IBM Data Science Professional Certificate

Applied Data Science Capstone

Prediction of the Car Accident Severity in Seattle City, Washington

Emre Can Okten

September 15th, 2020

1. Introduction

There are various factors contributing to the occurrence of a car accident in the highways. The main objective of this project is to create a machine learning model to predict the possibility of the car accident and how severe it would be according to given features such as weather and road conditions, so that people drive more carefully or even change their travel if they are able to.

This model can be utilized to warn people to be especially mindful when travelling and to notify relevant authorities to be prepared for an accident with a certain severity level in given conditions and take immediate action to mitigate the consequences of it.

2. Data

The required data for this project is acquired from Seattle Open Data Portal [1]. It includes all types of collisions in the city from 2004 to present with the label of accident severity which defines the fatality of an accident. Apart from the labelled data, the dataset consists of 37 total attributes providing various types of information about location, date/time, weather and light conditions.

Metadata form from the website provide a good description for all attributes. After investigating all the attributes in the dataset and performing the exploratory data analysis, it is concluded that the continuous and categorical variables having significant information related to the severity of the car accident will be used to train the machine learning model. Remaining attributes in the dataset will be removed to reduce the processing and memory load in the model development.

[1] https://data-seattlecitygis.opendata.arcgis.com/datasets/5b5c745e0f1f48e7a53acec63a0022ab_0

3. Methodology

In this project, python has been used as a programming language to deal with basic data frame and array operations, fundamental data manipulation techniques, exploratory data visualizations, model development and model evaluation. The required libraries used in the analysis are following:

1. pandas, numpy for basic operations and data manipulation
2. seaborn, matplotlib for visualization
3. sklearn, xgboost for model development and evaluation

