.909353	0.36332686
1.5019456	24.51828	31.92113	71.59426	7.149837	0.08977425
Unemployment	Minority				
7.733726	22.53687				
7.589820	15.21426				
17.525557	51.94382				

Using `census` data, the information is aggregated into county-level data by computing total population-weighted average of each predictor for each county to account for the population variability. Rows with missing data are filtered out. Some predictors such as `Men`, `Employed`, `Citizen` are converted from absolute values to percentages. A new predictor `Minority` is generated by adding `Hispanic`, `Black`, `Native`, `Asian`, and `Pacific`, which are removed after creating `Minority`. To avoid collinearity among predictors, if a set adds up to 100%, one predictor of that set is deleted. Consequently, `Walk`, `PublicWork`, `Construction`, `Women`, `White` are removed. The cleaned data is named `census.del`. The detailed description of each predictor is in Supplement.

Two dataset `census.subct` and `census.ct` are created to contain information in sub-county resolution and county level. In `census.subct`, two attributes `CountyTotal` and `weight` are added by calculating the total population in each county and the percentage of each sub-county region's population in county. Then the sub-county information is aggregated into county level with the weights of each sub-county regions to construct `census.ct`. The first three observations in `census.ct` are shown in Table III.

TABLE III. All Datasets

Dataset	Information
election_federal	Federal-level election data
election_state	State-level election
election	County-level election results
county_winner	Winner of each county
state_winner	Winner of each state
census.subct	Sub-county level census data
census.ct	County-level census data

III. EXPLORATORY DATA ANALYSIS

In this section, the dataset is explored to gain insights and find important predictors. Spatial analysis and principle component analysis are conducted.

A. Spatial Analysis

There are 32 candidates in total in the 2016 election. Figure 1 shows a bar chart of the logarithm of votes received by each candidate national-wise. In Figure 1, Hilary Clinton and Donald Trump received the most of the votes on a similar magnitude.

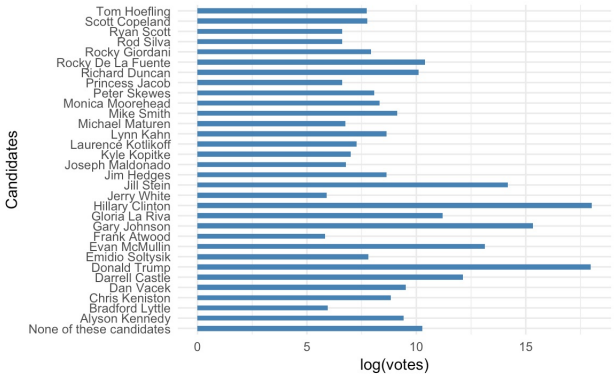

FIG. 1. Bar chart of the logarithm of total votes each candidate got

Figure 2 is the state and county maps of United States. The winning candidates of each state and county will be shown onto US map to have a more clear visual view.

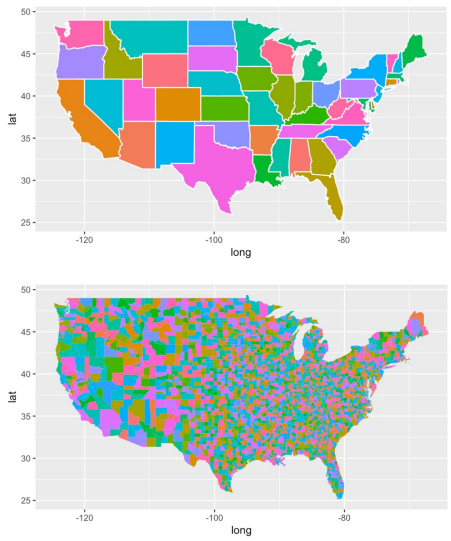

FIG. 2. State and county level maps

The winning candidate of each state and county are mapped (Figure 3). Trump won 31 out of 52 states and most of the counties.

