An Investigation of Ensemble Methods and Vote Forecasting

Jackie Excell
Department of Computer Science
Faculty of Science
University of Auckland
Auckland 1010, New Zealand
yexc128@aucklanduni.ac.nz

Francis Tang
Department of Computer Science
Faculty of Science
University of Auckland
Auckland 1010, New Zealand
ftan638@aucklanduni.ac.nz

Michael Thomson
Department of Computer Science
Faculty of Science
University of Auckland
Auckland 1010, New Zealand
mtho614@aucklanduni.ac.nz

***Abstract*— Democratic elections form the backbone of western society and as such the predictions of upcoming election results have received and continue to receive extensive focus by both the research community and the public. In this paper we test the validity of predicting the US Presidential results using ensemble methods, specifically Random Forest. We train our predictor on the 2012 election results along with population metrics provided by the US Department of Agriculture to 'predict' the result of the 2016 presidential election. We compare our predictor's result to other machine learning techniques, Adaboost, k-NN and SVM to validate our prediction.**

I. INTRODUCTION

The art of election prediction is a field of great interest to political scientists, statistical researchers, news media and the public. The goal of this paper is to build prediction models that will correctly predict the results of past US presidential elections. We will train our models on the results of the 2012 presidential elections and then test it on the 2016 presidential election results. The models we will build will use the following techniques. Random Forests, Adaboost, k-NN and SVM. We are most interested in the performance of the random forest model in comparison to the other three.

For our training set we combine the county election results of the 2012 elections with population metrics for those counties as compiled by the US Department of Agriculture. By doing this we are able to see a variety of attributes for the counties including metrics such as education level, gender distribution, age, ethnicity and more for a total of 78 different attributes. For our test set we perform the same data collection and cleaning of the 2016 election results, compiling the same set of attributes with the county level voting results.

We will train and test four models using the mentioned techniques with the goal of verifying the validity and robustness of the Random Forest model in comparison to Adaboost, SVM and k-NN. In doing so we will show how ensemble methods such as Random Forest are a good choice for researchers wanting to perform election prediction that are at worst comparable to other popular techniques and at best better than the currently popular techniques.

II. BACKGROUND

A. First-Past-The-Post voting

In a first-past-the-post voting system each voter gets a single vote that can be used to vote for a single candidate. Then, the candidate with the most votes is declared the winner. This means that the winner is not necessarily the person with the majority vote. For example, in a three-candidate case the voting distribution could be as follows:

- Candidate 1: 30%
- Candidate 2: 34%
- Candidate 3: 36% - Winner

In this case the majority of the voting pool did not vote for candidate 3, however candidate 3 is still the winner.

B. United States Presidential Elections

The Presidential Elections are done under a modified first-past-the-post system. The modification is such that there are what we describe as two rounds of voting for the president.

In the first round each voter gets a single vote and votes for their preferred candidate. This candidate can be someone on the ballot or a write in candidate. Which is a candidate not included on the ballot but whom can be voted for by writing the candidates name on the voting ballot. Each individual state then tallies these votes to determine the winning candidate for that state.

The electoral college representatives for that state then all vote for the winning candidate for that state. This is the second round of voting. Each state has a different number of votes in the electoral college, for example California has 55 votes. This means that the winning candidate voted for by California voters would receive 55 electoral college votes. In this round of voting the candidate with more than half, at least 270, of these votes is declared the winner.

Note, it is possible for any individual electoral college member to vote for any candidate they want. Though in practise this is almost never the case. As such this is not a consideration we take take into account in this paper

III. HYPOTHESIS

Our research hypothesis is as follows. We believe that the ensemble method of Random Forest can be trained on past election data to accurately predict the results of future

elections. We believe that the method presented in this paper is comparable to other popular prediction techniques that are used for vote prediction and forecasting in other fields. To test this hypothesis we will first conduct the training and testing of a Random Forest based vote forecaster, if the results of this first phase are of an acceptable level. that is we believe our model provides competitive accuracy measurements. Then we will perform the same training and testing using other popular methods and compare the results. We consider our hypothesis proven if the results from our random forest model are at least as good as the results from the subsequently created models. We will be using the same training and testing data sets for all created models.

IV. Related Works

A. Traditional Election Prediction

Due to the importance of elections worldwide there is correspondingly much literature around the techniques of predicting their results. According to [1], there are four main types of forecasting results: Structuralists, Aggregators, Synthesisers, and Judges. Structuralists usually apply standard regression algorithms to national level data to provide a final prediction before upcoming elections. For aggregators, during the campaign, they prefer to use national opinion polling data to give multiple forecasts as the campaign progresses. Synthesisers is a combination of the previous methods. They will start with some set of political and economic background data. Then they will then aggregate the national and state level data, then combine both to offer dynamically updating forecasts. The final type are Campaign observers or Judges. Judges make predictions based on polling data, official data, and also patterns within some other data attributes like weather and economic markets [1].

These four mainstream forecasting approaches have historically worked well. With the rapid development of artificial intelligence and the increasing volume of available data, recent years has seen machine learning approaches becoming embedded into existing election prediction techniques.

In 21st century, with the rapid development of artificial intelligence, there is increased motivation for researchers to apply machine learning techniques to vote forecasting.

B. Machine Learning and Election Prediction

[2] presents a new method for solving prediction problems, rather than comparing different models and theories to find the best one, this paper proposes a new approach to avoid finding the best model for prediction.

The approach [2] uses is an Ensemble Bayesian Model Averaging (EBMA). This approach assumes that there is no best model for making predictions, it averages across multiple different models to make the final predictions. According to the paper, EBMA evaluates its component models by their performance on predictions then generates different weights for its models.

The motivation for EBMA is in trying to make more accurate predictions by combining these component models which can be used to capture some different selectively accurate understandings of the data.

