Car Accident Severity Report

Asvin Sritharan

October 8, 2020

Introduction

A GPS navigation system provides directions to thousands of users daily. The goal of a navigation system is to be able to get a user from point A to point B in the most efficient way possible. Dealing with accidents puts a great strain on this goal. Accidents are commonplace in every city, creating complication for drivers and navigation systems alike. Thus, it is beneficial for the system to be able to detect the severity of an accident to determine if there are more efficient ways to route a user in a nonconventional way.

Problem

This project seeks to build a model which can successfully predict the severity of an accident.

Interest

The interested party in this problem would be a navigation company. A navigation company can use this model to determine whether it would be more efficient to let the driver pass through traffic or avoid it. Furthermore, it could be a safety measure as well, deterring drivers from driving towards sites of heavy collision. It would be beneficial for both the driver and the GPS company.

Data

Based on our description of the problem it is important that we get relevant accident information to build our model. Factors which we would need would include the location of the accident, the date, the weather, whether intoxication was involved and more.

To obtain this information, a car accident dataset from data collected in Seattle will be used. This data was collected as part of the IBM Capstone Project Course which provided us with the dataset.

The dataset contains columns: 'SEVERITYCODE', 'X', 'Y', 'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO','STATUS', 'ADDRTYPE', 'INTKEY', 'LOCATION', 'EXCEPTRSNCODE','EXCEPTRSNDESC', 'SEVERITYCODE.1', 'SEVERITYDESC', 'COLLISIONTYPE','PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'INCDATE','INCDTTM', 'JUNCTIONTYPE', 'SDOT_COLCODE', 'SDOT_COLDESC','INATTENTIONIND', 'UNDERINFL', 'WEATHER', 'ROADCOND', 'LIGHTCOND','PEDROWNOTGRNT', 'SDOTCOLNUM', 'SPEEDING', 'ST_COLCODE', 'ST_COLDESC','SEGLANEKEY', 'CROSSWALKKEY' and 'HITPARKEDCAR'. We must first drop columns which are redundant or appear in other forms. An example of such columns is the SEVERITYCODE and SEVERITYCODE.1 column. They hold the same values, so it is meaningless to include both columns in our dataset.

After the relevant variables are gathered, we then plot the variables against the SEVERITYCODE labels to determine if they have any influence on whether an accident is an injury causing accident or a property

damage causing. Dropping the irrelevant variables here, give us a more rigid set of variables which we can then use for training our model.

The data then needs its categorical variables factorized so they can be used in models. Once it is all done, the data will be normalized so variables do not have more influence over others in the model. Finally, the Decision Trees, KNN, SVM and Logistic Regression models will be created using the data and the test data will be made using 10% of the dataset. The model with the best scores across precision, recall and f1-scores will be the model selected as the best model.

Methodology

Our goal of this project is to successfully predict accidents by their true severity. To do this, we must have many samples for accident data. The dataset selected, contains data from Seattle which was provided by the IBM Applied Data Science Capstone Course.

We will plot categorical variables against the severity code categorical variable by using bar charts to compare shapes to see if there are differences in distribution. Side by side boxplots will be used to plot quantitative variables against the severity categorical variable to see differences in distribution of data. Variables which are variations of other variables, have no variation in fraudulent transactions or are insignificant will be excluded from the model.

After the unnecessary variables are dropped from our dataset, we factorize the categorical variables to give categories a numerical representation so that they can be modelled on. Normalization will take place so variables like state collision code, which have large number values, do not have higher impact on our model than other variables like category of purchase.

Finally, since this is a classification problem, we will build a model using decision trees, KNN, SVM and logistic regression for fraud detection. Each model will then be tested by calculating its precision, recall and f1-scores. We will not be focusing on precision or recall by itself; we will be considering models which have a high precision and recall score for greatest accuracy in our model. Models based around fraud detection may rely heavily on recall scores but since our goal is to detect the severity of an accident for consumer purposes, we will be focusing on all rounded model which excels in precision, recall and f1-score.

Data Cleaning

To begin data cleaning, variables such as identifier variables, duplicate variables, and variables with large amounts of missing data were dropped. This saw the removal of variables ObjectID, CollisionType, Inckey, SeverityCode.1, ColDetKey, ReportNo, ExceptRSNCode, ExceptRSNDESC, IncDate, SDOT_ColDesc, PedRowNotGrnt, and ST_ColDesc, which were variables that were unique identifier variables and duplicate variables for information available elsewhere in the dataset. Variables IntKey, InAttentionInd, SDOTColNum and Speeding were all columns which were dropped because they contained a large amount of null values.

