Predicting Car Accident Severity in Seattle, USA

IBM Data Science Capstone

19th September 2020

Ryan W

Table of Contents

1 Business Problem

1.1 Introduction

In the United States, road traffic crashes are a leading cause of death for people aged between 1 – 54 years. Globally, every day, almost 3,700 people are killed in road traffic crashes and are estimated to be the eighth leading cause of death.[1]

The impact of road traffic crashes is felt economically as well for countries, costing countries 3% of their gross domestic product.[2] A study conducted in the United States showed that in 2010 the total economic cost of motor vehicle crashes was $242 billion. These costs were derived from the 32,999 fatalities, 3.9 million non-fatal injuries, and 24 million damaged vehicles. However, this number increases significantly when quality-of-life valuations are considered for those involved in road traffic crashes, increasing the total value of societal harm from road traffic crashes to $836 billion.[3]

With such a large impact on both the lives of those living in the United States, as well as the country's economy, there is a keen interest to develop machine learning models to better predict causes of road traffic crashes to develop preventative measures.

1.2 Intended Audience

The following report will provide insight into various key indicators for road traffic crashes in the Seattle district of the United States. This information will prove a range of insights that can have direct applications for a range of industries such as the Seattle council, insurance and health care.

For the Seattle council, insights will provide information related to areas prone to areas to accidents and their cause effect, which can lead towards the development of interventions in speed management, infrastructure design and enforcement of traffic laws.

For insurance companies, insights will provide information to district areas that have high accident counts for both fatalities and minor crashes such as sideswiping of parked cars. This can lead towards requiring owners pay an increased premium on their car insurance to reduce the risk for the insurance company.

For the health care department, insights will provide information on areas that have high accident counts that require medical assistance and the type of treatment required. This can lead towards improved resource allocation in areas of high road traffic crashes, development of post-crash survival strategies and training for relevant staff.

---

[1] https://www.cdc.gov/injury/features/global-road-safety/index.html
[2] https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries
[3] https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812013

2 Data

2.1 Data Sources

Data that outlines the collisions in the Seattle area from the year 2004 to current can be found here and the metadata can be found here. The data is collected by the SDOT Traffic Management Division and is updated weekly. It includes data of all types of collisions across all types of transportation methods. The dataset, at the time of writing this report, contains 194,673 reported accidents. There are 37 different attributes that describe the accident:

Location Based Attributes (7):

Attribute	Data type, length	Description
SHAPE	Geometry	ESRI geometry field
ADDRTYPE	Text,12	Collision address type: • Alley • Block • Intersection
INTKEY	Double	Key that corresponds to the intersection associated with a collision
LOCATION	Text, 255	Description of the general location of the collision
JUNCTIONTYPE	Text, 300	Category of junction at which collision took place
SEGLANEKEY	Long	A key for the lane segment in which the collision occurred.
CROSSWALKKEY	Long	A key for the crosswalk at which the collision occurred.

Unique Accident Identifier Attributes (5):

Attribute	Data type, length	Description
OBJECTID	ObjectID	ESRI unique identifier
INTKEY	Long	A unique key for the incident
COLDETKEY	Long	Secondary key for the incident
EXCEPTRSNCODE	Text, 10	If the accident has sufficient data associated
EXCEPTRSNDESC	Text, 300	If the accident has sufficient data associated

Accident Details Attributes (22):

Attribute	Data type, length	Description
SEVERITYCODE	Text, 100	A code that corresponds to the severity of the collision: 3–fatality 2b–serious injury

		2–injury 1–prop damage 0–unknown
SEVERITYDESC	Text	A detailed description of the severity of the collision
COLLISIONTYPE	Text, 300	Collision type
PERSONCOUNT	Double	The total number of people involved in the collision
PEDCOUNT	Double	The number of pedestrians involved in the collision. This is entered by the state.
PEDCYLCOUNT	Double	The number of bicycles involved in the collision. This is entered by the state.
VEHCOUNT	Double	The number of vehicles involved in the collision. This is entered by the state.
INJURIES	Double	The number of total injuries in the collision. This is entered by the state.
SERIOUSINJURIES	Double	The number of serious injuries in the collision. This is entered by the state.
FATALITIES	Double	The number of fatalities in the collision. This is entered by the state.
INCDATE	Date	The date of the incident.
INCDTTM	Text, 30	The date and time of the incident.
SDOT_COLCODE	Text, 10	A code given to the collision by SDOT.
SDOT_COLDESC	Text, 300	A description of the collision corresponding to the collision code.
INATTENTIONIND	Text, 1	Whether or not collision was due to inattention. (Y/N)
UNDERINFL	Text, 10	Whether or not a driver involved was under the influence of drugs or alcohol.
PEDROWNOTGRNT	Text, 1	Whether or not the pedestrian right of way was not granted. (Y/N)
SDOTCOLNUM	Text, 10	A number given to the collision by SDOT.
SPEEDING	Text, 1	Whether or not speeding was a factor in the collision. (Y/N)
ST_COLCODE	Text, 10	A code provided by the state that describes the collision. For more information about these codes, please see the State Collision Code Dictionary.
ST_COLDESC	Text, 300	A description that corresponds to the state's coding designation.
HITPARKEDCAR	Text, 1	Whether or not the collision involved hitting a parked car. (Y/N)