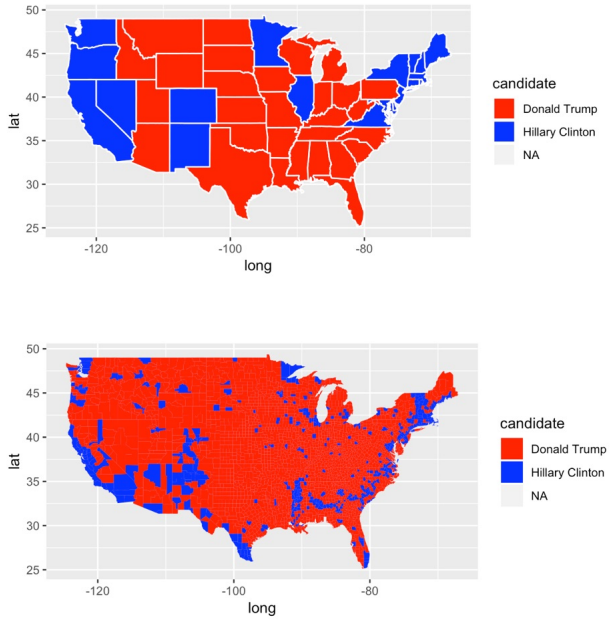

FIG. 3. Winning candidate in the state and county level

Demographics usually play an important role in election. The two demographic information shown here is the percentage of white people in a county and a county's total population (Figure 4). In the top graph, more red represent a higher percentage of white population and more blue for lower percentage of white. Bottom graph is the map of the logarithm of total population in each county. More red means a smaller population and more blue means a larger population. Grey areas in both graphs are due to missing values. The similarity between these two maps with the winner map in Figure 3 shows that race and population of a county can be important predictors in predicting election results. White people are more likely to support Trump and counties with bigger population are more likely to support Clinton.

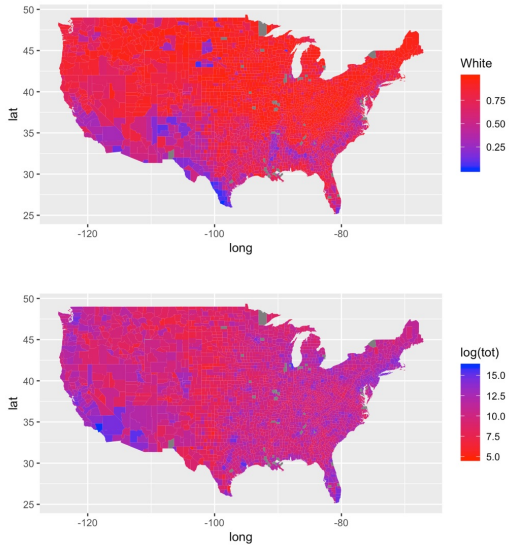

FIG. 4. Top graph is the map of percentage of population that is white in each county. Bottom graph is the map of the logarithm of total population in each county

B. Principle Component Analysis

Principle Component Analysis(PCA) for dimensionality reduction for both county and sub-county level data(`census.subct` and `census.ct`) is performed. The features are both centered and scaled before running PCA. If the data is not centered, the first principle component(PC) will always point to the mean, which is not very informative. Also, without scaling, different features with different scales, features with large values (e.g. `Income`, `IncomePerCap`, `IncomePerCapErr`) will be identified as important by explaining more variance. Clustering and classification using reduced data would be shown in the later part.

By checking the first two principle components of both county and sub-county level data(`census.subct` and `census.ct`), `IncomePerCap`, `ChildPoverty` and `Poverty` are found to have the three largest absolute values of the first principal component in the county level. And for sub-county level: `IncomePerCap`, `Professional` and `Income` have the three largest absolute values of the first principal component.