We were then left with data that was not as complete as we would have liked. There were still columns which had missing values. LightCond, RoadCond, Weather, UnderInfl, JunctionType, and AddrType had null values replaced with their respective mode values. We were left with X, Y and Location containing null values.

X and Y had missing data that we attempted to replace by matching it's location to other X and Y values in the table, however, we were only able to gain 4 data points by doing this leaving 5330 observations without X and Y values. It was then, we determined to drop X and Y considering we can use location as a variable to determine the approximate location of an accident. This was likely for the better as a more generalized, yet still accurate location was better and reporting whether an area was prone to accidents compared to X and Y values which represent the longitude and latitude of an accident. We had no null values besides those in 2677 observations in the location variable. We decided to drop these rows as we had a large amount of observations for car accident data.

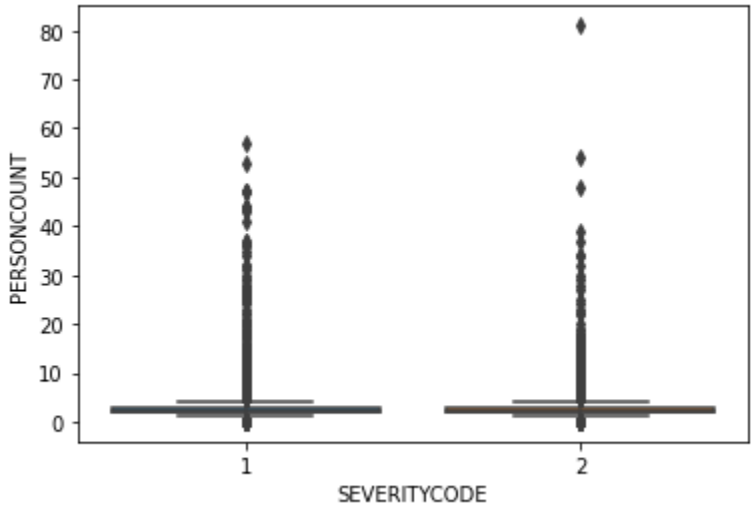
We then converted objects in the dataset which were marked as integers into their intended category: object. This was done to SeverityCode, SDOT_ColCode, SegLaneKey, and CrossWalkKey.

A final peek into each variable's unique values told us there were values which were labelled as 'Unknown' and the date variable 'INCDTTM' had values which contained the time and values which were missing the time. To resolve the data variable, we removed the time data from the date variable since we were unable to obtain specific time information regarding the accident. The Unknown values were found in variables JunctionType, Weather, RoadCond, and LightCond. JunctionType was resolved by replacing Unknown with the mode of the variable. Weather, RoadCond and LightCond were resolved by replacing values with the mode of values which were not unknown in the same day as the unknown values. The data was complete and data analyzation could begin.

Data Analyzation

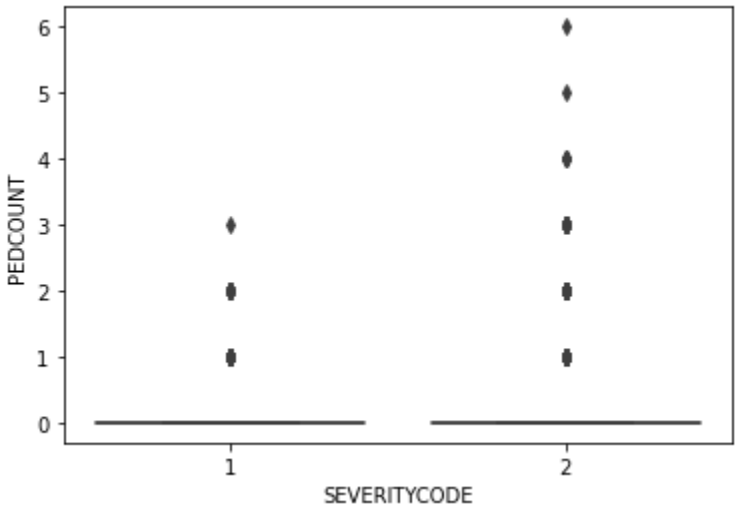
Each remaining variable was then plotted against the label variable: SeverityCode. The primary focus of each graph is to see if there is a difference in the variable between different severity codes which would allow us to include the variable in our model.

