Predicting the Severity of a Car Accident through Classification Models

Emilio Padrón Molina
emiliopm1997@gmail.com
IBM Data Science Professional Certificate

September 26, 2020

Introduction

Accidents happen around the world all the time, causing unnecessary deaths and injuries. Some of them can not be avoided such as natural disasters and lightning strikes; however, there are other types of accidents that can be prevented by technology just like smoke detectors foresee potential fires. Car accidents have been one of the most common types of accidents through out the XX and XXI centuries, and even though there are global and local efforts to decrease this statistic, the numbers are still alarming.

Giving that this problem concerns a big part of the population, some people might have asked themselves: Is there a way to predict a car accident before it happens? If there is, companies that use navigation systems, such as Uber, Google, and Rappi, could take advantage of this and implement a model that could not only optimize traveling time, but also prevent a potential accident.

In this report, a classification model will be built to try to predict the risk of getting in a car accident. Nevertheless, before building a model, it is important to consider the reasons why car accidents happen such as traffic, weather, or poor road conditions, and how this data can be obtained.

Data

Relevant data to carry out this project was obtained from the Seattle Police Department who recorded car collisions in their city. These records go from January 1st, 2004 to May 20th, 2020 and are composed of 194,673 incidents. In the best case scenarios, incidents contain 38 characteristics including the severity of the accident, weather conditions, and type of collision. Nevertheless, it is worth mentioning that neither all the features are significant nor all the incidents contain a full description (there are many features are missing).

After analyzing the dataset carefully, one can say that the most significant characteristic that relates to predicting the possibility of a car accident is the "Severity" column which only contains the values: 1 and 2. Number 1 implies that the accident only resulted in property damage while number 2 indicates that there was an injury involved and, therefore was more severe. Given this, a classification model can be built to predict the "Severity" through the most significant variables in the dataset.

At a first glance, it is easy to notice that many of the incident's characteristics are consequences of the accident itself such as "number of people involved", "number of pedestrians in the accident", and "number of vehicles involved", and therefore, are directly related to the "Severity". Given the redundancy of this information, these sort of columns can be dropped from the dataset. Additionally, there are other variables that work as keys of incidents but have evidently no relation to the severity of the accident, thus these could be deleted as well. After this filtering, the dataset that will be used for an exploratory analysis contains 10 features (collision type, alcohol/substance influences, weather, road conditions, address type, junction type, lack of attention, light conditions, and speeding, report number) and the target. Nonetheless, this doesn't mean that all of the features will be used to build the model, given that some may be deleted later.

Methodology

When reaching this part, it is important to first deal with duplicate rows and assign columns the correct data type. In fact, in this project, the "report number" column was kept to deal with the first issue, which resulted in 3 deletions. After this, the "report column" was dropped due to the null relation to the target. For assigning the correct data type, it is important to consider that in this specific case, all the features are categorical; however, the columns regarding "lack of attention", "alcohol/substance influences" and "speeding" can be treated as integers given their yes/no nature. In these specific cases, the values of "yes" were set to 1 and the values of "no" to 0.

Next on the list is dealing with missing values in the data set. For the columns regarding "collision type", "address type", and "junction type", missing values where assigned the category "other" or "unknown" given that their count is much smaller than the other categories of their respective features. On the other hand, when the columns "alcohol/substance influence", "weather conditions", "road conditions" and "light conditions" had a missing value for a specific incident, its respective row was dropped from the data set.

At this point, the data set was composed of 189,334 rows and 10 columns (one of them being the target). After looking at the counts of each "Severity" label, it was evident that the data was unbalanced; more specifically, the amount of 1s was more than double the amount of 2s. To fix this, an algorithm was created that would delete random rows that had a "Severity" value of 1 until the data was balanced, i.e. the number of 1s equaled the number of 2s. The new data set was then used to generate bar plots of each feature with the target (this was due to the categorical nature of the data).