Road Condition Attributes (3):

Attribute	Data type, length	Description
WEATHER	Text, 300	A description of the weather conditions during the time of the collision.
ROADCOND	Text, 300	The condition of the road during the collision.
LIGHTCOND	Text, 300	The light conditions during the collision.

3 Methodology

3.1 Data Cleaning

After a preliminary analysis of the data in the excel file, reading the related descriptions and viewing it in the notebook there were three immediate decisions that I made.

The first was that I removed all rows that had the data point of NEI (not enough information) in the column EXCEPTRSNCODE. This was because the data point indicated that the entire row had corrupt data.

```
# From reviewing the data, the columns "EXCEPTRSI
print("Total Rows:", len(df))
df["EXCEPTRSNCODE"].value_counts(dropna=False)
```

```
Total Rows: 194673

]: NaN    109862
     79173
NEI      5638
Name: EXCEPTRSNCODE, dtype: int64
```

```
df = df[df.EXCEPTRSNCODE != 'NEI']
print("Total Rows:", len(df))
df["EXCEPTRSNCODE"].value_counts(dropna=False)
```

```
Total Rows: 189035

]: NaN    109862
     79173
Name: EXCEPTRSNCODE, dtype: int64
```

With this completed, I then removed various columns that had no relevance to the prediction of an accident or were data points from after the accident and so irrelevant in the prediction. These columns were:

EXCEPTRSNCODE
EXCEPTRSNDESC
INTKEY
INCKEY
COLDETKEY
SEGLANEKEY
CROSSWALKKEY
SEVERITYCODE.1
REPORTNO
SEVERITYDESC
SDOTCOLNUM

ST_COLDESC
LOCATION
STATUS
X
Y
INCDATE
INCDTTM
SDOT_COLCODE
SDOT_COLDESC

```
# The accident columns of SEVERITYDESC
df = df.drop(['EXCEPTRSNCODE', 'EXCEPTRSNDESC', "INTKEY",'INCKEY', 'COLDETKEY', 'SEGLANEKEY', 'CROSSWALKKEY',
```

Lastly, I removed the null values from important columns that would assist in the ML models. This was completed by first assessing which columns had null values and how many because in the event that a column was primarily driven by nulls, then the column itself would then be deleted.

```
missing_data = df.isnull()
for column in missing_data.columns.values.tolist():
    print(column)
    print (missing_data[column].value_counts())
    print("")
```

```
SEVERITYCODE
False    189035
Name: SEVERITYCODE, dtype: int64

OBJECTID
False    189035
Name: OBJECTID, dtype: int64

ADDRTYPE
False    189032
True          3
Name: ADDRTYPE, dtype: int64

COLLISIONTYPE
False    184904
True       4131
Name: COLLISIONTYPE, dtype: int64
```

With the column information provided, I was able to deduce that the following columns needed their null values to be removed:

WEATHER
ROADCOND
ROADCOND
UNDERINFL
LIGHTCOND
ST_COLCODE
ADDRTYPE
COLLISIONTYPE
JUNCTIONTYPE

```
df = df.dropna(subset=['WEATHER','ROADCOND', 'ROADCOND', 'UNDERINFL','LIGHTCOND','ST_COLCODE','ADDRTYPE','COLLISIONTYPE','JUNCTIONTYPE'])
missing_data = df.isnull()
for column in missing_data.columns.values.tolist():
    print(column)
    print (missing_data[column].value_counts())
    print("")
```

Now with the unnecessary data points removed, the categorical data points needed to be converted to numerical, so that they could then be run through ML modelling. The first stage of this was identifying that a range of columns had Y or N responses and so converting those to 1 and 0. This was completed by checking what unique data points were in each of these columns to understand what needed to be converted to what.

```
]: df['UNDERINFL'].unique().tolist()
26]: ['N', '0', '1', 'Y']

]: df['PEDROWNOTGRNT'].unique().tolist()
27]: [nan, 'Y']
```

As we can see in the above screenshot, UNDERINFL has 4 different options of unique values and PEDROWNOTGRNT has only 1 and then empty cells.

