Predicting the probability of getting into a car accident in Seattle City

Francy Martin

September 21,

2020

1. Introduction

1.1 Background

Seattle city has really troubles to deal with the accidents data. When people are driving to another city for work or to visit some friends. It is rainy and windy, and on the way, they could come across a terrible traffic jam on the other side of the highway. Long lines of cars barely moving. As the people keep driving, police car start appearing from afar shutting down the highway. It is an accident and there's a helicopter transporting the ones involved in the crash to the nearest hospital. They must be in critical condition for all of this to be happening. Now, wouldn't it be great if there is something in place that could warn the people, given the weather and the road conditions about the possibility of they getting into a car accident and how severe it would be, so that the people would drive more carefully or even change their travel if they are able to. Well, this is exactly what you will be working on in this.

1.2 Problem

The objective of the project is describing fatality of accidents. Given that the number of accidents on busy roads is creating problems in the city's transportation. It has the necessity to determine under what circumstances a serious traffic accident is most likely to occur.

Among the factors to be evaluated will be the type of crash, and means of transport, the weather and the road conditions about the possibility of you getting into a car accident and how severe it would be.

2. Data cleaning

2.1 Data cleaning

Data The dataset contain 194673 observations (rows) and 38 attributes (columns). First the columns that have NaN were checked. Columns that had more than half like this were eliminated. Columns that contained duplicate information from other columns were also removed.

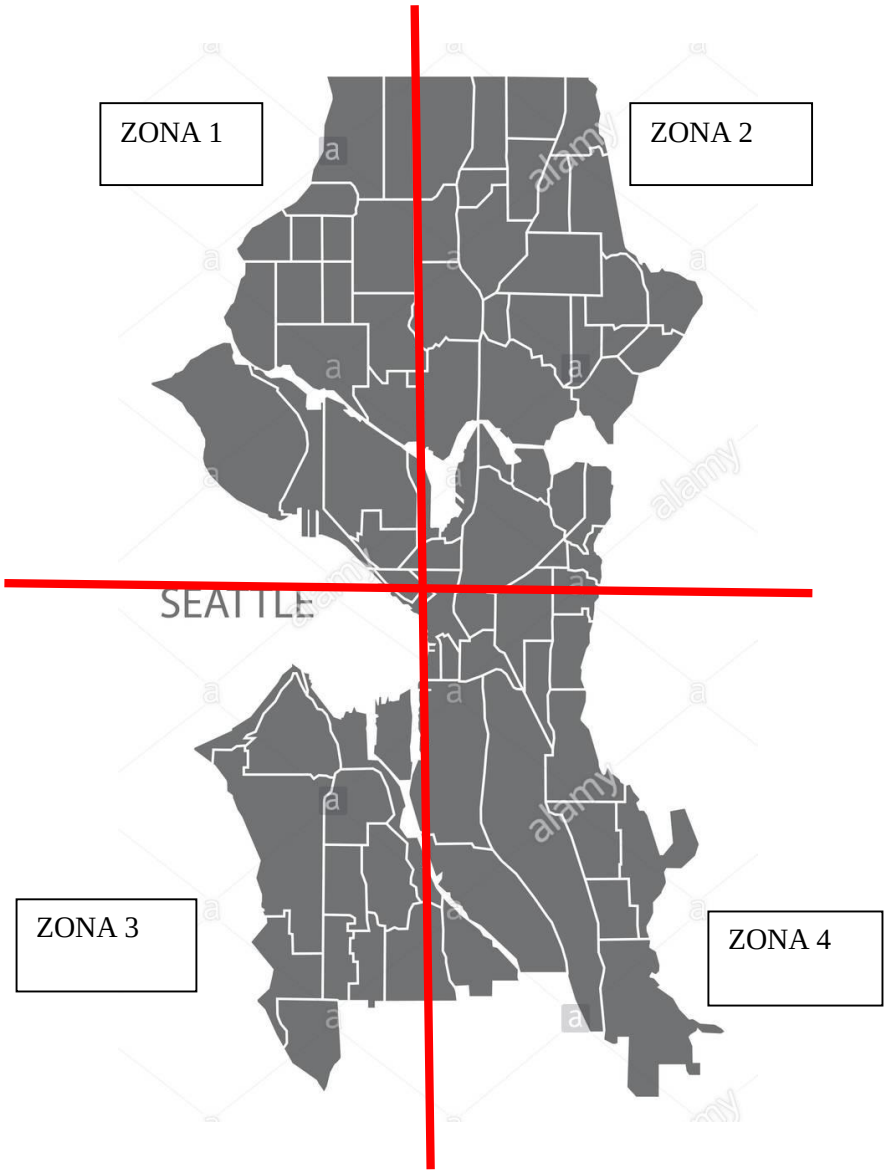
```
df.drop(['INCKEY', #irrelevant
         'OBJECTID', #irrelevant
         'REPORTNO', #irrelevant
         'STATUS', #irrelevant
         'LOCATION', #Duplicate Information,
         'SDOTCOLNUM', #Duplicate information
         'SDOT_COLCODE',
         'SDOT_COLCODE',
         'INTKEY', #irrelevant
         'HITPARKEDCAR',
        'COLDETKEY',
        'EXCEPTRSNCODE',
        'EXCEPTRSNDESC',
        'SEVERITYCODE.1',
        'INATTENTIONIND',
        'SDOTCOLNUM',
        'SPEEDING', #many missing data
        'ST_COLCODE',
        'ST_COLDESC',
        'SEGLANEKEY',
        'ST_COLCODE',
        'CROSSWALKKEY',
        'PEDROWNOTGRNT',
         'UNDERINFL',
         'PERSONCOUNT',
         ], axis = 1, inplace = True)
```

