2024 Presidential Election Forecast

Brett Anwar
Computer Science Department
Emory University
Atlanta, United States
brett.anwar@emory.edu

Xavier Pierce
Computer Science Department
Emory University
Atlanta, United States
xavier.pierce@emory.edu

Izana Melese
Computer Science Department
Emory University
Atlanta, United States
izana.melese@emory.edu

***Abstract*—This project aims to use historical US County Presidential Election returns from 2000 to 2020 alongside socio-economic indicators such as demographics, employment rates, and median household income, to predict the outcomes of the 2024 Presidential election across US counties.**

Keywords—voting, data analysis, election, random forest, swing votes, demographics, linear regression, sentiment analysis, gradient boosting

I. Introduction

Predicting election outcomes is a complex challenge that has significant implications for political strategy, governance including foreign policy, trade agreements, and global partnerships. Businesses also have a vested interest in election outcomes considering they have the capability to influence market stability, investor confidence, and economic policies. Secondhandedly, election predictions have the capability to shape citizens' confidence about the democratic process and their beliefs on how their lives may change as a result of an election outcome. The motivation behind this project is to derive meaningful patterns and insights from historical election data and socio-economic indicators to forecast future election results. Such predictions can help in understanding the shifting political landscape and the factors that drive electoral preferences. This project involves analyzing a vast array of data, including past election results and various socio-economic indicators, to identify patterns and factors with a significant impact on voting behavior. The ability to accurately predict election outcomes at the county level can provide valuable insights for political campaigns, policy makers, and analysts. It can identify key battleground counties, explain the socio-economic drivers of voting behavior, and develop strategies tailored to the needs and preferences of specific populations.

II. Related Work

After some research, the team recognizes a lack of data mining experiments and/or projects related to predicting election results based solely on historic demographic data.

A. Split Ticket

The most prominent piece of related work is a political analysis website, Split Ticket. Split Ticket uses historic demographic and electoral data to train models used for an election forecast, then making its best attempt at inferring the actual margin for current candidates using current election data. It must be noted that the CEO of this forecaster has highlighted the high potential for bias to "creep into even a code-based approach to elections" [3].

B. Swing Vote Analysis Experimentation

Another particular experiment seeks to predict the election results of the 2024 Indian General Elections. The research calculates what it calls "swing parameters" using techniques like linear regression, Naive Bayes, Random Forest, Time Series, and more [2]. This research identifies "swing parameters" as reasons for swing voting, satisfaction with the current candidate, and current needs; these are parameters and techniques we should potentially include in our own research and experiment.

C. Sentiment Analysis Experimentation

As it stands, most approaches to our selected problem deal in sentiment analysis on social media sites such as Twitter or Facebook. Seeing as many experiments highlight "...the relationship between social media data and political results…" [5], it is likely that our initial proposed approaches, utilizing solely demographic and electoral data analysis, will not suffice for an accurate solution to our problem. If this is the case, we may opt for including some sentiment analysis in our methodologies in an effort to provide an accurate solution.

III. Proposed Approaches

We have contemplated using logistic regression and Random Forest models for predicting the 2024 Presidential election outcomes at the county level.

A. Logistic Regression

Logistic regression for its simplicity and effectiveness in binary classification problems which makes it suitable for predicting a binary outcome (Republican or Democrat).

B. Random Forest

The Random Forest model is great at handling high-dimensional data and reducing the risk of overfitting, providing us with robust alternatives that can incorporate the multitude of factors influencing electoral outcomes. The ability to automatically handle non-linear relationships and wide feature spaces aligns well with our goal of utilizing different data sources like demographics, income levels, employment rates, and historical election results. By constructing forests of decision trees trained on different subsets, this approach can inherently account for county-level variations and uncover localized patterns that drive voting behavior.

C. Sentiment Analysis

We must also consider implementing sentiment analysis on Twitter, Reddit, and/or Facebook data to further enhance the model's predictive capability by gauging public sentiment and potential shifts in voter preferences. We still would like to do more research to find the model that is most applicable to our task.

D. *Gradient Boosting*

Gradient Boosting is an ensemble technique involving the combination of multiple individual models to create a stronger aggregate model. This model employs decision trees sequentially to minimize prediction errors and improve overall model performance.

E. Support Vector Machine (SVM)

The Support Vector Machine model is an effective method for binary classification tasks, making it suitable for predicting election outcomes as either Republican or Democrat at the county level. SVM works by finding a hyperplane that separates data points from different classes, with a focus on maximizing the margin between them. This characteristic makes it particularly useful in handling complex, high-dimensional data. The choice of kernel function, such as linear, radial basis function (RBF), or polynomial, allows SVMs to handle non-linear classification problems which we expect to be the case for our datasets.