The next step was to fill all nan values with N. And to then check that there weren't any null values present in the data set.

```
df[['PEDROWNOTGRNT','INATTENTIONIND','SPEEDING']] = df[['PEDROWNOTGRNT','INATTENTIONIND','SPEEDING']].fillna('N')

missing_data = df.isnull()
for column in missing_data.columns.values.tolist():
    print(column)
    print (missing_data[column].value_counts())
    print("")

df.head(10)
```

And lastly, to then convert all categorical values to be numerical, even the "0" and "1" from the UNDERINFL column.

```
]: df['UNDERINFL'].replace(to_replace= ['N', 'Y'], value=[0, 1], inplace = True)
   df['UNDERINFL'].replace(to_replace= ['0', '1'], value=[0, 1], inplace = True)
   df['PEDROWNOTGRNT'].replace(to_replace= ['N', 'Y'], value=[0, 1], inplace = True)
   df['INATTENTIONIND'].replace(to_replace= ['N', 'Y'], value=[0, 1], inplace = True)
   df['SPEEDING'].replace(to_replace= ['N', 'Y'], value=[0, 1], inplace = True)
   df['HITPARKEDCAR'].replace(to_replace= ['N', 'Y'], value=[0, 1], inplace = True)
   df.head()
```

This process was then verified with a check of both null values and also the correlation between each column, which wouldn't be possible if a column contained categorical values.

```
df.isnull().sum()
```

```
SEVERITYCODE        0
OBJECTID            0
ADDRTYPE            0
COLLISIONTYPE       0
PERSONCOUNT         0
PEDCOUNT            0
PEDCYLCOUNT         0
VEHCOUNT            0
JUNCTIONTYPE        0
INATTENTIONIND      0
UNDERINFL           0
WEATHER             0
ROADCOND            0
LIGHTCOND           0
PEDROWNOTGRNT       0
SPEEDING            0
ST_COLCODE          0
HITPARKEDCAR        0
dtype: int64
```

```
df.corr()
```

	SEVERITYCODE
SEVERITYCODE	1.000000
OBJECTID	0.037631
PERSONCOUNT	0.123792

With this all completed, the data is now cleaned and can be processed to have the categorical values transformed to numerical. However, with the data cleaned, various initial graphs were created to get a sense of the impact that each column may have on the severity of the accident. This was to assess if there are any immediate trends, as well as identify if any of the selected columns aren't useful in the ML models.

3.2 Visual Data Exploration

The first exploration I conducted was the junction type versus the number of collisions, categorised with Severity 1 and Severity 2. From the Figure 1 below, it can be observed that the highest number of overall collisions occur at Mid-Block (not related to intersection), however, the highest number of collisions with a Severity 2 occur At Intersection (intersection related).

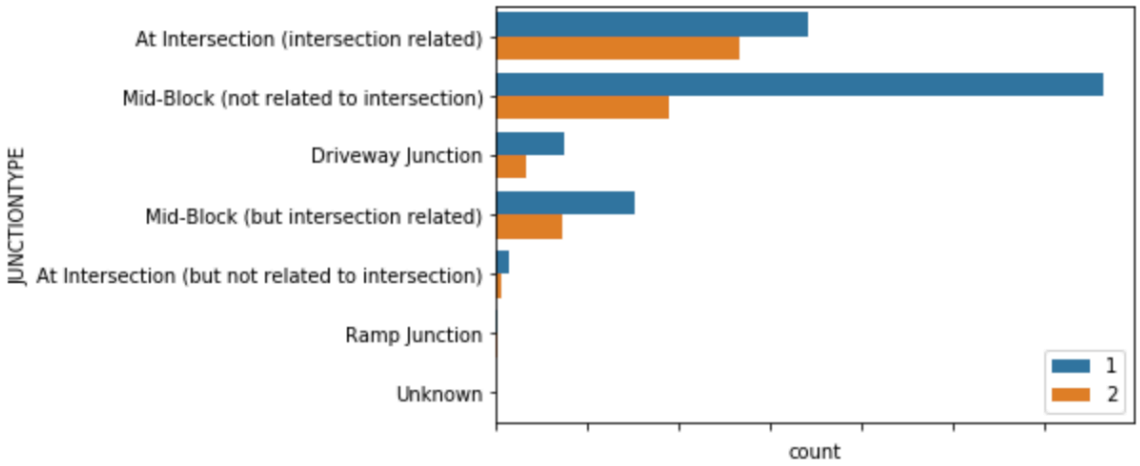


Figure 1: Junction Type versus Number of Collisions, with Severity categorisation

The light condition versus number of collisions, with categorisation can be seen in Figure 2 below. From the figure, it can be observed that the highest number of collisions of both severities occur during daylight. This may be a result of increased driving traffic as well as good conditions make the driver more confident and less focused on their surroundings. Similar results can be seen in Figure 3 that shows the weather versus number of collisions,

with categorisation and Figure 4 that shows the road conditions versus number of collisions, with categorisation.

