# Capstone Project: Seattle Collision Data

**Applied Data Science Capstone on Coursera**

**By Jared Best**

## 1 Introduction / Business Problem

The goal of this capstone project is to predict injury collisions (SEVERITYCODE = 2) and property damage collisions (SEVERITYCODE = 1) using classification models and the data set explained in the following section.

This report is relevant to those interested in predicting injury and property damage collisions.

## 2 Data

The data used in the current work is from a data set titled "Collisions—All Years". The data set includes information on all collisions provided by the Seattle Police Department (SPD) and recorded by Traffic Records. This includes all types of collisions. Collisions will display at the intersection or mid-block of a segment.

For further information on the data set, refer to the following table.

**Table 1. Data set summary**

| | |
|---|---|
| **Title** | Collisions—All Years |
| **Abstract** | All collisions provided by SPD and recorded by Traffic Records. |
| **Description** | This includes all types of collisions. Collisions will display at the intersection or mid-block of a segment. |
| **Timeframe** | 2004 to Present |
| **Update Frequency** | Weekly |
| **Keyword(s)** | SDOT, Seattle, Transportation, Accidents, Bicycle, Car, Collisions, Pedestrian, Traffic, Vehicle |

## 2.1 Download links

- The data set can be found via this link.
- Its metadata can be found via this link.

# 3 Methodology

Exploratory data analysis was first performed to gain a deeper understanding of the data set.

The following figure was generated to first compare the raw count values of the severity code types. Note that SEVERITYCODE = 1 corresponds to property damage collision and SEVERITYCODE = 2, injury collision.

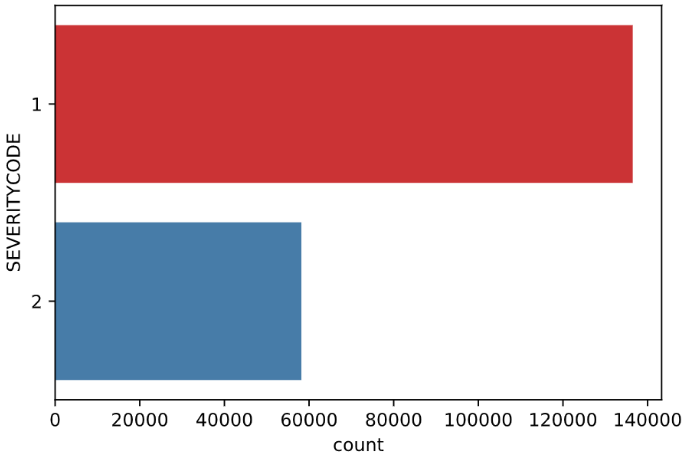


**Figure 1. Collision count by SEVERITYCODE**

Here we observe that number of property damage collisions is more than double that of injury collisions, indicating an unbalanced situation in our data.

We can gain a further understanding of the data set when we group the collisions by the collision address type (ADDRTYPE); refer to the figure below.

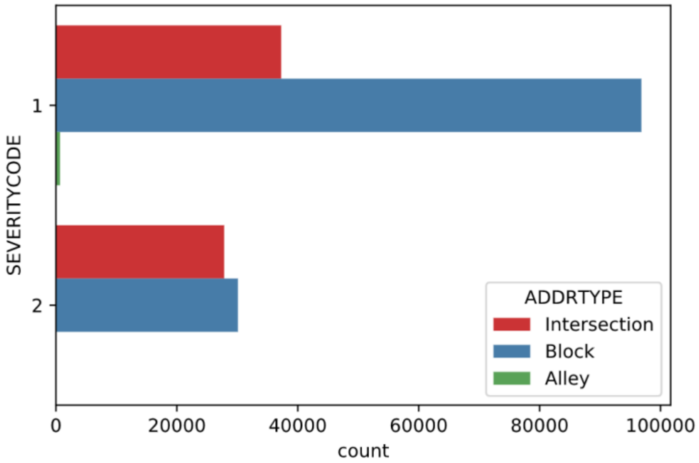


**Figure 2. Collision count by SEVERITYCODE (grouped by ADDRTYPE)**

This figure shows that property damage collisions occur at the highest frequency at mid-block locations.

