Predicting the Severity of Car Accidents

Stephanie Graziano

September 29, 2020

1. Introduction

1.1. Background

People are driving more now than they ever have been; whether it be commuting, seeing friends, road tripping, or rushing to get the kids to soccer. For our routine commutes, we all have the time it takes to get to one place from another programmed in our heads and typically don't leave much wiggle room for accidents or other delays. However, weather and road conditions change day to day and each year there are tens of thousands of car accident deaths in the U.S. These changing weather conditions play a significant role in the outcome of a car accident. Therefore, it is beneficial to accurately predict whether and how severe a car accident will be based on weather and road conditions.

1.2. Problem

Data that might contribute to determining car accident severity might include road condition, weather, light condition, speed, and other metrics that describe the type of impact. This project aims to predict the severity of a car accident based on these attributes in Seattle, Washington.

1.3. Interest

Commuters would be interested in an accurate prediction of car severity for daily commute planning. The Public Authority department would also be interested in this information to develop a warning system.

2. Data Acquisition and Cleansing

2.1. Data Source

Accident severity data can be found in this Kaggle dataset from the Seattle DOT. Data is from 2004 to present.

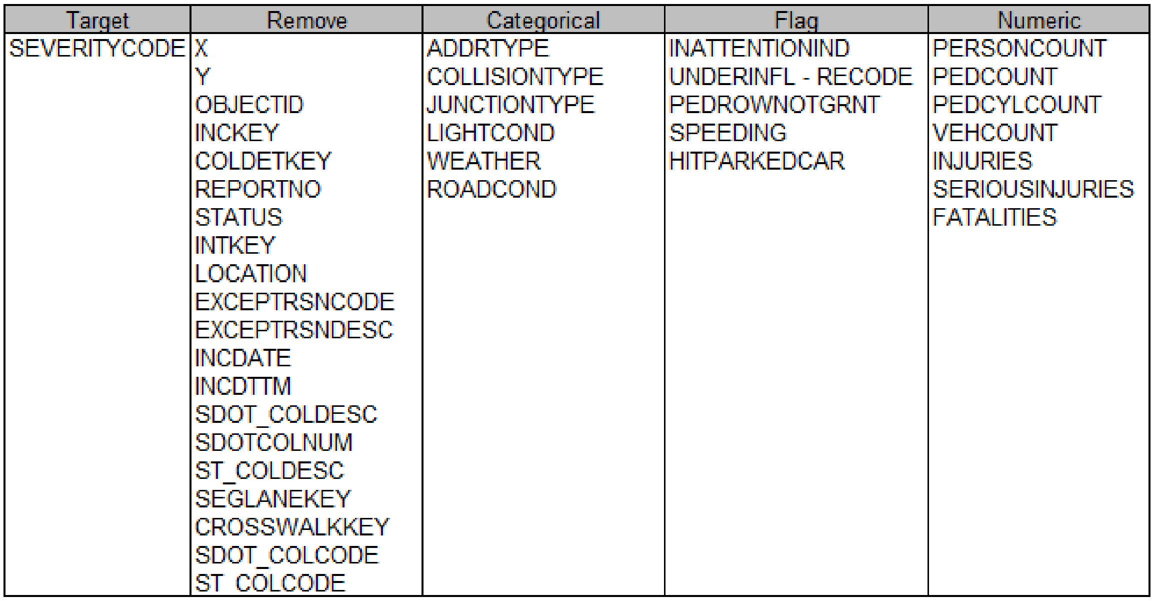
2.2. Data Cleansing

In its original form, this data is not fit for analysis. There are many columns that are not relevant for use in this model and therefore will be removed. Also, most of the features are of object type, when they should be numerical. We must use one-hot encoding to convert the features to our desired data type. Other flag columns include a combination of object and numerical data which much also be consistently converted to 0 or 1. MinMaxScaler will be used to scale numerical values to a given range. Lastly, there is some missing data in the columns which must be filled in using the most frequent value or mean data point.

2.3. Feature Selection

The target variable will be SEVERITYCODE because it is used measure the severity of an accident from 0 to 5 within the dataset.

