Arianna Dunham
DATS 6103
Dr. Jafari
May 3, 2021

Individual Report

Introduction

Our group decided to work with D.C. crash data to determine if the available features can be used to predict the level of injury sustained in a D.C. accident. We posed this as a binary classification problem with the targets being minor injury or major injury/fatality and features being: age, vehicle type, ticket issued, impaired, speeding, license plate state, and person type.

Individual Work

We first began working with two datasets. While we were working with these two datasets, we worked on all aspects of the project collaboratively, splitting up each portion of the code. However, we discovered that the way we were working with the two datasets couldn't work and we had to re-start. From there, we split up the sections. Lydia, who had found the dataset and set up the Git Repo, wrote the script to read our data (ReadData) and created our GUI. I did EDA, Random Forest, Logistic Regression (with Logistic Regression Hyperparameter Optimization). RyeAnne took pre-processing, statistics, XGBoost, Naïve Bayes, and Voting Classifier. We decided to write each model as a class so they could be easily called in the GUI. RyeAnne created the structure for this and I followed suit. In the ReadData, EDA, and pre-processing scripts, there are a couple of lines from all of us that were from our previous work with the two datasets. All lines are marked by the author.

In terms of deliverables, outside of code scripts, we all collaborated on the presentation PowerPoint evenly. I edited the video of our presentation. For our final report, we all contributed to the introduction of the dataset. I wrote EDA, the sections about Random Forest and Logistic Regression, and the conclusion. RyeAnne also added to the conclusion and wrote the sections on pre-processing, statistics, XGBoost, Naïve Bayes, and Voting Classifier. Lydia wrote the portion about the GUI.

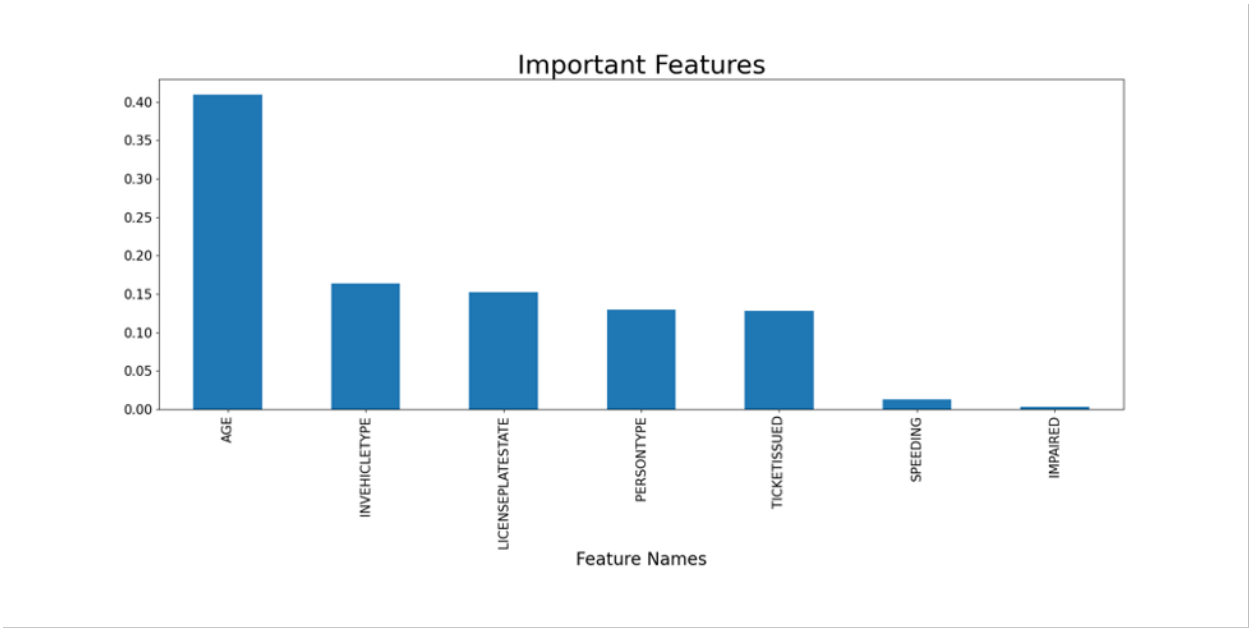
While we split up a lot of the work individually, we would meet at least once a week to discuss our work. This allowed us to be collaborative and add our ideas to the portions we weren't doing ourselves. We regularly checked on each other's works and added comments and suggestions.