The following table shows the results of averaging the severity codes for each address type. These results indicate that injury collisions occur most at intersections.

**Table 2. Mean SEVERITYCODE by ADDRTYPE**

| | **SEVERITYCODE** |
|---|---|
| **ADDRTYPE** | |
| Intersection | 1.427524 |
| Block | 1.237115 |
| Alley | 1.109188 |

We can also perform data analysis to see if there are temporal changes; refer to the following figure.

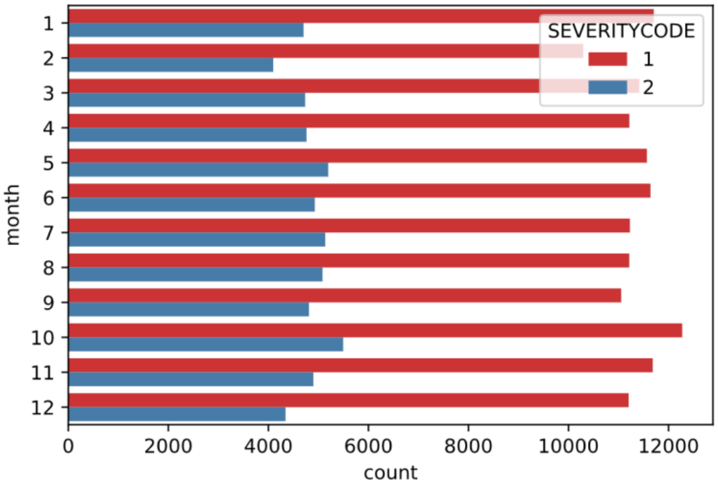
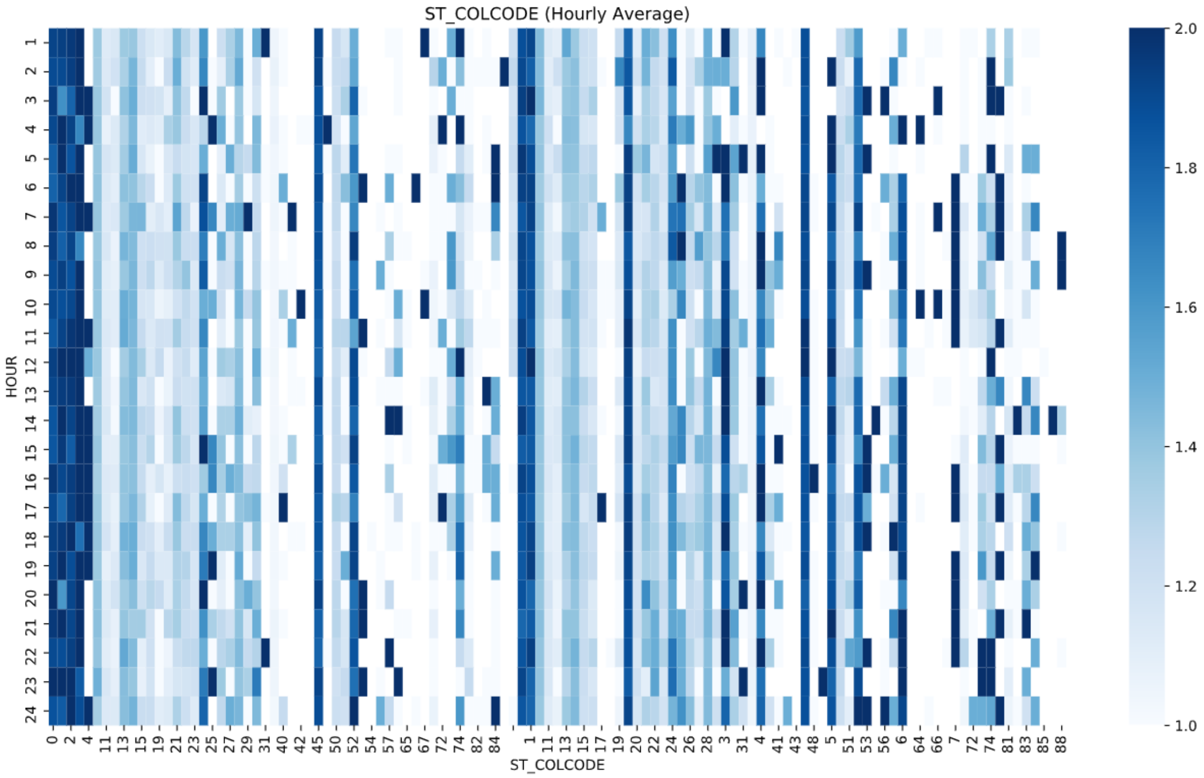
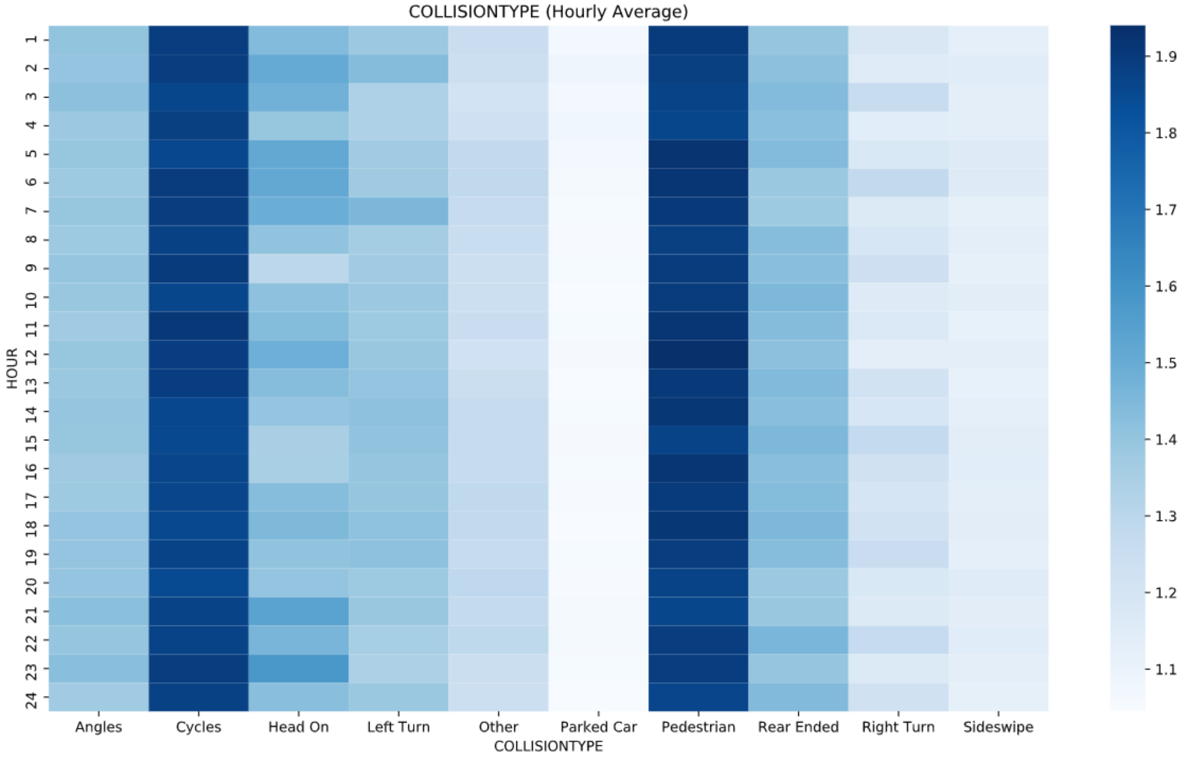


