Predicting Severity of Collisions in Seattle

Garima Anand

September 12, 2020

1. Introduction

1.1 Background

With an increase of people on the road, there has been an increase in the number of accidents, drastically impacting traffic flow as well as the safety of drivers. Over the years, the Seattle Police Department has collected data on kinds of collisions in an attempt to determine prevention measures. Collisions caused by factors such as speeding, being under the influence of drugs, and inattention can be reduced by harshly penalizing such behaviors. However, collisions due to unpredictable factors such as the weather conditions and the road conditions are more difficult to prevent. Therefore, a model needs to be developed to predict possible collisions so that police department can alert the drivers of potential collisions so they can be prepared. Subsequently, these models need to be evaluated for accuracy by comparing the target variable with predictions of the models.

1.2 Interest

The main audience for the study is the Seattle Police Department, Emergency Medical Services (EMS), as well as drivers. The goal of the models is to enable the police department and the EMS to predict the severity of the collisions based on the conditions present and deploy the necessary protocols. Additionally, the results of the deployment of the model can be used to alert drivers of the severity of a potential collision so that the drivers can plan accordingly beforehand.

2. Data Acquisition

2.1 Data Source

The data of the collisions was provided by the Seattle Police Department and recorded by Traffic Records, found on the Seattle Geo Data open source website. The dataset chosen for the study consists of 38 features, and 194,673 observations or entries. The columns in the dataset include metadata such as the location of the collision, weather conditions, road conditions, and the time of the day the collision occurred. The target variable for the model generated will be the severity of the collisions provided by the "SEVERITYCODE" column. The column contains severity codes for varying severity levels of the collisions:

- 3: fatality of parties involved
- 2b: serious injury of parties involved
- 2: injury of parties involved
- 1: property damage of parties involved
- 0: unsubstantial collision

Features such as the weather conditions, road conditions, and the time of day of the collision can be extracted from the data in order to train the model and predict the target variable.

2.2 Data Cleaning

Attributes from the dataset were not used to make the machine learning models such as the number of vehicles involved in the collision were removed from the dataset. Independent attributes chosen for the model were based primarily on addressing the impact of them on the severity of the collision such as weather and road condition. Additionally, the relevant attributes such as speeding was one hot encoded.

3. Methodology

3.1 Exploratory analysis

For the dataset, the target variable of the model consists of two severity codes; therefore, the dataset was used to train binary classification models. To understand the distribution of the target variable, I determined the counts for severity codes, which were 1 and 2.

```
1    136485
2     58188
Name: SEVERITYCODE, dtype: int64
```

The dataset had an unequal distribution of the two classes, which would ultimately cause the model to generate a biased prediction. In order to create balance in the distribution, I decided to resample the dataset, specifically utilizing the undersampling method. Undersampling randomly removed observations from the majority class in order to match the observations in the minority class.

```
1 after resampling:  58188
2 after resampling:  58188
```

In order to determine the features to extract from dataset, I determined the breakdown of the collisions that occurred in different conditions such as time of day, road conditions, and weather. Using these breakdowns, I chose road and weather conditions as the independent features to predict the severity of the collisions because these features seemed to have a more significant influence on the model's prediction accuracy.

```
Clear                       67977
Raining                     20494
Overcast                    16865
Unknown                      6842
Snowing                       503
Other                         434
Fog/Smog/Smoke                359
Sleet/Hail/Freezing Rain       66
Blowing Sand/Dirt              32
Severe Crosswind               15
Partly Cloudy                   4
Name: WEATHER, dtype: int64
```

```
Dry                 76076
Wet                 29266
Unknown              6819
Ice                   690
Snow/Slush            561
Other                  71
Standing Water         62
Sand/Mud/Dirt          46
Oil                    36
Name: ROADCOND, dtype: int64
```

3.2 Feature Preparation

Each feature was extracted from the dataset using the one hot encoding technique to convert the categorical variables that were different conditions in the feature to binary variables and were saved in a separated data frame. Additionally, the features were normalized in order to standardize the data; these features were used to train the models one at a time.

