**Capstone Project Report - Car Accident Severity**

**Applied Data Science Capstone By IBM/Coursera**

**Introduction: Business Problem**

The project is to predict the severity of the Accidents in Seattle in certain areas and their causes. By observations through the given dataset containing information and stats about the types and causes of the accidents, I have to predict the severity of the accident depending upon traffic delay, cars or buses involved and etc which would cause or play important role in prediction.

In the given dataset, the data is provided like overspeeding, number of cars, buses, weather and road conditions. For prediction of severity we apply machine learning on dependent variable i.e severity. Therefore, Supervised learning will be used for this prediction.

**Data**

By analyzing the data we have certain insights from the data which is provided:

1. Types of injuries
2. Location of Accidents
3. Weather/Road/Driver's Conditions
4. Cars/buses/pedestrians involved and their injury severity.

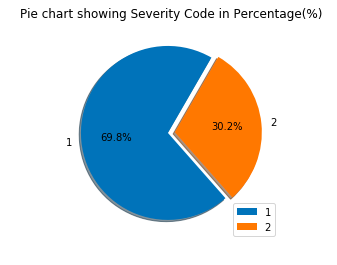
To help in predicting severity of accidents, we create dependent and independent variables and then apply train\_test\_split to split the data so that it can be applied to machine learning algorithm which we will apply. We have to predict severity so it will be our dependent variable.

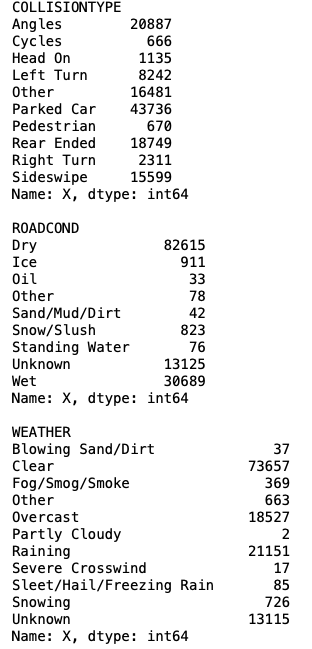
Machine Learning algorithm should be chosen wisely to predict more accurately and give less complicated results i.e results which are easily understandable.

