Aum Narkar
Professor Koehler
Data Bootcamps
16 December 2024

Data Bootcamp Final Project Write-Up

Project Introduction:

One of the most pressing issues in society today is how to deal with bias and discrimination within the police force. The most frequent contact between citizens and police in the United States comes in the form of a traffic stop, yet before the Stanford Open Policing Project, there was no systematized data collection and analysis of these interactions. Minorities, both racial and otherwise, have long reported uncomfortable and undeservedly targeted feelings regarding traffic stops, highlighting the urgent need to assess this data for meaningful insights. My goal with my final project is to conduct exploratory data analysis of the traffic stops data to discover certain variables that correlate with certain stop outcomes. With these results, I will aim to create a predictive model to forecast the outcomes of traffic stops given important input variables. The three classifiers I'll be using are the multinomial logistic regression, the decision tree classifier, and the random forest classifier. By analyzing the features the model deems important and its overall accuracy, I hope to determine whether or not there is data-driven evidence of discriminatory practices within law enforcement. If proven to br discriminatory, the results of this analysis could be useful in several ways. Results can be used by advocacy groups pushing for police reform to support their calls for change, pointing to concrete evidence of inequities that demand attention. This data could also be used by law enforcement agencies themselves to reexamine and revise internal policies that will encourage practices toward fairness and equity in interactions with the public. Even in legal settings, the model's outcomes might be useful for lawyers in making court cases, as means of showing patterns of discrimination, establishing new precedents, or challenging unjust verdicts. This project has the potential to serve as an important resource for social equity and reform within law enforcement.

Dataset Description:

My data comes from the Stanford Open Policing Project, which has stop information recorded for 29 states currently. To conserve space and time, I split the US into its 4 key regions (West, Midwest, South, Northeast) and picked states whose data had ethnic diversity as well as limited null values. I sampled 1000 rows at random from each state and then combined them into one DataFrame that I used for my analysis throughout the project. Sample features in this dataset include subject age, subject sex, subject race, vehicular details, stop date and time, whether or not a search was conducted, and if contraband was found. A lot of the columns are repetitive, having the data in a raw format, and there were several null values across a variety of categories so the data did require significant cleaning, especially when attempting to implement the multinomial logistic regression model.

Exploratory Data Analysis:

Race

The pie chart of demographic backgrounds shows that that race is not equally distributed across our dataset. So in our following analysis, we will focus on proportions as a point of comparison rather than raw metrics, to account for differences in race-based population counts.

By standardizing for population size, it becomes evident that Black and Hispanic drivers are far more likely to get searched if stopped.

However, this alone doesn't provide any evidence of discrimination because it might also be possible that a confounding variable exists. Factors that correlate with contraband might also correlate with race and make minorities more likely to be searched.

Using the Outcome Test

To navigate this challenge, I usd Gary Becker's outcome test (won the Nobel Prize in Economics) to determine bias in traffic search procedures. Essentially, we will look at search rates in tandem with actual hit rates (whether or not contraband was actually found). If more contraband was found for minority groups, it means that the search procedures were warranted. If not, it may be an indication of discriminatory practices.

Looking at the side-by-side comparison of our outcome test, it appears I found some evidence for discrimination in search practices. Most prominently with the Black group, we see that they have much lower hit rates than the White group yet are 5x as likely to be searched. Although not to as great an extent, we see the same pattern with the Asian/Pacific Islander population, who have a hit rate of 0, but still get searched. Interestingly enough, our dataset suggests that roughly 1 in every 5 searches of a white individual yields contraband, yet they are searched the least often.

Gender:

Next we looked to gender to identify any signs of discrimination that might be helpful in building our predictive model. In our dataset we found that males were more likely to get stopped in general, twice as likely to get arrested, and had similar warning rates when compared to their female counterparts. Conducting a similar outcome test, I found that although males are searched more often, that is justified by a higher hit rate, which provides weak evidence for discrimination. However, realizing that males are typically associated with more severe stop outcomes may prove to be useful for our predictive model.

Age:

First, I looked at a distribution of ages across all of the stops and it was skewed to the right, which makes sense. Stops often occur with more reckless drivers or drivers more prone to crime and those activities are often correlated with younger ages as well. Once I created a boxplot distribution of ages for each stop outcome, it showed the fact that more severe stop outcomes, such as citations and arrest had lower age distributions, most likely due to similar reasons. Understanding that a lower age makes an individual more likely to have a severe stop outcome might be an important factor in my predictive model.