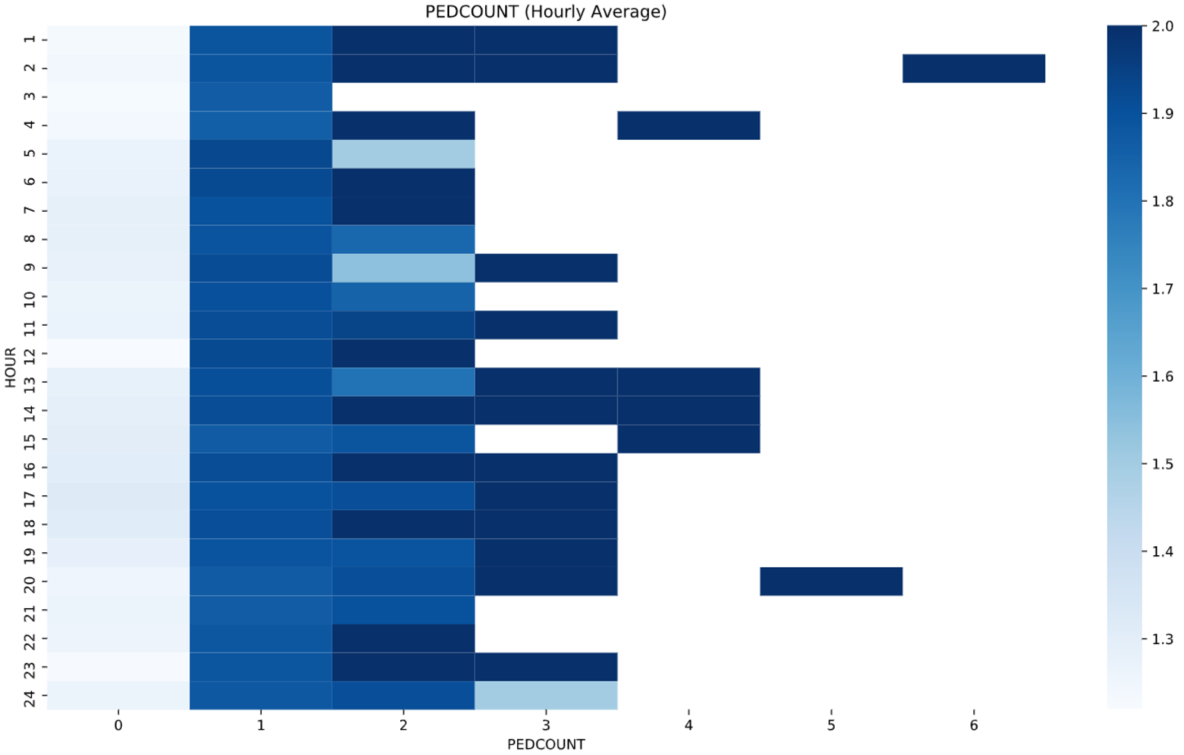
**Figure 3. Collision count by month (grouped by SEVERITYCODE)**