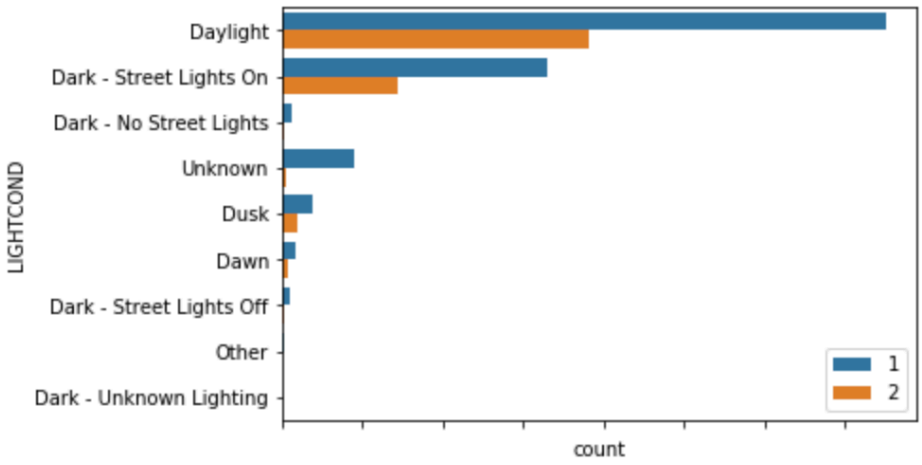


Figure 2: Light condition versus Number of Collisions, with Severity categorisation

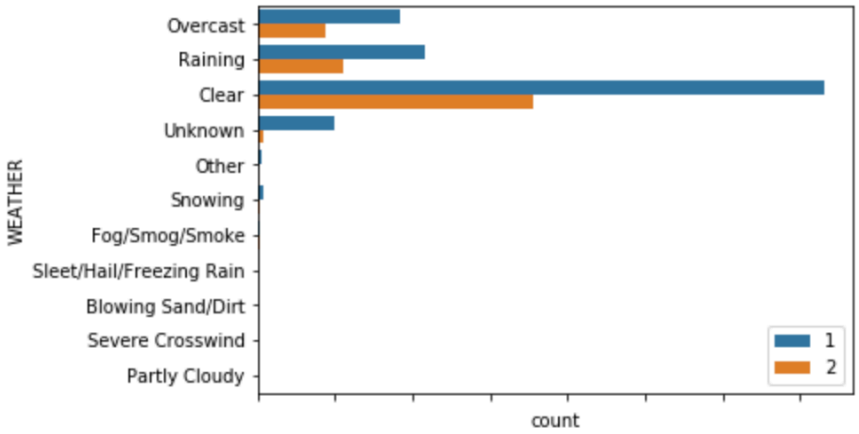


Figure 3: Weather versus Number of Collisions, with Severity categorisation

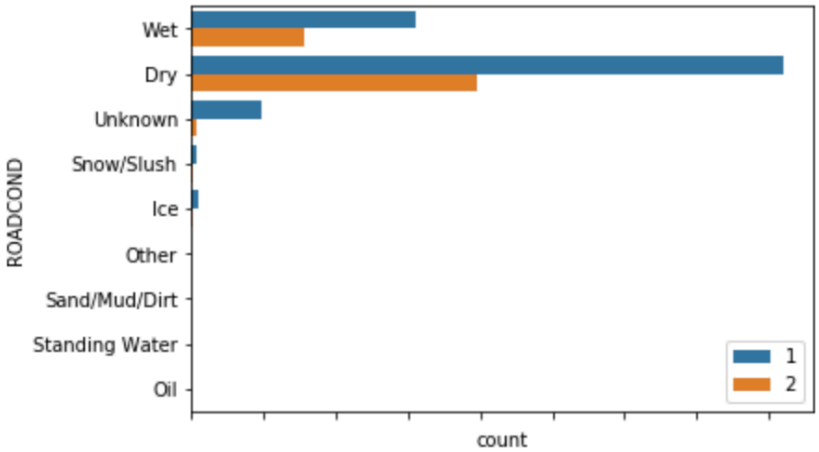


Figure 4: Road condition versus Number of Collisions, with Severity categorisation

Certain effects of the driver were reviewed such as Speeding in Figure 5, Under the Influence or Drugs/Alcohol in Figure 6 and whether the Pedestrian was given right of way in Figure 7. However, they didn't provide any information that was immediately useful.