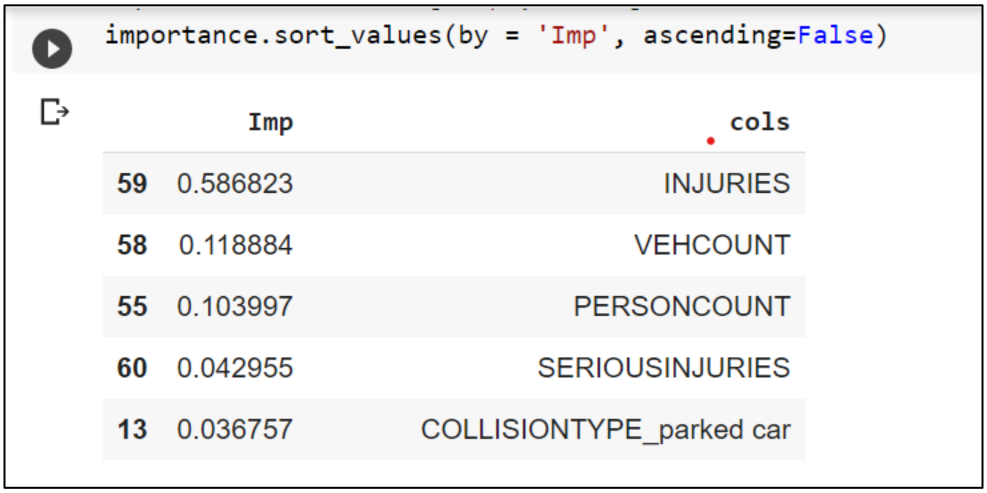
I started with a wide range of features to analyze the data:



Target	Remove	Categorical	Flag	Numeric
SEVERITYCODE	X	ADDRTYPE	INATTENTIONIND	PERSONCOUNT
	Y	COLLISIONTYPE	UNDERINFL - RECODE	PEDCOUNT
	OBJECTID	JUNCTIONTYPE	PEDROWNOTGRNT	PEDCYLCOUNT
	INCKEY	LIGHTCOND	SPEEDING	VEHCOUNT
	COLDETKEY	WEATHER	HITPARKEDCAR	INJURIES
	REPORTNO	ROADCOND		SERIOUSINJURIES
	STATUS			FATALITIES
	INTKEY			
	LOCATION			
	EXCEPTRSNCODE			
	EXCEPTRSNDESC			
	INCDATE			
	INCDTTM			
	SDOT_COLDESC			
	SDOTCOLNUM			
	ST_COLDESC			
	SEGLANEKEY			
	CROSSWALKKEY			
	SDOT_COLCODE			
	ST_COLCODE			

After running a Random Forest Classifier on the data, I noticed that there were certain features that were too predictive that skewed the data. Those features were then removed:

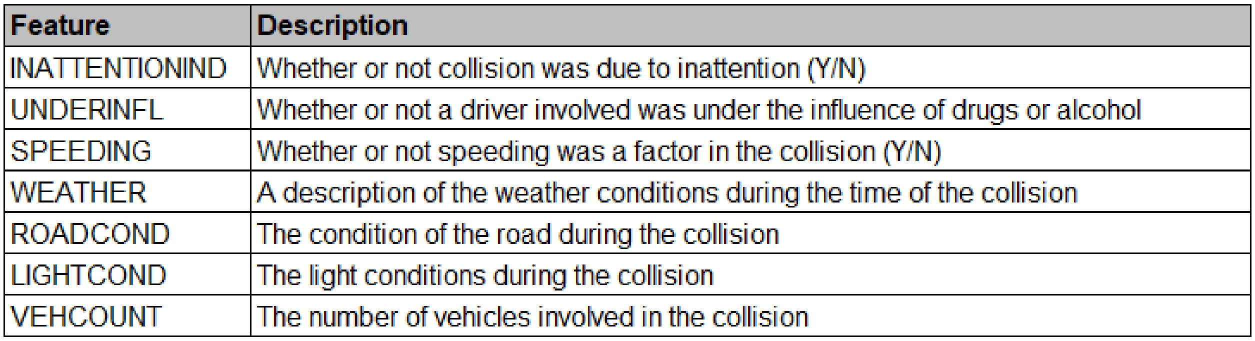
- INJURIES
- VEHCOUNT
- PERSONCOUNT



```
importance.sort_values(by = 'Imp', ascending=False)
```

```
         Imp                   cols
59  0.586823               INJURIES
58  0.118884               VEHCOUNT
55  0.103997            PERSONCOUNT
60  0.042955        SERIOUSINJURIES
13  0.036757  COLLISIONTYPE_parked car
```

After a complete analysis, seven (7) features were selected: three (3) categorical, three (3) flags, and one (1) numeric. We can now use this data in our analysis and machine learning models.