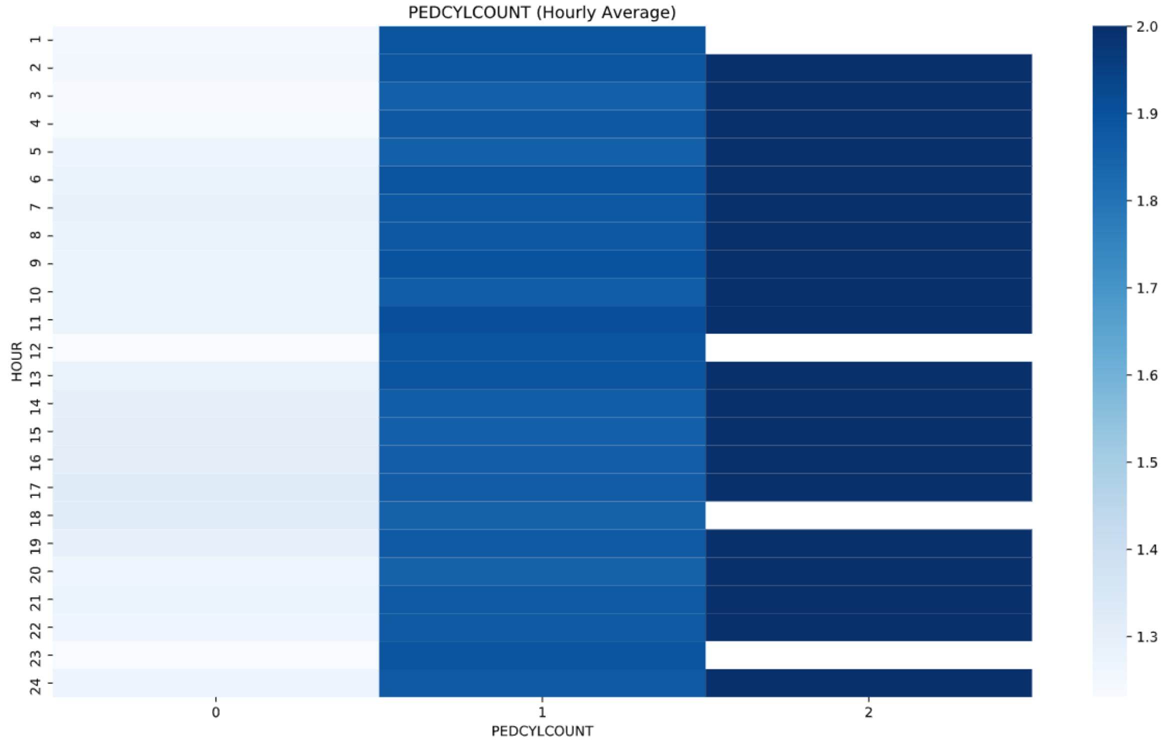
Here we see a relatively uniform distribution of both collision types throughout the year. In other words, no specific month has a substantially higher or lower value for the number of collisions by type.

To better visualize the relationships between data attributes, the following heatmaps were generated. A darker color indicates a higher percentage.