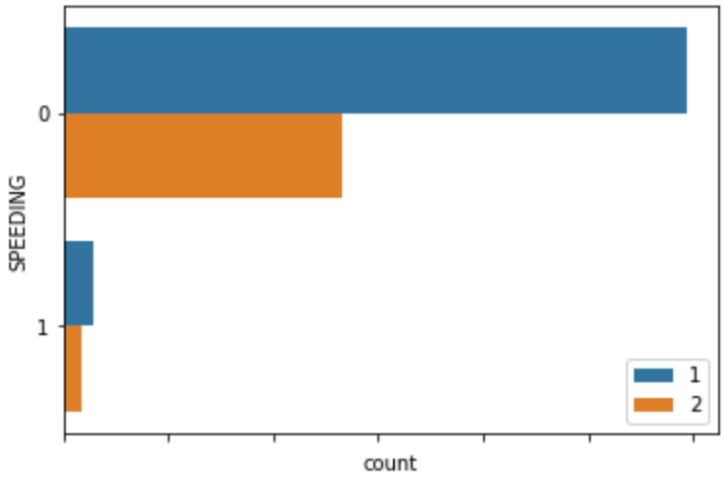


Figure 5: Speeding versus Number of Collisions, with Severity categorisation

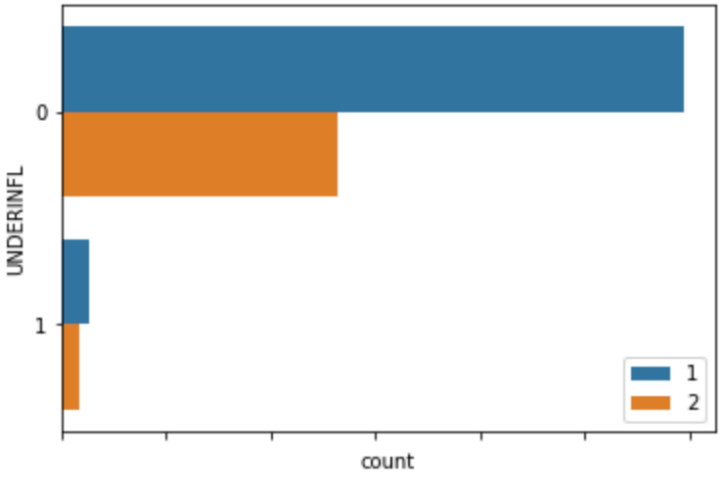


Figure 6: Under the influence of Drugs/Alcohol versus Number of Collisions, with Severity categorisation

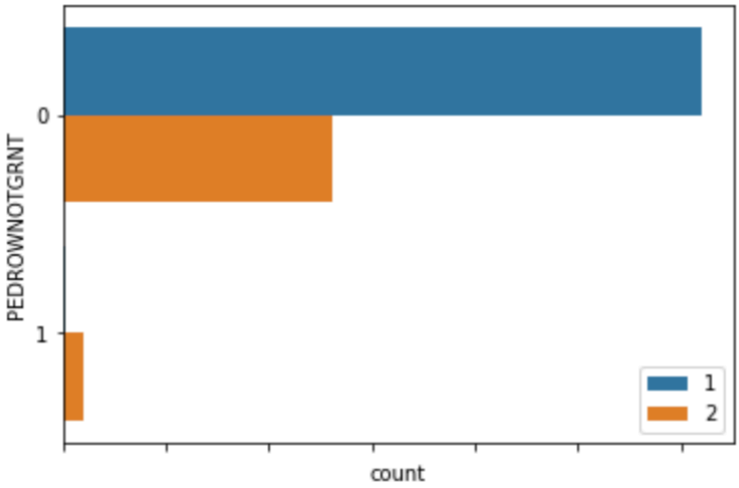


Figure 7: Pedestrian right of way versus Number of Collisions, with Severity categorisation

3.3 One Hot Encoding

All categorical columns that had 3 or more values had to have their values transformed to a numerical type. This was achieved through one hot encoding, by giving each unique value its own subset column and representing the value as a 1 and the other unique values as 0. The columns that received this treatment were:
WEATHER
ADDRTYPE
COLLISIONTYPE
ROADCOND
LIGHTCOND
JUNCTIONTYPE