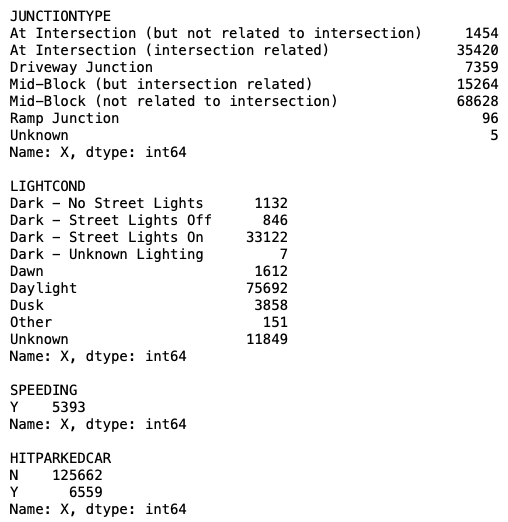
**Data Cleaning**

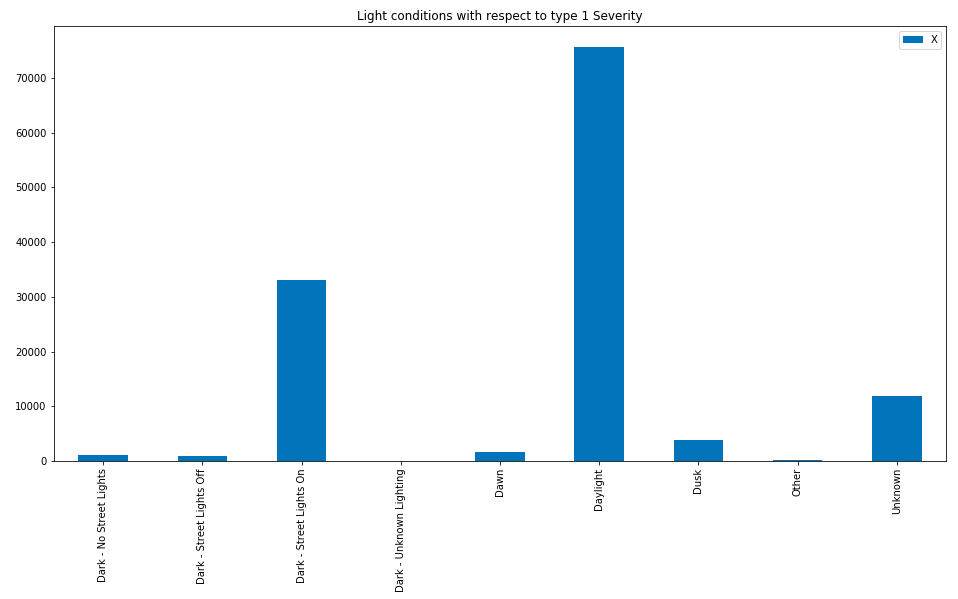
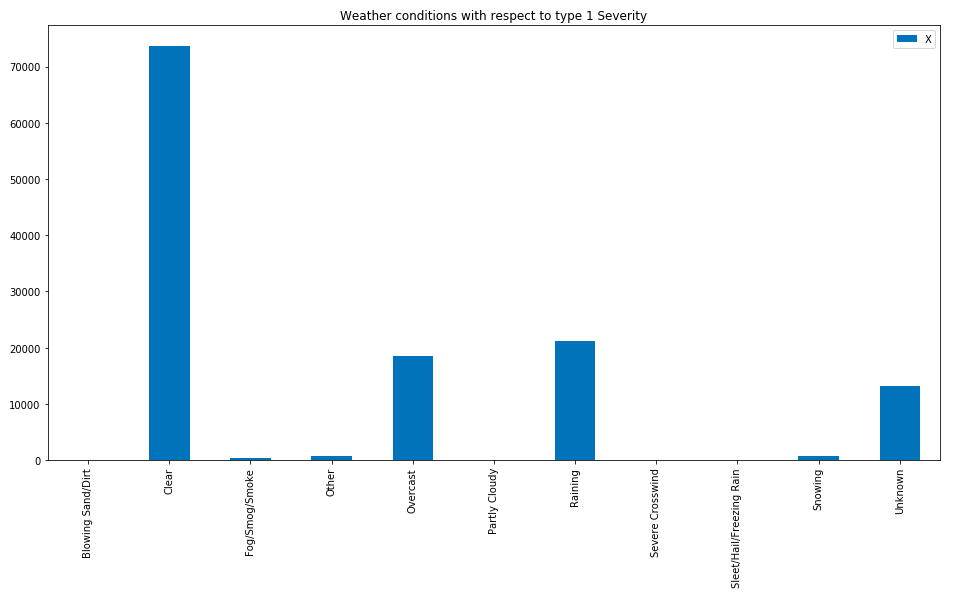
Data is cleaned i.e utilising only required attributes which would help us in predicting severity of accidents.

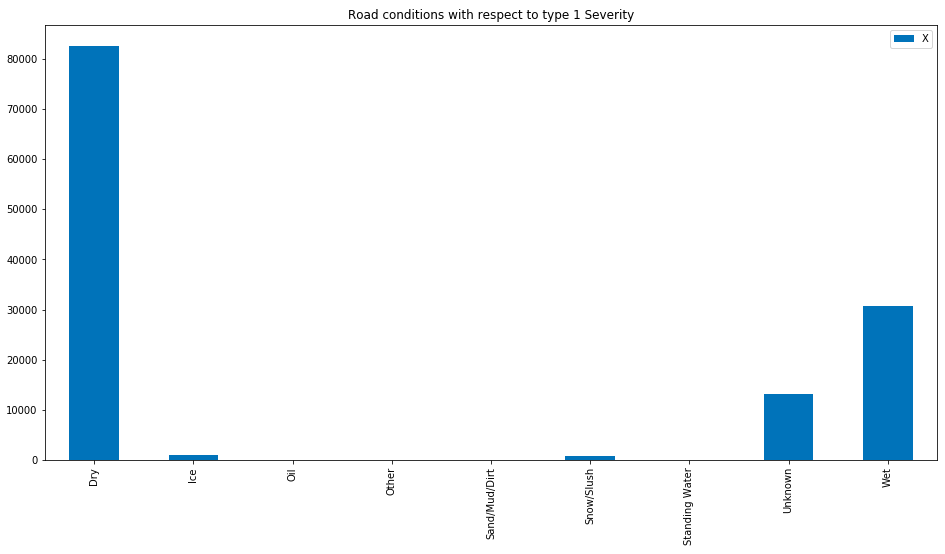
Firstly we select the attributes which would help in predicting the severity and then apply severity to Group by which would then show the columns of respective severity good and would help us in determining our dependent variable. After applying goodbye to the respective columns we print the dataFrame according to that attribute.

