**PROJECT REPORT**

**ON**

**ACCIDENT RISK PREDICTION SYSTEM**



Submitted by-

Team 11

Aaradhy Sharma - 001040799

Akshay Khandelwal - 001563326

**Abstract**

This report consists of the final findings of the Supervised Machine Learning algorithms used for the prediction of the occurrence and severity of a road accident in the United States of America.

The dataset used for the project has been taken from Kaggle. The data is then cleaned as per the requirement and then various models were built in Jupyter Notebook (Anaconda 3) to examine the models’ performance on certain parameters.

After a preliminary study of the available algorithms and data review, it became apparent that the problem fell under non-linear Classification category. The study focuses on various algorithms by using classifiers like- Logistic Regression Classification, K-Nearest Neighbor Classification, Support Vector Machine Classification, Decision Tree Classification, Naïve Bayes Classification, and Random Forest Classification.

The major finding is that the machine learning approach should be suitable for this problem due to many aspects, like a total of 49 features in the dataset.

**Table of Contents**

* Introduction
  + Background
  + Motivation
  + Goal
* Methodologies and Algorithms
  + Data Review
  + Software and Libraries used
  + Data Cleaning
  + Feature Selection
  + Exploratory Data Analysis
  + Methods used
    - Logistic Regression
    - K-Nearest Neighbors
    - Support Vector Machine
    - Naïve Bayes
    - Decision Tree
    - Random Forest
  + Feature Engineering
  + Feature Selection
* Description of Dataset
* Results and Analysis
* Conclusion
* Reference

**Introduction**

**Background**

Accident Risk Prediction system aims to predict the occurrence of a probable accident based on the data of past occurred Accidents in the United States of America from year 2016 to 2020. This system has ‘Severity’ as the target variable and its predictions are based on a variety of intrinsic and contextual attributes such as location, time, natural language description, weather, period-of-day, and points-of-interest.

**Motivation**

Reducing traffic accidents is an important public safety challenge. Though there are many studies already available on traffic accident analysis and prediction. But we found those working on small-scale datasets that do not provide broad coverage of impact and applicability. To get the correct analysis- it is important to include vital contextual information such as environmental stimuli and terrain knowledge. Thus, we decided to work on this project including all essential factors affecting the risk of accidents, so to predict and prevent a probable accident, and eventually save lives.

**Goal**

We wish to present this system to predict occurrence of a traffic accident, by providing risk analysis in simpler visualizations formats for quick and easy understanding. Our goal is to get helpful results like some mentioned below-

* Hotspot regions with most numbers of traffic accidents in a season/time.
* Change in rate of accident occurrence on factors of environmental stimuli.
* Factors affecting the severity of the accidents.

**Methodology & Algorithm**

**Data Review**

The dataset containing records of all the states of the USA had 3.5 million records (1GB size), so the model is applied on three subsets of dataset after data cleaning, these are-

* All records of the Dataset
* Records for the State of California
* Records for the State of South Carolina

**Software and Libraries used**

The dataset is downloaded from Kaggle. And Jupyter notebook is used with several Libraries

That are-

* Numpy
* Pandas
* Matplotlib
* Seaborn
* Metrics
* Mean Squared Error
* Scikitlearn Tree
* DecisionTreeClassifier
* train\_test\_split
* Standard Scaler
* r2\_score
* accuracy\_score
* sklearn.ensemble
* RandomForestClassifier
* plotly.graph\_objects
* sklearn.preprocessing
* LabelEncoder
* OneHotEncoder

And several others for classification models.

**Data Cleaning**

Data provided was highly heterogenous with many missing values as data is from the year 2016 to 2020. Therefore, many rows had NAN values which can compromise the models, so to sort it, all the rows with NAN values are removed as it is a huge dataset with many records.

**Feature Selection**

#### The data originally had 45+ columns this can lead to too much noise in the database many of those columns have empty values too so to prevent it from disturbing the database final model is built on a data frame with 28 columns.

**Exploratory Data Analysis**

After Cleaning the data and sorting it we have done feature selection on it using various pandas and matplotlib commands. Firstly, we have shown a histogram of all relevant columns. using hist() then other correlation heatmaps are formed then several other attributes are represented using matplotlib. Such as Severity of accident by year, City most prone to accident etc.