IV. System Design and Implementation

A. Tools and Technology

We will use the Python programming language, renowned for its simplicity and powerful suite of data analysis and machine learning libraries. Each chosen package offers specific functionalities that are crucial for different stages of our analysis, from data manipulation to visualization and modeling. alongside packages and tools that are simple to integrate and easy to use.

a. Numpy

NumPy is the fundamental package for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently.

We utilize NumPy for its efficient array operations, which are essential for handling and performing calculations on numerical data. Its functions are used for tasks like data transformation, normalization, and other preprocessing steps that require mathematical operations.

b. Pandas

Pandas is an open-source data analysis and manipulation tool, offering data structures and operations for manipulating numerical tables and time series. It's particularly well-suited for handling and analyzing input data in various formats, such as CSV or SQL databases.

Pandas is our primary tool for data preprocessing, including loading, cleaning, and transforming the election and demographic datasets. It enables us to filter rows, merge datasets, handle missing values, and create new features with its powerful DataFrame API.

c. Seaborn

Seaborn is a Python visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn is particularly useful for exploring and understanding data through visualizations.

It aids in understanding the distribution of key features, identifying patterns, and visualizing relationships between variables. Plots such as histograms, box plots, and scatter plots are used to analyze the demographic factors and their relationship with voting patterns.

d. Scikit-learn

Scikit-learn is one of the most popular machine learning libraries for Python. It features various classification, regression, and clustering algorithms, including support vector machines, random forests, gradient boosting, k-means, and logistic regression, among others. It is designed to interoperate with NumPy and Pandas. Scikit-learn is the backbone of our modeling phase.

We use it to train various machine learning models, perform cross-validation, and evaluate model performance. Its comprehensive suite of preprocessing tools also supports feature selection, encoding of categorical variables, and data normalization or standardization for the modeling process.

B. Datasets

The primary dataset for this analysis includes county-level results from US presidential elections spanning from 2000 to 2020. This dataset contains detailed information on votes received by each candidate in every county, alongside metadata such as the election year, state, and county names.

To enrich our analysis and improve the accuracy of our predictions, we also integrate a demographic dataset that includes variables such as median household income, unemployment rate, educational attainment, and racial composition for each county. This data is sourced from the United States Census Bureau and is aligned with the election dataset based on county code (fip) and year, where available.

Below we list the datasets that were used in the training and testing of the predictive model. The presented datasets require extensive preprocessing before a training or testing model can be applied.

Dataset	**# Entries**	**Source**
County Presidential Returns 2000-20	72,617	Link
Unemployment 2000-22	3,198	Link
Median Household Income 2004-20	15,990	2004: Link 2008: Link 2012: Link 2016: Link 2020: Link
Education Dataset (2000, 2008-12, 2017-21)	72,153	Link

C. Preprocessing

a. Attribute Selection

Irrelevant attributes including state, office, version, and mode are removed. State_po (state postal code) and county_fips (county identification number) are retained until demographic data integration is complete.

b. Row Filtering

Entries for minor party candidates ("Other", "Libertarian", and "Green") are removed to focus the analysis on the major parties, though this step can be adjusted based on the analysis' scope.

c. Feature Engineering

A new feature representing the ratio of “candidatevotes” to “totalvotes” is created to normalize the data, providing a percentage representation of votes received by each candidate.

d. Data Integration

The election dataset is merged with unemployment, data, aligning on county_fips and year. This step enriches the dataset with additional features that may influence voting patterns.

e. Data Cleaning

Missing values are handled appropriately, either through imputation or removal, depending on their volume and potential impact on the analysis. Categorical variables (e.g., state) are encoded as necessary.

D. Modeling

a. Training

For training the predictive models, the following steps are undertaken:

i. Feature Selection: A subset of features, including the vote ratio and demographic characteristics (income, unemployment rate, etc.), are selected based on their relevance and potential predictive power.

ii. Model Selection: The analysis begins with a logistic regression model as a baseline. This simple model will be our initial model and we will look to output a 0 for republican and 1 for democrat based on the various features. Subsequently, more complex models like decision trees, random forests, and gradient boosting machines (e.g., XGBoost, LightGBM) are explored for improved performance. Using the gradient boosting will allow for us to minimize our prediction error and increase the accuracy and AUC-ROC of our model.

iii. Training Process: Models are trained on a preprocessed and cleaned version of the County Presidential Returns 2000-20 dataset with additional augmented features from other listed datasets, with performance monitored via metrics appropriate to classification tasks (accuracy, precision, recall, F1 score).

b. *Testing*

The final phase of the modeling process involves evaluating the trained models on a separate test dataset using Random Forest, SVM and Gradient Boosting to assess their predictive performance in a real-world scenario:

i. Performance Evaluation: Models are evaluated based on their accuracy, precision, recall, and F1 score on the test dataset. These metrics provide a comprehensive view of model performance, especially in handling imbalanced classes.

ii. Cross-validation: To ensure the models generalize well to new data, k-fold cross-validation is employed. This technique also aids in model selection and hyperparameter tuning.

iii. Model Comparison: The performance of different models is compared to select the best-performing model based on the chosen metrics. This comparison considers both the predictive performance and the interpretability of the models.