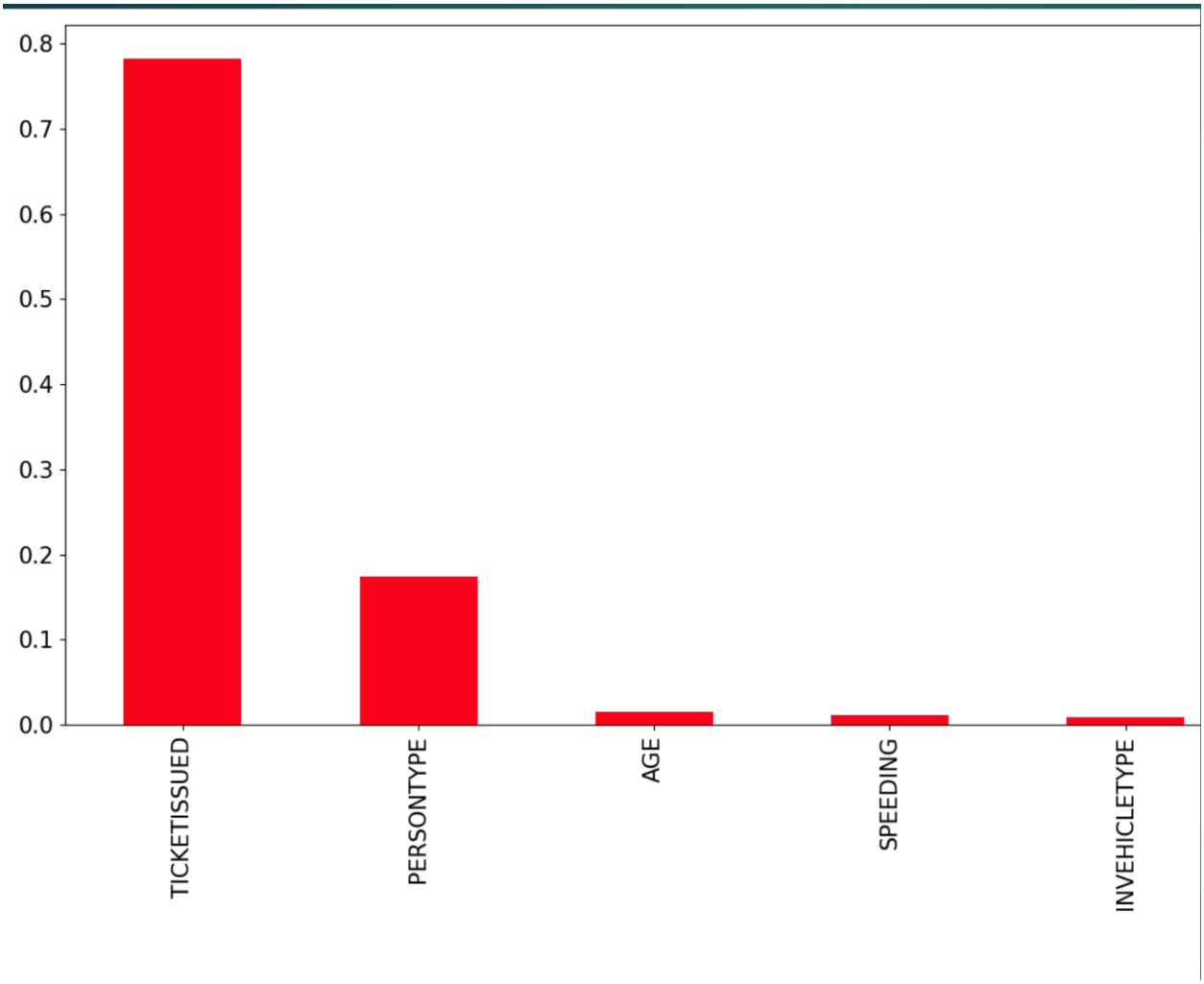
Results

Overall, our models had AUCs between .6 and .7. As you can see in the table below, XGBoost had the highest AUC. However, when we look at sensitivity Random Forest had the highest and is closely followed by Logistic Regression. Our Voting Classifier had the highest specificity. Because our target classes are highly imbalanced, these are important metrics. Naïve Bayes was our worst model because it was not highly sensitive to our features.

	Naïve Bayes	XGBoost	Random Forest	Logistic Regression	Voting Classifier
Overall Accuracy	95.5%	74.8%	74.9%	77.3%	76.3%
AUC	0.70	0.768	0.74	0.664	---
Sensitivity	17.0%	66.7%	74.7%	74.3%	61.9%
Specificity	97.5%	75.1%	63.5%	48.7%	76.8%

Both of our decision tree models checked for important features and produced different results. As you can see in the chart below (blue bar chart), our Random Forest model found age to be the most important feature. On the other hand, ticket issued was the most important feature in our XGBoost model and age was not a very significant feature in this case (depicted in the red chart below).





The difference in important features is notable considering that these were our strongest two models. Since we wanted to predict what, if any, of these features might indicate the type of injury sustained in an accident, the vast difference between the important feature results means we can't yet draw a conclusion about which features are the most important.

Summary

Our models certainly predict injury sustained better than a random guess, but there is room for improvement and higher accuracy. Our overall best model was XGBoost, followed by Random Forest. We believe that a reason we don't have higher AUCs with these models is because our features are vague and describe a wide range of categories. For example, the vehicle type column doesn't specify the make and model of a car, which could provide us with more insight and could give our models more information to learn from. Additionally, the ticket issued category doesn't specify what the ticket was issued for. These could include violations such as texting while driving, not wearing a seatbelt, or running a red light—to name a few. Again, these more specific categories could lead to stronger models.

Code Percentage

Total lines: 125 + 25 + 34 + 73 = 257
Written myself: 110 + 6 + 2 + 10 = 128
Copied (includes modified): 15 + 19 + 32 + 63 = 129
Modified: 15 + 0 + 8 + 42 =65
Calculation: (129-65)/(129+128) = 64/257 =.25 * 100 = 25% of code was copied

References (from my portions of the combined report)

U.S. Department of Transportation National Highway Traffic Safety Administration, *The Economic and Societal Impact of Motor Vehicle Crashes*, 2010. Revised: May 2015. [PDF]. Accessed March 11, 2021. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812013

District Department of Transportation, The Vision Zero Initiative. [online]. Accessed March 11, 2021. Available: https://ddot.dc.gov/page/vision-zero-initiative

Donges, Niklas, *A Complete Guide to The Random Forest Algorithm*, BuiltIn, (2019). [online]. Accessed April 25, 2021. Available: https://builtin.com/data-science/random-forest-algorithm

Hoffman, Juilen I.E., Logistic regression Analysis, Science Direct, (2019). [online]. Accessed April 25, 2021. Available: https://www.sciencedirect.com/topics/medicine-and-dentistry/logistic-regression-analysis