3.1 Exploratory data analysis

In order to have a deeper understanding of the dataset, fundamental data exploration techniques will be performed by means of shape, info, dtypes and describe methods in pandas library.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 194673 entries, 0 to 194672
Data columns (total 38 columns):
SEVERITYCODE      194673 non-null int64
X                 189339 non-null float64
Y                 189339 non-null float64
OBJECTID          194673 non-null int64
INCKEY            194673 non-null int64
COLDETKEY         194673 non-null int64
REPORTNO          194673 non-null object
STATUS            194673 non-null object
ADDRTYPE          192747 non-null object
INTKEY            65070 non-null float64
LOCATION          191996 non-null object
EXCEPTRSNCODE     84811 non-null object
EXCEPTRSNDESC     5638 non-null object
SEVERITYCODE.1    194673 non-null int64
SEVERITYDESC      194673 non-null object
COLLISIONTYPE     189769 non-null object
PERSONCOUNT       194673 non-null int64
PEDCOUNT          194673 non-null int64
PEDCYLCOUNT       194673 non-null int64
VEHCOUNT          194673 non-null int64
INCDATE           194673 non-null object
INCDTTM           194673 non-null object
JUNCTIONTYPE      188344 non-null object
SDOT_COLCODE      194673 non-null int64
SDOT_COLDESC      194673 non-null object
INATTENTIONIND    29805 non-null object
UNDERINFL         189789 non-null object
WEATHER           189592 non-null object
ROADCOND          189661 non-null object
LIGHTCOND         189503 non-null object
PEDROWNOTGRNT     4667 non-null object
SDOTCOLNUM        114936 non-null float64
SPEEDING          9333 non-null object
ST_COLCODE        194655 non-null object
ST_COLDESC        189769 non-null object
SEGLANEKEY        194673 non-null int64
CROSSWALKKEY      194673 non-null int64
HITPARKEDCAR      194673 non-null object
dtypes: float64(4), int64(12), object(22)
memory usage: 56.4+ MB
```

Figure 1 Output of info() method

As the output of info method, it can be seen that dataset has 194.672 entries excluding headers and 38 columns in total. Since all 38 columns will not be used in the machine learning model, it must be decided which attributes to keep and which ones to remove before proceeding into data manipulation phase to reduce memory/data load from the system and have a cleaner dataset.

3.2 Data Wrangling

After investigating the attribute information in the metadata of the dataset and looking at the content in the cells it can be confidently said that the following attributes will not serve the goal of the project and will not be used in the feature set of our machine learning model since they include additional information such as registration information and ID codes.

'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO', 'INTKEY', 'LOCATION', 'EXCEPTRSNCODE', 'STATUS', 'PEDROWNOTGRNT', 'EXCEPTRSNDESC', 'SEVERITYCODE. 1', 'SEVERITYDESC',

'INCDTTM', 'ST_COLCODE', 'HITPARKEDCAR', 'SDOT_COLCODE', 'SDOT_COLDESC', 'SDOTCOLNUM', 'ST_COLDESC', 'SEGLANEKEY', 'X', 'Y', 'CROSSWALKKEY'.

The new shape of the dataset is (194673, 15). Remaining attributes will serve as relevant information for our machine learning model and correlation will be investigated between them and the dependent variable.

Since dataset has an accident date attribute, it can be converted into the day of week and see if there is a correlation between the accident severity and the day of the week.

It is clearly seen that some attributes have significant amount of NaN values as well as inconsistent binary categorization labels such as having both 0-1 and N-Y. The missing data should be properly treated, and inconsistent values should be corrected. The steps for dealing with the missing data will be following:

1. Replacing 'Unknown' and 'Other' values with NaN in the attributes WEATHER, ROADCOND, LIGHTCOND, JUNCTIONTYPE.
2. Replacing NaN values with 0 and N in the attributes SPEEDING, INATTENTIONIND, UNDERINFL.
3. Replacing all N values with 0 and all Y values with 1 in the attributes UNDERINFL, SPEEDING, INATTENTIONIND.

Categorical variables that are more likely to be features for the model is investigated by grouping by the target variable. Results for the investigation of the categorical variables are following:

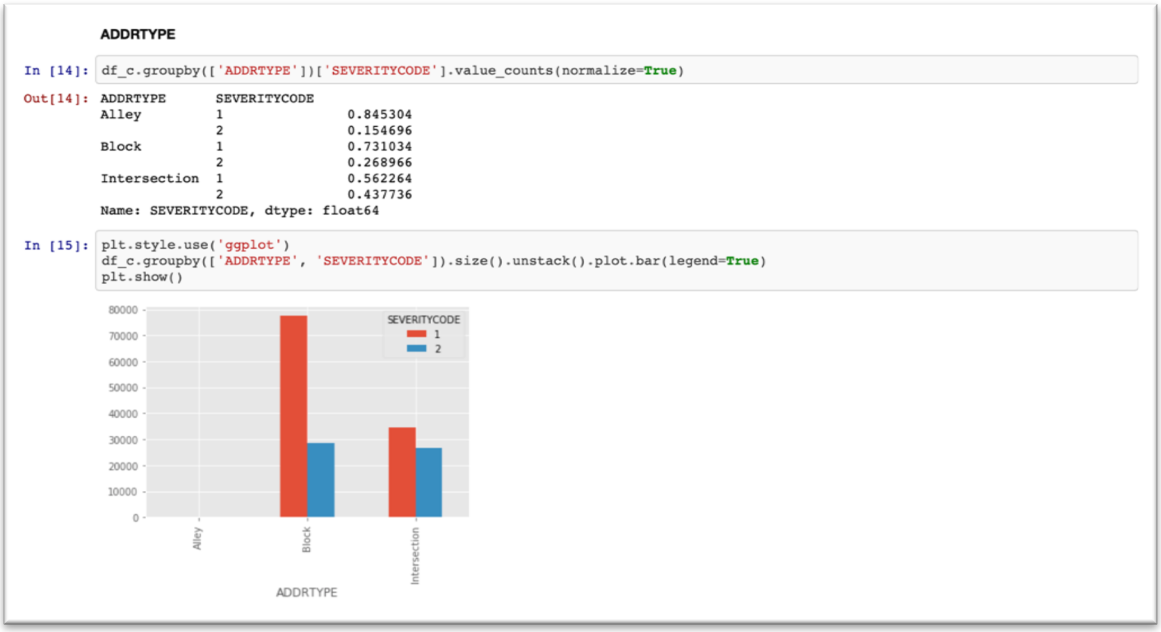

Figure 2 Investigation of ADDRTYPE attribute by SEVERITYCODE

COLLISIONTYPE

```
In [16]: df_c.groupby(['COLLISIONTYPE'])['SEVERITYCODE'].value_counts(normalize=True)
```

```
Out[16]: COLLISIONTYPE  SEVERITYCODE
         Angles         1              0.603006
                        2              0.396994
         Cycles         2              0.883259
                        1              0.116741
         Head On        1              0.559856
                        2              0.440144
         Left Turn      1              0.599850
                        2              0.400150
         Other          1              0.733017
                        2              0.266983
         Parked Car     1              0.920676
                        2              0.079324
         Pedestrian     2              0.899952
                        1              0.100048
         Rear Ended     1              0.563682
                        2              0.436318
         Right Turn     1              0.787933
                        2              0.212067
         Sideswipe      1              0.862071
                        2              0.137929
         Name: SEVERITYCODE, dtype: float64
```