PersonCount



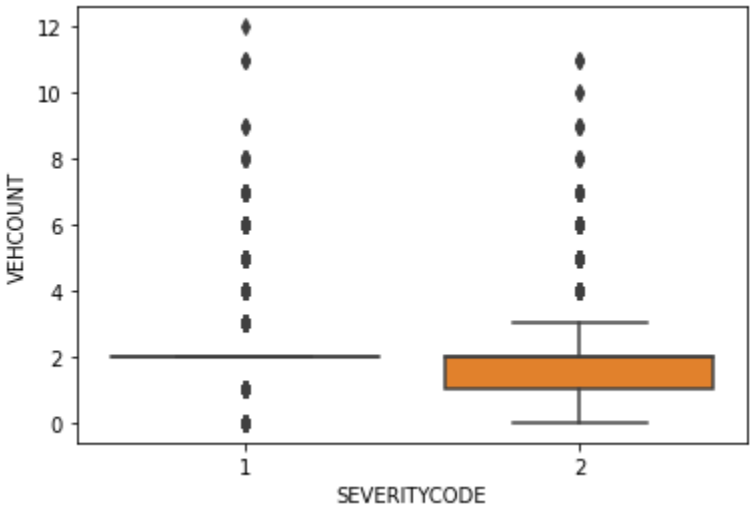
The person count variable has a different number of outliers in severitycode=2, so this can help push a model to labelling an accident with very high person counts as accidents with injury. This variable was kept in our model.

PedCount



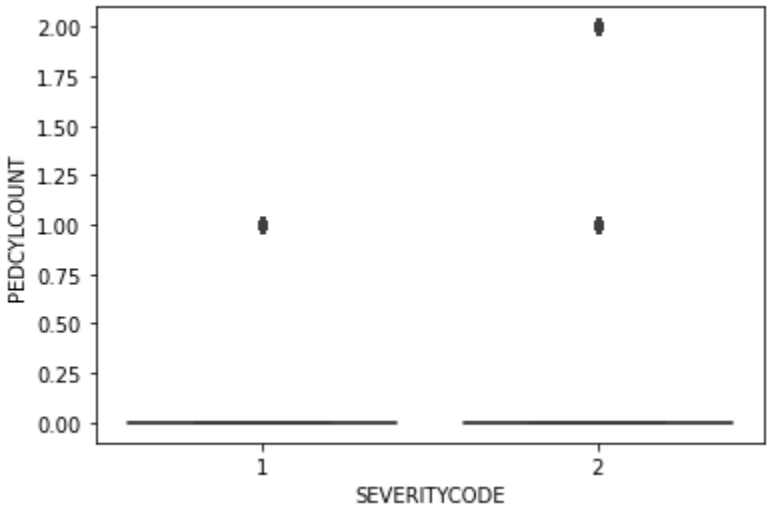
The outliers are different in this variable as well, so this variable was kept in our model. Having high pedestrian counts can push our model to label an accident as an accident with injury.

VehCount



There is much higher variance in vehicle count for the more serious accidents with the boxplot also showing max and min values more spread out than the low severity case. This variable was kept in our model because of this.

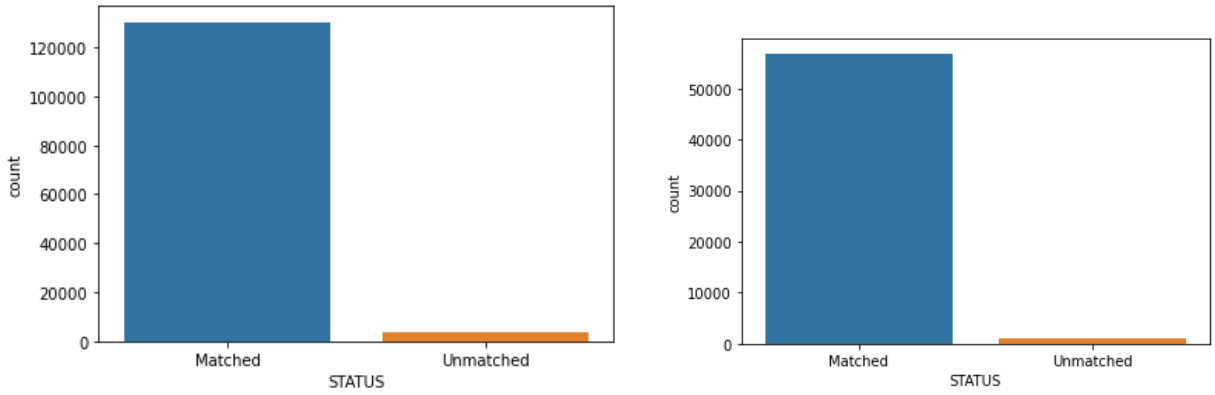
PedCycleCount



The outliers with 2 pedestrian cyclists involved in accidents have been injury related accidents so this will help push the model to make these values lean towards the side of injury related accidents. This variable will be kept in our model.