#### **Models Used**

The Dependent Target Variable is Severity which contains numerical digits from 0 to 4, which meaning-

* Severity Class 0 - no accident, or no injury
* Severity Class 1 - accident less severe
* Severity Class 2 - a severe accident
* Severity Class 3 - a severe accident with loss of live
* Severity Class 4 - accident with loss of lives and system failure

As the Target Variable has 5 classes, the prediction model used is Classification.

The dataset with its matrix of features (independent variables) is trained on various Classification models to predict the class value of the dependent target variable.

Different types of classifiers used in the project are-

* Logistic Regression Classification
* Support Vector Machine Classification
* Naïve Bayes Classification
* K-Nearest Neighbor Classification
* Decision Tree Classification
* Random Forest Classification

**Logistic Regression Classification**

This classifier is used to transform its output using the logistic sigmoid function to return a probability value. Logistic Regression was not much success with the accuracy of the prediction of the target value of our dataset.

**Support Vector Machine Classification**

It is a type of supervised learning algorithm which use**s** kernel trick to transform data and then based on these transformations it finds an optimal boundary between the possible outputs. As this classifier takes a lot of computing time, thus we have done training on limited number of records (20000) due to the limitation of the personal laptop.

**Naïve Bayes Classification**

This classifier is based upon Naïve Bayes Theorem which works on strong independence assumptions between the features. As it works best with a linear dataset, we didn’t get good results with this classifier.

**K-Nearest Neighbor Classification**

This classifier predicts the probability upon checking the K number of nearest neighbors. Applying this algorithm gave a promising accuracy score on our dataset.

**Decision Tree Classification**

Decision Tree is used in all three subprojects that are California, South Carolina and entire USA. Decision tree is highly useful in classification problems where the total number of features is very high. As well as total number of rows is high too. A decision tree is represented as upside down where its root is at the top of the tree then it splits into branches and when it cannot further split then the end branch is called as decision. Growing a decision tree requires to choose features and conditions to select optimal tree which has maximum prediction. The tree is grown arbitrary.

**Random Forest Algorithm Classification**

Random forest is on par with decision tree in terms of getting result both have given satisfactory results. With accuracy score reaching as high as 88%.

The random forest is flexible algorithm which is easy to use and take very less time as compared the likes of SVM. In decision tree only one tree is made but in random forest, our algorithm randomly creates a specified number of decision trees. And chooses tree which is best for our model.

#### **Feature Selection**

We selected features that have the maximum contribution to our output by this way we can reduce the compute time of our model. The more details on Feature selection are in the section of Data Preprocessing in this report.

**Visualization**

Visualization is done at classification models as well as EDA. Where bar graph, line graphs, pie charts etc. are used to represent data as well as data frames have been used Matplotlib,Seaborn and Plotly.graph are used.

**Dataset**

The Dataset consists information of accident happened in 49 states of the United States of America From February 2016 to June 2020. The total number of accidents recorded in the given dataset is about 3.5 million.

The dataset consists of one .csv file which is around 1.24 GB and has total 49 Attributes,

Of which:

* 17 attributes are String,
* 13 attributes are Boolean,
* 12 attributes are Decimal,
* And 7 are others.

List of Attributes:

|  |  |  |  |
| --- | --- | --- | --- |
| S.no | Attributes | Description | Nullable |
| 1 | ID | This is a unique identifier of the accident record. | No |
| 2 | Source | Indicates source of the accident report (i.e. the API which reported the accident.). | No |
| 3 | TMC | A traffic accident may have a [Traffic Message Channel (TMC)](https://wiki.openstreetmap.org/wiki/TMC/Event_Code_List) code which provides more detailed description of the event. | Yes |
| 4 | Severity | Shows the severity of the accident, a number between 1 and 4, where 1 indicates the least impact on traffic (i.e., short delay as a result of the accident) and 4 indicates a significant impact on traffic (i.e., long delay). | No |
| 5 | Start\_Time | Shows start time of the accident in local time zone. | No |
| 6 | End\_Time | Shows end time of the accident in local time zone. End time here refers to when the impact of accident on traffic flow was dismissed. | No |
| 7 | Start\_Lat | Shows latitude in GPS coordinate of the start point. | No |
| 8 | Start\_Lng | Shows longitude in GPS coordinate of the start point. | No |
| 9 | End\_Lat | Shows latitude in GPS coordinate of the end point. | Yes |
| 10 | End\_Lng | Shows longitude in GPS coordinate of the end point. | Yes |
| 11 | Distance(mi) | The length of the road extent affected by the accident. | No |
| 12 | Description | Shows natural language description of the accident. | No |
| 13 | Number | Shows the street number in address field. | Yes |
| 14 | Street | Shows the street name in address field. | Yes |
| 15 | Side | Shows the relative side of the street (Right/Left) in address field. | Yes |
| 16 | City | Shows the city in address field. | Yes |
| 17 | County | Shows the county in address field. | Yes |
| 18 | State | Shows the state in address field. | Yes |
| 19 | Zipcode | Shows the zipcode in address field. | Yes |
| 20 | Country | Shows the country in address field. | Yes |
| 21 | Timezone | Shows timezone based on the location of the accident (eastern, central, etc.). | Yes |
| 22 | Airport\_Code | Denotes an airport-based weather station which is the closest one to location of the accident. | Yes |
| 23 | Weather\_Timestamp | Shows the time-stamp of weather observation record (in local time). | Yes |
| 24 | Temperature(F) | Shows the temperature (in Fahrenheit). | Yes |
| 25 | Wind\_Chill(F) | Shows the wind chill (in Fahrenheit). | Yes |
| 26 | Humidity(%) | Shows the humidity (in percentage). | Yes |
| 27 | Pressure(in) | Shows the air pressure (in inches). | Yes |
| 28 | Visibility(mi) | Shows visibility (in miles). | Yes |
| 29 | Wind\_Direction | Shows wind direction. | Yes |
| 30 | Wind\_Speed(mph) | Shows wind speed (in miles per hour). | Yes |
| 31 | Precipitation(in) | Shows precipitation amount in inches, if there is any. | Yes |
| 32 | Weather\_Condition | Shows the weather condition (rain, snow, thunderstorm, fog, etc.) | Yes |
| 33 | Amenity | A [POI](https://wiki.openstreetmap.org/wiki/Points_of_interest) annotation which indicates presence of [amenity](https://wiki.openstreetmap.org/wiki/Key:amenity) in a nearby location. | No |
| 34 | Bump | A POI annotation which indicates presence of speed bump or hump in a nearby location. | No |
| 35 | Crossing | A POI annotation which indicates presence of [crossing](https://wiki.openstreetmap.org/wiki/Key:crossing) in a nearby location. | No |
| 36 | Give\_Way | A POI annotation which indicates presence of [give\_way](https://wiki.openstreetmap.org/wiki/Tag:highway%3Dgive_way) in a nearby location. | No |
| 37 | Junction | A POI annotation which indicates presence of [junction](https://wiki.openstreetmap.org/wiki/Key:junction) in a nearby location. | No |
| 38 | No\_Exit | A POI annotation which indicates presence of [no\_exit](https://wiki.openstreetmap.org/wiki/Key:noexit) in a nearby location. | No |
| 39 | Railway | A POI annotation which indicates presence of [railway](https://wiki.openstreetmap.org/wiki/Key:railway) in a nearby location. | No |
| 40 | Roundabout | A POI annotation which indicates presence of [roundabout](https://wiki.openstreetmap.org/wiki/Tag:junction%3Droundabout) in a nearby location. | No |
| 41 | Station | A POI annotation which indicates presence of [station](https://wiki.openstreetmap.org/wiki/Key:station) in a nearby location. | No |
| 42 | Stop | A POI annotation which indicates presence of [stop](https://wiki.openstreetmap.org/wiki/Key:stop) in a nearby location. | No |
| 43 | Traffic\_Calming | A POI annotation which indicates presence of [traffic\_calming](https://wiki.openstreetmap.org/wiki/Key:traffic_calming) in a nearby location. | No |
| 44 | Traffic\_Signal | A POI annotation which indicates presence of [traffic\_signal](https://wiki.openstreetmap.org/wiki/Tag:highway%3Dtraffic_signals) in a nearby location. | No |
| 45 | Turning\_Loop | A POI annotation which indicates presence of [turning\_loop](https://wiki.openstreetmap.org/wiki/Tag:highway%3Dturning_loop) in a nearby location. | No |
| 46 | Sunrise\_Sunset | Shows the period of day (i.e. day or night) based on sunrise/sunset. | Yes |
| 47 | Civil\_Twilight | Shows the period of day (i.e. day or night) based on [civil twilight](https://en.wikipedia.org/wiki/Twilight#Civil_twilight). | Yes |
| 48 | Nautical\_Twilight | Shows the period of day (i.e. day or night) based on [nautical twilight](https://en.wikipedia.org/wiki/Twilight#Nautical_twilight). | Yes |
| 49 | Astronomical\_Twilight | Shows the period of day (i.e. day or night) based on [astronomical twilight](https://en.wikipedia.org/wiki/Twilight#Astronomical_twilight). | Yes |

