Car Accident Severity Report

Sreeja P Sreenivasan

1. Introduction

Every day millions of people are involved in car accidents. If there was an algorithm to predict the severity of an accident, it could enable faster aid to arrive at the scene of the accident. For example, it could help inform police officers what kind of accident and to send the right kind of help.

In this project, we are attempting to predict the severity of vehicle accidents. We utilize the data given by using the number of people and vehicles involved in the accident and what kind of accident had occurred, whether it was only damage to property or damage to passengers as well.

2. Data

The data set used in this project can be found at

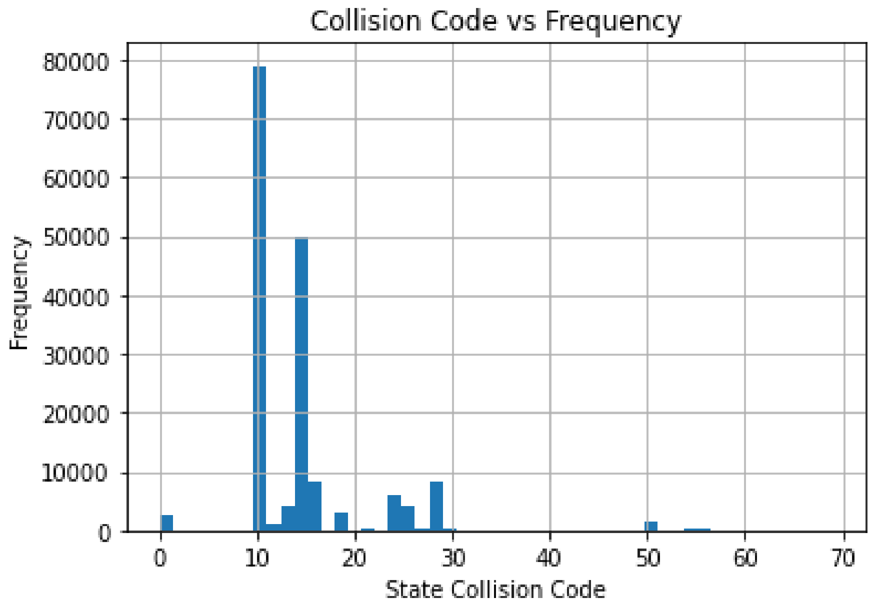
https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP07 01EN/version-2/Data-Collisions.csv.

This data set contains the driving conditions, the number of people and vehicles involved in the crash, and the severity of the crash. There were several problems regarding this data set. Some entries were missing crucial data required for this algorithm. For example, some columns were filled with an "Unknown" in the number of vehicles or persons injured. To solve this, those rows were dropped.