From these visual representations, it was notable that for most feature categories, the amount of severe and non-severe accidents were not significantly different. Nevertheless, for the "light conditions", "road conditions", "junction types" and "weather" features, the unknowns clearly favored the target label of 1. This could generate problems when building a model because the unknown values would naturally influence it to select the target label that carried more weight, resulting in erroneous predictions. To fix this problem, it was decided to return to the unbalanced data set, delete the rows corresponding to unknowns for these four features, and then balance the data once again. The reason for continuing this way was based on the idea of loosing the least amount of data as possible.

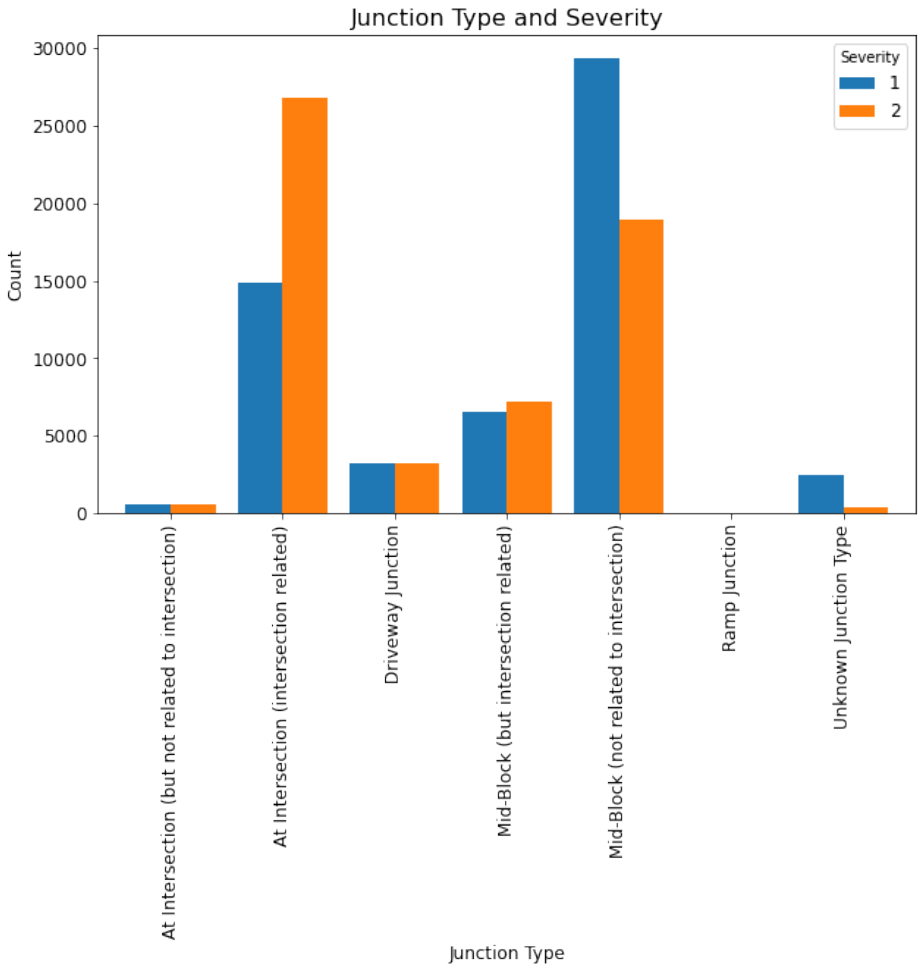


Figure 1: Counts of junction types considering the severity of the accident.