**Data Source**

The dataset has been taken from Kaggle:

US Accidents (3.5 million records) A Country wide Traffic Accident Dataset (2016-2020) <https://www.kaggle.com/sobhanmoosavi/us-accidents>

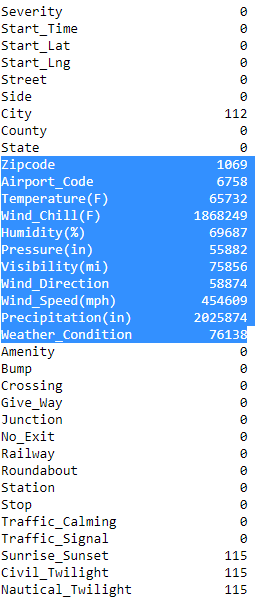
**Analysis**

**Data Exploration**

The given dataset has a large number of rows that is 3.5 million and 49 columns. Our aim was to predict severity of accident which is how long traffic is stop due to the accident, this can predict how worst was the accident. On Initial inspection, the huge number of values in attribute were actually null so to sort this we used data cleaning and feature selection along with correlation matrix to get understanding of the data. We have dropped several columns and rows and our final data has 28 columns as compare to original 49 columns.

**Data Cleaning and Feature Selection**

Firstly we found column with zero values



To sort This, we remove some columns as well as dropped nan value by command drop.na ()