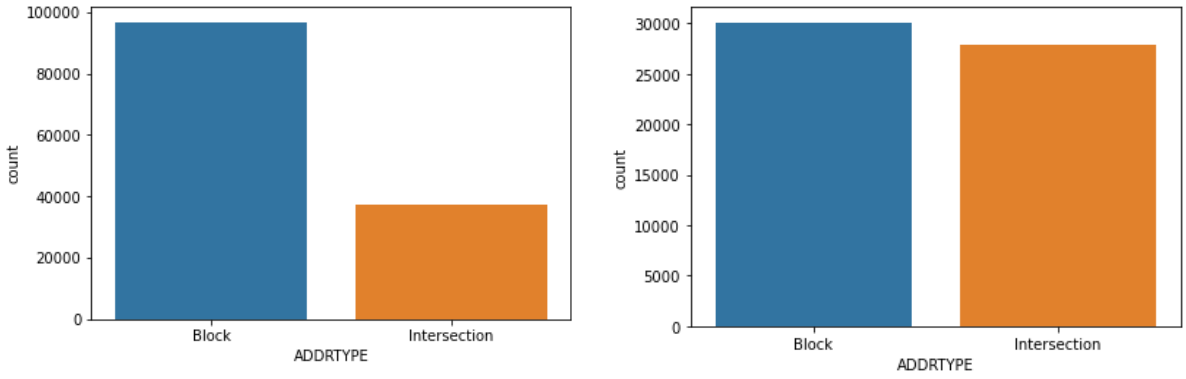
This concludes the boxplot analyses, and the following graphs are categorical variables which will be displayed in box plots. The left chart represents the variable plotted against SeverityCode=1, the right chart represents the variable plotted against SeverityCode=2.

Status



The graphs show a lot of similarity and not much difference comparing severity codes so this variable will be left out of the model.

AddrType



There are a lot more accidents which are severe which occur in intersections. In non severe accidents we see a lower bar for intersection. There is a clear difference between both groups so we will include this variable in our model.

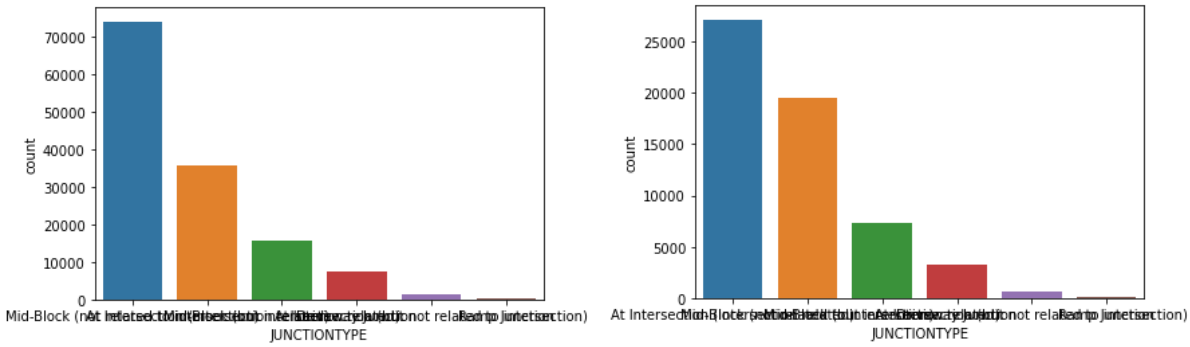
Location

The location variable contained a large amount of values so using a chart will not be helpful in determining if there is high similarity between variables. Thus, we investigated the value counts of the variable between severity codes. There is an apparent difference between locations which have had property damage because of collisions and locations which have had injuries because of collisions. Aurora Ave N between N 117th Pl and N 125th St has the highest number of injury related collisions but is not seen in the top 5 of the collisions which caused property damage. Therefore, we will include this variable in our model.

INCDTTM

This is the date variable which we cleaned up earlier. This variable is like the location variable in that there are too many unique values to use a bar chart effectively. We investigated the value count to determine if there are high similarities between severity codes. Seeing the spread of data on the tables allows us to see if there are days in which accidents were at a high. We see that there is some overlap with dates like 2006-11-02 and 2005-05-18, but there are also different dates, so we will include this variable into our model