EBMA works in the following way. We have k models: M_1, M_2, M_3, ... , M_k, they will be used to make predictions. Each model has its own probability which comes from a prior distribution, and the outcome is associated to each individual Probability Density Function (PDF) conditional on M_k. So, after applying the Bayes model, the predictive distribution of the outcome y^t will be as following:

$$p(y^t) = \sum_{k=1}^{K} p(y^t|M_k)p(M_k|y^t) \tag{1}$$

We can treat this probability density function as the weighted prediction. In [2], 10 forecasting models and 16 presidential elections from 1948 to 2008 are applied to predict the 2012 US presidential election. The following table was generated to list the weights, Root Mean Squared Error (RMSE) and Mean Average Error (MAE) of different models. When Abramowitzs model seems to get a better accuracy, Lockerbie and Hibbs are heavily down-weighted by EBMA.

Ensemble Weights and Fit Statistics for Calibration-Period Performance (1948–2008)

	ENSEMBLE WEIGHT	RMSE	MAE
Ensemble		0.859	0.696
Abramowitz	0.674	0.981	0.769
Berry/Bickers	0.006	0.808	0.750
Campbell (Trial Heat)	0.047	1.610	1.252
Cuzán (FPRIME short)	0.178	1.800	1.357
Erikson/Wlezien	0.012	1.775	1.549
Hibbs	0.004	2.806	2.240
Holbrook	0.015	2.144	1.734
Lewis-Beck/Tien (Jobs)	0.039	1.264	1.050
Lockerbie	0.009	3.943	3.329
Norpoth/Bednarczuk	0.015	2.411	2.129

The second column contains the weight assigned to each component model in the final ensemble. The other columns show two fit statistics to evaluate the relative performance of each component model and the ensemble across the calibration period. EBMA tends to place higher weight on better performing models, but the relationship is not monotonic.

Fig. 1. Ensemble Weights and Fit Statistics for Calibration-Period Performance [2]

Instead of using these weights to determine the best model, they are used by EBMA to better mix the results from the component models.

Figure 2 illustrates the predictive PDFs results came from EBMA. The EBMAs PDFs data from 2004 to 2008 are

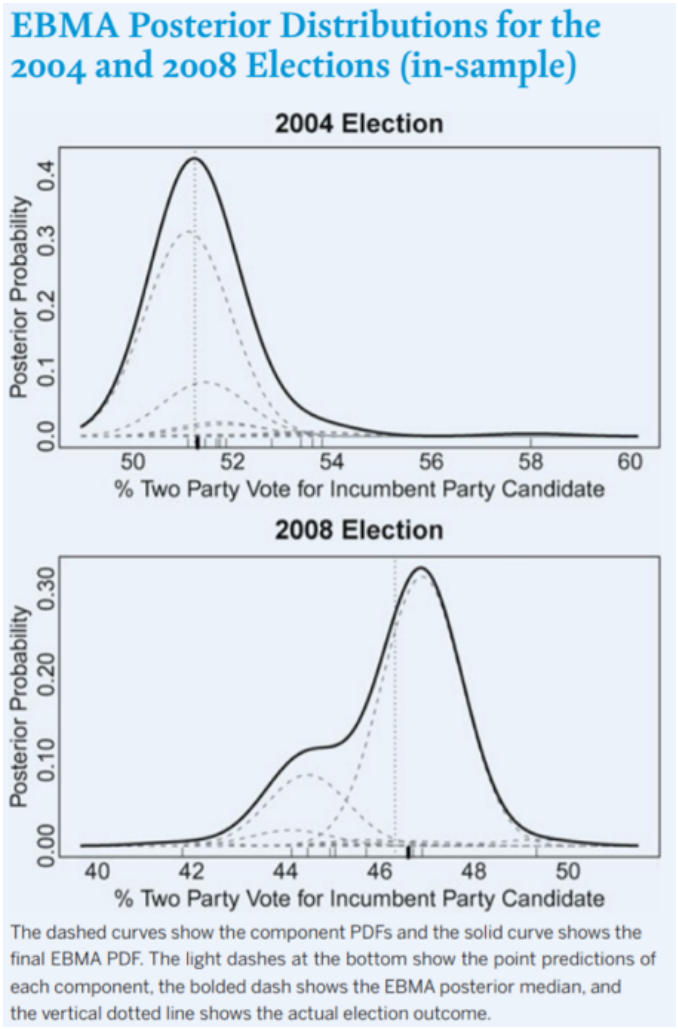

Fig. 2. EBMA Posterior Distributions for the 2004 and 2008 Elections (in-sample) [2]

plotted in bold black lines. Scaled predictive densities of component models are plotted in dashed lines. At the bottom of the plot, light dashes represent the point predictions of each of the component models and bold dashes represents the ensemble model.

From these two plots, we can from visual inspection determine that EBMA may not provide the closest prediction to the actual results (vertical dotted line), but it is always close. Using the data in the above tables and plots, this paper predicts that the vote for the Democratic candidate Obama, will be 50.3% with a 95% credible interval ranging from 46.4% to 52.5%. We found the actual result on Wikipedia, which shows that this prediction is accurate.

[2] uses ensemble methods by weighting each of the component models. This approach is intended to avoid over-fitting and also collates successful parts of different models.

V. Methodology

A. Data-sets

There are not many official, easy to find, data-sets that are publicly available on the internet. Many are performed by various private entities and need specific authorisation to access. For our project, we choose to use data from multiple official resources and combine them into a usable and relevant data set - media releases, government websites and university databases, which make their data both available and reliable. The United States Department of Agriculture Economic Research Service (USDA-ERS) [3] offers details on the demographics of each county in the nation - including education (figure 3), population (figure 4), poverty (figure 5), income, and employment (figure 6). Initial polling and the final result of every single U.S. presidential election is sourced from The Guardian [4] website (figure 7).

Predictions should not be made solely by looking at the historical party-vote distribution. We must also consider other attributes which could influence voting patterns. When making our models, the related current year polling data and prior election final results are all useful to improve the accuracy and performance in conjunction with other demographic data.