Before this was completed, unnecessary unique values such as Unknown and Other were identified in the columns and so were removed during the process. This was due to the fact that they would not have any positive effect on the models created.

```
Feature = df[['OBJECTID','PERSONCOUNT','PEDCOUNT','PEDCYLCOUNT','VEHCOUNT','INATTENTIONIND','UI
Feature = pd.concat([Feature,pd.get_dummies(df[['WEATHER','ADDRTYPE','COLLISIONTYPE','ROADCOND
Feature.drop(['LIGHTCOND_Unknown','ROADCOND_Unknown','ROADCOND_Other','WEATHER_Unknown','WEATHI
Feature.head()
```

01:

3.4 Modelling

The Seattle Collisions data set has two specific categories of severity, Severity 1 and Severity 2. The purpose of the models is to predict the likelihood of either severity based on the conditions of the accident. To achieve the best prediction model, I used a range of models and compared them across each other. The models that I used were:

- Random Forest
- Decision Tree
- Logistic Regression

- Gradient Boost
- Naïve Bayles

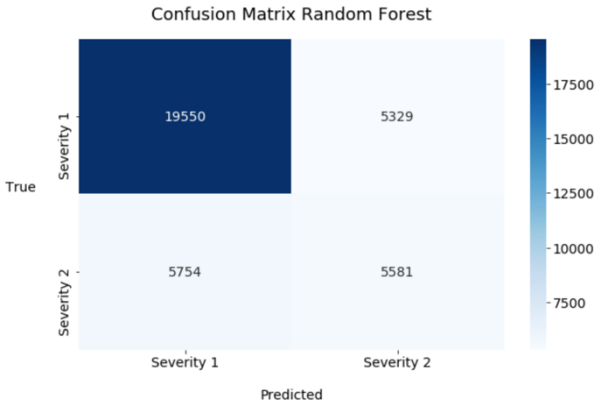
These models were then assessed based on the Jaccard Similarity score and the F1 score, with a visual comparison of a confusion matrix.

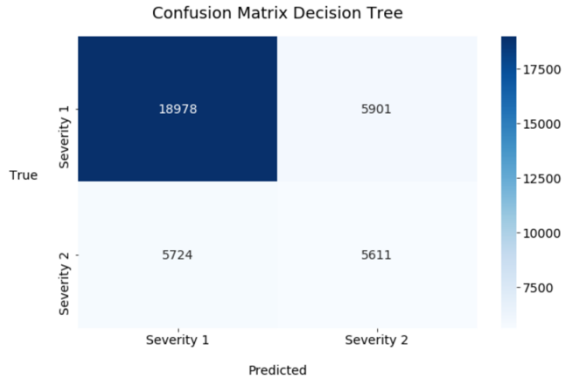
4 Results

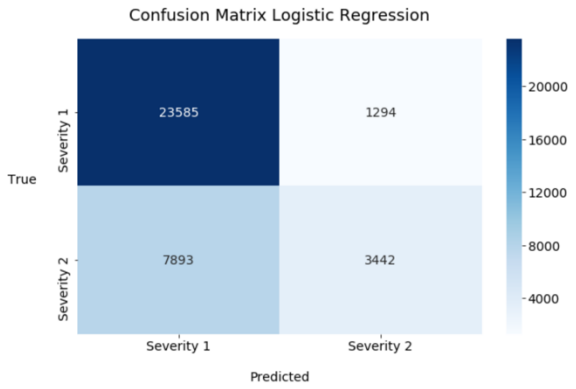
The results of each ML model can be seen in Table 1 as a summary of Jaccard Similarity Score and F1 Score. As well as in Figure 8, as a summary of each confusion matrix. For the logistic regression, an initial test to assess which logistic regression solver produced the highest Log Loss accuracy. It resulted in the SAGA model.

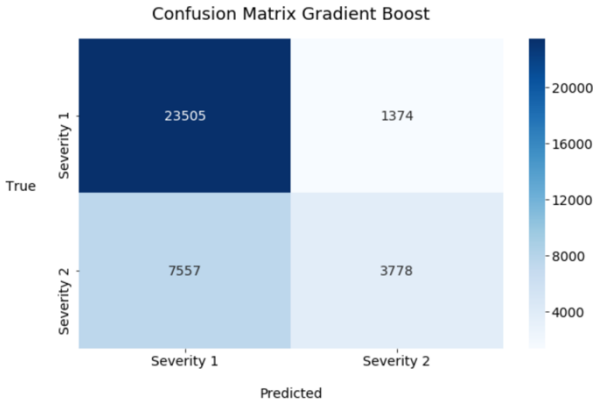
Table 1: Summary Results for Each Model

	Jaccard Similarity Score	**F1 Score**
Random Forest	0.694	0.692
Decision Tree	0.679	0.680
Logistic Regression	0.746	0.709
Gradient Boost	0.753	0.721
Naïve Bayes	0.739	0.691