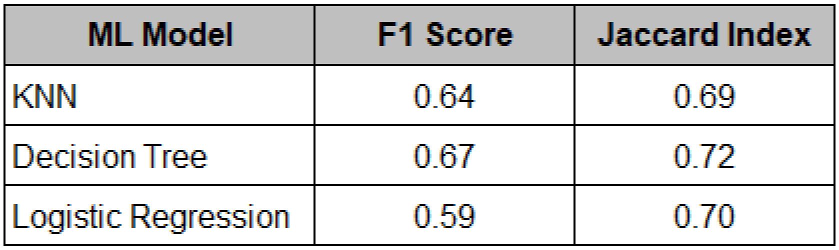
Feature	Description
INATTENTIONIND	Whether or not collision was due to inattention (Y/N)
UNDERINFL	Whether or not a driver involved was under the influence of drugs or alcohol
SPEEDING	Whether or not speeding was a factor in the collision (Y/N)
WEATHER	A description of the weather conditions during the time of the collision
ROADCOND	The condition of the road during the collision
LIGHTCOND	The light conditions during the collision
VEHCOUNT	The number of vehicles involved in the collision

3. Methodology

After balancing the target variable, and standardizing the input features, the data was ready for building machine learning models. The machine learning models used were K Nearest Neighbor, Decision Tree Classifier, and Logistic Regression. K Nearest Neighbor is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (based on distance). Logistic Regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. The Decision Tree Classifier breaks down a data set into smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. The reason why these classification methods were chosen is because the Support Vector Machine (SVM) model is inaccurate for large data sets, while this data set has more than 180,000 rows filled with data.

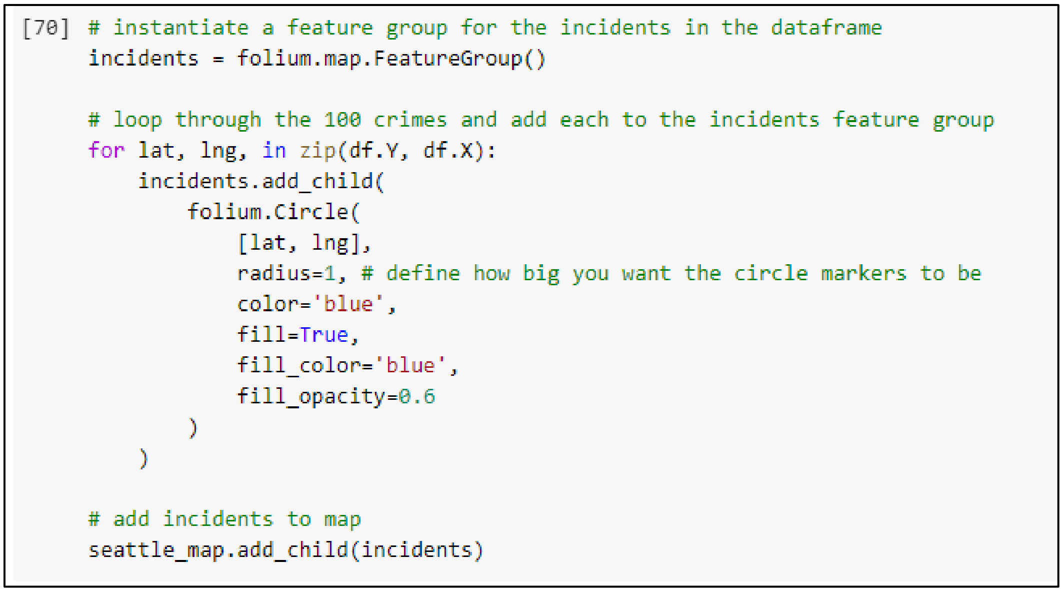
4. Results

The results of the evaluation models are shown below in the table. By comparing all the models by their Jaccard and F1 Scores, we can see a clearer picture in terms of the accuracy of the three models individually and how well they perform. Based on the results, the Decision Tree Classifier is the best model to predict the severity of a car accident with an F1 Score of 67% and a Jaccard Score of 72%.