B. Statistical Models and Analysis

1) Statistical Models: We employ four classification methods to train four models: Random Forests, AdaBoost, k-Nearest Neighbours (k-NN) and Support Vector Machines (SVM), which are all widely used in the field of data-mining and machine learning.

a) Random Forests (RF): Random Forests [5] are a popular ensemble machine learning method, and have been proven to make highly accurate predictions while also avoiding over-fitting the data. Random Forest uses the general bootstrapping method, randomly picking n samples from the training data-set with replacement. Then, uses the samples to construct n random multiple decision trees. When making predictions, the result takes the majority votes over all of the sub-trees.

The advantage of using Random Forest is by randomly growing non-pruned trees, having the best split at each node and making an ensemble prediction, the variance is decreased with very low bias. Therefore, cross-validation is not necessary. However, of use to this paper, to keep the comparison on the same level, 10-fold cross validation is used on the Random Forest model as well.

b) AdaBoost (ADA): AdaBoost, short for Adaptive Boosting, is another widely used ensemble machine learning method. ADA method starts with building a very simple decision tree, then calculating the mis-classification rate. Aiming to correct the mis-classification rate, second tree is built and a new mis-classification rate is calculated. By doing this, new trees keep adding on until the there is a 'perfect' prediction or a maximum number of trees are added determined by a set limit. The goal is that each tree is slightly better than random guessing, and by combine them together the final classifier is able to make very good predictions[6].

The advantage of Adaboost is that it performs very well on binary classification problems, which is essentially what FPP voting is in a two party system. Also, similar to Random Forest, it can handle the over-fitting problem very well.

Educational attainment for adults age 25 and older for the U.S., States, and counties, 1970-2016
Sources: Census Bureau, 1970, 1980, 1990, 2000 Censuses of Population, and the 2012-16 American Community Survey 5-yr average.
For defintions of rural classifications, see USDA, Economic Research Service - http://www.ers.usda.gov/topics/rural-economy-population/rural-classifications.aspx
This table was prepared by USDA, Economic Research Service. Contact: Tim Parker, tparker@ers.usda.gov

FIPS Code	State	Area name	Less than a high school diploma, 1970	High school diploma only, 1970	Some college (1-3 years), 1970	Four years of college or higher, 1970	Percent of adults with less than a high school diploma, 1970	Percent of adults with a high school diploma only, 1970	Percent of adults completing some college (1-3 years), 1970	Percent of adults completing four years of college or higher, 1970
00000	US	United States	52,373,312	34,158,051	11,650,730	11,717,266	47.7	31.1	10.6	10.7
01000	AL	Alabama	1,062,306	468,269	136,287	141,936	58.7	25.9	7.5	7.8
01001	AL	Autauga County	6,611	3,757	933	767	54.8	31.1	7.7	6.4
01003	AL	Baldwin County	18,726	8,426	2,334	2,038	59.4	26.7	7.4	6.5
01005	AL	Barbour County	8,120	2,242	581	861	68.8	19.0	4.9	7.3
01007	AL	Bibb County	5,272	1,402	238	302	73.1	19.4	3.3	4.2
01009	AL	Blount County	10,677	3,440	626	404	70.5	22.7	4.1	2.7
01011	AL	Bullock County	4,245	958	305	314	72.9	16.5	5.2	5.4
01013	AL	Butler County	8,353	2,459	499	541	70.5	20.7	4.2	4.6
01015	AL	Calhoun County	30,535	13,804	3,823	3,921	58.6	26.5	7.3	7.5
01017	AL	Chambers County	13,616	4,816	927	847	67.4	23.8	4.6	4.2
01019	AL	Cherokee County	6,126	1,878	440	329	69.8	21.4	5.0	3.8
01021	AL	Chilton County	10,285	2,805	538	415	73.2	20.0	3.8	3.0
01023	AL	Choctaw County	5,720	1,922	378	362	68.2	22.9	4.5	4.3

Fig. 3. Education Data of Each County [3]

Population estimates for the U.S., States, and counties, 2010-17 (see the second tab in this workbook for variable name descriptions)
These data were posted to the ERS website (at https://www.ers.usda.gov/data-products/county-level-data-sets/download-data.aspx) April 2018. Contact: Ti

FIPS	State	Area_Name	CENSUS_2010_POP	ESTIMATES_BASE_2010	POP_ESTIMATE_2010	POP_ESTIMATE_2011
00000	US	United States	308,745,538	308,758,105	309,338,421	311,644,280
01000	AL	Alabama	4,779,736	4,780,135	4,785,579	4,798,649
01001	AL	Autauga County	54,571	54,571	54,750	55,199
01003	AL	Baldwin County	182,265	182,265	183,110	186,534
01005	AL	Barbour County	27,457	27,457	27,332	27,351
01007	AL	Bibb County	22,915	22,919	22,872	22,745
01009	AL	Blount County	57,322	57,324	57,381	57,562
01011	AL	Bullock County	10,914	10,911	10,880	10,675
01013	AL	Butler County	20,947	20,946	20,944	20,880
01015	AL	Calhoun County	118,572	118,586	118,466	117,785
01017	AL	Chambers County	34,215	34,170	34,122	34,031
01019	AL	Cherokee County	25,989	25,988	25,973	25,993

Fig. 4. Population Data of Each County [3]

Poverty estimates for the U.S., States, and counties, 2016 (see second tab in this workbook for variable name descriptions)
Source: U.S. Census Bureau, Model-based Small Area Income & Poverty Estimates (SAIPE) - https://www.census.gov/programs-surveys/saipe.html