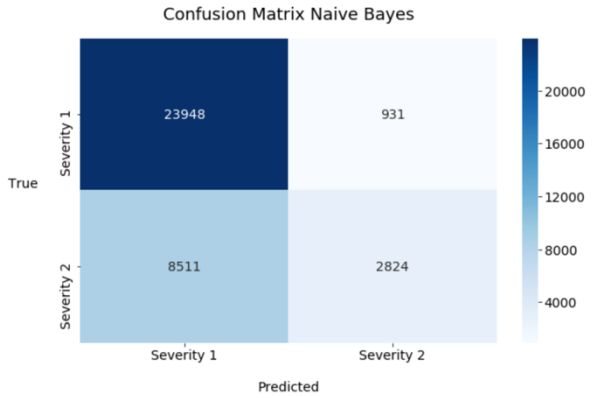


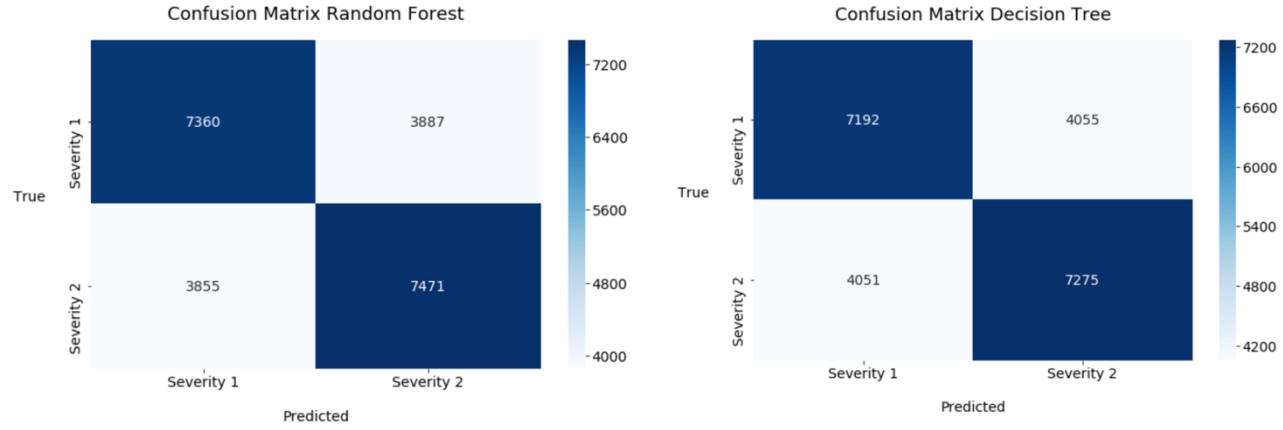
Figure 8: Collection of Confusion Matrices for the ML Models

As seen in Table 1, from the Jaccard Similarity and F1 Scores, it can be deduced that the Gradient Boost ML Model is the best predictor for the severity of an accident. However, after the completion of each ML model, it was observed that the models struggled to accurately predict Severity 2 accidents. This is evident from the confusion matrices in Figure 8, given the significant proportion of cases found in Severity 1 (and resulting accuracy), however, a severe lack in Severity 2 (and resulting incorrect predictions).

This was investigated and deduced that due to the significantly higher number of Severity 1 accidents, there was an unbalance in the data. As such, to resolve the issue, under sampling was undertaken to drop the number of entries of the dominant category (Severity 1) until both categories had that same number of samples. The new results of each ML model can be seen in Table 2 as a summary of Jaccard Similarity Score and F1 Score. As well as in Figure 9, as a summary of each confusion matrix.

