The Prediction of Traffic Collision Severity

A Focus on Seattle, U.S.A.

Michel Soto Osegueda

August 2020

Table of Contents

1. Introduction

1.1 Problem and Data Description

Accidents while driving lead to many problems: traffic, lost time, money, health, health insurance, loss of a loved one, and many more. Governments try to prevent accidents to make the country a safer place to live and transit. As part of this effort, my study seeks to be able to reduce the amount and severity of accidents by predicting when a severe accident has more probabilities of happening, based several conditions like: road, weather and light conditions, location where it took place, etc. The deliverable will be a machine learning algorithm that can predict the severity of an accident, if it occurs.

The purpose is to be able to have a tool to use, so the information may be used to warn people and drivers who are in a high-severity-risk area at a certain moment, and accidents could be prevented.

The data is a table of a little under 200,000 rows and 38 columns. Each row is an accident. Columns are attributes of the accident, such as: severity code, road conditions, light conditions, weather forecast, parked car, cars involved, speeding, and others.

Severity code will be used as the dependent variable, and many of the other will be the independent variables. Data wrangling and cleaning will be performed to transform all strings into floats or integers. Then, the data will be fitted in three Machine Learning Methods: K-Nearest Neighbor, Decision Tree, and Support Vector Machine. The accuracy of each method will be calculated and compared to choose the best method to work the algorithm.

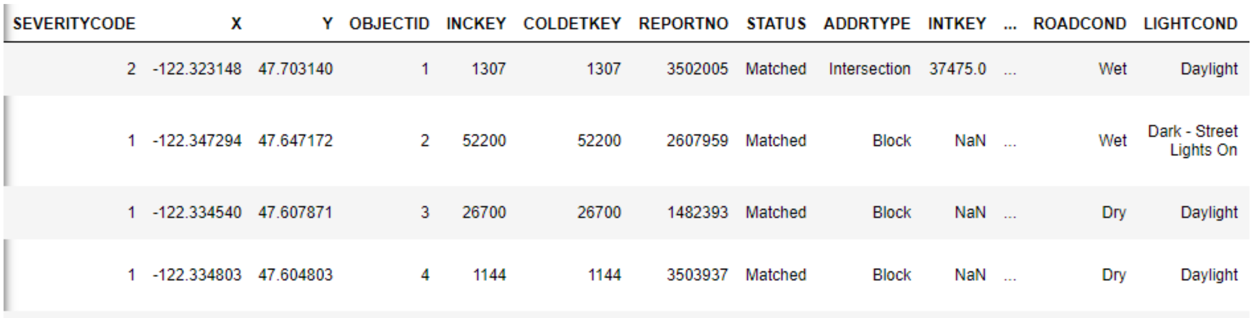
2. Exploring the Data

2.1 Data Source

The data was provided by Coursera but sourced from the Seattle Police Department, and it contains information about traffic collisions in the Seattle Area from 2004 to Present. Online, the table is constantly fed new data and can be downloaded in this link.

2.2 Overview of the table

The table is has more than 194,000 rows and 38 columns. Each row is a traffic collision, and contains 38 different attributes (or columns) that describe the causes, the incident, and some of the consequences.



SEVERITYCODE	X	Y	OBJECTID	INCKEY	COLDETKEY	REPORTNO	STATUS	ADDRTYPE	INTKEY	...	ROADCOND	LIGHTCOND
2	-122.323148	47.703140	1	1307	1307	3502005	Matched	Intersection	37475.0	...	Wet	Daylight
1	-122.347294	47.647172	2	52200	52200	2607959	Matched	Block	NaN	...	Wet	Dark - Street Lights On
1	-122.334540	47.607871	3	26700	26700	1482393	Matched	Block	NaN	...	Dry	Daylight
1	-122.334803	47.604803	4	1144	1144	3503937	Matched	Block	NaN	...	Dry	Daylight

Table 1. First 5 rows and some of the columns in the data.

The most important column is SEVERITYCODE, and it measures the severity of a traffic collision, based on its characteristics. This study will create a Machine Learning model that will be able to predict the severity of collisions that have not happened yet, based the most important of the characteristics. The data provided needs cleaning to be able to fit and predict any outcome.