The features with opposite signs in a certain principle component are also checked because it will give us information about what that principle component is

capturing. Here, the interest is in the first principle component. In county level data, `Men`, `Income`, `IncomeErr`, `IncomePerCap`, `IncomePerCapErr`, `Professional`, `Transit`, `OtherTransp`, `WorkAtHome`, `Employed`, `PrivateWork`, `SelfEmployed` and `FamilyWork` have positve signs while `Citizen`, `Poverty`, `ChildPoverty`, `Service`, `Office`, `Production`, `Drive`, `Carpool`, `MeanCommute`, `Unemployment`, and `Minority` have negative signs. This implies that the first component represents the contrast between different races, occupations, being employed or not and transportation modes. In sub-county level data, `Men`, `Citizen`, `Income`, `IncomeErr`, `IncomePerCap`, `IncomePerCapErr`, `Professional`, `Drive`, `WorkAtHome`, `MeanCommute`, `Employed`, `SelfEmployed`, and `FamilyWork` have positve signs while `Poverty`, `ChildPoverty`, `Service`, `Office`, `Production`, `Transit`, `Carpool`, `OtherTransp`, `PrivateWork`, `Unemployment`, and `Minority` have negative signs. This implies that the first component represents the contrast between races, income, transportation modes and occupations, which is very similar to county level results.

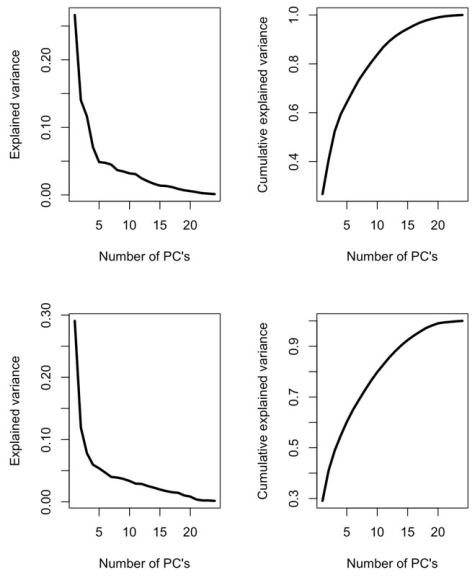

FIG. 5. Top graph is the PVE and cumulative PVE for PCA of county data. Bottom graph is the PVE and cumulative PVE for PCA of sub-county data

With PCA, a lower dimensional representation that captures main features of data can be found. Figure 5 shows the proportion of variance explained (PVE) and cumulative PVE for both county and sub-county analyses. Here the numbers of minimum number of PCs needed to capture 90% of the variance for the county and sub-county data are 13 and 14, respectively. Figure 5 shows the proportion of variance explained (PVE) and cumulative PVE for both county and sub-county analyses.

IV. METHODS

In the Methods section, clustering and classification methods including decision tree, Logistic regression, LASSO regression, KNN, LDA, QDA, SVM, random forest and boosting are conducted. The comparison of all methods is in Results section.

A. Clustering

Clustering identifies groupings in data without known labels. Here, data is clustered into 10 clusters using the `census.ct` and also the first two PCs respectively. Before clustering, both original data and the first two PCs are scaled. Figure 6 shows the results of the clustering based on these two datasets. The top graph is the result of clustering using `census.ct` and bottom graph is with the first two PCs. From the top graph, most of the counties clustered into one group while in the bottom graph the size of the clusters seems more even.