FIPStxt	State	Area_Name	POVALL_2016	CI90LBAll_2016	CI90UBALL_2016	PCTPOVALL_2016	CI90LBALLP_2016	CI90UBALLP_2016
00000	US	United States	44,268,996	44,022,086	44,515,906	14.0	14.0	14.0
01000	AL	Alabama	814,197	796,927	831,467	17.0	17.0	18.0
01001	AL	Autauga County	7,444	6,255	8,633	14.0	11.0	16.0
01003	AL	Baldwin County	24,005	20,132	27,878	12.0	10.0	14.0
01005	AL	Barbour County	6,787	5,551	8,023	30.0	25.0	35.0
01007	AL	Bibb County	4,099	3,194	5,004	20.0	16.0	25.0
01009	AL	Blount County	8,033	6,506	9,560	14.0	11.0	17.0
01011	AL	Bullock County	2,841	2,155	3,527	33.0	25.0	41.0
01013	AL	Butler County	4,880	4,033	5,727	25.0	21.0	29.0
01015	AL	Calhoun County	19,057	16,226	21,888	17.0	15.0	20.0
01017	AL	Chambers County	6,656	5,264	8,048	20.0	16.0	24.0
01019	AL	Cherokee County	4,273	3,327	5,219	17.0	13.0	21.0

Fig. 5. Poverty Data of Each County [3]

Unemployment and median household income for the U.S., States, and counties, 2007-17

Sources: Unemployment - Bureau of Labor Statistics - LAUS data - https://www.bls.gov/lau/
Median Household Income - Census Bureau - SAIPE data - https://www.census.gov/did/www/saipe/
For definitions of rural classifications, see USDA, Economic Research Service - https://www.ers.usda.gov/topics/rural-economy-population/rural-classifications
This table was prepared by USDA, Economic Research Service. Contact: Tim Parker, tparker@ers.usda.gov. Last modified 5/24/2018.

FIPStxt	State	Area_name	Civilian_labor_force_2007	Employed_2007	Unemployed_2007	Unemployment_rate_2007
00000	US	United States	152,191,093	145,156,134	7,034,959	4.6
01000	AL	Alabama	2,175,612	2,089,127	86,485	4.0
01001	AL	Autauga County, AL	24,383	23,577	806	3.3
01003	AL	Baldwin County, AL	82,659	80,099	2,560	3.1
01005	AL	Barbour County, AL	10,334	9,684	650	6.3
01007	AL	Bibb County, AL	8,791	8,432	359	4.1
01009	AL	Blount County, AL	26,629	25,780	849	3.2
01011	AL	Bullock County, AL	3,653	3,308	345	9.4
01013	AL	Butler County, AL	9,099	8,539	560	6.2
01015	AL	Calhoun County, AL	54,861	52,709	2,152	3.9
01017	AL	Chambers County, AL	15,474	14,469	1,005	6.5
01019	AL	Cherokee County, AL	11,984	11,484	500	4.2
01021	AL	Chilton County, AL	19,737	19,067	670	3.4

Fig. 6. Employment Data of Each County [3]

State Postal	County Name	FIPS	Obama vote	%	Romney vote	%
AK	Alaska	0	91,696	41.6	121,234	55
AK	Alaska	2000	91,696	41.6	121,234	55
AL	Alabama	0	793,620	38.4	1,252,453	60.7
AL	Autauga	1001	6,354	26.6	17,366	72.6
AL	Baldwin	1003	18,329	21.6	65,772	77.4
AL	Barbour	1005	5,873	51.3	5,539	48.3
AL	Bibb	1007	2,200	26.2	6,131	73.1
AL	Blount	1009	2,961	12.3	20,741	86.5
AL	Bullock	1011	4,058	76.3	1,250	23.5
AL	Butler	1013	4,367	46.1	5,081	53.6
AL	Calhoun	1015	15,500	33.5	30,272	65.5
AL	Chambers	1017	6,853	47.1	7,596	52.2

Fig. 7. General Election Results by County [3]

However, Adaboost is sensitive to noise and outliers. If there is significant noise in the data, it can significantly mislead the final result, which is a solvable problem for this research as we will develop our own data-sets by aggregating different resources together.

c) k-Nearest Neighbours (k-NN): k-Nearest Neighbours method is a non Parametric lazy learning algorithm. Just as the name implies, the prediction is made based on the k closest instances. Each instance can have n dimensional space, the distance of each dimension is calculated using the standard Euclidean distance[7].

k-NN is known to be very efficient when the training data-set is large. Also, it is resistant to noise and outliers in the training data. However there are some disadvantages to this approach, firstly, the computing cost is very high. Secondly, if some variables have very big range, normalisation is often needed. Due to the different ways of normalisation, the accuracy can be very different depending on the normalisation used.

d) Support Vector Machines (SVM): Support Vector Machines[8] can be both supervised learners and unsupervised learners. For the training instances, each of them is marked as one of two categories that they belong to. The model is trained as a non-probabilistic binary linear classifier. The gap between the two categories is mapped as far as possible. This is so when making predictions, the examples that are mapped in the same space are predicted in the same category. When data is unlabelled, SVM uses an unsupervised learning approach. Where it is to find the natural clustering of the data into groups, and then to map the new data in these groups in order to make a prediction.

Some of the advantages of SVM are as follows. First, it has a regularisation parameter, this makes it easier to avoid overfitting. Secondly it uses a kernel function, and some problems can be solved via engineering this kernel. However, it takes very long time to train the model on large data sets and difficult to interpret variable weights and individual impact.

2) Tools: Tableau, Python and R are our tools of choice for data processing and analysis.

a) Tableau: Tableau is a software which focuses on business intelligence. Its powerful visualisation and data processing tools benefit not only business analysts but also data scientists.

Importantly for our work, Tableau includes support for visualising county level voting data using available geographical databases.

b) IPython: Python [9] is one of the most commonly used programming languages for machine learning and statistical analysis. In order to get a better visualisation of both data and algorithms, IPython (Jupyter Notebook) [10] has been used as the programming and visualisation application for Python based analysis.

There is a wealth of Python machine learning packages, several of which we are using in this project. Scikit-learn is one of the most popular and stable machine learning library for Python, the included packages such as Random Forest, Decision Trees, Support Vector Machines (SVM), Logic Regression, Adaboost, etc. will be used to build statistical models and train the data. Popular scientific computing libraries like Numpy [11] and Pandas [11] will also be used for data processing.