I also looked at warning rates per age group and found that senior citizens were considerably more likely to be let off with a warning during a stop, which also makes sense since most of their errors probably weren't ill-intentioned.

Time of Year/Time of Day:

When analyzing the dates and times, I looked for two main features: intraday patterns and seasonal patterns. When looking for intraday patterns, I compared citation rates by time of day with arrest rates and drew an interesting, yet somewhat predictable insight: more arrests happened during the later hours of the night while citations occurred more frequently during the daytime. This makes sense because the more severe, arrestable crimes are likely to take place during the night time and the more mild offenses, such as non-moving violations, are more prone to occurring in the day. However, it still shows that the time of stop can be a useful factor in predicting a stop's outcome. On a similar vein, I compared search, arrest, warning, and citation rates per month and didn't see any clear usable pattern, suggesting that it probably wouldn't help the final predictive model accurately classify different stop outcomes.

Regional Analysis:

When analyzing on the basis of region, I found that if a stop occurred in the Midwest, it had a 0% chance of being an arrest and if a stop occurred in the South it had a 0% chance of being a warning. Although most likely due to fluctuations in individual state data, it still does provide insight, showing that certain regions lean more towards certain stop outcomes, and will serve extremely useful for predictive purposes.

Models, Methods, and Interpretations:

Given the nature of this predictive task, the way I compared the performance of each model was accuracy in relation to a baseline accuracy. I derived the baseline accuracy from the most common stop

outcome's frequency and compared each of my models' results to that baseline to see if the model added any value.

Additionally, for each model, I looked through the main parameters that may influence the model's effectiveness, and did my best to GridSearch through possible permutations to find the best version. However, due to computational limitations I had to limit the variety of parameters I tested and as a result may not have found the most effective model for each variation.

I used the same pipeline for each model, starting with encoding the categorical data and then feeding the encoded data to the model being used.

Decision Tree & Random Forest:

For the decision tree and random forest models, the baseline accuracy was 71.55%, coming from the most common category of citation.

A decision tree aims to split the dataset with features that makes each individual grouping of data after the split as homogenous as possible. It does so by using metrics such as the Gini index or entropy, both measures of randomness in a given grouping, and aims to minimize these values with each additional split. To do so, it picks the highest impact features that clearly segregate the data, and provides the user with feature importances to show with variables that helped reduce the dataset's randomness the most. Decision trees are also very prone to overfitting, leading to poor generalization, which is why I experimented with the max_depth parameter to ensure generalization to the test set.

According to the grid search, the best parameters for this model were keeping the criterion as the Gini index and keeping the model to a max_depth of 6 and that yielded a score of 0.853 from the grid search, for which I left the scoring parameter as accuracy, and had an accuracy of 84.197% on the test dataset. Overall, the decision tree model definitely added value, boosting our prediction power by nearly 16%. Furthermore, analyzing the feature importances, we can see that a heavy weightage (~0.36) was given to the latitude of the stop and the department_id (~0.10), with subject_age being the 5th-most important variable at ~0.02 importance, not using many of the variables we previously mentioned in the EDA.

A random forest operates on the same principle as a decision tree, but uses an ensemble of decision trees to make decisions. It takes the majority vote of the trees to make each decision and as a result normally accounts for more variability in the data. This can be particularly useful to uncover complicated relationships within the dataset and oftentimes helps boost the predictive power of a lone decision tree.

When grid searching for the random forest, I chose not to include the number of estimators because I believed that its default of 100 trees would be enough to take into account the dataset's complexity while still conserving computational power. I still grid searched across max_depth and criterion and this time received different results, with the best estimator being a max_depth of 9 and entropy being used

as the criterion. Unsurprisingly, my random forest model ended up performing better than my singular decision tree, with a grid score of 0.844 but an 84.829% prediction accuracy on the test dataset. Looking at feature importances, it seemed pretty evenly dispersed across a multitude of features, which makes sense given that there were 100 trees in this forest, but also suggests that there weren't as many patterns to pick up in the data.

A common trend I observed across both models was that they were better at correctly identifying warnings, both at almost 95% accuracy and, but significantly worse at identifying citations, often misclassifying them as warnings. That suggests that both models favored predicting warning as the stop outcome, highlighting a possible weakness or oversight in the modelling process.