```
In [17]: plt.style.use('ggplot')
         df_c.groupby(['COLLISIONTYPE', 'SEVERITYCODE']).size().unstack().plot.bar(legend=True)
         plt.show()
```

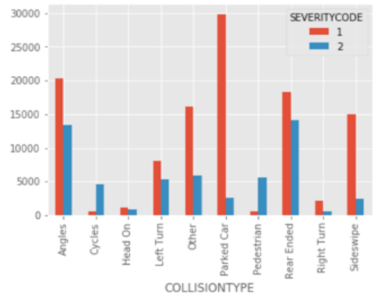

Figure 3 Investigation of COLLISIONTYPE attribute by SEVERITYCODE

JUNCTIONTYPE

```
In [18]: df_c.groupby(['JUNCTIONTYPE'])['SEVERITYCODE'].value_counts(normalize=True)
```

```
Out[18]: JUNCTIONTYPE                                    SEVERITYCODE
         At Intersection (but not related to intersection)  1              0.665737
                                                            2              0.334263
         At Intersection (intersection related)             1              0.559100
                                                            2              0.440900
         Driveway Junction                                  1              0.685691
                                                            2              0.314309
         Mid-Block (but intersection related)               1              0.670669
                                                            2              0.329331
         Mid-Block (not related to intersection)            1              0.754463
                                                            2              0.245537
         Ramp Junction                                      1              0.679739
                                                            2              0.320261
         Name: SEVERITYCODE, dtype: float64
```

```
In [19]: plt.style.use('ggplot')
         df_c.groupby(['JUNCTIONTYPE', 'SEVERITYCODE']).size().unstack().plot.bar(legend=True)
         plt.show()
```

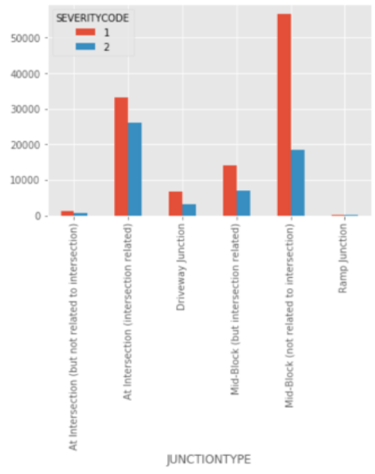

Figure 4 Investigation of JUNCTIONTYPE attribute by SEVERITYCODE

WEATHER

```
In [20]: df_c.groupby(['WEATHER'])['SEVERITYCODE'].value_counts(normalize=True)
```

```
Out[20]: WEATHER                  SEVERITYCODE
         Blowing Sand/Dirt        1             0.714286
                                  2             0.285714
         Clear                    1             0.669823
                                  2             0.330177
         Fog/Smog/Smoke           1             0.662963
                                  2             0.337037
         Overcast                 1             0.676920
                                  2             0.323080
         Partly Cloudy            2             0.600000
                                  1             0.400000
         Raining                  1             0.658540
                                  2             0.341460
         Severe Crosswind         1             0.720000
                                  2             0.280000
         Sleet/Hail/Freezing Rain 1             0.745283
                                  2             0.254717
         Snowing                  1             0.800247
                                  2             0.199753
         Name: SEVERITYCODE, dtype: float64
```

```
In [21]: plt.style.use('ggplot')
         df_c.groupby(['WEATHER', 'SEVERITYCODE']).size().unstack().plot.bar(legend=True)
         plt.show()
```

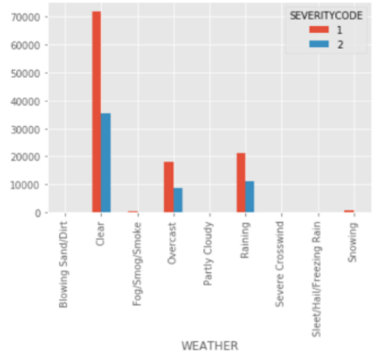

Figure 5 Investigation of WEATHER attribute by SEVERITYCODE

ROADCOND

```
In [22]: df_c.groupby(['ROADCOND'])['SEVERITYCODE'].value_counts(normalize=True)
```

```
Out[22]: ROADCOND        SEVERITYCODE
         Dry             1             0.670431
                         2             0.329569
         Ice             1             0.758231
                         2             0.241769
         Oil             1             0.578947
                         2             0.421053
         Sand/Mud/Dirt   1             0.650000
                         2             0.350000
         Snow/Slush      1             0.812880
                         2             0.187120
         Standing Water  1             0.732673
                         2             0.267327
         Wet             1             0.662207
                         2             0.337793
         Name: SEVERITYCODE, dtype: float64
```

