PREDICTING CRASH SEVERITY IN VIRGINIA

Jack Liu
School of Engineering
University of Virginia
Charlottesville, VA 22904
jml7ctd@virginia.edu

Jerry Liu
School of Engineering
University of Virginia
Charlottesville, VA 22904
jyl3xf@virginia.edu

January 8, 2021

1 Abstract

Driving is a ubiquitous part of our everyday lives, yet it is also a task that comes with its dangers. Unfortunately, thousands of car accidents happen each day in the United States [1]. Many factors can increase the chances of an accident occurring and determining what they are can provide us with a way to predict and hopefully prevent crashes in the future. In order to determine which factors are most important in predicting crash occurrence and severity, a machine learning model can be used to classify different crashes. In this study, the performance of different classification algorithms such as logistic regression, decision trees, random forest, and neural networks was compared and the random forest classifier was found to be the best working model to classify crash severity.

2 Introduction

Our ML4VA project involves predicting crash severity in Virginia. It is a practical application of machine learning that can be used in real-world scenarios. In NOVA, Charlottesville, and throughout all other parts of Virginia, car accidents are a serious issue, and many lives and costs can be saved if we can predict crash severity. We can potentially use our machine learning model to create an app that can warn motorists if the current driving conditions make crashes more deadly or provide the data to legislators so that they know where to best focus their efforts on to reduce serious car accidents.

There has already been some related work in this area in the past. For example, in 2013, RongjieYu and Mohamed Abdel-Aty used a support vector machine (SVM) and a classification and regression tree (CART) to evaluate the risk of crashes given real-time data [2]. The CART was first used to analyze the most important features to train the SVM in order to lower the dimensionality of the dataset. Downstream average speed, crash location average speed, crash location standard deviation of occupancy, crash locations standard deviation of volume were the four features selected this way. They found that limiting the SVM to only these factors helped reduce overfitting compared to training with all the features of their dataset. Furthermore, they found that their SVM had overall comparable performance to logistic regression models and artificial neural networks. However, what was interesting was that their RBF SVM models' performance improved as the sample size decreased which is important in the application of crash prediction as there can often be few crashes and thus few data points for the model to train on.

3 Method

The first step in our project was obtaining the dataset used to train the machine learning models. We obtained the data from the Virginia Department of Transportation Smarter Roads website. From there, we were able to gather data on crash reports, speed limits, and road traffic data for the past year throughout the state. This data was originally contained in shapefiles which are meant to be used with GIS software, however, we exported the relevant data into CSV files for easier processing. Once we had the CSV files, the next step was to combine our three datasets which was done using merge operations on identical columns between the three CSV files. We then chose a set of features from our data to

train our model on as some features were not relevant such as the object ids from the shapefile. A few examples of features we chose were the weather condition, time of day, whether alcohol was present, and the distribution of traffic between trucks, cars, busses, etc. The rest of the pre-processing of data was handled by a pipeline that we set up. The pipeline contained two components to handle numerical and categorical data separately. For numerical data, we used Simple Imputer to fill in any missing values with the median and then applied a Standard Scalar to bring each feature to a common scale. The categorical data was encoded using a One Hot Encoder. After some initial experimentation with this data, we realized that the labels for crash severity were severely skewed as most of the crash data were non-serious incidents. In order to balance things out, we combined crash severities together so that we only had severe and non-severe groups instead of multi-class so that each class would have more members. We additionally performed oversampling so that we could obtain an equally split training set for the model. These techniques were used in all further experimentation moving forward. Once all of the data processing was completed, the data was used to train four separate machine learning models. This included logistic regression as our baseline, decision tree, random forest classifiers, and a neural network. The results of these models and comparisons between them are explained below.

4 Experiments

Initially, we tried using a multi-class logistic regression classifier for our model. The trained model has a weighted f1 score of 0.58. The confusion matrix for this model overestimates the case where no injury has occurred in a car accident. (The columns are the actual labels and the rows are the predicted labels. From left to right and top to bottom, the labels are K (Fatal Injury), A (Incapacitating Injury), B (Non-Incapacitating Injury), C (Possible Injury), and O (No Injury).

Figure 1: Confusion matrix for our model

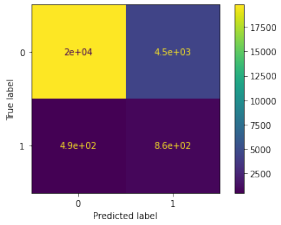
```
[[    5,     3,     2,     0,     0],
[   40,   103,    89,    12,    69],
[   56,   301,   509,    79,   202],
[    0,     0,     0,     1,     0],
[   58,   781,  4570,  1907, 16846]]
```

We later decided to turn the multi-class classification problem into a binary classification problem with the labels of K and A in the class of severe/fatal injury and with the labels of B, C, and O in the class of minor/no injury. We did this because car crashes that result in fatal injuries share many similarities to car crashes that result in severe injuries while crashes that result in minor or no injuries would share many similarities as well. Moreover, since our minor/no injury class was still over-represented in our dataset, we implemented oversampling on our minority class where we copied random training examples of the fatal/severe class and put them in our training set so that there was an equal number of fatal/severe injury training examples as minor/no injury training examples. Our baseline test on our binary dataset was logistic regression. The logistic regression model had the poorest performance of our models and had a large number of false positives. The second model we used was a Decision Tree Classifier which had a better performance than our logistic regression model as it had a larger number of true negatives (minor/no injury class), but it also had less true positives and more false negatives. Our third model Random Forest Classifier performed very similarly to our Decision Tree Classifier, although a little better. The Random Forest Classifier had the highest performance overall. Lastly, we used a neural network with 5 hidden layers with 128 nodes each, using the ELU activation function, and had a learning rate of 0.001. We thought the neural network would perform well since we had a lot of data to learn from, but it had the second-lowest performance.