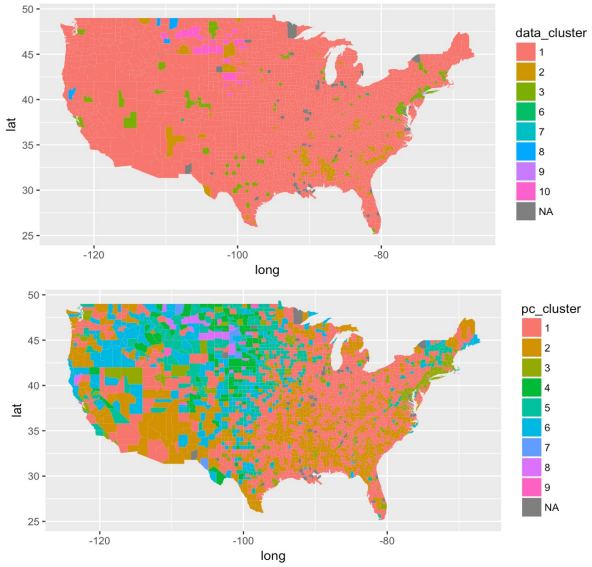

FIG. 6. Top graph is the map of clusters from clustering of original data. Bottom graph is the map of clusters from clustering of the first two PCs data.

One can check which clustering has a result more related to voting behavior by looking at a specific county: San Mateo County. The winning candidate for San Mateo county is Clinton. If the clustering captures features related to voting behavior, the winning candidate of most of the other counties in the cluster with San Mateo will be Clinton too. In the results from dataset `census.ct`, 39.58% counties have Clinton as their winning candidate while clustering using the first two PCs find 35.2% counties voting for Clinton. The clustering with original data without dimensionality reduction seems to put San Mateo in a more appropriate clusters. The similarity of vot-

ing behavior is better captured in the clustering with `census.ct`. This result is shown in Figure 6 too. The top map has a more similar structure with Figure 3. This result is not surprising. The first two PCs only capture a part of the variance but not all. In clustering, there might be other important information that is not contain in PCs but in the original data.

B. Classification

To perform classification, `county_winner` and `census.ct` are combined into one dataset and then partitioned into 80% training and 20% testing set randomly.

In this section, the models are trained with training set and evaluated with testing set. classifications using decision tree, logistic regression and LASSO, KNN, LDA, QDA, SVM, Random forest and Boosting will be explored.

1. *Decision Tree*

In decision tree, a tree structure is build and the labels are predicted for each leaf. A tree with no restraint on size (Figure 7) is built at first. The original tree has 82 nodes and it's hard to get useful information from looking at the tree structure. Thus, a 10-fold cross validation is performed to prune our tree. The goal of pruning is to find a tree which balances the misclassification error and tree size. Figure 8 is the pruned tree with seven nodes.

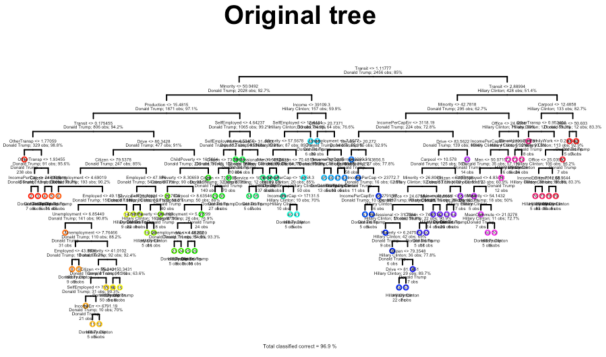

FIG. 7. Tree without pruning

Overall, 85% of the counties in the training set vote for Trump. In the pruned tree, `Transit`, `Minority` `Production`, `Income` appear to be splitting variables. Counties with less public transportation, less Minority are more likely to have Trump to be their winning candidate. In the counties with more than 50% Minority and less than 39109 dollars median house income have Clinton to be their winning candidate.

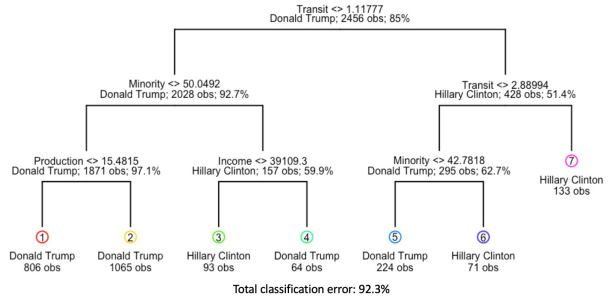

FIG. 8. Pruned tree with seven nodes

2. *Logistic regression and LASSO regression*

Here, a logistic regression is conducted to predict the winning candidate in each county. To control for overfitting, a LASSO regression is used too.