```
In [23]: plt.style.use('ggplot')
         df_c.groupby(['ROADCOND', 'SEVERITYCODE']).size().unstack().plot.bar(legend=True)
         plt.show()
```

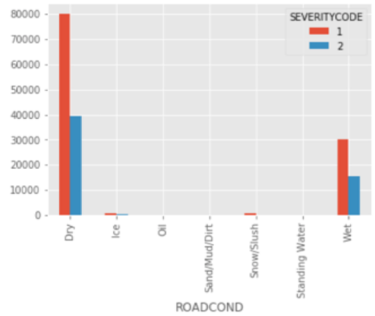

Figure 6 Investigation of ROADCOND attribute by SEVERITYCODE

```
LIGHTCOND

In [24]: df_c.groupby(['LIGHTCOND'])['SEVERITYCODE'].value_counts(normalize=True)

Out[24]: LIGHTCOND                SEVERITYCODE
         Dark - No Street Lights  1             0.765359
                                  2             0.234641
         Dark - Street Lights Off 1             0.717972
                                  2             0.282028
         Dark - Street Lights On  1             0.692114
                                  2             0.307886
         Dark - Unknown Lighting  1             0.625000
                                  2             0.375000
         Dawn                     1             0.662168
                                  2             0.337832
         Daylight                 1             0.659025
                                  2             0.340975
         Dusk                     1             0.660065
                                  2             0.339935
         Name: SEVERITYCODE, dtype: float64

In [25]: plt.style.use('ggplot')
         df_c.groupby(['LIGHTCOND', 'SEVERITYCODE']).size().unstack().plot.bar(legend=True)
         plt.show()
```

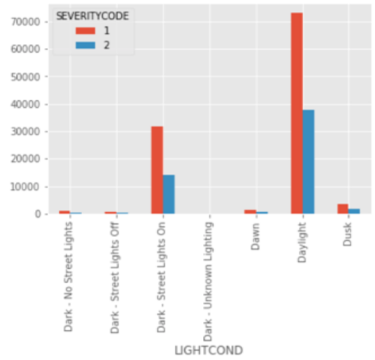

Figure 7 Investigation of LIGHTCOND attribute by SEVERITYCODE

The correlation between the accident severity and all of the categorical variables can be seen in the above figures, so these attributes will be used together with the binary categorical variables as independent variables in our machine learning model.

3.3 Model development

Since there is a categorical data in the label column, supervised classification algorithm will be used for model development. The data will be trained first and then the corresponding evaluation metrics will be calculated by means of the predicted values and the correct categorization labels in the test set.

In the feature set selection phase, the continuous variables and binary categorical variables are added into a new Feature data frame.

1. Continuous variables: 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT'
2. Binary Categorical variables: 'UNDERINFL', 'SPEEDING'

One-hot encoding technique is used to convert categorical variables to binary variables, appending them to the feature data frame. Since some of the attributes do not include significant amount of data after one-hot encoding, they will be dropped. This operation will remove the cluster from the data and save time in the model development phase.

Label set is selected as SEVERITYCODE attribute.

Feature set is normalized before model development in order to reduce data redundancy and improve data integrity. Then the feature set and label set are split into train and test set since a supervised machine learning algorithm is being used. Used classification methods are given below:

1. Decision Tree
2. XGBoost
3. Logistic Regression

4. Results

The evaluation metrics calculated for three different classification methods can be seen in the chart below.

Method	Jaccard Index	F1-score	LogLoss
Decision Tree	0.7221	0.6535	NA
XGBoost	0.7334	0.6928	NA
Logistic Regression	0.7322	0.6915	0.5208

5. Discussion

Considering all the calculated evaluation metrics, it is obvious that the most accurate classification machine learning model to predict the severity of an accident in Seattle is XGBoost with a 74.34% accuracy with Jaccard Index.

Since the accuracy of the model developed with Logistic Regression, which is used to calculate the probability of the binary classification, is too close to XGBoost model, there exists the chance of guessing the probability of an accident severity with a 73.22% accuracy as well.

F1-score has been calculated to utilize the advantage of confusion matrix, which is that it shows the model's ability to correctly predict or separate the classes. Although it has a decent precision and recall scores, its weighted average is too low compared to Jaccard Index.

6. Conclusion

The main goal of this project was to analyse the Seattle car accident dataset and conclude a prediction model based on the given criteria such as day of the week, weather, road and light condition and different locations so that the travellers in the city has an idea about which factors and conditions to pay attention to before travelling.

At the end of the model development phase, 3 different classification models, all of which can predict the severity of a car accident with almost 73% accuracy, are developed. By the help of these models, drivers in Seattle will have the opportunity of paying more attention to given conditions and reduce both the frequency and the severity of a car accident.