As the USDA data-sets usually contain more than 100,000 rows, which made the traditional data processing tools less efficient. But with Pandas [11], the data-sets can be converted into Pandas DataFrames which makes data aggregation much simpler.

c) R: As one of the most popular statistical and graphical programming languages, R [12] contains a large number of libraries which can be used solve a wide variety of data analysis problems. In this project, we use the ”mlbench”[13], ”caret”[14] and ”pROC” [15] packages.

C. Steps

1) Data Wrangling: The raw data that we have gathered from publicly available sources cannot immediately be used. We must firstly pre-process the data to ensure that it is statistically usable. Firstly, null entry’s and data in the wrong format need to be corrected prior to any kind of processing. During this process some attributes may need to be merged or divided, and new properties may need to be considered. Some of the data may also require transformation.

In this project, we combine the demographics of each county with their general election votes by their unique

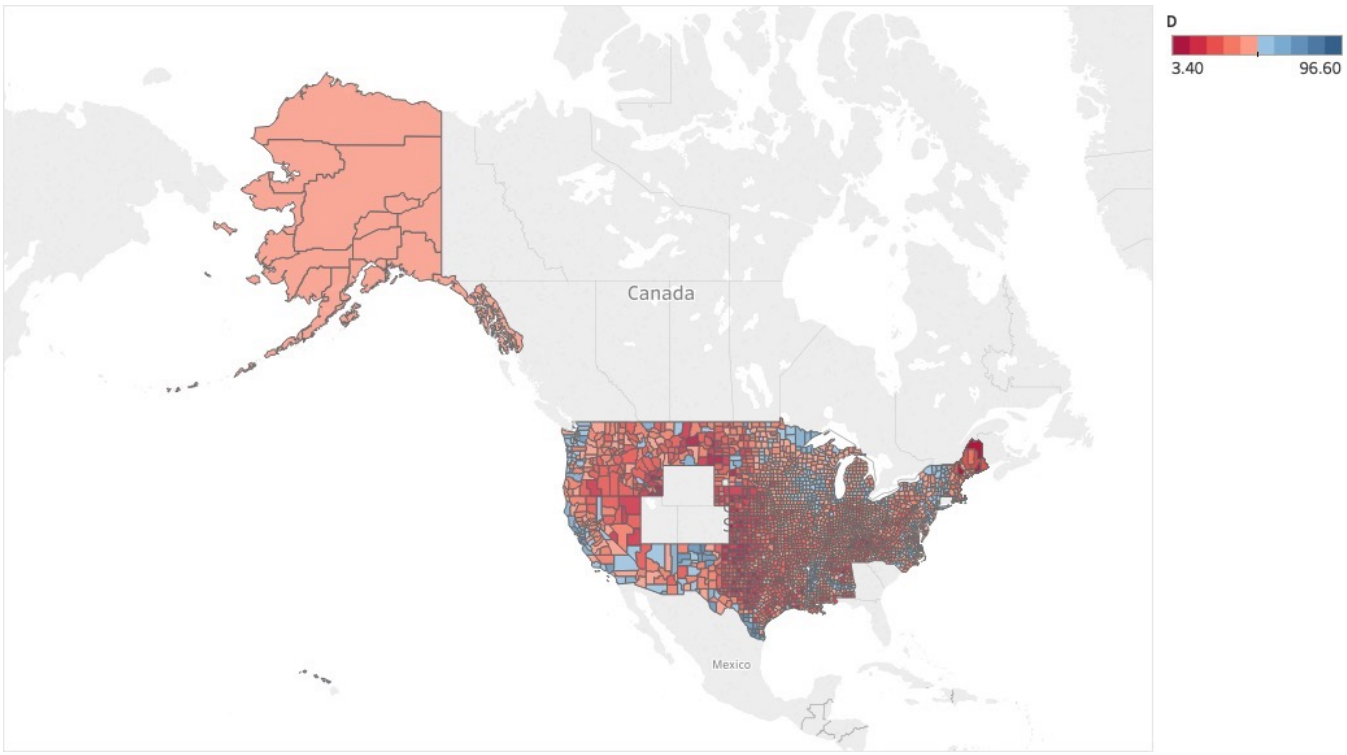

Fig. 8. 2012 U.S. General Election Results by County [4]