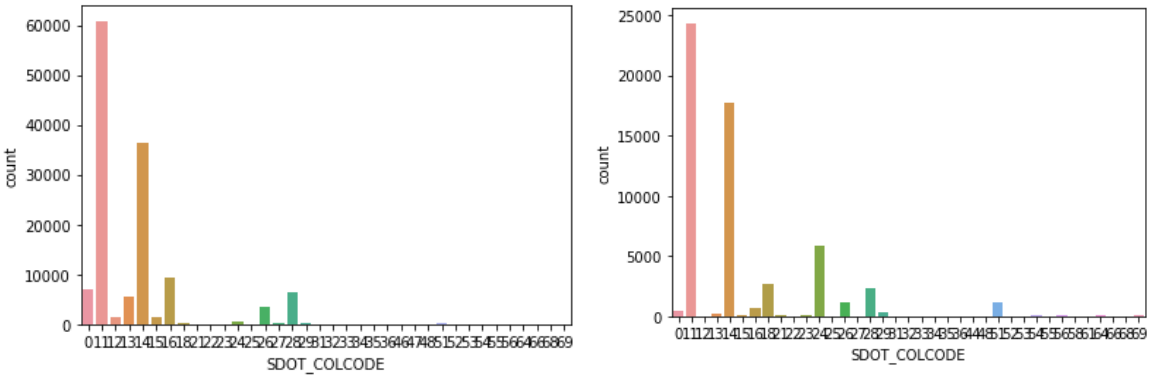
JunctionType



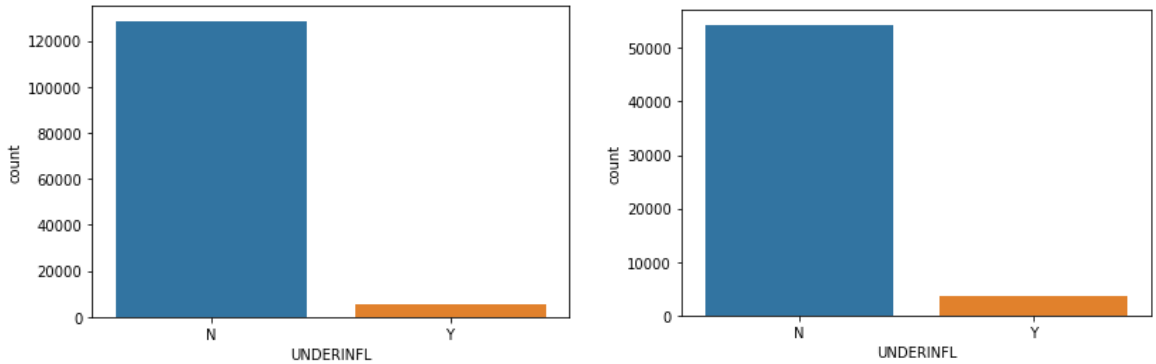
We can see that there are differences in relevancy of junctions between injury related incidents and property related incidents. Injury related incidents have the highest frequency in at intersection collisions while property damage related incidents have the highest frequency in mid block collisions. We will include this variable in our model.

SDOTColCode

This is another variable which contains large amounts of values but using a bar chart lets us see the difference in distribution between two severity codes. In the value counts, the state collision code has common values in the same positions in both groups in terms of frequency. Values such as 11 and 14 are in the top 2 spots in both severities. However, we notice differences in values as we go down the table so we will consider this variable in our model since there does not appear to be a high level of correlation between the two severities.