After importing the necessary libraries, the dataset was split into test and train data for the models. The models chosen for the study were the K Nearest Neighbor, Decision Tree, and Logistic Regression.

3.3 Model Development for Each Feature

The K Nearest Neighbor algorithm was chosen since it classifies cases based on their similarity to other cases and the best K value or number of nearest neighbors to consider was determined by calculating the accuracy of the predicted values to the actual values of the test set. Using this value, the model was built again using the best K found earlier and the accuracy metric was determined.

For the Decision Tree model, a similar procedure was utilized in order to determine the best maximum depth value of the decision tree model based on the accuracy of the predicted values of the model to the actual values of the test set. The maximum depth of the decision tree is the maximum length branch of the tree, from the root to the leaf node. It determines how many splits in the tree are necessary to get to an accurate prediction and avoid overfitting[1].

Finally, the logistic regression model was utilized in order to predict the target variable. Similar to the Support Vector Machine, the inverse of the regularization parameter of the logistic regression algorithm prevents the algorithm from overfitting to the training data[2], ultimately allowing more accurate predictions when compared to the actual test data.

After the models were developed for each feature, the models were evaluated by comparing the actual labels in the test result with the label predicted by the model using the Jaccard Index, F1 Score, and Log Loss. The F1 Score was chosen because it is an ideal parameter to show that a classifier has a good apt with both recall and precision, which are important concepts to demonstrate the accuracy of a model. The Log Loss and the Jaccard Index were also determined in order to further portray the performance of the classifier algorithms.

4. Results

4.1 Evaluation Metric Reports

Algorithm	Jaccard	F1-Score	LogLoss
KNN	0.563782	0.563638	NA
Decision Tree	0.546209	0.543917	NA
LogisticRegression	0.548922	0.494079	0.672254

Figure 1: Evaluation Metric Report for the Weather feature

Algorithm	Jaccard	F1-Score	LogLoss
KNN	0.584121	0.582869	NA
Decision Tree	0.559042	0.492537	NA
LogisticRegression	0.556416	0.494928	0.671102

Figure 2: Evaluation Metric Report for the Road Condition feature

4.2 Model Performance

Reports of the evaluation metrics were determined for the features of the study as shown in Figures 1 and 2 for the weather feature and road condition feature, respectively. For both of the features chosen, the K Nearest Neighbor was the most accurate model in terms of predicting the target variable when looking at the Jaccard Index and the F1-Score. Comparing the Log Loss evaluation metric between the weather feature and the road condition feature, the road condition feature was slightly better than the weather feature because the Log Loss measures how far the prediction is from the actual label.

5. Discussion

Based on the evaluation metric reports, K Nearest Neighbor algorithm for the road condition feature seemed to be a more accurate predictor of the severity of collisions. The accuracy values for the features were not as high as I expected them to be, which might be due to the resampling of the data used for feature extraction. However, using these models, the results demonstrate that the road condition feature enabled the model to more accurately predict the target variable, which are is the severity of the collisions. Using the results of the models with the dataset of the collisions in Seattle, the road conditions are a better predictor of predicting potential kinds of collisions; therefore, I would recommend to the Seattle Police Department to allocate resources to warn drivers about the road conditions so that drivers can plan accordingly and drive more carefully.

6. Conclusion

In this project, I analyzed the impact of certain features on predicting the severity of collisions in Seattle using the open source dataset provided by the Seattle Police Department. The severity levels included in the dataset were the property damage or injury to parties involved; therefore the dataset was used to train binary classification models. I built three kinds of classification models for each feature and evaluated the models for accuracy using three evaluation metrics. These models were relatively beneficial in terms of predicting the severity of the collisions; however, the results of the accuracy evaluations suggesting there might be additional model development necessary using different algorithms and other independent attributes to better predict the target variable.

7. References

1. https://towardsdatascience.com/understanding-decision-trees-for-classification-python9663d683c952
2. https://scikitlearn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html