Then correlation matrix is formed with data

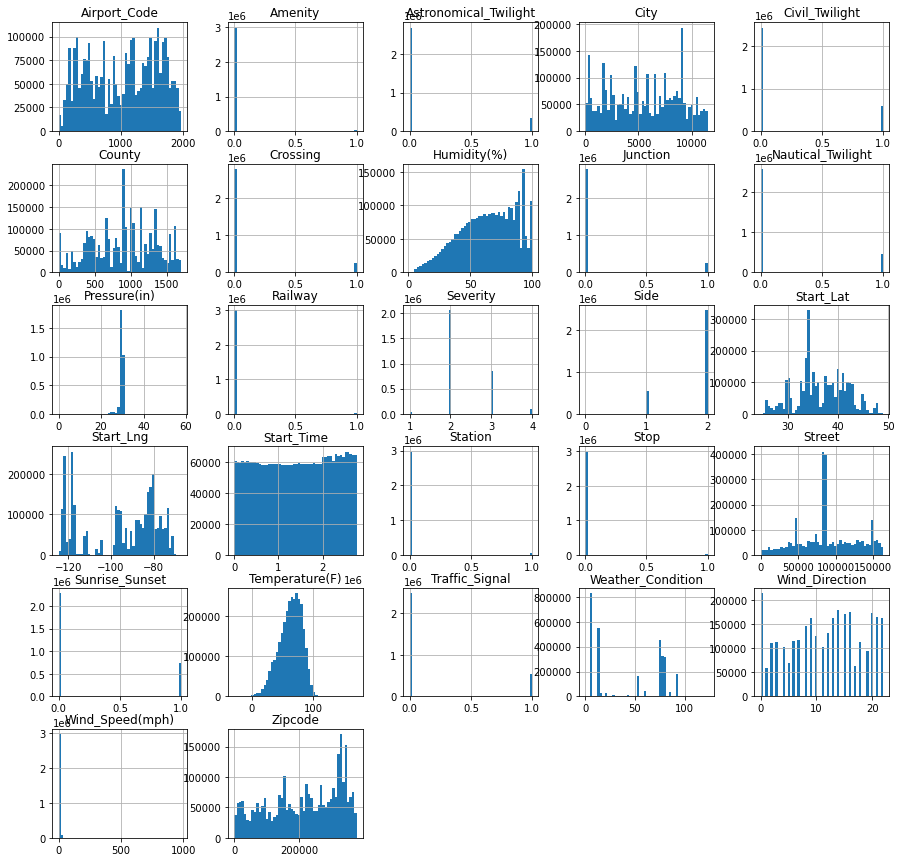


Then least correlated attributes are dropped. That is attribute whose value is closer to zero.

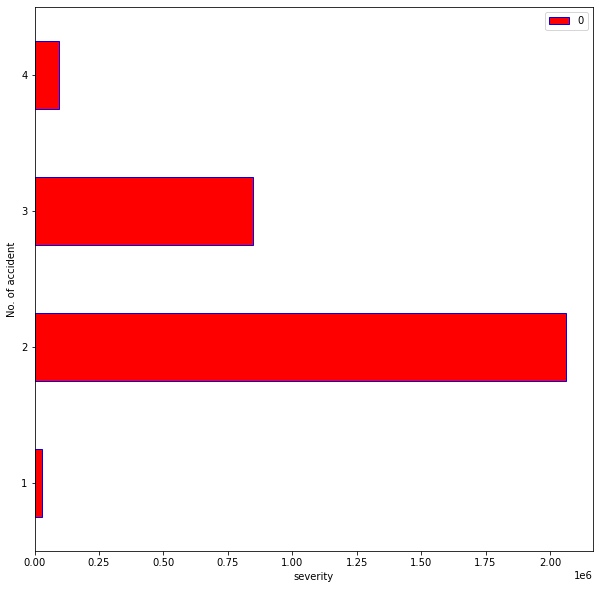
Then after checking for nan we apply one hot coding to get data in range.

**EDA**

Exploratory data analysis has we have made a Histogram to see all columns and there effects

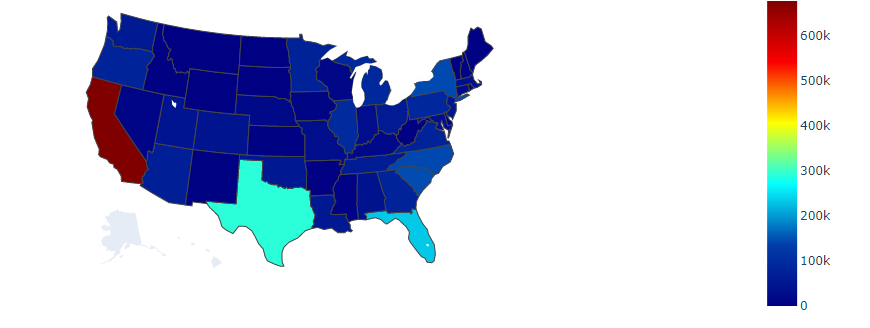
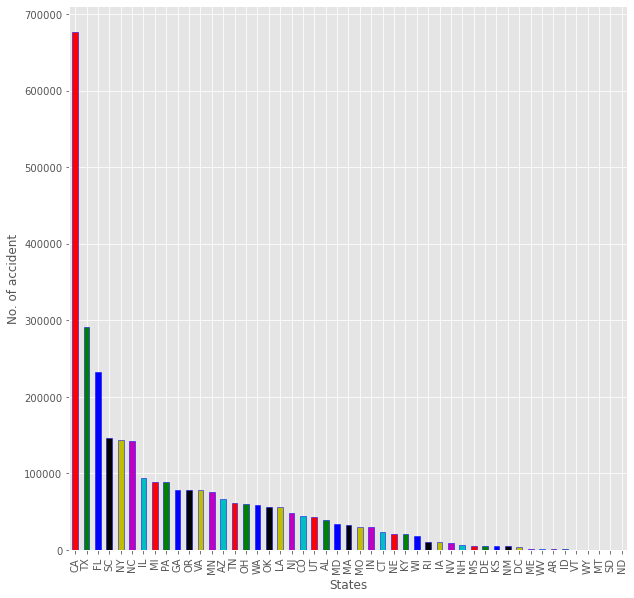


Histogram shows value change in columns across data.