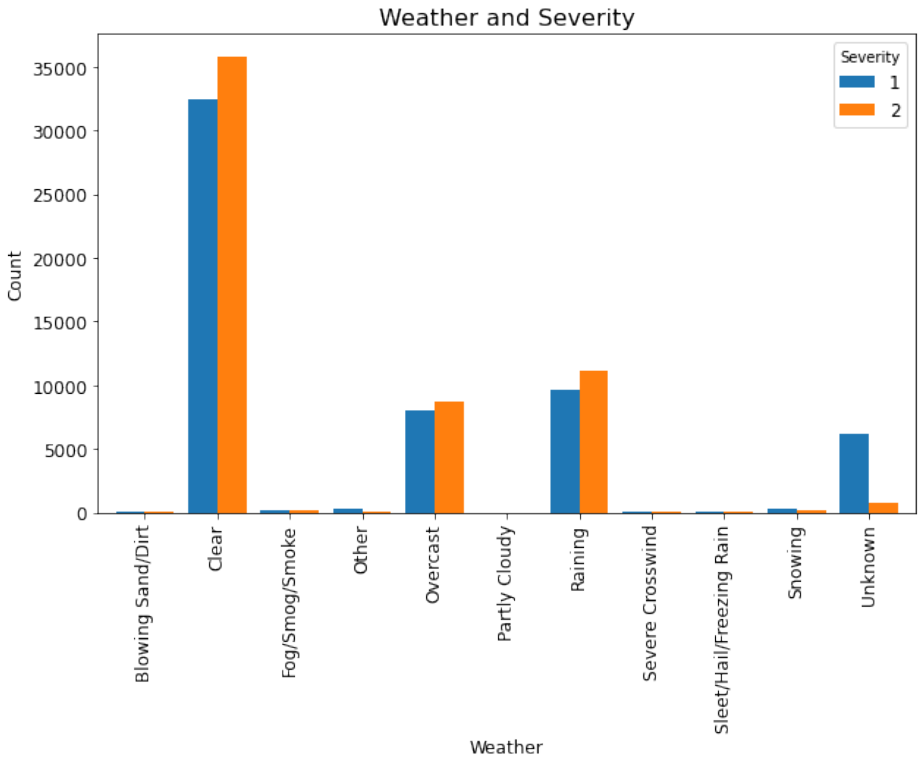


Figure 2: Counts of weather categories considering the severity of the accident.

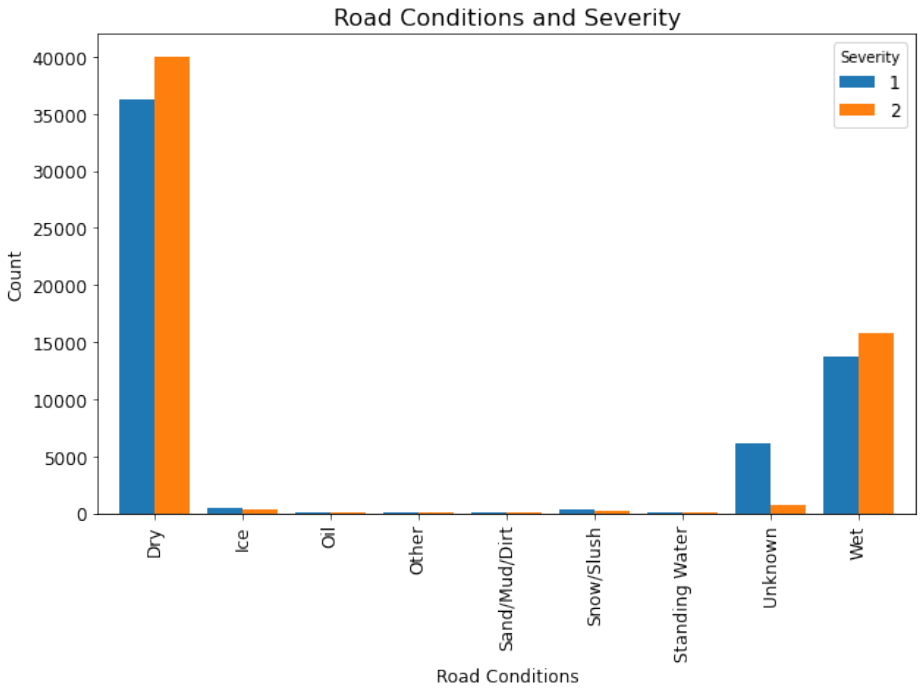


Figure 3: Counts of road condition categories considering the severity of the accident.