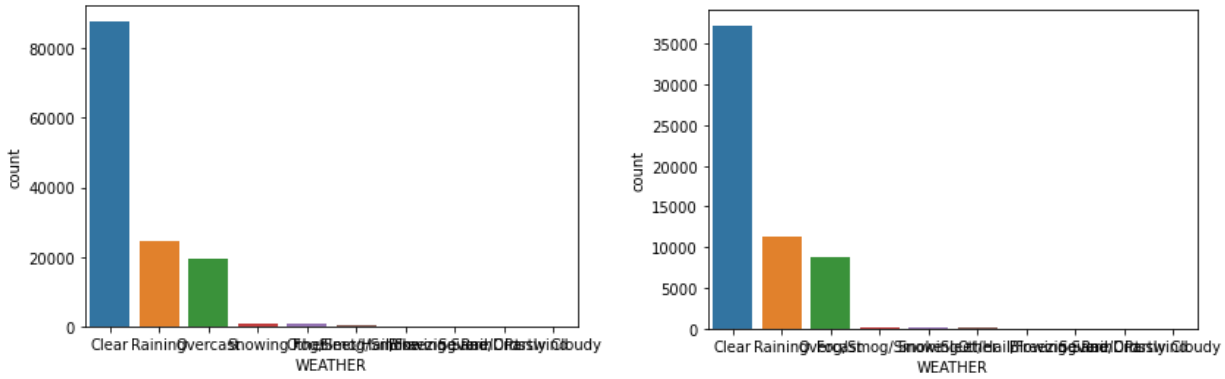


UnderInfluence



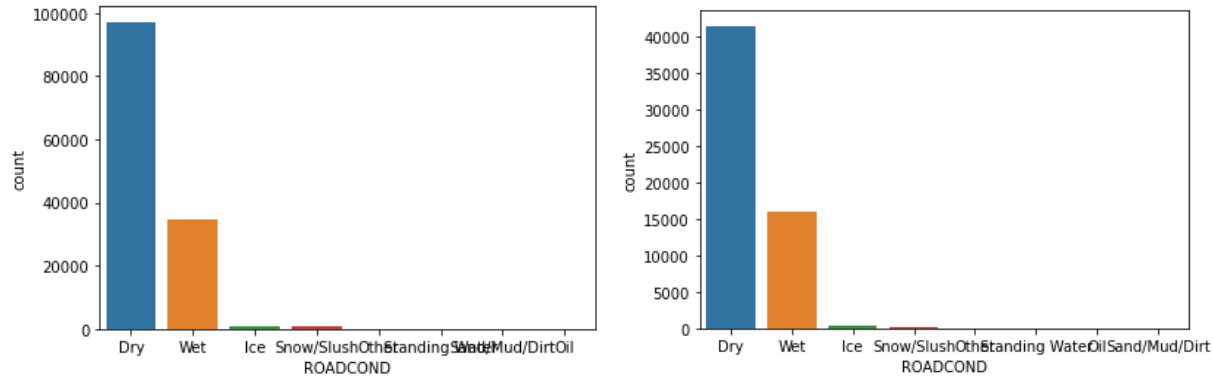
In more severe accidents, we see that the bar for yes is slightly higher than that of accidents which only cause property damage. There are almost 3 times more non under influence counts in the property damage data but there are less than 2 times more under the influence counts in property damage accidents. This indicates that those who are driving under the influence are more likely to cause accidents which cause injury.

Weather



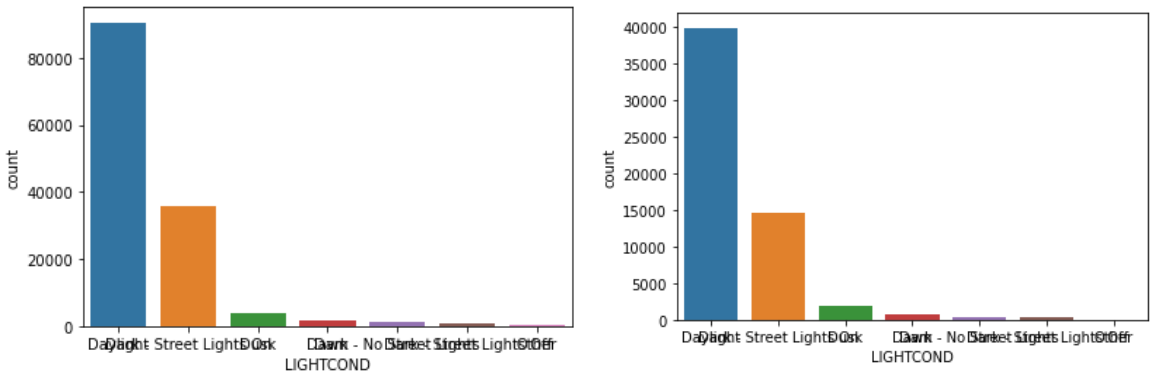
It is difficult to see that there is some variation between severity codes through the graphs, however we investigated the value counts to see if there is a difference between codes. We can see that the weather variable is highly correlated, but it has different rankings for different weather between injury and property damage collision cases. Snowy weather is more likely the cause of an accident in property damage accidents compared to snow while fog is more likely to be the cause of an accident in injury accidents compared to snow. It may be beneficial to include this variable in the model.

RoadCond



The road condition is very similar in both cases so we will leave out this variable from our model.

LightCond

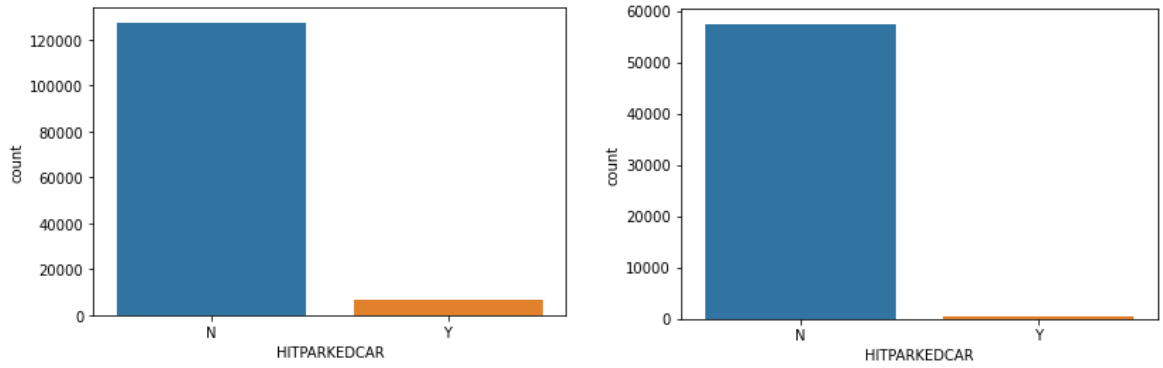


The Lighting condition does not differ among different accident severities so we will leave this variable out of our model

StColCode

It is difficult to tell the worthiness of this variable when using a bar chart due to the high number of unique values, as result the value counts were investigated. The highest frequency values are different between different severity codes so this variable will be included in the model.