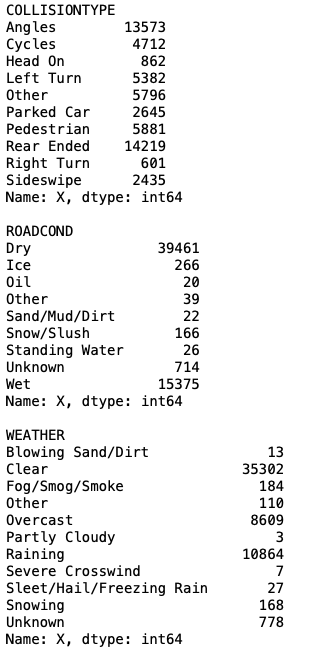
We have also created a pie chart which which would show the percentage which is occupied by a particular security code.

Dataset with severity 1:

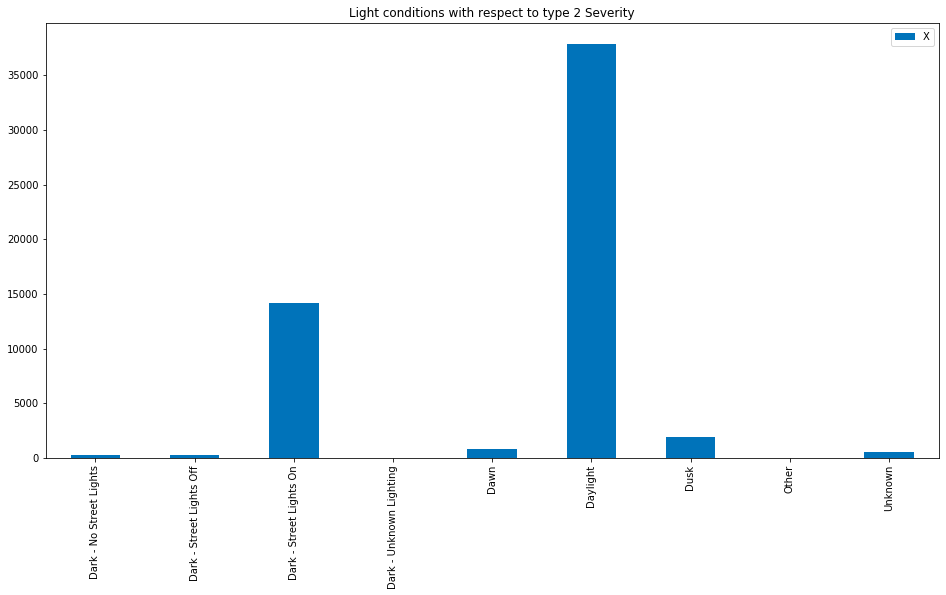
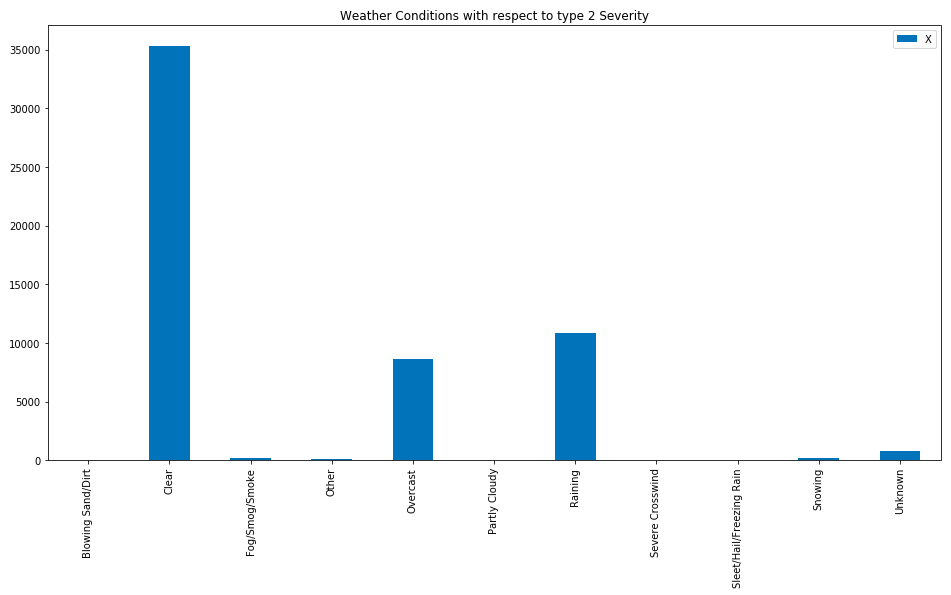
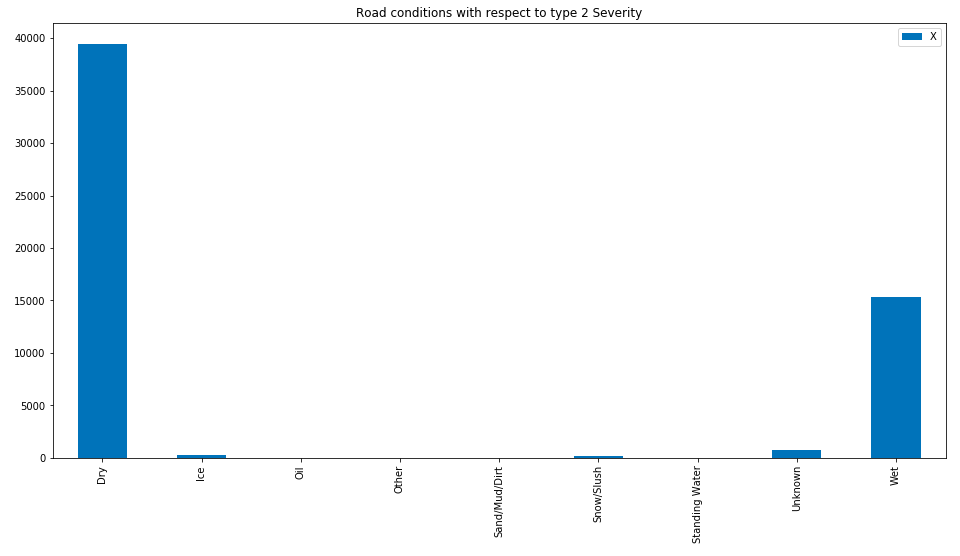