5 Results

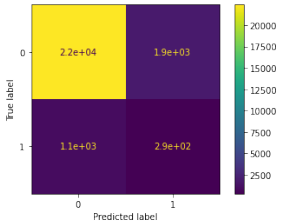
Firstly, our logistic regression model had the lowest metrics with a recall of 80.6 %, a precision of 93.3 %, and a f1 score of 85.5 %.

Figure 2: Confusion matrix for Logistic Regression



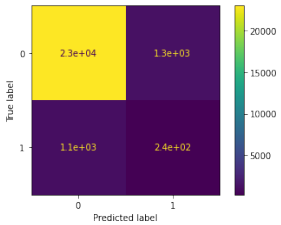
Our Decision Tree Classifier had the second best metrics with a recall of 88.5 %, a precision of 91.2 %, and a f1 score of 89.8 %.

Figure 3: Confusion matrix for Decision Tree



The Random Forest Classifier had the best performance with a recall of 90.7 %, a precision of 91.2 %, and a f1 score of 91.0 %.

Figure 4: Confusion matrix for Random Forest



The neural network had the second lowest performance with a recall of 86.2 %, a precision of 91.7 %, and a f1 score of 88.7 %.

Figure 5: Confusion matrix for Neural Network

```
[[21666  2620]
 [  908   439]]
```

From the feature importances of the Random Forest Classifier, the top five most influential features in determining if a crash results in a fatal/severe injury or a minor/no injury are: 'VDOT DISTR' = 9.38 %, 'COLLISION' = 8.43, 'BELTED UNB' = 6.55 %, 'MOTOR NONM' = 4.55 %, and 'INTERSECTI' = 4.13 %. Some other notable features are 'PERCENT TR' = 3.07 % as it was the most important category of traffic and 'ALCOHOL NO' = 2.15 %, and 'NIGHT' = 1.02 % as they were on the lower end of the spectrum.

6 Conclusion

Something interesting about all our models is the false positives (predict fatal/severe when it's actually minor/none) always outnumbered the false negatives (predict minor/none when it's actually fatal/severe) in the confusion matrices.

This is a consequence of our oversampling during our training when we wanted our model to see more copies of the fatal/severe class so it is not overwhelmed by the majority class, which is minor/no injury. This is good because it is better to predict an accident will be fatal/severe even though it is actually minor rather than predict that an accident will be minor even though it is actually fatal/severe.

One of the key takeaways from this project was the feature importances shown in the results section above. One highlight was the very high weight of whether or not a diver was belted on whether or not a crash resulted in severe injury. This helps to reinforce the idea that you should always buckle up when you drive no matter the situation. On the other end of the spectrum, it was interesting to see some factors like day/night showing up as a feature that did not contribute much to crash severity. However, after some thought, this made sense as low light conditions may make a crash more likely, it does not necessarily contribute to a more severe crash. As a consequence, the feature importances gives us a powerful analysis tool for evaluating what factors cause the most severe crashes rather than seem important and thus helps to point out what issues are most pressing to address to keep our drivers safe.

Although we have made good progress towards using machine learning to keep drivers safe, there is still much more that can be done. With regards to our project, the next step would be to take the results we have obtained and bring it to legislators and others who are able to create actionable change. As for continued research, one area of focus would be to develop better ways of handling the small and skewed nature of our dataset. Thankfully most crashes reported are not severe, but this is not helpful to our machine learning model. Even with the change to binary classification and oversampling, it was difficult for the model to learn from the widely unbalanced data. Developing a more rigorous model to handle these cases would be extremely useful for this application as well as other areas in machine learning in general.

7 Contribution

Since the initial proposal, one of our group members has dropped out of the class and we have continued on the project as a group of two. Jack did the initial data discovery and investigation as well as some data cleaning. He also was primarily responsible for the write up of this report. Jerry finished up the data cleaning by creating the pipeline and he also was in charge of designing our machine learning models. We both worked equally on the video presentation as well as this final paper. Overall, both of us have no concerns about either partner's level of contribution and are happy with what we were able to accomplish together.

8 Appendix

A Jupyter Notebook and Data Files

Jupyter Notebook Link: `https://colab.research.google.com/drive/1vnaa7EATtigaVkg9aMYqkTjFKnsVoDfZ#scrollTo=9ryd74LQk-1c`
Crash Data: `https://drive.google.com/file/d/15OdMePSn1yEur5MunRLXxQXf2Ozoh6wp/view?usp=sharing`
Speed Limits: `https://drive.google.com/file/d/1mm7JZlgICu0u9J8wW6fuD-I5OWj5hc-4/view?usp=sharing`
Traffic Data: `https://drive.google.com/file/d/112QqCS_-IjgElQK9MIST1B_kukFsw2F3/view?usp=sharing`

References

[1] Bill Widmer. 50 car accident statistics in the u.s. & worldwide, Aug 2020.

[2] Rongjie Yu and Mohamed Abdel-Aty. Utilizing support vector machine in real-time crash risk evaluation. *Accident Analysis & Prevention*, Dec 2012.