ML Model	F1 Score	Jaccard Index
KNN	0.64	0.69
Decision Tree	0.67	0.72
Logistic Regression	0.59	0.70

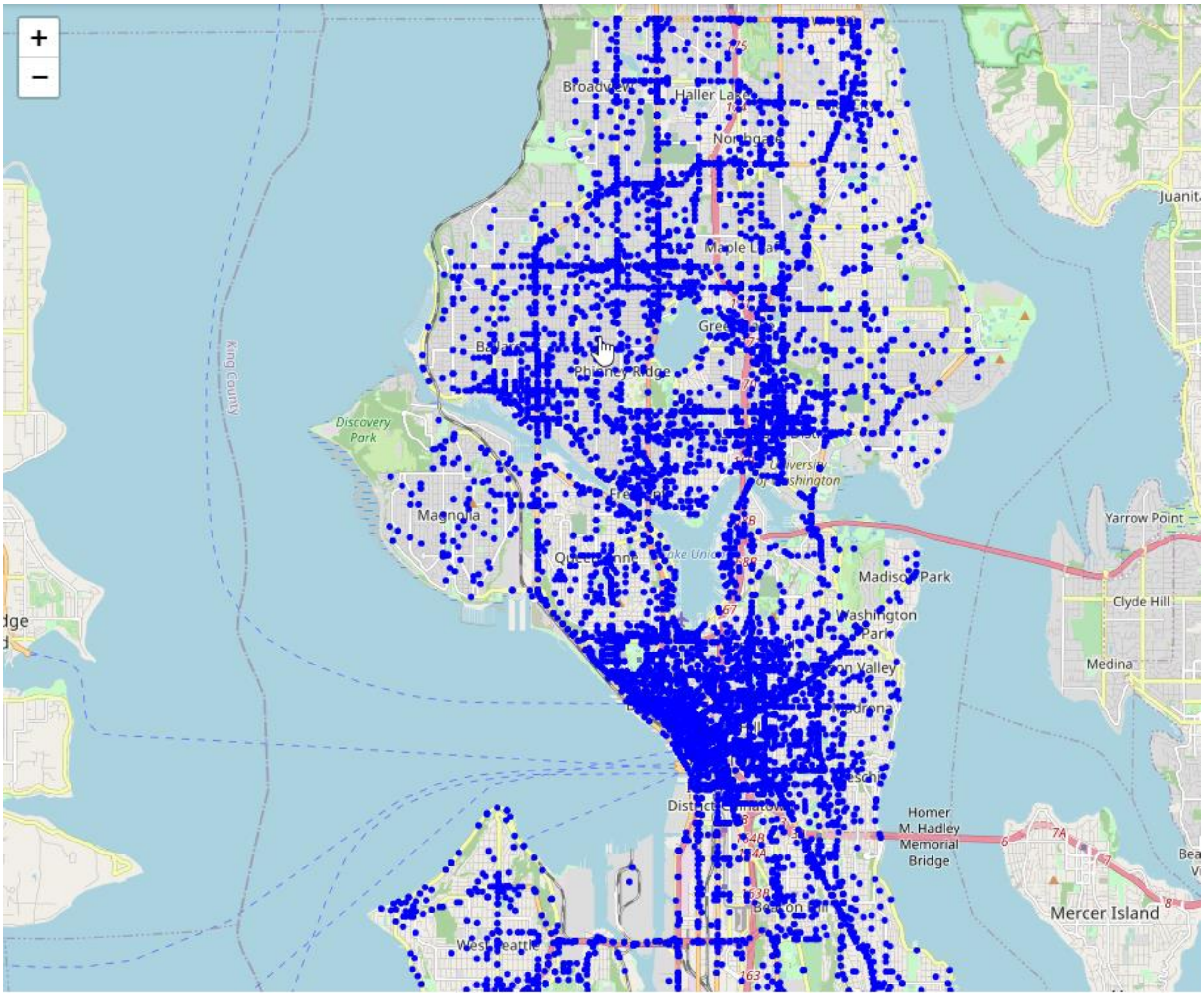
Most accidents happen in Belltown:



```
[70] # instantiate a feature group for the incidents in the dataframe
     incidents = folium.map.FeatureGroup()

     # loop through the 100 crimes and add each to the incidents feature group
     for lat, lng, in zip(df.Y, df.X):
         incidents.add_child(
             folium.Circle(
                 [lat, lng],
                 radius=1, # define how big you want the circle markers to be
                 color='blue',
                 fill=True,
                 fill_color='blue',
                 fill_opacity=0.6
             )
         )

     # add incidents to map
     seattle_map.add_child(incidents)
```



5. Conclusion

These models could have performed better if a few more things were present and possible.

- A balanced dataset for the target variable.
- More instances recorded of all accidents taken place in Seattle, Washington.
- Minimized missing values within the dataset for feature variables.

6. Recommendations

The Public Authority department of Seattle, Washington can assess how many of these accidents have happened where road or light conditions were not ideal, for that specific area, and develop a warning system to alert the public of higher risk in accident severity. Drivers would also be able to use this information when planning their commute.

Sources

https://www.kaggle.com/jonleon/seattle-sdot-collisions-data/