3. Methodology

3.1 Data Cleaning

3.1.1 Missing and NaN Values

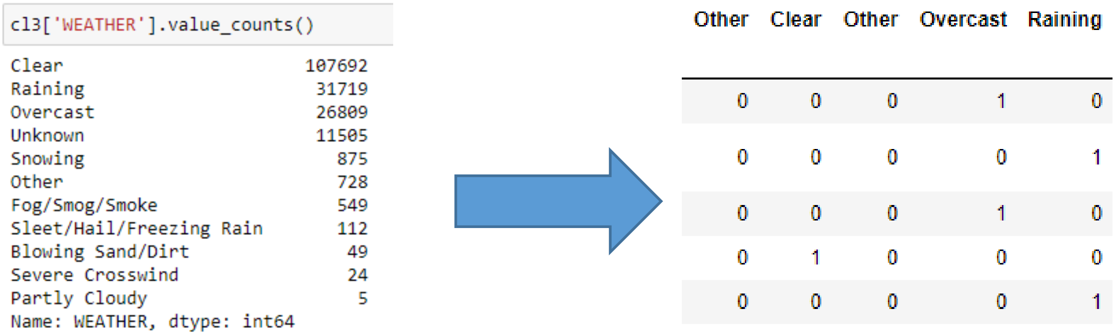
Columns with more than 50% of missing data are not usable and were deleted. Rows with missing values were either deleted, or replaced with the average of the column, so we would affect the results the least possible.

3.1.2 Other, Unknown, and small categories.

Inside many of the columns had many choices with very small counts. They were changed to be included in "Other" category. Rows with data stated to be unknown were deleted.

3.2 Feature Selection

Once the missing, NaN and unknown values were cleaned, columns with two answers were replaced by 1s and 0s; columns with three or more categories were converted to one column per category, using the moth DUMMYS, resulting in as many columns (of 1s and 0s) as they were categories in the original column, and finally the original column was dropped (Figure 1).



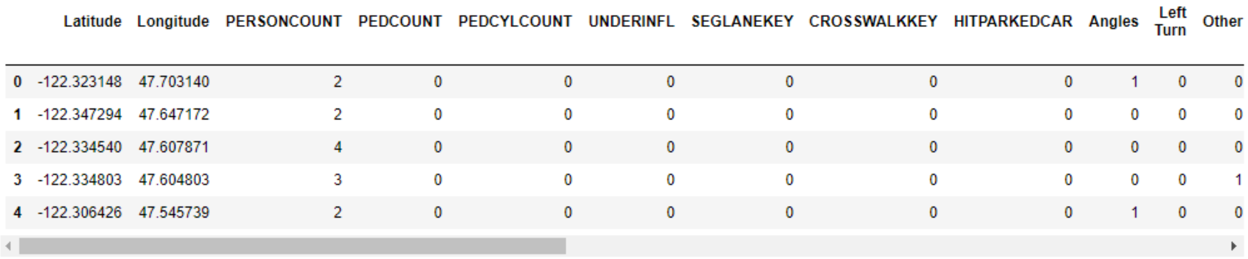
```
cl3['WEATHER'].value_counts()
```

```
Clear                       107692
Raining                      31719
Overcast                     26809
Unknown                      11505
Snowing                        875
Other                          728
Fog/Smog/Smoke                 549
Sleet/Hail/Freezing Rain       112
Blowing Sand/Dirt               49
Severe Crosswind                24
Partly Cloudy                    5
Name: WEATHER, dtype: int64
```

Other	Clear	Other	Overcast	Raining
0	0	0	1	0
0	0	0	0	1
0	0	0	1	0
0	1	0	0	0
0	0	0	0	1

Figure 1. WEATHER column has many answers, and was changed to many columns of 1s and 0s.

After repeating the process for all the selected features, the result is a table with no strings, and ready to be normalized (Table2)



	Latitude	Longitude	PERSONCOUNT	PEDCOUNT	PEDCYLCOUNT	UNDERINFL	SEGLANEKEY	CROSSWALKKEY	HITPARKEDCAR	Angles	Left Turn	Other
0	-122.323148	47.703140	2	0	0	0	0	0	0	1	0	0
1	-122.347294	47.647172	2	0	0	0	0	0	0	0	0	0
2	-122.334540	47.607871	4	0	0	0	0	0	0	0	0	0
3	-122.334803	47.604803	3	0	0	0	0	0	0	0	0	1
4	-122.306426	47.545739	2	0	0	0	0	0	0	1	0	0

Table 2. Features are ready to be normalized

3.3 Machine Learning Models

3.3.1 K-Nearest Neighbor

After normalization, the data was first fitted in a K-Nearest Neighbor model, using a "for loop" to calculate 60 values for K on our KNN model, so the results would give us an array of the best achievable accuracy. Then, the resulting values were graphed to visualize it.