V. *Experimental Results*

A. *Model 1 - Random Forest*

The Random Forest model provides a baseline for predicting county-level election outcomes based on historical election data and socio-economic indicators. The Random Forest approach demonstrates an initial accuracy of 75%, reflecting its capacity to handle high-dimensional data and account for various feature interactions. However, its performance is limited by the availability and depth of county-level data, particularly in terms of capturing nuanced, localized trends. Future iterations may benefit from additional socio-economic features and finer-tuned hyperparameters to improve prediction accuracy.

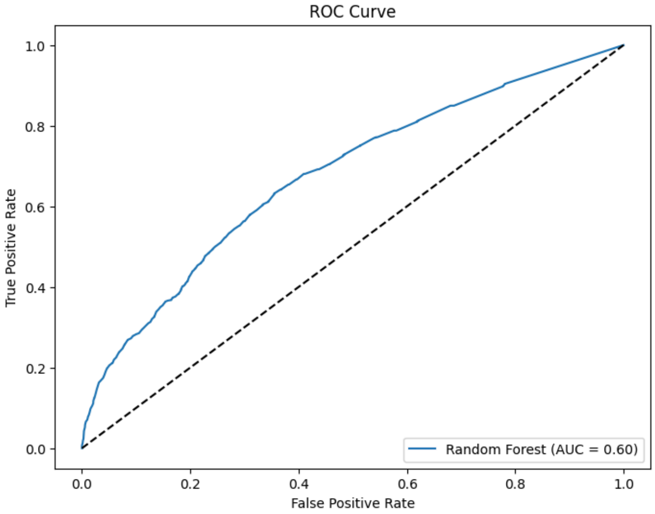

Figure 1: This curve illustrates the trade-off between the true positive rate and the false positive rate for the Random Forest model.

B. *Model 2 - SVM*

The SVM model offers a more refined approach, and with our implementation we treat the year as a categorical variable through one-hot encoding. The model employs a Radial Basis Function (RBF) kernel, enhancing its ability to handle non-linear relationships.

Kernel: Radial Basis Function (RBF)*:*
where gamma is a parameter that defines influence of input vectors.

$$K(x,x') = exp(-\gamma \; || \; x - x'||2)$$

The inclusion of year as a variable and this more sophisticated kernel led to a significant performance improvement. The SVM model achieved a solid accuracy, though further optimization and the inclusion of additional features may further boost its predictive power.

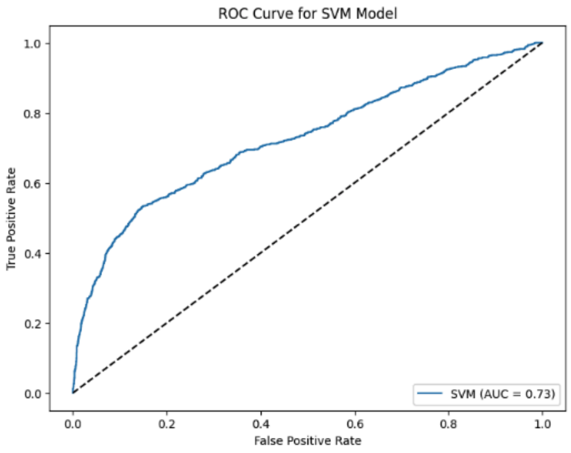

Figure 2: The SVM curve demonstrates the true positive rate versus the false positive rate for the SVM model with a Radial Basis Function kernel.

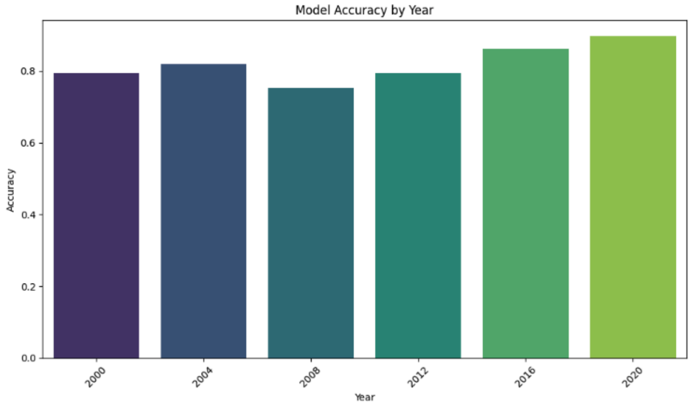

Figure 3: This figure demonstrates the SVM models accuracy across distinct years (2000, 2004, 2008, 2012, 2016, 2020) with the best performance on the 2020 subset.