HitParkedCar



There are a lot fewer yes responses to injury related collisions even factoring in the higher sample count in non injury cases, so it would appear to be beneficial including this variable in our model. It logically

makes sense that collisions which involve hitting a parked car would only cause property damage in many cases. This backs up that theory.

Variable Selection

The data analysis allowed us to conclude that variables: 'PERSONCOUNT', 'PEDCOUNT', 'VEHCOUNT, 'PEDCYLCOUNT', 'ADDRTYPE', 'LOCATION', 'INCDTTM', 'JUNCTIONTYPE', 'SDOT_COLCODE', 'UNDERINFL', 'WEATHER', 'ST_COLCODE', and 'HITPARKEDCAR' were good variables to use in our model. Other variables were dropped.

The categorical variables in our selected variables were factorized into numerical values representing the categories of each variable so they can be used in training and testing our model. The dataset, now containing all numerical values, were normalized to avoid variables of high category counts having higher influence over those with lower category counts. The normalized matrix was split into training and testing groups which were then used to train out models.

Model Selection

Our problem is a classification problem. We must choose between KNN, SVM, Decision Trees or Logistic Regression for our model. Each type of model was trained and the model with the best solution was chosen.

Decision Trees

Depth = 1	Precision	Recall	F1-score
Macro Avg	.61	.63	.6
Weighted Avg	.69	.61	.63
Depth = 2	Precision	Recall	F1-score
Macro Avg	.67	.67	.57
Weighted Avg	.77	.57	.58
Depth = 3	Precision	Recall	F1-score
Macro Avg	.67	.69	.66
Weighted Avg	.73	.68	.69
Depth = 4	Precision	Recall	F1-score
Macro Avg	.66	.69	.66
Weighted Avg	.73	.68	.69
Depth = 5	Precision	Recall	F1-score
Macro Avg	.66	.69	.66
Weighted Avg	.73	.68	.70
Depth = 6	Precision	Recall	F1-score
Macro Avg	.67	.70	.67
Weighted Avg	.74	.70	.71
Depth = 7	Precision	Recall	F1-score
Macro Avg	.67	.71	.67
Weighted Avg	.74	.70	.71
Depth = 8	Precision	Recall	F1-score
Macro Avg	.68	.71	.67
Weighted Avg	.75	.69	.70

Depth = 9	Precision	Recall	F1-score
Macro Avg	.68	.71	.67
Weighted Avg	.75	.68	.69
Depth = 10	Precision	Recall	F1-score
Macro Avg	.67	.70	.66
Weighted Avg	.75	.68	.69
Depth = 50	Precision	Recall	F1-score
Macro Avg	.62	.63	.61
Weighted Avg	.69	.63	.65
Depth = 100	Precision	Recall	F1-score
Macro Avg	.61	.63	.61
Weighted Avg	.68	.63	.65

We tested our data on many depths for our decision trees. Ultimately, the best score was obtained by the tree with a max depth of 7. We received an overall average f1-score of 0.7. The model was able to correctly identify 72% of all injury causing accidents and 69% of all property damage accidents. The model was also 85% correct when it labelled an accident as a property damage accident, and 50% correct when it labelled an accident as an injury related accident.

K Nearest Neighbours

k = 1	Precision	Recall	F1-score
Macro Avg	.62	.64	.62
Weighted Avg	.69	.64	.66
k = 2	Precision	Recall	F1-score
Macro Avg	.64	.63	.64
Weighted Avg	.69	.70	.70
k = 3	Precision	Recall	F1-score
Macro Avg	.64	.67	.64
Weighted Avg	.71	.66	.67
k = 4	Precision	Recall	F1-score
Macro Avg	.65	.66	.65
Weighted Avg	.70	.70	.70
k = 5	Precision	Recall	F1-score
Macro Avg	.65	.68	.65
Weighted Avg	.72	.67	.68
k = 6	Precision	Recall	F1-score
Macro Avg	.66	.67	.66
Weighted Avg	.72	.70	.71
k = 7	Precision	Recall	F1-score
Macro Avg	.66	.68	.65
Weighted Avg	.73	.67	.69
k = 8	Precision	Recall	F1-score
Macro Avg	.66	.68	.66
Weighted Avg	.72	.70	.70
k = 9	Precision	Recall	F1-score