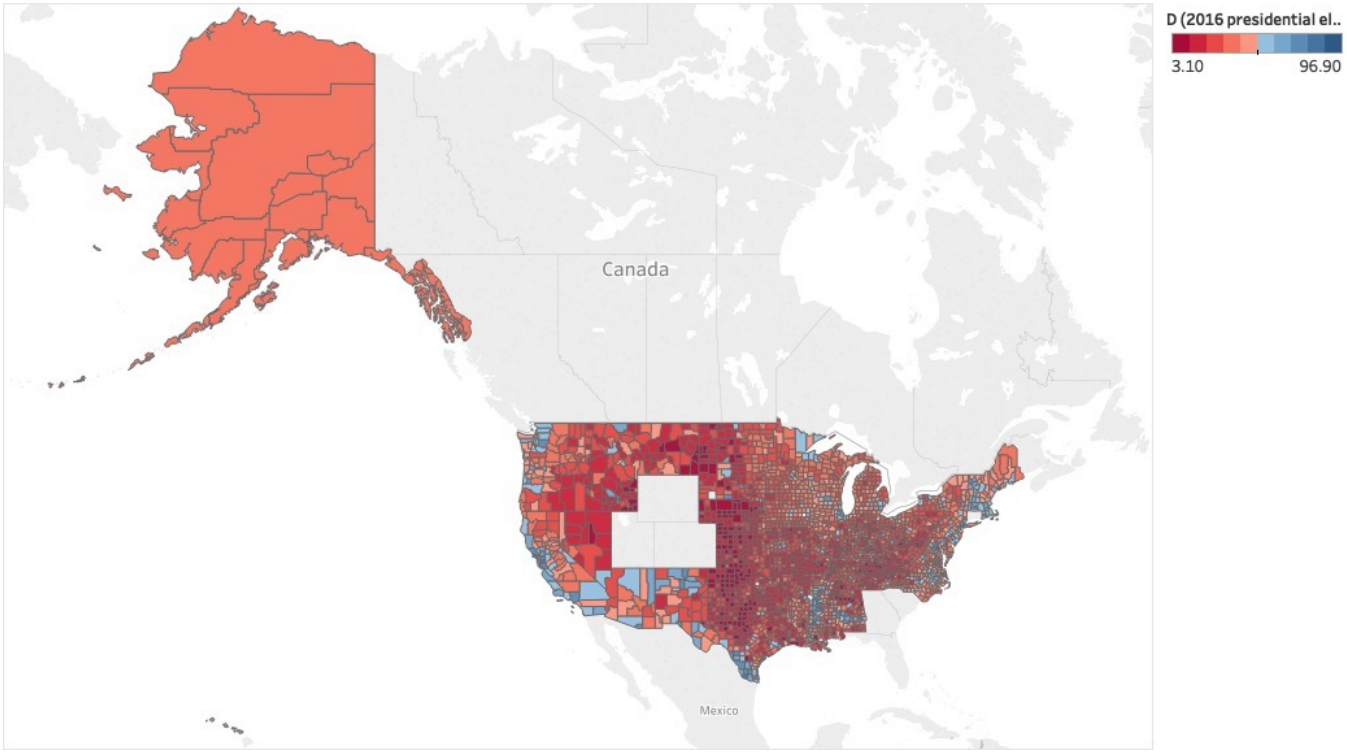

Fig. 9. 2016 U.S. General Election Results by County [4]

identifier - FIPS. For each election result data, we find the related demographics for it. For example, for the 2012 election final results data demographics of the same counties from 2010 to 2012 are combined. The combined data-set has FIPS (county identifier), a large set of relevant demographics, vote counts and a binary attribute to indicate the final elected party of each county (0 for Republican and 1 for Democratic) which is what our models will predict.

2) Exploratory Data Analysis (EDA): EDA is a compulsory part in data analysis. It summarises a data-sets characteristics and usually comes with direct and understandable visualisations [16]. These help analysts know what features and models can suit the data-set.

In this project, we used Tableau to visualise the voting distributions by county level. The election result data found on The Guardian [4] website (figure 7) were imported into Tableau and sorted by its corresponding county. The visualisations (figure 8 and figure 9) illustrates the how 'blue' - Democratic or 'red' - Republican counties are by comparing their vote counts of both parties.

3) Features:

a) Feature Selection: It is tempting that to simply apply machine learning algorithms on the given data-sets as is. However, this would lead to the consideration of too many features of the data. Normally this will increase computational cost and complexity, without appreciable gain in model correctness.[17] Choosing partial features to be specifically analysed, also known as feature selection will often improve the performance of machine learning algorithms. Feature selection provides more a interpretable result model as it reduces the number of parameters being considered. Feature selection can be performed by using algorithms like Chi-square, Information Gain, Support Vector Machines (SVM), Random Forest, Gini Index, and Gain Ratio [17].

We adopted all possible demographics from USDA data-sets, which has more than 20 attributes. This motivates us to apply feature selection process to pick several 'most important' features among the demographics for prediction. After we train out a model, we can apply that model to get a list of feature importance, which will be explained in detail in the Section VI.

b) Feature Engineering: There are often not enough interesting information in a data-set by only looking at the provided features. This makes feature engineering particularly important. It provides an opportunity for an analyst to use domain knowledge to create new features to adapt machine learning algorithms [18]. Although it is expensive and difficult, it is still worthy to do when the data-sets are lacking meaningful features [18].

We can easily found population, education, employment/unemployment rates and poverty data of each county from USDA ERS website. But we also needs more data like house income and racial composition which were published on multiple resources and requires us to pre-process and merge them into our current data-sets.

4) Models and Algorithms: Applying models and algorithms to the data-sets are the core part of the whole data analysing process. By training different models and algorithms on the data-sets, and comparing to the test data-sets, performance and accuracy can be determined at the end of this step.

According to the No Free Lunch theorem [19] which simply states that there is no one model or classifier that works the best for every single problem. This motivated us to try as many algorithms as we can to find the best for a particular problem. In this case, in order to find the 'best' model, we need to validate multiple algorithms of varying complexity to assess their predicting performance and accuracy [17].

In order to predict 2016 election based on the 2012 data, we used county demographics data and voting information from 2012 as our training data-sets to make prediction on Democratic party (1 as win and 0 as lose). For k-NN and SVM model, we use formula below to normalise the training data.

$$\frac{x - min(x)}{max(x) - min(x)} \tag{2}$$

To keep the comparison on the same level, we used 10-fold cross validation and repeated it three times for all four models. That is, the training dataset is divided into 10 parts, each time, 9 parts are used as the training data and 1 part will be the testing data. Each part will be chosen to test the model once. And we repeated this process three times.

- In Random Forest model, we chose 500 trees to grow and 38 variables randomly as candidates at each split.
- In Adaboost model, we use the default 100 iterations and the maximum depth tree growth is 3. We implemented the SAMME algorithm by setting coeflearn = 'zhu' to build the best model.
- In k-NN model, to find the best k, we did cross validation with grid of ks, k = 7 are picked at last as the best k.
- In SVM model, the tuning parameter 'sigma' was held constant at a value of 0.01893442 and C = 1.

After training the models, we compared the performance of the four models. Then, we applied the 2016 election data to these four models to make prediction. By comparing the real result and our predicted result, The prediction performance are evaluated as well.