Multinomial Logistic Regression:

The multinomial logistic regression model operates on the same principle as the binary logistic regression model but it allows for multi-class identification. It builds on binary logistic regression by modeling the probabilities of each class as a function of input features, and assigns the highest probability to the predicted class. This model tends to be effective for linearly separable data, but its performance may struggle when complex or non-linear relationships exist within datasets. It also uses optimization solvers to minimize the loss function and find the best-fit model parameters. Each solver approaches this optimization task differently, and their suitability depends on the size of the data, feature characteristics, and computational constraints. For my solvers, I focused on SAG and SAGA because those work well for large datasets and dense features, so I thought they would aid in handling this predictive task's complexity.

For this model, a lot of data cleaning was required because unlike the other models, it doesn't navigate null values well and as a result I had to drop a significant portion of the dataset, which definitely might have impacted accuracy. Additionally, I had to ensure that the target column encoded the string stop outcomes into numeric values to use for the logistic regression. As a result of the significant changes to the dataset, I recalculated the baseline accuracy for this model and the most common category was citation at 79.67%

One of the key parameters was max_iter, and although I didn't have the time nor computational resources to grid search for the ideal number, I was able to conduct some basic trial-and-error to find that 1500 allowed for a majority of models to avoid the convergence error and give the model enough iterations to find an optimal set of parameters.

While grid searching, I looked at the solvers used, as mentioned previously, as well as tried to combat the issue of class imbalance in my dataset with class weights. The main thing I wanted to avoid was having the logistic regressor simply guess the majority class, and as a result add no value beyond the baseline model. I tried 4 different variations in increments of 0.25, meaning that at certain points I had the minority classes (arrest and warnings) weigh 4 times as much as the majority class (citations). Based

on the grid search, the best estimator was one that weighed the minority classes 4 times as much and used the SAGA solver and it yielded a score of 0.831. However, when presented with the test set, the model performed poorer than the train set, predicting the correct category at 79.145%, performing worse than our baseline. Analyzing feature importances for this model, it becomes clear that this model really wasn't well implemented for the dataset, splitting only by time values and even those feature importances were essentially 0.

Furthermore, the fact that the train set's score was higher suggests that the model may have overfit to the train data and as a result wasn't as generalizable to unseen data. Its poor performance may also be due to the resource constraints I dealt with in trying to save time but also test as many relevant permutations as possible, so it may also be that with better computational power and more permutations explored, the model might better learn the data.

Conclusion and Next Steps:

Conclusion:

Looking through the feature importances, it seems that we did not find any conclusive evidence of discrimination amongst traffic stops. Most of our models found stronger patterns with variables that weren't explored in our EDA and most feature importances were under 0.10, indicating that they didn't have a strong impact on the model's final prediction. However, looking at the other studies conducted regarding police discrimination, the Washington Post's database identifies numerous key insights. It reveals that more than a third of unarmed individuals fatally shot by police were black, despite them making up only 13% of the U.S. population. Young black men are disproportionately affected, and black victims represent a significant portion of cases involving unarmed individuals, which comprised just 1 in 18 fatal shootings between 2016 and 2022. Examples such as this show that police bias is still a critical issue to investigate and address. Through the power of data journalism, projects like this can be expanded on or approached in a different light to yield significant results that can change the course of social injustices.

Next Steps:

Although our modelling portion may not have agreed with our initial data exploration, there are several next steps we can take to improve this project with additional time:

- This project primarily focused on the use of quantitative data to model the average traffic stop, but I think by incorporating qualitative data relating to speech, such as officer tone, driver tone, sentiment of the discussion, I might be able to create more features in my dataset that would give more context into the stop and thus improve the predictive accuracy of each of my models
- One reason the logistic regressor didn't work well was due to class imbalance, and although I tried combatting that with the weight hyperparameter, I didn't have sufficient computing power

to try all of the necessary combinations. Keeping this limitation in mind, improving my sampling procedure in the very beginning to combat this imbalance might have helped the logistic regressor yield more insight.

- In terms of modelling, I'd love to explore using an artificial neural network to model this data. Not only would it help potentially add the speech recognition feature mentioned before, but I think it would be able to capture deeper relationships in the data more effectively. Since we are modelling human behavior, it can be difficult to get accurate predictions, but a neural network might be effectively able to parse the data for underlying patterns.