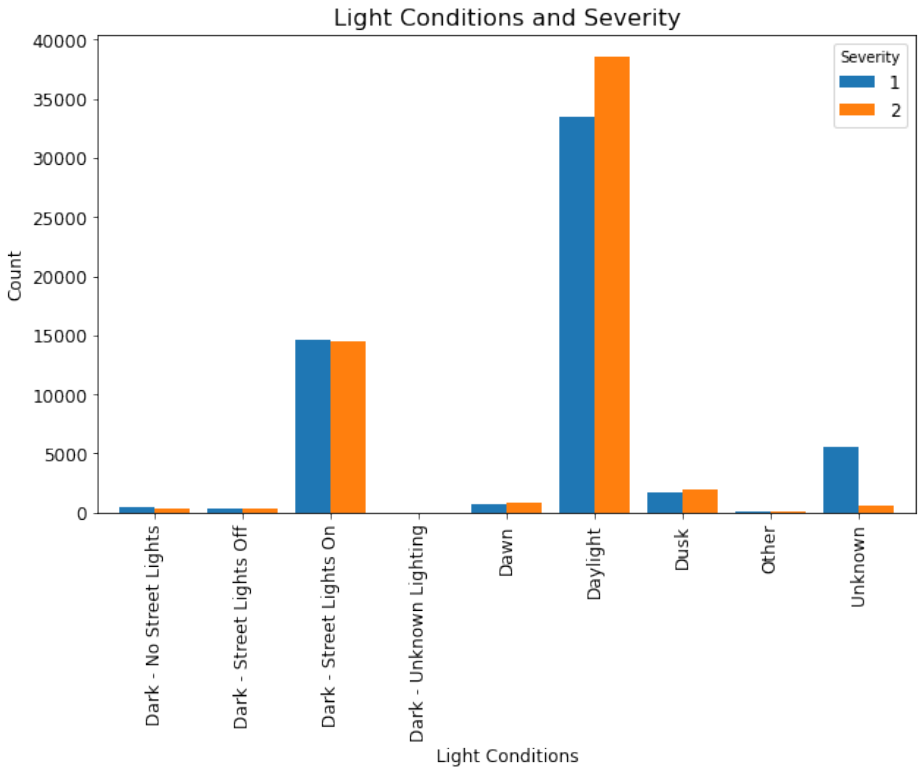


Figure 4: Counts of light condition categories considering the severity of the accident.

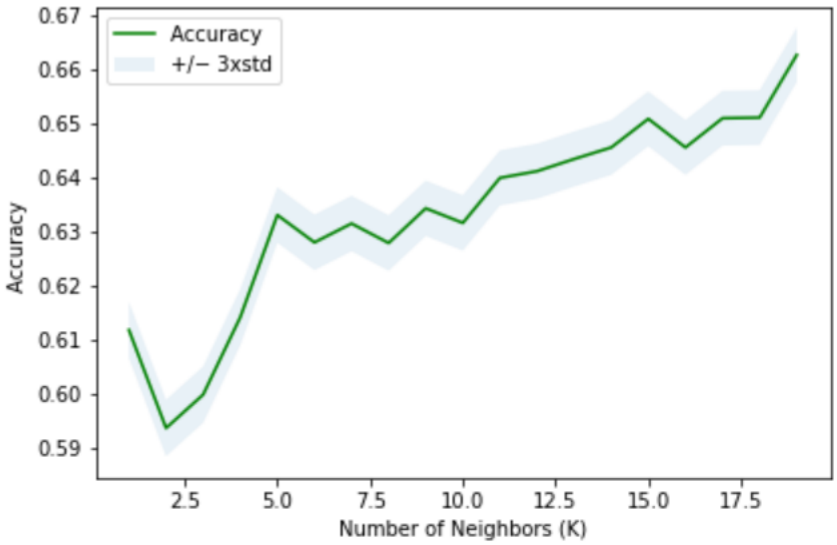
After a new balanced data set was built, it was necessary to extract the dummies in within the features: "address type", "collision type", "junction type", "weather", "road conditions", and "light conditions". This resulted in a data set with 50 columns and 111,026 rows that was ready for the train-test-split. For this, the testing data set was selected to be 20% of the entire collection to have more data to train on.