In logistic regression, probability of the response is modeled based on predictors. From the fitted model, `Citizen`, `IncomePerCap`, `Professional`, `Service`, `Office`, `Production`, `Drive`, `Carpool`, `Employed`, `PrivateWork`, `FamilyWork`, `Unemployment` and `Minority` are significant predictors. To interpret them, for example, if the citizen proportion increases by one percentage and other factors fixed, then the probability of Hillary Clinton wins the county becomes $e^{0.011} = 1.011$ times of the original probability. If the minority proportion increases by one percentage and other factors fixed, then the probability of Hillary Clinton wins the county becomes $e^{0.127} = 1.135$ times of the original probability. With one percentage increase in the proportion of people who drive, the probability of Hillary Clinton being the county winner becomes $e^{-0.181} = 0.834$ times of the original one. To compare with the results from decision tree, some of the factors are consistent with the results before, including: `Minority`, `Production`.

For LASSO regression, a K-fold cross validation is performed to select the best regularization parameter from 0.0001, 0.0005, 0.001 and 0.005. The best parameter selected by cross validation is 0.0005. From the fitting result, except `Men`, `Income`, `ChildPoverty` and `SelfEmployed` and `OtherTrans`, other predictors have non-zero coefficients. The predictors with zero coefficients are all not significant in the unpenalized logistic regression. The non-zero coefficients are quite close in two models.

3. *KNN, LDA, QDA, SVM, random forest and boosting*

Here, more classification methods are conducted.

K-Nearest Neighbors Classification(KNN) is a non-parametric approach with no assumptions about the shape of the decision boundary. Also, KNN does not show which predictors are important. In KNN, a 10-fold cross validation is performed to find the best number of

neighbors, which in this case, is 15.

Linear Discriminant Analysis(LDA) and Quadratic Discriminant Analysis(QDA) model the distribution of the predictors first and then uses Bayes' theorem to estimate the probability of the response. LDA has a linear boundary and QDA assumes a quadratic decision boundary. Thus QDA models a wider range of problems than linear methods.

Support Vector Machine(SVM) is a classifier defined by a separating hyperplane. SVM has a cost function which controls the amount of slack and it controls the bias-variance tradeoff. We performed cross validation to find the optimal cost among 0.001, 0.01, 0.1, 1, 5, 10, 100. The final result sets the cost at 10.

Random forest is an extension to decision trees with bagging. It de-correlates the trees. Random forests grows many classification trees. Each tree gives a classification, and the forest chooses the classification with the most votes over all the trees in the forest.

With variable importance plot, the top 5 most important variables are selected in decreasing order of importance based on Model Accuracy and Gini value. From Figure 9, `Minority`, `Transit`, `Professional`, `Unemployment`, `Service` and `IncomePerCap` are the top most important predictors in the random forest.

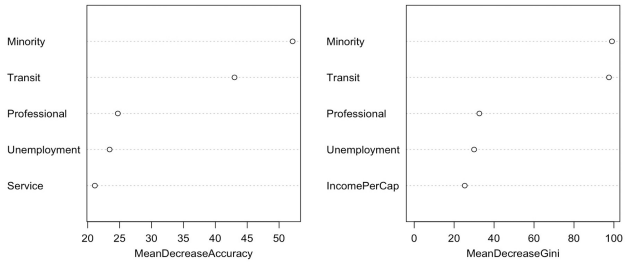

FIG. 9. Variance importance for random forest

Boosted trees fit trees multiple times sequentially, each trying to correct its predecessor. By checking the relative influence table, high influence variables can be picked out. Here, `Transit`, `Minority`, `Unemployment`, `Income` and `Professional` are the top five highest relative influence variables.