Eliminated Columns

Then the rows that did not have values of the columnhas corresponding to the coordinates, and type of collision were eliminated. All this to eliminate the null and irrelevant values. Finally, it was replaced by the frequency in the columns 'WEATHER' and 'ROADCOND', cause it is most likely to occur.

In addition, the YEAR, MONTH, and TIME columns were created, replacing the previously existing date columns. To see if depending on these the probability of an accident was higher.

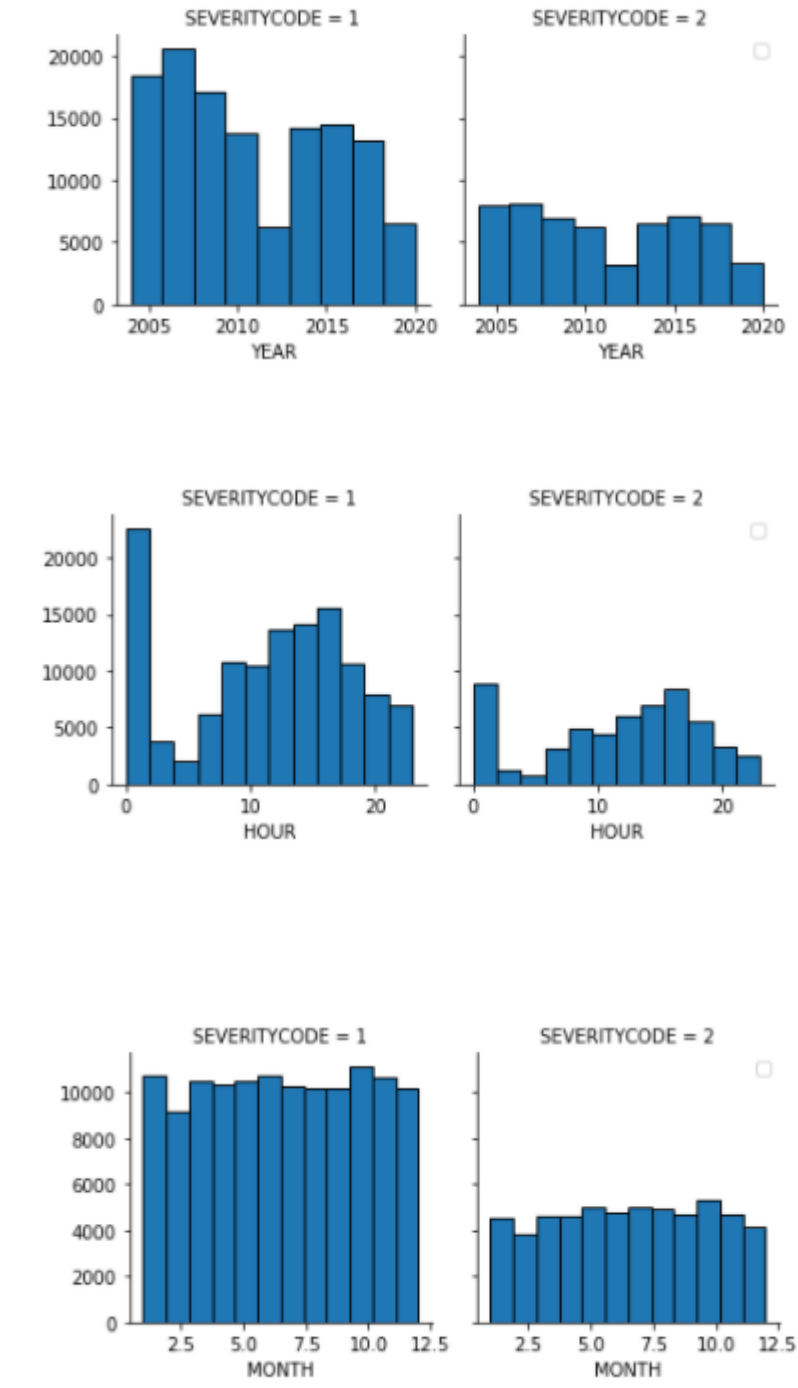
In addition, the coordinates were grouped into four areas as shown in the figure below. In such a way that each data will be grouped in one of these Zones and know which of these is more prone to accidents.



3. Exploratory Data Analysis

3.1 Relationship between 'YEAR', 'MONTH', 'HOUR' and "SEVERITY"