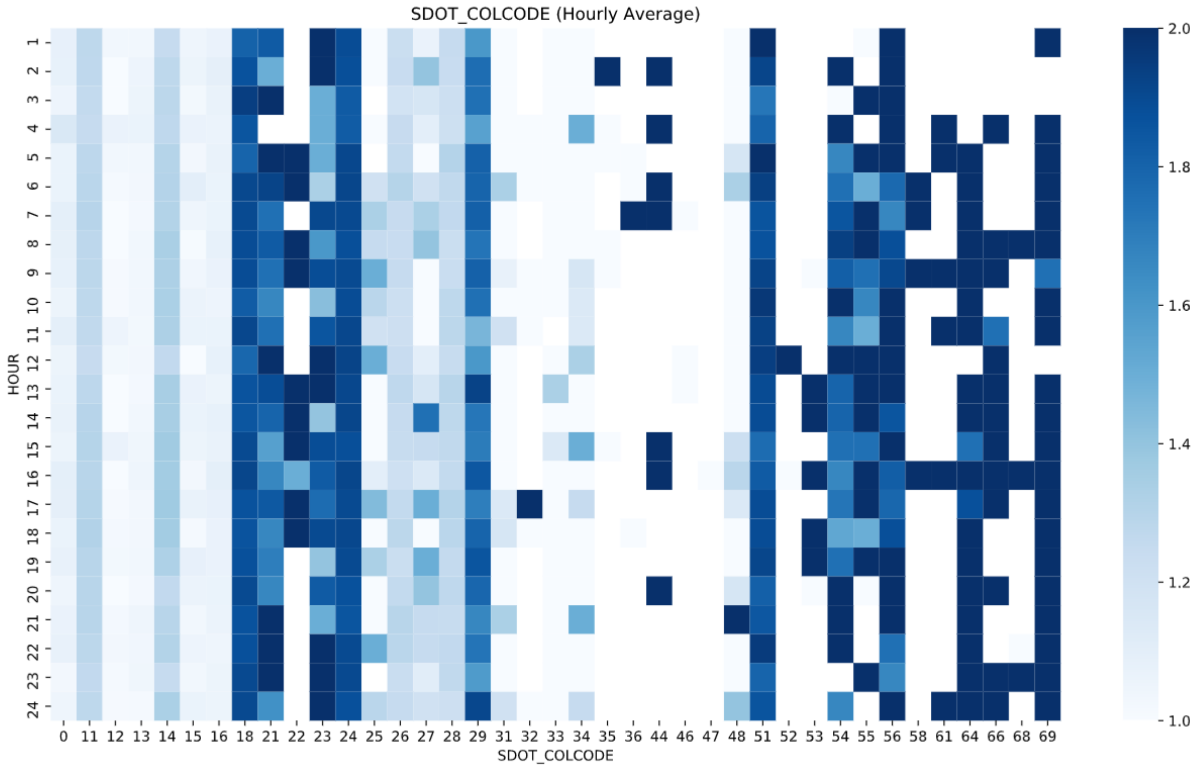



COLLISIONTYPE (Hourly Average)
HOUR
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
Angles
Cycles
Head On
Left Turn
Other
Parked Car
Pedestrian
Rear Ended
Right Turn
Sideswipe
COLLISIONTYPE
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1

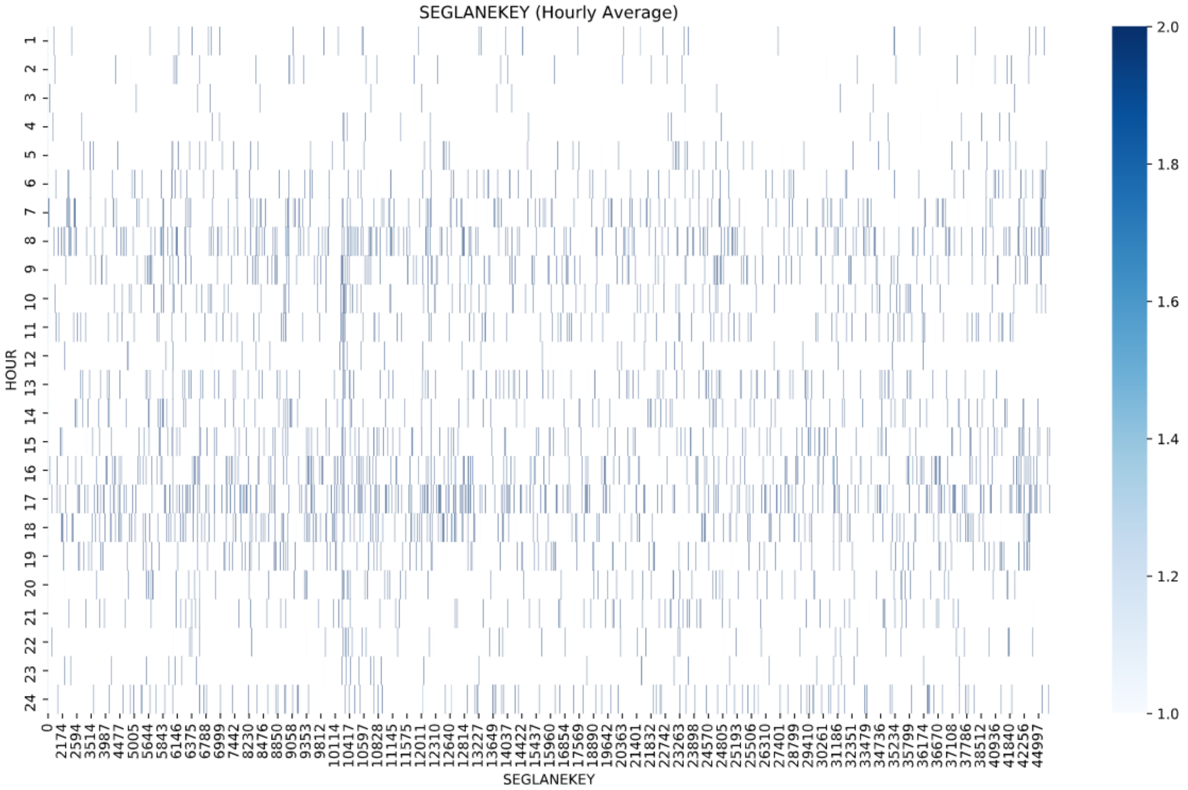

PEDCOUNT (Hourly Average)
HOUR
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
0
1
2
3
4
5
6
PEDCOUNT
2.0
1.9
1.8
1.7
1.6
1.5
1.4
1.3