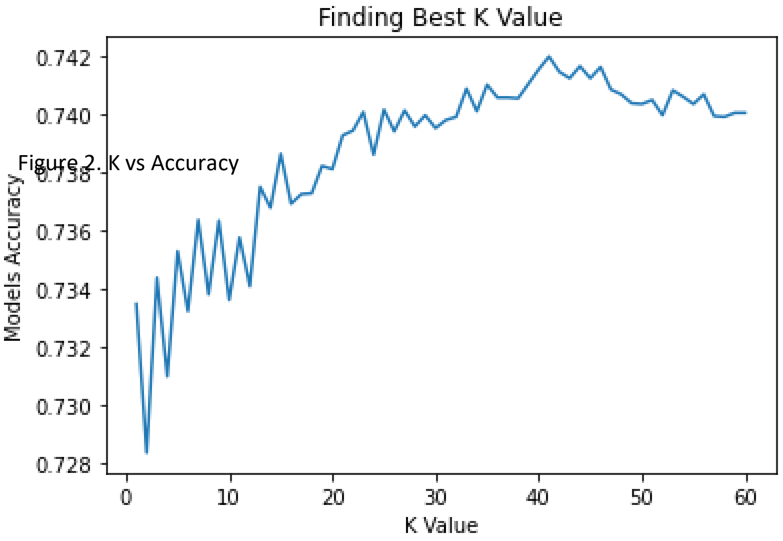
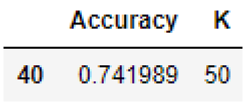


Figure 2. K vs Accuracy

From the array of resulting Ks, the best value of accuracy (the highest in the graph) is:



	Accuracy	K
40	0.741989	50

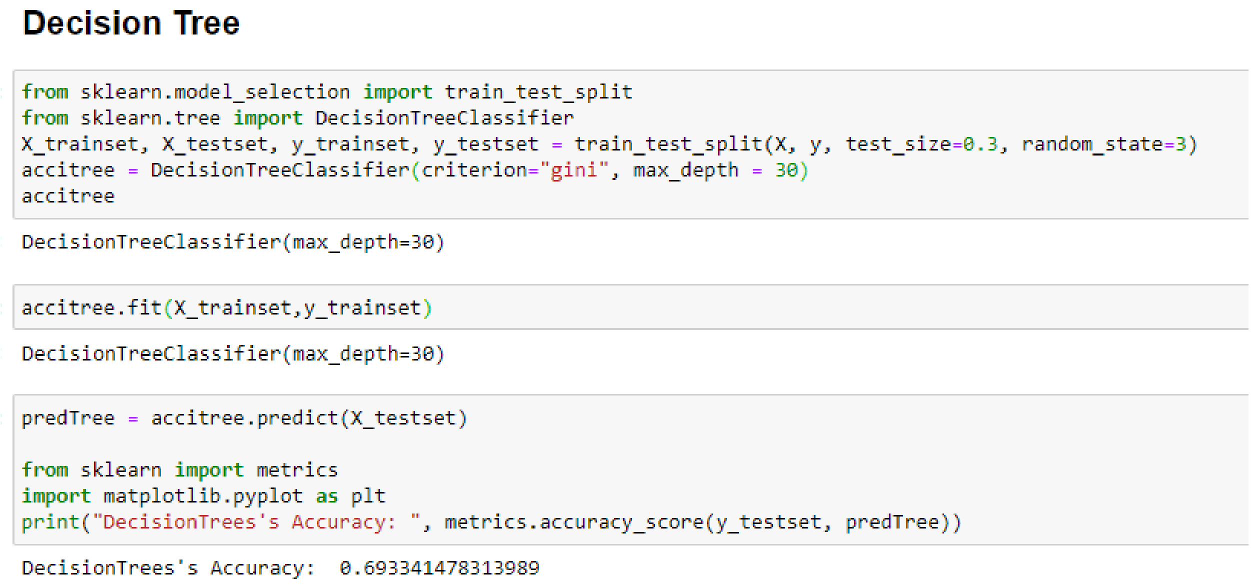
Table 3. Best K and Accuracy

The accuracy of the model K-Nearest Neighbor, when used to predict the severity of new collisions, is 74.1989%.

In other words, if we use this model to alert the drivers when the conditions are such, that there is a high chance of a severe accident from happening, 74% of the time our model would be right, and an alert could potentially save the involved and rest of the drivers many problems.

3.3.2 Decision Tree

The data was also fitted and run on the Decision Tree model, with the objective of comparing the accuracy of several models and choose the best one for our problem.