It see that the month of the year does not influence the fatality of the accident. While it is seen that as the years go by, the number of accidents decreases, then only the last 5 years will be taken into account for the study. Finally, a strong relationship is observed between the time of day and the number of accidents.



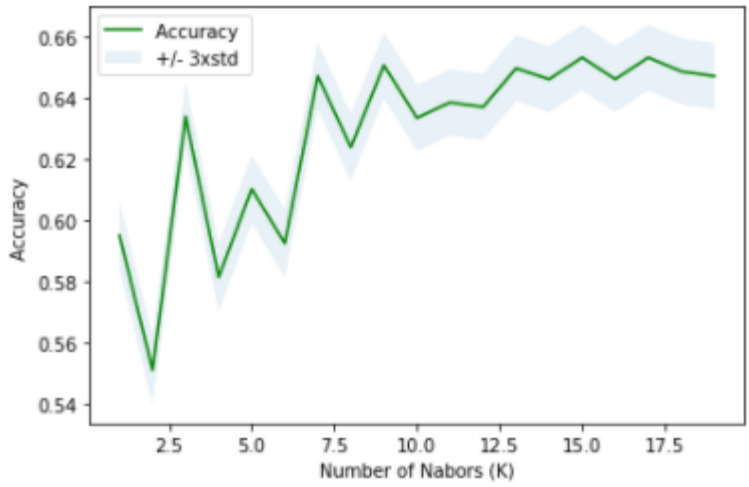
Box plot of improvement of players of different ages.

4. Predictive Modeling

In this case, given that the variables are not continuous, that is, they are discrete. A Classification method had to be used. In this case, K nearest Neighbors was chosen.

4.1 Regression models

4.1.1 K Nearest Neighbor




```
The best accuracy was with 0.6530303030303031 with k= 15
```

jaccard KNN: 0.6530303030303031

F1-score KNN: 0.6259979935509098