Table 2: Summary Results for Each Model with Under Sampling

	Jaccard Similarity Score	**F1 Score**
Random Forest	0.657	0.657
Decision Tree	0.641	0.641
Logistic Regression	0.701	0.698
Gradient Boost	0.716	0.714
Naïve Bayes	0.632	0.606



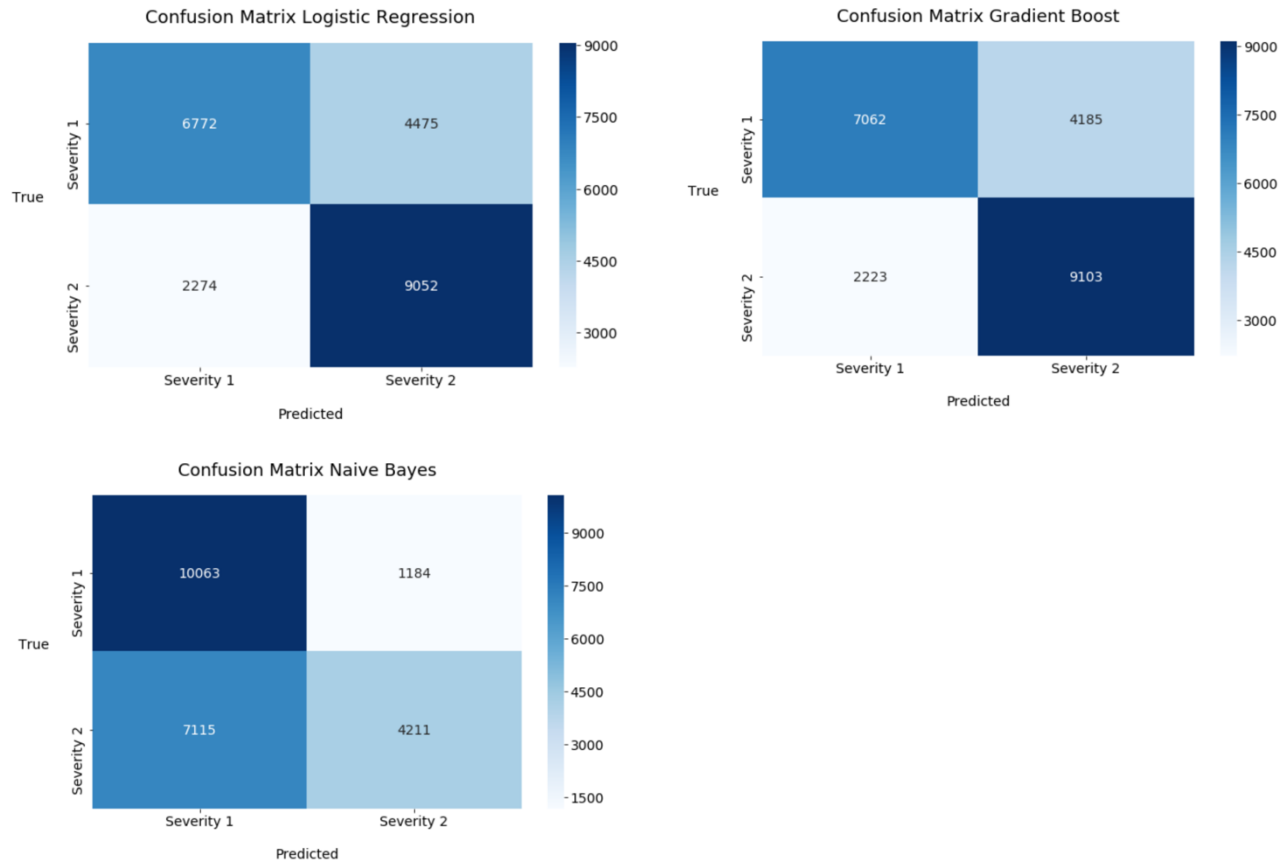


Figure 9: Collection of Confusion Matrices for the ML Models with Under Sampling

5 Discussion

After applying the under-sampling, the models performed better in making unbiased classification except for the Naïve Bayles model. While there were reductions across each model in Jaccard Similarity and F1 scores, as seen in Table 2, their performance in classifying Severity 2 increased dramatically, as seen in Figure 9.

It has been observed that the ideal model to predict the Severity of the accident is the Gradient Boost model, with a Jaccard Similarity Score of 0.716 and F1 Score of 0.714.

While the under sampling proved effective, it should be noted that a deeper consideration towards the connection between accidents in Seattle areas and any relevant supporting projects from the council may be show interesting connections to reduce the severity of accidents.

6 Conclusion

In this study, I analyzed the relationship between a range of location based, road conditions and accident details to the resulting severity of the accident. From the models built, the ML Model of Gradient Boost produced the best results with a Jaccard Similarity Score of 0.716 and F1 Score of 0.714. This model can be useful for the Seattle council, as

they will be able to better understand the influence that variables can have on the severity of an accident and use that information to develop of interventions in speed management, infrastructure design and enforcement of traffic laws.

For insurance companies, this can be useful to improve the questionnaires asked of car owners submitting for insurance premiums and how high they should be. For example, if they drive significantly during the day and in areas that have high intersection accidents, they would look to increase the premium to cover the business.