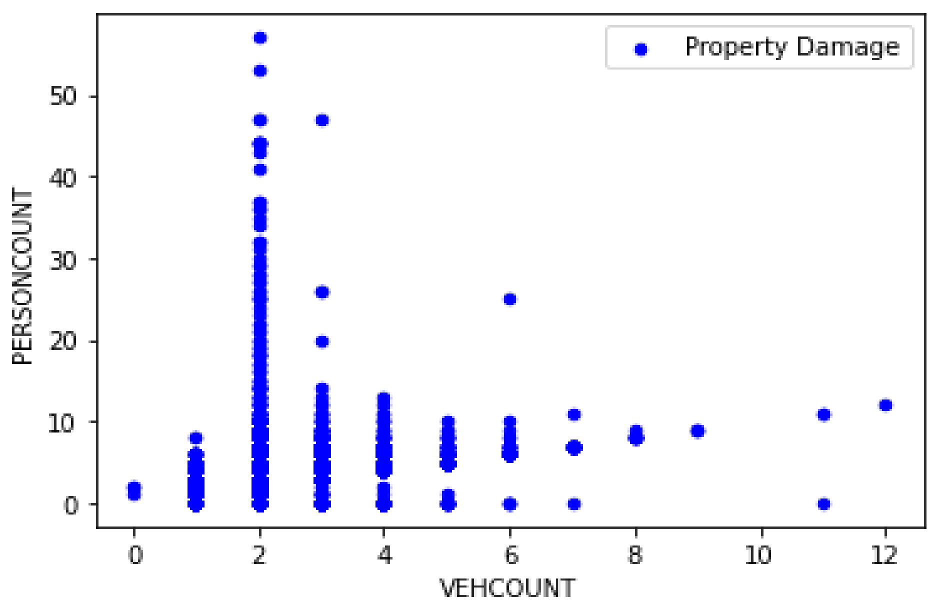
3. Methodology

After cleaning up the data,Three graphs were plotted . One was a simple histogram plotting the severity code against its frequency. This plot gives us the idea that a severity code of 1 was most common among the data set.

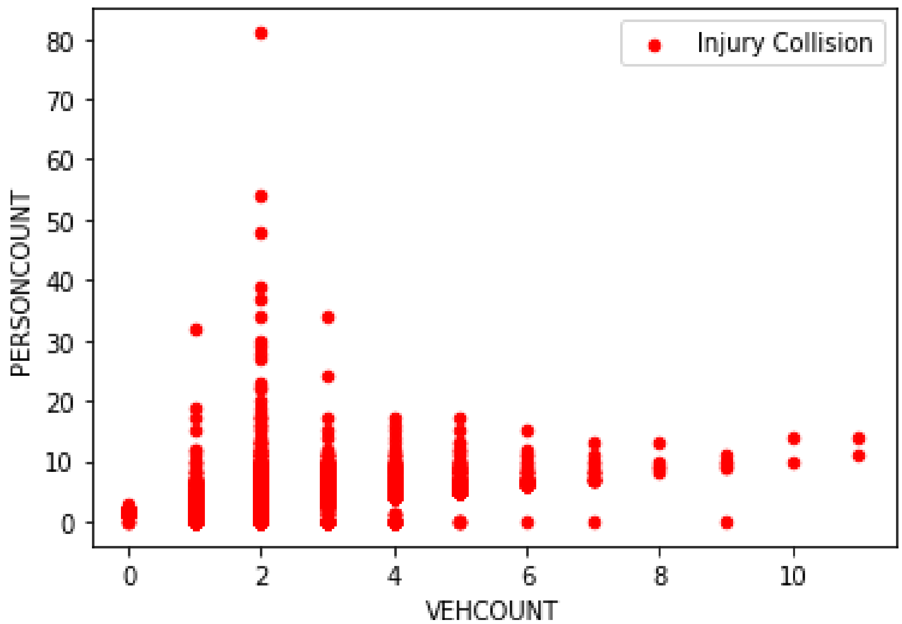
Another plotted graph was the collision code and the frequency. Similarly to the one above, it plots the amount of a specific collision code. However, in contrast to the one above, the collision code is more specific, stating that the collision of code 10 or 11, which is either "entering at an angle" or "both going straight, both moving,sideswipe" was the highest among the rest.



The last two graphs plotted were scatter plots of damage only to property and a collision involving human injuries taking into account the number of people and vehicles involved. Looking below we see that property damage accidents mostly involve two or more vehicles and multiple people.



The last graph below (red) is the same as the one above (blue) except it graphs the collisions that involve injuries.