Bar charts showing composition of severity 1:





For Severity type 2:



**Model:**

We apply multiple linear regression model to predict severity type (supervised learning). As we have only 2 different possibilities (if severity is or 2) we need a huge amount of data to the train group (80%) and we let the 20% of out dataset to test the model.

X = df1[["PEDCOUNT","PEDCYLCOUNT","VEHCOUNT","PERSONCOUNT"]]

y = df1[[“SEVERITYCODE"]]

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(X,y, test\_size=0.2, random\_state = 1)

print(x\_train.shape,y\_train.shape)

print(x\_test.shape, y\_test.shape)

(155738, 4) (155738, 1)

(38935, 4) (38935, 1)

from sklearn.linear\_model import LinearRegression

lr = LinearRegression()

lr.fit(x\_train,y\_train)

print('Intercept: \n', lr.intercept\_)

print('Coefficients: \n', lr.coef\_)

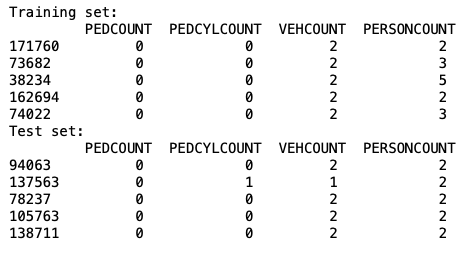
Intercept:

[1.12]

Coefficients:

[[0.59 0.62 0.01 0.05]]

print("Training set:\n",x\_train.head())

print("Test set:\n",x\_test.head())

**Maps:**

!wget --quiet https://data.seattle.gov/api/views/vwtx-gvpm/rows.json -O world\_countries.json

!pip install folium

import folium

seattle\_geo = r'world\_countries.json' # geojson file

# creating a plain world map

world\_map = folium.Map(location=[0, 0], zoom\_start=2, tiles='Mapbox Bright’)

world\_map = folium.Map(location=[47.6062, -122.3321], zoom\_start=14, tiles='OpenStreetMap')

# display map

world\_map

df\_map = df1[["X","Y","SDOT\_COLDESC"]]

df\_map = df\_map.fillna(0)

from folium import plugins

# let's start again with a clean copy of the map of San Francisco

world\_map = folium.Map(location = [47.6062, -122.3321], zoom\_start = 11, tiles='Stamen Toner')

incidents = plugins.MarkerCluster().add\_to(world\_map)

# looping through the dataframe and add each data point to the mark the cluster

for lat, lng, label, in zip(df\_map.Y, df\_map.X, df\_map.SDOT\_COLDESC):

folium.Marker(

location=[lat, lng],

icon=None,

popup=label,

).add\_to(incidents)

world\_map

**Conclusion**

Considering all the results from our model and maps we conclude that:

1. As the dataset had many attributes and data, we use only significant amount of data for our calculations and models.
2. Accidents were distributed randomly and it was not easy to predict.
3. Weather conditions do not contribute for our project as most of the accidents occurred on clear weather.
4. Through our horizontal bar charts we obtain some observations which would help us predict severity of the accidents.