In this project, it was decided to construct four main classification models to predict the target "Severity". This models were: KNN, Decision Tree, SVM, and Logistic Regression. It is worth mentioning that the parameters within each model were changed as an attempt to optimize the model.

Results

For the KNN model, an algorithm was built to calculate the k values that made the model have the best accuracy; as it can be seen from Fig.5, this happens at $k = 19$. It is worth noting that even though the accuracy grew for greater ks, it was not significant enough to risk overfitting; therefore, a KNN model with $k = 19$ was constructed with the training set. Additionally, Decision Tree, SVM, and Logistic Regression models were constructed to determine the best type. For the Decision Tree models, the parameter "criterion" was changed, for which the "gini" proved to have the best outcome; in the case of the SVM models, "rbf" as "kernel" and 0.1 as "C" (being the inverse of regularization parameter)

showed the optimal results; and for the Logistic Regression models, a "C" of 1 and "solver" of "liblinear" produced the most improved form.




```
The best accuracy was with 0.6625759963972079 with k= 19
```

Figure 5: Accuracy of the model for different ks.

Moreover, for these optimal models, a table with Jaccard and F-1 scores was build by comparing the models' predictions from testing features with the testing targets. Also, for the Logistic Regression, the Log Loss was calculated with the probability of the prediction and the testing targets. All of these results were the following:

Algorithm	Jaccard	F1-score	LogLoss
KNN	0.48	0.67	NA
Decision Tree	0.47	0.67	NA
SVM	0.49	0.68	NA
Logistic Regression	0.46	0.67	0.58

Figure 6: Accuracy scores for the four analyzed models.

Discussion

After analyzing the results of each model with their respective optimal parameters, it can be seen that even though the SVM model with a kernel of RBF and C of 0.1 showed the evaluation metrics with a Jaccard score of 0.49 and F1-score of 0.68, the accuracy scores from all are rather similar. Also, it is worth noting that in general none of the models are

very precise given that their score values are moderately low. All of this could be improved by through different approaches.

The first proposal resides on acquiring more data with a target label of 2, i.e. data of more severe accidents. This would not only make the initial data set more balanced but also larger. Furthermore, one could also try to acquire more information from the incidents themselves like: age of the driver, traffic conditions, type of the car. This is based on the idea that one of these features might cause a greater weight in the classification models and increase their accuracy. On the other hand, if acquiring more data is not viable, one could also explore the "incidents date/time" feature from the initial data set and specifically analyze the time of the incident. Even though many transformations are required to follow this path, some type of significant correlation between "Time" and "Severity" could also improve the classification models and make them more precise.

Conclusion

This project was done with the objective of building a classification model that could predict the severity of a car accident that could be implemented in navigation applications to optimize trips and prevent accidents. To do this, a data set built by Seattle Police Department from 2004 to 2020 was used to extract significant features of these incidents including weather, road conditions, light conditions and others. The data from this new set was cleaned and the missing values were dealt with depending on the type of information. Also some bar plots were constructed to further clean the data before balancing it with respect of the target variable, i.e. the severity of the accident. Additionally, dummy variables were obtained from several features and added to the data that was then divided into a training and testing splits.

Afterwards, the training data was used to build KNN, Decision Tree, SVM, and Logistic Regression models to determine the best one. The SVM model proved to best model of all with a Jaccard score of 0.49 and F1-score of 0.68. However, the accuracy scores of this model were neither significantly different from the others nor large enough say it is precise. That being mentioned, the are some actions that can be taken to fix these issues based on obtaining new data or including more information from the original set. By doing so, a new relation could appear in the form of a correlation that carry enough weight in the models to make them more accurate and, therefore, proceed to an implementation.