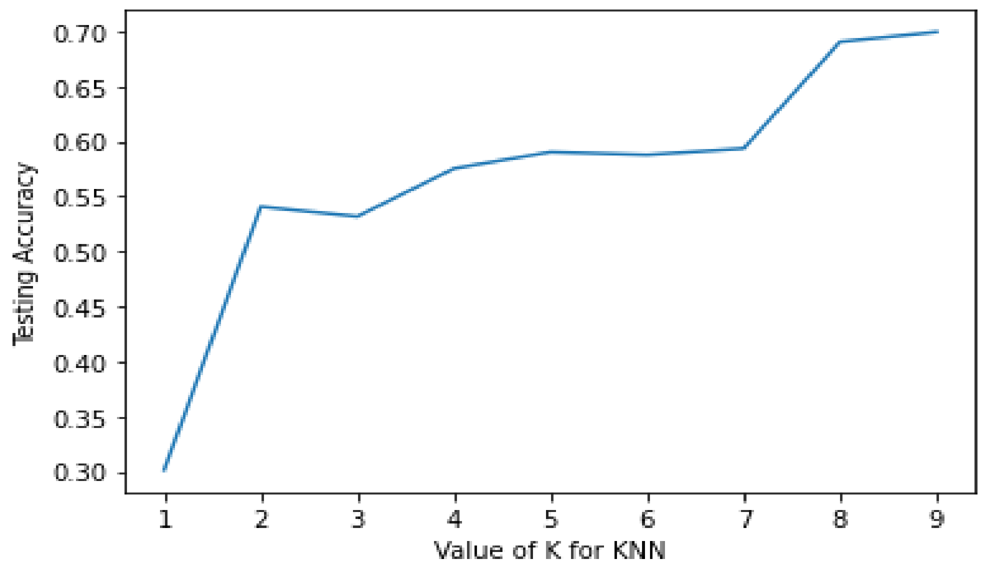


To predict the accident severity, I had implemented both a decision tree and K-nearest neighbor (KNN). However, the plotted decision tree had looked cluttered, and not much information could be taken from it.

Instead, the KNN implementation was much easier and helpful. I had first tested the accuracy for the value of K between 1 and 9 inclusive to see which one would result in a higher accuracy value

Test set Accuracy at k= 1 : 0.30121733966745845

Test set Accuracy at k= 2 : 0.5406360424028268
Test set Accuracy at k= 3 : 0.5316246382802811
Test set Accuracy at k= 4 : 0.5752497371188223
Test set Accuracy at k= 5 : 0.5901874914465581
Test set Accuracy at k= 6 : 0.5876661840200079
Test set Accuracy at k= 7 : 0.5934058393659725
Test set Accuracy at k= 8 : 0.69045830202855
Test set Accuracy at k= 9 : 0.6993428081193953



4. Results

I had found that a K value of 9 resulted in the highest accuracy value at around 0.699. I then predicted the value of y_hat and it had produced 12 correct values out of 20.

```
In [72]: X= df[["VEHCOUNT", "PERSONCOUNT", "SDOT_COLCODE", "SEGLANEKEY"]].values
         y = df["SEVERITYCODE"].values
         print("Actual values of the test cases: " + str(y[0:20]))

         Actual values of the test cases: [2 1 1 1 2 1 1 2 1 2 1 1 1 2 2 2 2 2 1 2]
```

```
In [76]: k = 9
         KNN = KNeighborsClassifier(n_neighbors = k).fit(X_train, y_train)
         y_hat = KNN.predict(X)
         print("Predicted values using k = 9: " + str(y_hat[0:20]))

         Predicted values using k = 9: [1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1]
```

```
In [77]: print("KNN F1-Score: " + str(f1_score(y, y_hat, average = "weighted")))
         print("KNN Jaccard Score: " + str(jaccard_score(y, y_hat)))

         KNN F1-Score: 0.601255208417448
         KNN Jaccard Score: 0.6717197413185398
```

Using the KNN model, I had gotten around 60.125% accuracy for predicting the car accident severity.

5. Discussion

Based on the results, I believe if I had incorporated more variables to predict the target variable, the severity, the accuracy would be higher. Additionally,

this project led to thinking, what if instead of predicting car accident severity after accidents had occurred, what if we had used previous car crashed and the road, weather, lighting conditions, and speeding and other data to predict what is the likelihood one would get into a car crash given those conditions.

6. Conclusion

In this study, I analyzed the relationship between the number of people injured, the number of vehicles damaged, what kind of collision, and the severity of the collision. I built classification models, specifically the KNN model, to predict the severity of a car accident