Macro Avg	.66	.68	.65
Weighted Avg	.73	.67	.68
k = 10	Precision	Recall	F1-score
Macro Avg	.66	.68	.66
Weighted Avg	.72	.69	.70

The model with the highest average f1-score and best results from all our KNN models was the model with a K value of 8. There were many models which got an average f1-score of 0.70, like the K value = 8 model. The distinguishing factor is that the k value = 8 model had the highest recall score when considering all outputs. The model was able to predict 73% of all property damage accidents and 63% of all injury accidents while being 82% correct in it's prediction when it labelled an accident as a property damage accident and 50% correct when it labelled an accident as an injury related accident.

SVM

SVM	Precision	Recall	F1-score
Macro Avg	.67	.70	.66
Weighted Avg	.74	.68	.69

The SVM model was able to predict 75% of all injury related accidents and was 49% correct when it labelled an accident as an injury related accident. Additionally, it was able to predict 65% of all property damage accidents and was 85% correct when it labelled an accident as property damage. The f1 score was 0.74 and 0.59 for property damage accidents and injury accidents respectively earning it an average f1 score of 0.68.

Logistic Regression

Logistic Regression	Precision	Recall	F1-score
Macro Avg	.63	.65	.63
Weighted Avg	.7	.65	.67

The logistic regression model was able to predict 66% of all property damage accidents and 65% of all injury related accidents. It was also 81% correct when it labelled an accident as a property damage accident and 46% correct when it labelled an accident as an injury related accident. This model did worse than the SVM model but did a decent job overall as a severity predictor.

The decision tree model was the best model we were able to create since it has the highest precision and recall of all the models we have created. This will be the model which should be chosen to perform accident severity prediction with the available data.

Results

Our analysis was able to show us that the relevant variables we should use in our models were 'PERSONCOUNT', 'PEDCOUNT', 'VEHCOUNT, 'PEDCYLCOUNT', 'ADDRTYPE', 'LOCATION', 'INCDTTM', 'JUNCTIONTYPE', 'SDOT_COLCODE', 'UNDERINFL', 'WEATHER', 'ST_COLCODE', and 'HITPARKEDCAR'. Then we dropped all other variables and factorized the categorical variables so that they can be used in fitting

and testing for our models. A range of values were used in our decision tree and knn models for the number of trees and K neighbours, respectively. An SVM and Logistic Regression model was also created for the data but ultimately the decision tree with a depth of 7 was the most successful model.

The decision tree was able to give us a precision score of 0.85 and a recall score of 0.69 for the prediction of a property damage accident and a precision score of 0.50 and a recall score of 0.72 for the prediction of an injury accident. Other models were able to obtain similar results, with the second-best model being a KNN model with a k value of 8. The KNN model was able to give us precision and recall scores of 0.82 and 0.73 for property damage accidents, and 0.50 and 0.63 for injury causing accidents.

The decision tree's accuracy in plain English, can identify 69% of all property damage accidents and 72% of all injury causing accidents. Furthermore, when it did predict an accident as a property damage accident, it was correct 85% of the time. In the case of injury causing accidents, it was correct 50% of the time. These were collectively, the highest numbers we were able to obtain from any model, giving us confidence that this is the best model to use given the data we have.

Discussion

Our findings do not solely rely on recall as our focus is not to save individuals who are in the accident, but to assist GPS navigation systems in predicting routes around the accident. As a result, it is important for us to factor in precision, recall and the f1-score. Our findings were able to point us towards the direction of our decision tree with a depth of 7 branches. This was able to provide us with the most accurate model of all our models we created.

It is important to note that our model is not as reliable as we had hoped considering the model still makes a mistake in identifying 31% of property damage accidents and 28% of injury related accidents, but it does point us towards the right direction. It would be helpful to include more information to better predict the severity of an accident. Car information, number of officers, number of ambulances, and number of tow trucks could help us increase the reliability of our models thus increase the reliability in a GPS not directing drivers towards traffic.

Conclusion

The purpose of this project was to be able to create a model which would be able to correctly label an accident as a property damage accident or an injury related accident. Determining this severity would ultimately be able to assist GPS companies in routing users away from an accident scene if the accident is severe enough. Our model was able to detect 72% of all injury related accidents and 69% of all property related accidents, however it still has room for improvement. This report should be used as a base point for which an even reliable model can be created upon. Ultimately, the model was limited due to the lack of information available to us by the given data. With a greater amount of information, a more reliable model should be able to be created, giving great benefit to the company providing directions, and the customers using the software following the directions.