VI. Results

A. Classifier Performance

a) Accuracy and Kappa value: Figure 10 shows side-by-side box plots of our four constructed models accuracy and Kappa statistic distributions over the three iterations of the 10-fold cross validation. So we have 30 re-samples for each model. Our result show that all four models have very high accuracy and Kappa values. With median values of 0.85 and 0.68, respectively. Overall, random forests displays the highest average accuracy and Kappa value of the four models, followed by Adaboost, SVM, then k-NN. Also,

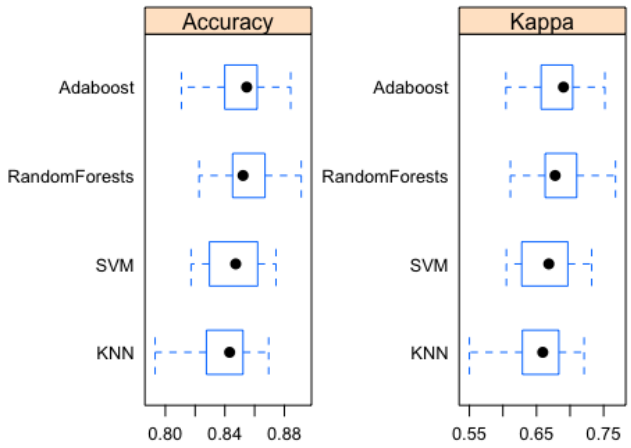

Fig. 10. Models performance comparison

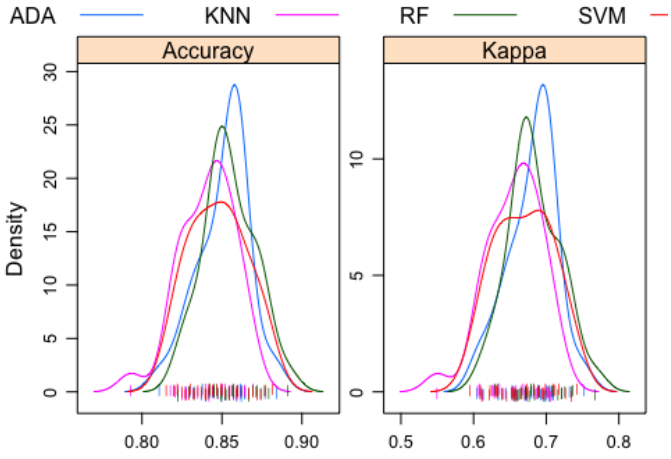

Fig. 11. Models density comparison

SVM expresses less variance than other models showing a more precise but less accurate result.

Figure 11 shows the density distribution plots of the four models. We can see that there is a big overlap between all four models. the distribution for the Random forest model is very similar to that of the Adaboost model while the SVM model is closer to the k-NN model. Adaboost shows the highest peak around 30 and 15 for accuracy and Kappa value respectively. This implies that for the Adaboost model the accuracy values are distributed around 0.86 and the Kappa values are distributed about 0.69. On the other hand, the SVM model has the lowest peaks of about 17 and 7 for accuracy and Kappa values. For our SVM models the accuracy values are mainly distributed about 0.83 to 0.85 and the Kappa values lie between 0.63 and 0.7.

We also performed a pairwise comparison of the four models. From this We show that the random forest model is more accurate than the k-NN model(p-value = 0.02834). The Random Forest model can be up to 0.014 more accurate than

```
p-value adjustment: bonferroni
Upper diagonal: estimates of the difference
Lower diagonal: p-value for H0: difference = 0

Accuracy
             RandomForests Adaboost  KNN       SVM
RandomForests                0.003443  0.013637  0.008058
Adaboost     1.00000                   0.010194  0.004615
KNN          0.02834       0.19723               -0.005579
SVM          0.33695       1.00000   1.00000

Kappa
             RandomForests Adaboost  KNN       SVM
RandomForests                0.002247  0.027169  0.016129
Adaboost     1.00000                   0.024922  0.013882
KNN          0.05627       0.10056               -0.011040
SVM          0.44083       0.81367   1.00000
```

Fig. 12. Pairwise models comparison

k-NN model. We also have evidence that the Random Forest model has a better Kappa statistic than the k-NN model(p-value = 0.056). So we can conclude that Random Forest model provides a statistically significant improvement over the k-NN model (95%).

From this set of comparisons we do not have evidence showing that Adaboost model, k-NN model and SVM model are significantly different from each other.

b) Efficiency: Training our models with 10-fold cross validation repeating 3 times has proven to be time consuming. During the process of training our classifiers, as expected the k-NN method takes the least amount of time to build which means this method is very good to process with large amounts of data. However as mentioned previously k-NN also demands significant preprocessing of the data. Finally, Adaboost takes the most time, and we do not suggest using this method to analyse big data sets.

B. Prediction Accuracy, Sensitivity and Specificity

TABLE I
PREDICTION ACCURACY, SENSITIVITY AND SPECIFICITY

Method	Accuracy	Sensitivity	Specificity
Random Forests	0.93	0.89	0.93
AdaBoost	0.92	0.83	0.94
k-Nearest Neighbours	0.89	0.67	0.98
Support Vector Machines	0.93	0.66	0.98

We apply 2016 county level data as our testing data set to make predictions on the results of whether the result if for or against the Democratic party. From table I we can see, all four models have very high accuracy, especially the Random Forest and SVM model (93%). All four models are very good predictors of true negatives which is the Democratic party losing (Specificity is greater 93%). However, there is a problem for k-NN model and SVM model which is the predictive power for the true positive result (Democratic won) is not very good (Sensitivities are 67% and 66%).

C. Prediction ROC and AUC

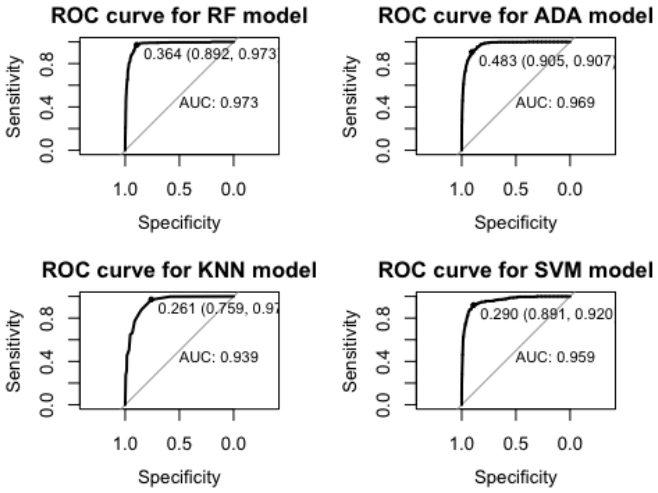

Fig. 13. Models ROC comparison

ROC or Receiver Operating Characteristic is a curve plotting two parameters: true positive rate and false positive rate. AUC is the area under the ROC curve. Figure 13 shows all four models perform very well on the ROC curve and the AUC values are all fairly high, so we cannot tell that there is any significant difference among the four models in this case.