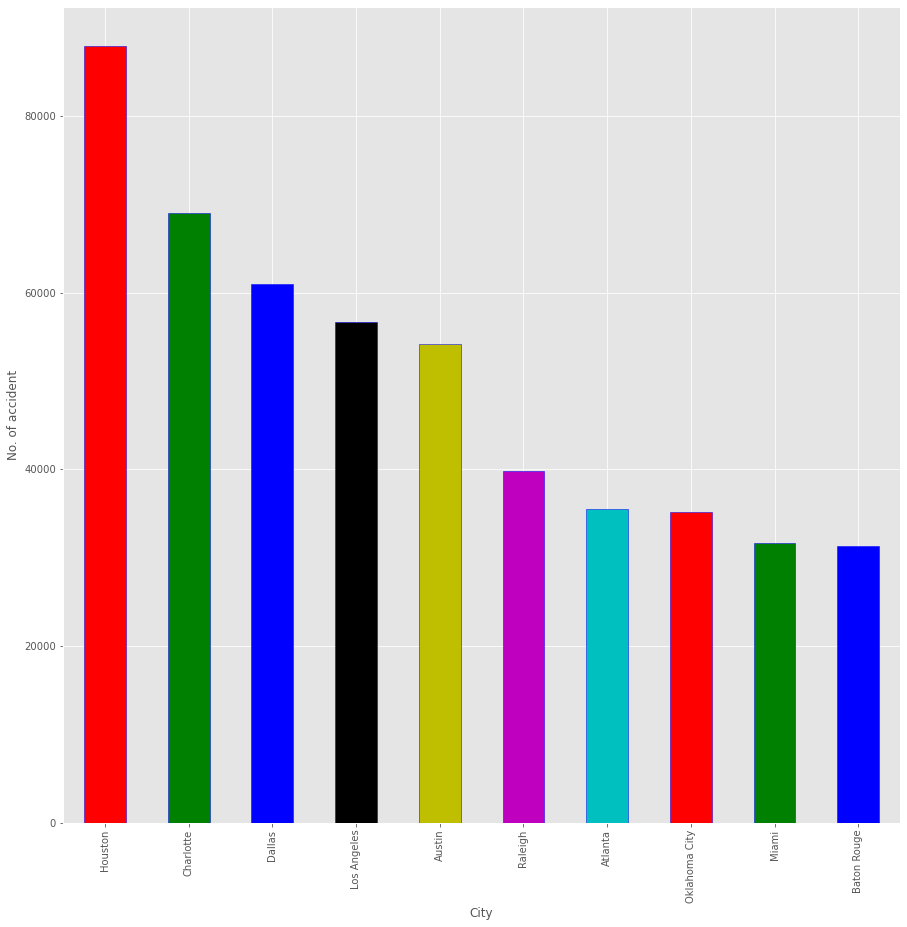


Given image Shows Severity of accident across USA. I can be noted as most of the accident occur in criteria 2 where traffic is stop for a while accident 4 are generally worse as traffic is stop for most time.

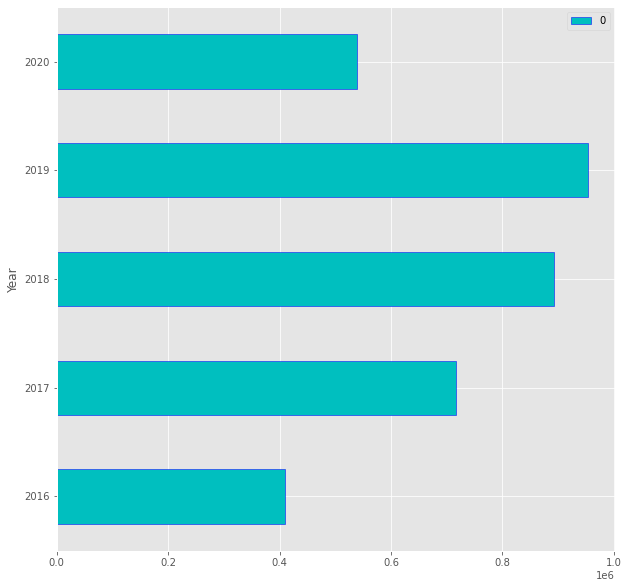
State Bar plot on no. of accidents



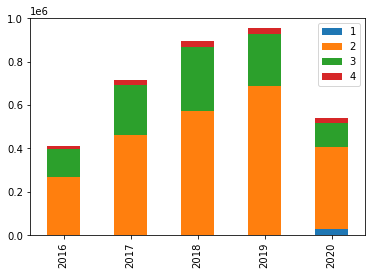
USA state wise graph of accidents and map