TABLE IV. The five highest relative influence variables in boosting

Variables	Relative Influence
Transit	46.78386
Minority	36.30344
Unemployment	2.99464
Income	2.32431
Professional	2.26640

In conclusion, `Transit`, `Minority`, `Income`, `Professional` are picked out by most of the classification methods as important variables.

V. RESULTS

Training and test errors of election prediction using nine different classifiers are listed in Table V. Decision tree, logistic and LASSO regression have their pros and cons. Decision tree is easy to interpret and fast to run. However, it typically has very high variance. A small change in the data might lead to learning a very different tree. Logistic regression is also easy to interpret. However, the assumption for it might not hold for some data and when an indicator has complete separation, it might cause over-fitting. LASSO logistic has low variance but if the true model is not sparse, the model will have high bias. Also, they concern different aspects of the question. Decision tree answers for what are the important factors and the other two methods focus on the effect of each factor on the response.

Logistic regression and LASSO logistic regression have almost the same performance in this particular study. However, LASSO classifier is supposed to have low variance due to shrinking coefficients of five non-important predictors to zero, leading to a simpler model. Decision tree has slightly higher errors then the two logistic regressions. This comparison of area under the receiver operating characteristic (ROC) curve also demonstrate this from the perspective of the trade-off between false positive rate and true positive rate.

TABLE V. Training and test errors using original data

Methods	Train.error	Test.error
Tree	0.077	0.085
Logistic	0.066	0.080
LASSO	0.068	0.080
KNN	0.131	0.145
LDA	0.066	0.081
QDA	0.073	0.089
SVM	0.019	0.062
Random forest	0.001	0.052
Boosting	0.001	0.043

Generally, the other eight classifiers expect KNN have competitive performance both in training and test errors. The KNN (k=15) classifier has a relatively high training error (i.e. 0.131) and test error (i.e. 0.145), which might be caused by the high dimensionality of the data. With this large dataset (around 3000 observations), QDA and LDA works well, implying that the normality assumption on the conditional probability holds here. QDA has a slightly worse performance than LDA. SVM classifier with the radial kernel beats the mentioned classifiers above.

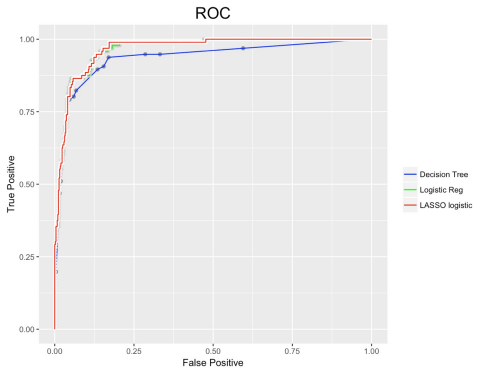

FIG. 10. ROC curves for decision tree, logistic and LASSO regression

Based on the errors, the ensemble classifiers (random forest and boosting) outperform all other methods. The aggregation of weaker learners' (single decision tree) learning results contributes to their high accuracy. Their training errors are really low (around 0.001) while their test errors are 10 times training errors but still lower than test errors of other algorithms. Actually the fact that the test error higher than the training error is common using other classifiers. Comparing classification errors, the ensemble classifiers (such as random forest and boosting trees) are recommended for future election prediction.

Further, a more explicit comparison of the distribution of test errors among the nine classifiers is shown in Figure 10. The most eye-catching impression is that the KNN classifier has much higher test errors than other classifiers. The LASSO logistic regression has relatively more high errors than logistic regression since some predictors are dropped but slight lower variance. LDA has lower bias and variance than QDA here. Also, random forest and boosting have both low bias and variance, which beat other classifiers in this problem.