D. Feature Importance

As discussed, feature importance is a crucial consideration for data analysis as critical features have to be selected in order to obtain better predictions. Figure 14 gives us an illustration of the top 12 most important features generated by the Random Forest model.

The result is very intriguing, the higher an attribute was ranked, the more uncertain the vote for the attribute will be. For example, it seems like white people and people who have a bachelors degree have very high importance. This could mean that their votes are 'swing' votes in the general election - they do not always vote for the same single party, and are prone to voting for different parties across different elections.

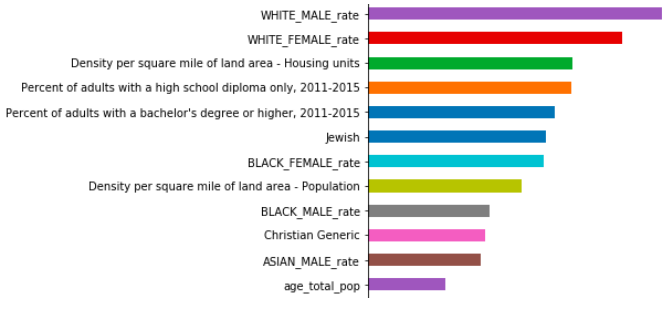

Fig. 14. Feature Importance by Random Forest

VII. Future work

In this paper the Random Forest technique gives a very satisfying result for predicting the 2016 election results using data from 2012, it makes us think maybe methodologies like [2] propose can be transformed to our data-sets to make even more accurate predictions. Further, it is possible to assign weights to different models and then to make an ensemble to generate predictions. Because of time and resource constraints this is not done in this paper, however it would be worthwhile to extend the results from this paper into such an experiment.

As mentioned, we perform predictions based on only a single set of training data. That of the 2012 presidential election results, a more robust model could be built using a larger amount of past voting results. We found that this approach to be too time consuming for the scope of this project and leave the pursuit of this to future work.

VIII. Conclusion

Predicting elections is a widely researched field the world over, there are various of methods which have been proven to work very well. However, one of less explored fields is applying machine learning techniques to forecast results regards to national elections. And even less go down to the county level using accompanying demographic data to make predictions. This project predicts 2016 U.S. election results by using county level previous election and official demographic data to train models, which are four widely used mainstream in the field of machine learning and data mining - Random Forests, AdaBoost, k-nearest neighbours (k-NN) and Support Vector Machines (SVM). After comparison of the performance of each model, the 'best' model for predicting future U.S. election is Random Forest. We show that ensemble methods can perform very competitively in multi-attribute involved binary predictions such as presidential election predictions.

References

[1] M. S. Lewis-Beck and M. Stegmaier, "US Presidential Election Forecasting: Introduction," *PS: Political Science Politics*, vol. 47, no. 2, p. 284288, 2014.

[2] J. M. Montgomery, F. M. Hollenbach, and M. D. Ward, "Ensemble predictions of the 2012 us presidential election," *PS: Political Science amp; Politics*, vol. 45, no. 4, p. 651654, 2012.

[3] USDA. (2018) United states department of agriculture, economic research service. [Online]. Available: https://www.ers.usda.gov/data-products/county-level-data-sets/download-data/

[4] "U.s. 2012 election county results data - the guardian," https://www.theguardian.com/news/datablog/2012/nov/07/us-2012-election-county-results-downloaddata, accessed: 2015-09-30.

[5] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[6] B. Kégl, "The return of adaboost. mh: multi-class hamming trees," *arXiv preprint arXiv:1312.6086*, 2013.

[7] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[8] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[9] P. S. Foundation, "Python language reference, version 2.7," 2010.

[10] F. Pérez and B. E. Granger, "Ipython: a system for interactive scientific computing," *Computing in Science & Engineering*, vol. 9, no. 3, 2007.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2008, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org

[13] F. Leisch and E. Dimitriadou, *mlbench: Machine Learning Benchmark Problems*, 2010, r package version 2.1-1.

[14] M. Kuhn, "Building predictive models in r using the caret package," *Journal of Statistical Software, Articles*, vol. 28, no. 5, pp. 1–26, 2008. [Online]. Available: https://www.jstatsoft.org/v028/i05

[15] X. Robin, N. Turck, A. Hainard, N. Tiberti, F. Lisacek, J.-C. Sanchez, and M. Mller, "proc: an open-source package for r and s+ to analyze and compare roc curves," *BMC Bioinformatics*, vol. 12, p. 77, 2011.

[16] Wikipedia contributors, "Exploratory data analysis — Wikipedia, The Free Encyclopedia," 2018, [Online; accessed 17-September-2018]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Exploratory_data_analysis&oldid=857784051

[17] T. Susnjak, D. Kerry, A. Barczak, N. Reyes, and Y. Gal, "Wisdom of crowds: An empirical study of ensemble-based feature selection strategies," in *AI 2015: Advances in Artificial Intelligence*, B. Pfahringer and J. Renz, Eds. Cham: Springer International Publishing, 2015, pp. 526–538.

[18] Wikipedia contributors, "Feature engineering — Wikipedia, The Free Encyclopedia," 2018, [Online; accessed 17-September-2018]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Feature_engineering&oldid=849817035

[19] David H. Wolpert, "No free lunch theorem," 2018, [Online; accessed 30-September-2018]. [Online]. Available: https://www.no-free-lunch.org