PEDCYLCOUNT (Hourly Average)



SDOT_COLCODE (Hourly Average)



These heatmaps do provide a visual indication on how these data are structured. For further information on how these were created, refer to the corresponding Jupyter notebook submitted with this report.

# 4 Models

This focus of this project is on predicting injury and property damage collisions; therefore, this is a classification problem.

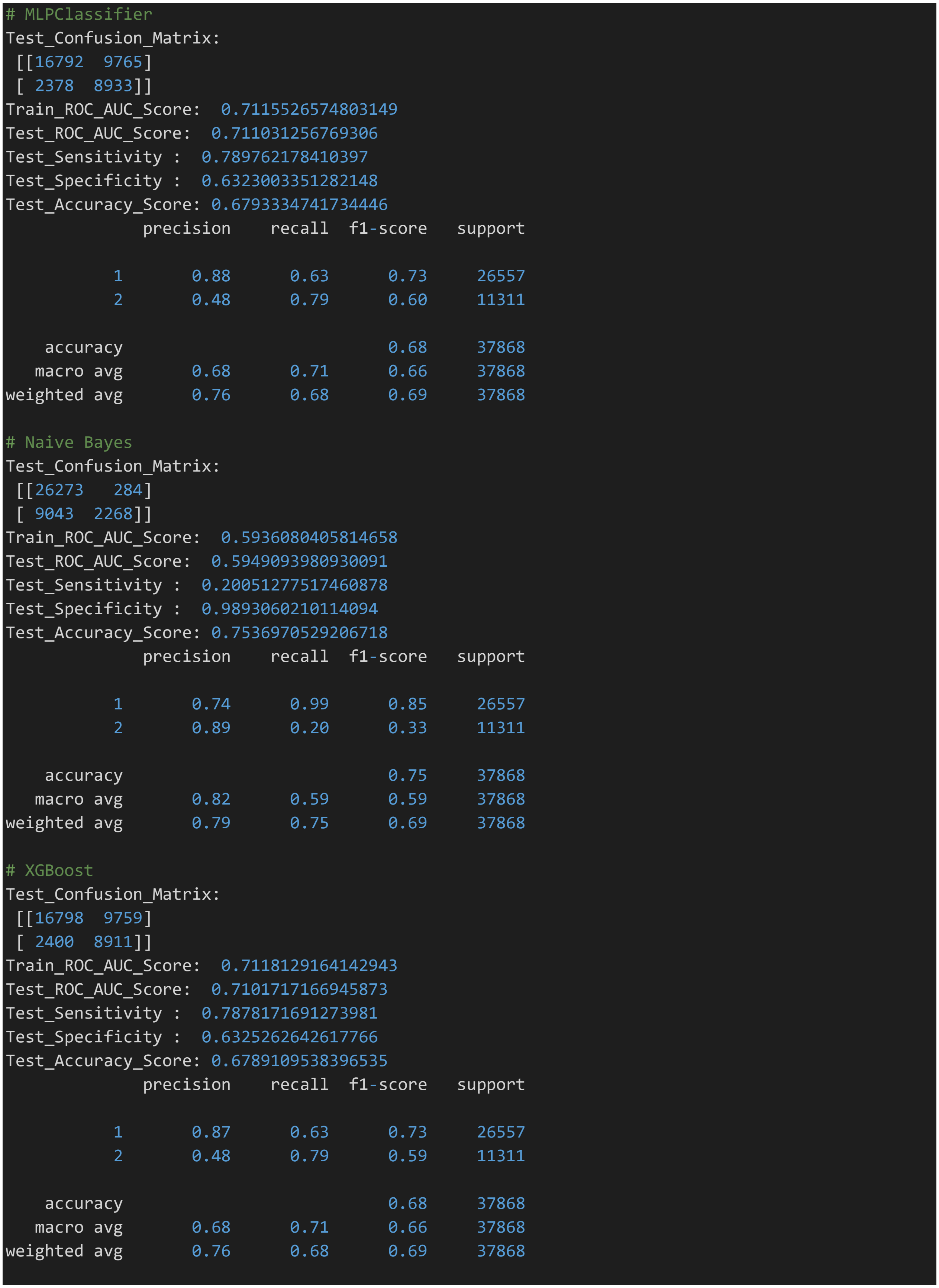
The following models are evaluated:

- MLPClassifier
- Naive Bayes
- XGBoost

These were selected due to their performance and popularity in the literature.

# 5 Results & Discussion

The following is the output from our three models, which shows for each model the confusion matrix, Train_ROC_AUC_Score, Test_ROC_AUC_Score, Test_Sensitivity, Test_Specificity, Test_Accuracy_Score, and the overall accuracy, macro average, and weighted average.



```
# MLPClassifier
Test_Confusion_Matrix:
 [[16792  9765]
 [ 2378  8933]]
Train_ROC_AUC_Score:  0.7115526574803149
Test_ROC_AUC_Score:  0.711031256769306
Test_Sensitivity :  0.789762178410397
Test_Specificity :  0.6323003351282148
Test_Accuracy_Score: 0.6793334741734446
              precision    recall  f1-score   support

           1       0.88      0.63      0.73     26557
           2       0.48      0.79      0.60     11311

    accuracy                           0.68     37868
   macro avg       0.68      0.71      0.66     37868
weighted avg       0.76      0.68      0.69     37868

# Naive Bayes
Test_Confusion_Matrix:
 [[26273   284]
 [ 9043  2268]]
Train_ROC_AUC_Score:  0.5936080405814658
Test_ROC_AUC_Score:  0.5949093980930091
Test_Sensitivity :  0.20051277517460878
Test_Specificity :  0.9893060210114094
Test_Accuracy_Score: 0.7536970529206718
              precision    recall  f1-score   support

           1       0.74      0.99      0.85     26557
           2       0.89      0.20      0.33     11311

    accuracy                           0.75     37868
   macro avg       0.82      0.59      0.59     37868
weighted avg       0.79      0.75      0.69     37868

# XGBoost
Test_Confusion_Matrix:
 [[16798  9759]
 [ 2400  8911]]
Train_ROC_AUC_Score:  0.7118129164142943
Test_ROC_AUC_Score:  0.7101717166945873
Test_Sensitivity :  0.7878171691273981
Test_Specificity :  0.6325262642617766
Test_Accuracy_Score: 0.6789109538396535
              precision    recall  f1-score   support

           1       0.87      0.63      0.73     26557
           2       0.48      0.79      0.59     11311

    accuracy                           0.68     37868
   macro avg       0.68      0.71      0.66     37868
weighted avg       0.76      0.68      0.69     37868
```

AUC, ROC, Sensitivity, Specificity, Precision, and Accuracy are compared for model evaluation. In the three models used in the current study, Naive Bayes has the highest Accuracy of 0.75; however, it has the lowest AUC, ROC score. Furthermore, its sensitivity is also low (at 0.2). Despite its high specificity value, Naive Bayes will not be considered. XGBoost and MLPClassifier both perform well, but as MLPClassifier slightly outperforms XGBoost, MLPClassifier will be selected as a final model.

# 6 Conclusion

The purpose of this project is to predict injury collisions and property damage collisions. The final model selected is the MLPClassifier, which provides AUC, ROC score of .71 and a detection rate of injury collisions equal to 79% and of property damage collisions equal to 63%. Its overall accuracy is 68%. This model will inform relevant stakeholders to predict injury and property damage collisions.