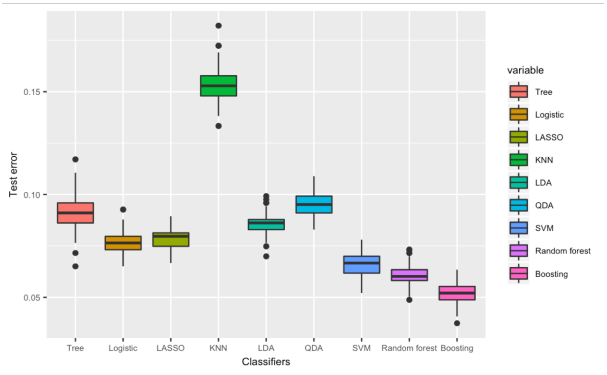

FIG. 11. Boxplot of bootstrap test errors

VI. DISCUSSION

In the previous sections, the election and census data is analyzed by exploratory data analysis, clustering and various classification methods. Several important predictors are found, such as, `Transit`, `Minority`, `Income`, `Professional`. Although dimensionality reduction was performed, the data used in all classification methods above is the full set of predictors. Using reduced data sometimes can improve classification performance. Thus, all the classification methods are conducted again on the first 13 PCs. The first 13 PCs are picked because it is the minimum number that capture 90% of the variance. Table VI shows the results of training and test errors. Compared to the models trained with original data, most errors here are larger except for KNN. This might be caused by the fact that PCs are only capturing 90% of the variance and have missed some infromative message. As for KNN, both training and test errors are much better than previous models using `census.ct`. The reason of this might be that using the first 13 PCs, the reduced dimension, can lead to better performance. For all the other models, although the errors are larger, the model is supposed to be smaller due to fewer predictors, which implies that those models should have lower variance.

TABLE VI. Training and test errors using the first 13 PCs

Methods	Train.error	Test.error
Tree	0.044	0.107
Logistic	0.087	0.098
LASSO	0.087	0.098
KNN	0.078	0.081
LDA	0.096	0.101
QDA	0.101	0.119
SVM	0.036	0.076
Random forest	0.001	0.089
Boosting	0.007	0.078

In the Methods section, classification has been done to predict county winners. Here, linear regression is used to predict the total votes for the two most important candidates (Clinton and Trump) by county. That means, the total votes for both Clinton and Trump are fitted as two linear models using census data separately.

`Citizen`, `IncomePerCap`, `Professional`, `Service`, `Office`, `Transit`, `MeanCommute`, `PrivateWork`, `SelfEmployed`, `Minority` are found to be significant when predicting Clinton's votes. Similarly, `Citizen`, `Professional`, `Service`, `Office`, `PrivateWork`, `Unemployment` are significant factors for regressing Trump's votes.

To compare this method with classifiers used, the candidate with higher votes would be predicted as the winning candidate for each county and the error is computed by comparing the winning candidate predicted and true value. The training error of this method is 0.2247 and the test error is 0.2512, larger than classifiers. This comparison shows that classification on county winner might be

more appropriate than linearly regressing vote amount when predicting election results.

VII. CONCLUSION

In this paper, the 2016 election is analyzed and predicted with both election and census data. Exploratory data analysis, clustering, and various classification methods are conducted to investigate and predict the election. In the exploratory data analysis, counties with a higher percentage of white people are found to prefer to support Trump and counties with bigger population tend to support Clinton. With classification, `Transit`, `Minority`, `Income`, `Professional` are found as important predictors. Comparison of various methods are shown in results section. KNN does not perform well when trained with original data but well when with reduced 13 PCs data, which might be related to better performance of KNN with data with lower dimensional data. Also, random forest and boosting have both low bias and variance, which beat other classifiers in this problem. Thus, they are recommended for future election prediction. Finally, linear models are fitted to predict total votes of each candidate. Compared to classification methods, linear models have worse performance in predicting the winning candidate of each county.

Election prediction is a complex problem. In this paper, we only used census data to perform prediction on election results. Combining other information, for example, polling data, might improve our prediction. Also, analysis with other models, for example, neural networks, might be worth conducting.

[1] Pradeep Mutalik, Why (Almost) Everyone Was Wrong, (available at https://www.quantamagazine.org/why-nate-silver-sam-wang-and-everyone-else-were-wrong-20161109/)