Decision Tree

```
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, y, test_size=0.3, random_state=3)
accitree = DecisionTreeClassifier(criterion="gini", max_depth = 30)
accitree
```

```
DecisionTreeClassifier(max_depth=30)
```

```
accitree.fit(X_trainset,y_trainset)
```

```
DecisionTreeClassifier(max_depth=30)
```

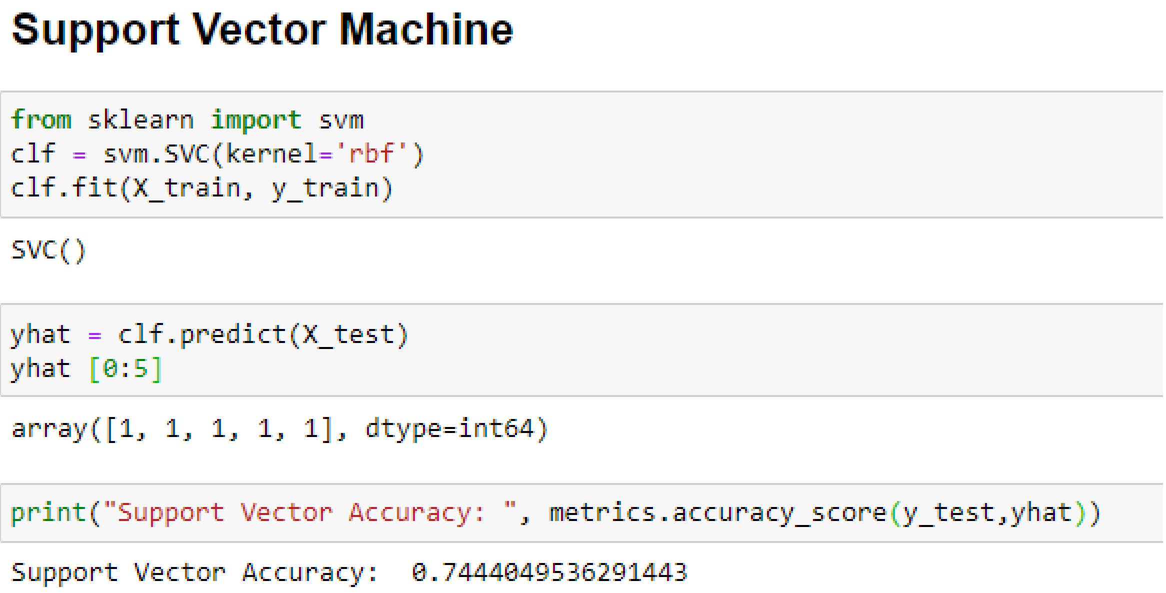
```
predTree = accitree.predict(X_testset)

from sklearn import metrics
import matplotlib.pyplot as plt
print("DecisionTrees's Accuracy: ", metrics.accuracy_score(y_testset, predTree))
```

```
DecisionTrees's Accuracy:  0.693341478313989
```

The accuracy of the Decision Tree Model, when used to predict the severity of traffic collisions in Seattle, is 69.33%.

3.3.3 Support Vector Machine

The data was also fitted in a Support Vector Machine model, to test its accuracy and find the best model for our problem.



Support Vector Machine

```
from sklearn import svm
clf = svm.SVC(kernel='rbf')
clf.fit(X_train, y_train)
```

```
SVC()
```

```
yhat = clf.predict(X_test)
yhat [0:5]
```

```
array([1, 1, 1, 1, 1], dtype=int64)
```

```
print("Support Vector Accuracy: ", metrics.accuracy_score(y_test,yhat))
```

```
Support Vector Accuracy:  0.7444049536291443
```

The accuracy of this model, to predict the severity of accidents in the Seattle Area, is 74.44%, making this the best fitting model for our data, with the best accuracy.

4. Results

4.1 Model and Accuracy

With the data fitted in three different models, the best model to predict the severity of collisions is chosen by the highest accuracy score.

Model	Accuracy
K-Nearest Neighbor	0.741989
Decision Tree	0.693341
Support Vector Machine	0.744404

To predict the severity of traffic collisions in the Seattle Area, the best Machine Learning Model (of the three tested in this study) for the data and problem is the **Support Vector Machine**, with a 74.44% accuracy.

4.2 Proposed Applied Solution

With this model, a government institution in charge of transit can create an alarm and notify drivers and people in the area there is high risk of a possible very sever accident. The notification could make drivers drive extra carefully that night and a sever collision was possibly prevented, along with all the other consequences and problems derived from the collision.

5. Discussion

The depth of this study is not meant to be an advanced one, but given the data available, it would be very useful if more people have access to this data. There is probably a lot that can be done for all the highways and locations, if you do the effort of taking it nationally.

6. Conclusions

Collision severity can be predicted, with an acceptable accuracy. Preventing collisions saves lives, economic, health, and many other lingering problems to the people involved, and the closest to them. Using this study, an application or software can be created to trigger an alarm when the conditions of a severe collision are met, before it happens.

An application is needed to use the proposed model to trigger the alarm, and warn people near the risk location.