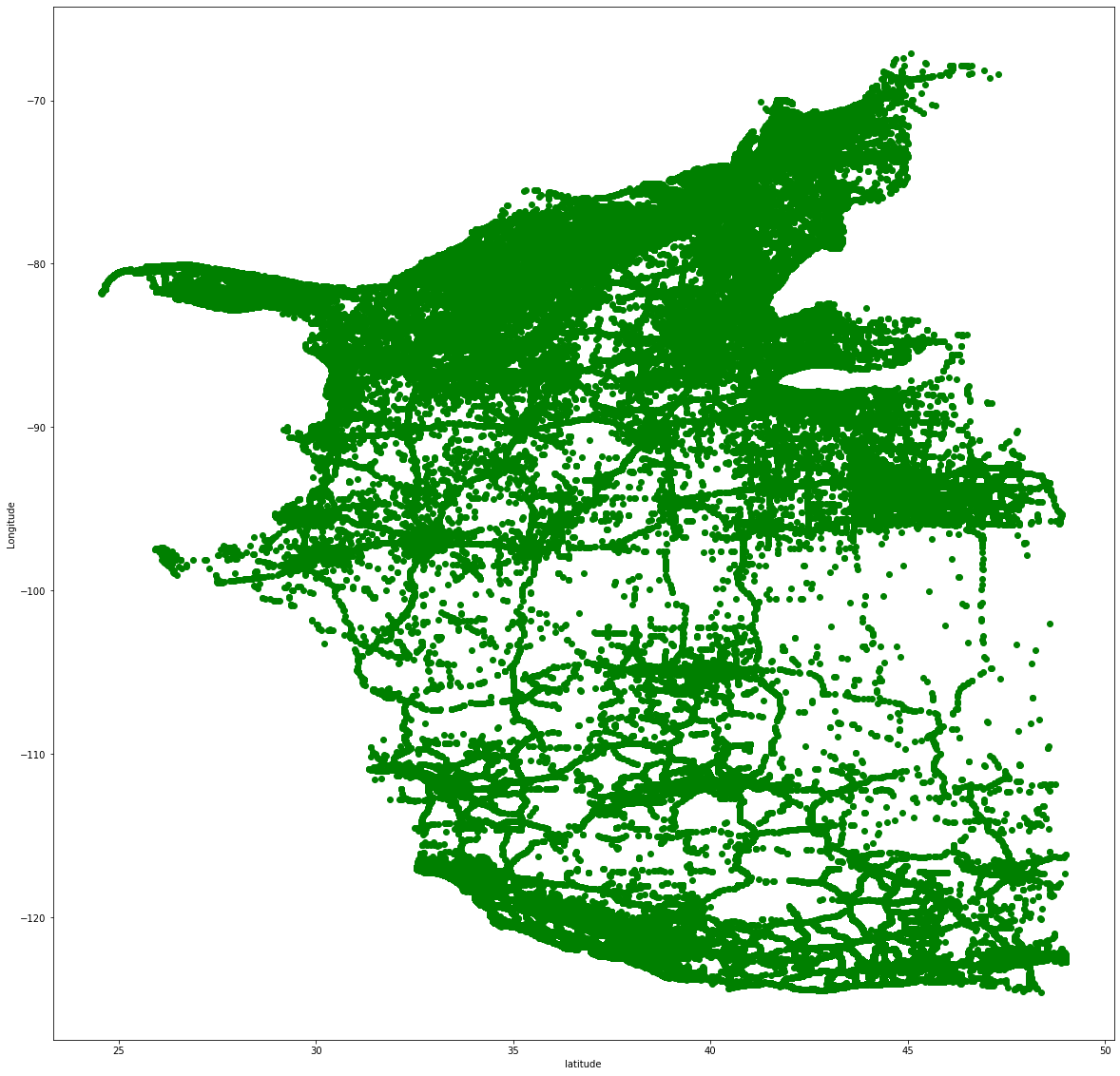
Most Accident-Prone City In USA as in given figure it is clear that Houston has most no. of accident as compare any other city in usa



In the above Graph it is looking Like accident in 2020 has decreased but our data is only till may 2020. And also in 2020 there was lockdown due to Covid-19, hence no. of accident is low.