C. Model 3 - Gradient Boosting

The Gradient Boosting model takes a sequential approach, gradually minimizing prediction errors by adding weak learners, or shallow decision trees, to the ensemble. This model achieved an accuracy score of 81.48% and an ROC AUC of 0.744, indicating high reliability in classifying county-level election outcomes. The model uses a loss function defined as:

$$\mathbf{L(y, f(x)) = \sum(y_i - f(x_i))^2}$$

, where $\mathbf{y_i}$ represents the actual target values and $\mathbf{f(x_i)}$ denotes the predicted values from the model.

To update the model's predictions, the Gradient Boosting algorithm incorporates a learning rate, with subsequent iterations following the equation:

$$\mathbf{f_{new}(x) = f_{old}(x) + learning\ rate \times (residuals)}$$

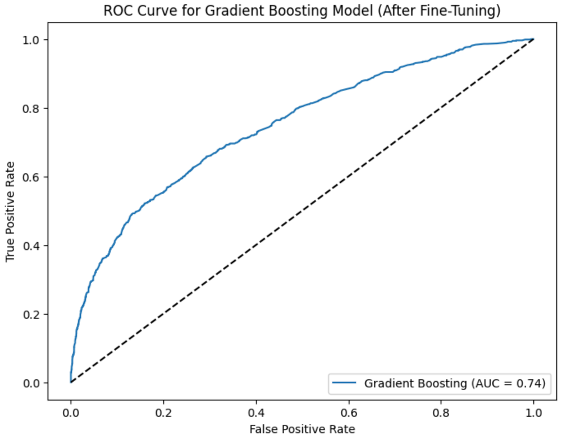

Figure 4: The Gradient Boost curve highlights the true positive rate against the false positive rate for the model, achieving an AUC of 0.744.

A grid search process was employed to fine-tune hyperparameters, leading to an optimal configuration of 200 estimators and a maximum depth of 3.

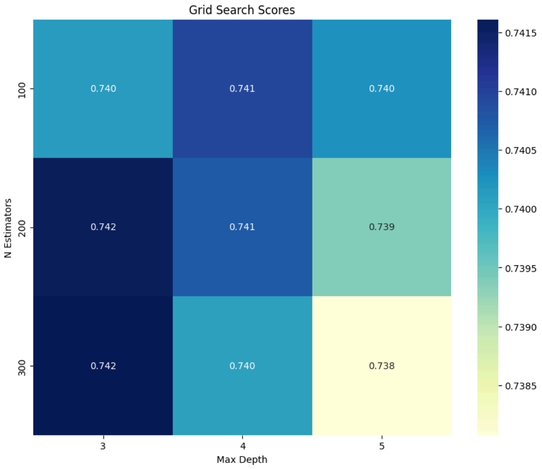

Figure 5: The heat map illustrates the performance of the Gradient Boosting model across various combinations of hyperparameters, including the number of estimators and maximum depth. The intensity of each cell represents the model's accuracy, helping to identify the optimal configuration.

This approach highlights the model's ability to manage complex relationships between features and indicates room for further improvement with additional county-level data.

VII. Conclusions

Our models demonstrate significant predictive capability, with college education (% individuals w/ a bachelor's degree or higher) emerging as the strongest predictor. Our **worst** model performs with an accuracy of 75%. Applying the models to 2024 election data, and incorporating more county-specific data on income, race, religion, and age distribution, will likely enhance the accuracy of our **best** model beyond the current 82% .

References

[1] Alvi Q, Ali SF, Ahmed SB, Khan NA, Javed M, Nobanee H. "On the frontiers of Twitter data and sentiment analysis in election prediction: a review." PeerJ Comput Sci. 2023 Aug 21;9:e1517. doi: 10.7717/peerj-cs.1517. PMID: 37705657; PMCID: PMC10495957.

[2] P. Parida, S. Sinha, A. P. Agrawal and R. Singh Yadav, "Predicting the General Election 2024 using ML and data analytics," 2023 4th International Conference for Emerging Technology (INCET), Belgaum, India, 2023, pp. 1-9, doi: 10.1109/INCET57972.2023.10170638.

[3] R. Leven, "Should we take election forecasts seriously? A computer scientist says yes | CDSS at UC Berkeley," data.berkeley.edu. https://data.berkeley.edu/news/should-we-take-election-forecasts-seriously-computer-scientist-says-yes (accessed Feb. 19, 2024).

[4] E. Culliford, "How political campaigns use your data," Reuters, Oct. 12, 2020. Available: https://www.reuters.com/graphics/USA-ELECTION/DATA-VISUAL/yxmvjjgojvr/

[5] Hazim Moawi, "Predicting Voting Behaviors and Election Results Using Digital Trace Data and Twitter," Social Science Research Network, Jan. 2023, doi: https://doi.org/10.2139/ssrn.44