Accident in years by severity in given years as the graph depict accident with severity level 2 is highest over the years.



Longitude and Latitude are ploted this gives us general idea of accident locations.

To get the best model we apply the model to the state of California, South Carolina and the entire USA excluding Alaska

**Data Preprocessing**

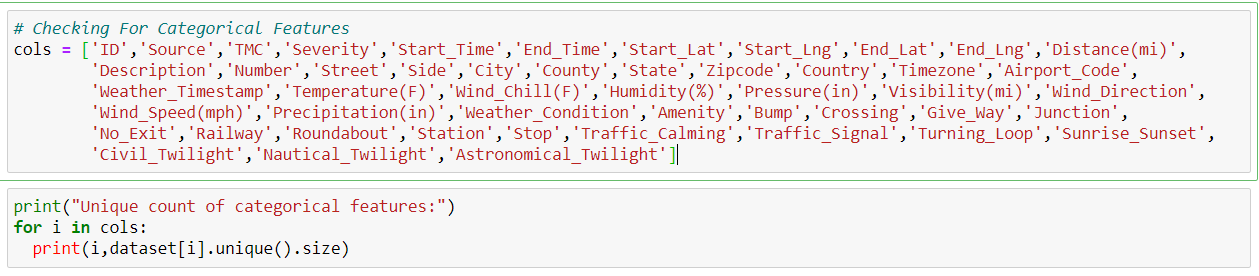
* Dropping useless Features

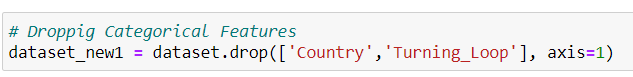
The Features that are not governing a direct relationship with the prediction of the Target Variable are dropped. Feature ‘ID' doesn’t provide any information about accident’s occurrence. Features ‘TMC', ‘Source', ‘Number', ‘End\_Lat’, ‘End\_Lng’, ‘Description’, ‘Distance(mi)’, ‘End\_Time’, ‘Timezone’, ‘Weather\_Timestamp’ can only be collected only after the accident has already happened and hence cannot be predictors for serious accident prediction. Moreover, we have Start Time feature in the dataset so thus no need of End Time and Duration. Same way, having Start Location Feature in the dataset make End Location Feature useless.

U,{03f246d2-c10b-4d49-bc36-c187ddb57ba2}{67},10,6.666666666666667

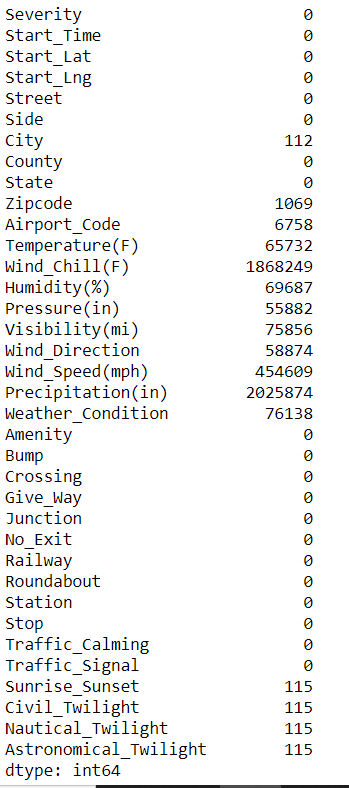
* **Dropping Categorical Features**

On finding the number of unique values in all the Features,

  
The outcome shows one unique value in all the records for the Feature ’Country‘ (that is USA, as the data is of the USA), and for Feature ’Turning\_Loop’ it is always False. Thus such Features have same value aren’t of our use and thus dropped from the dataset.



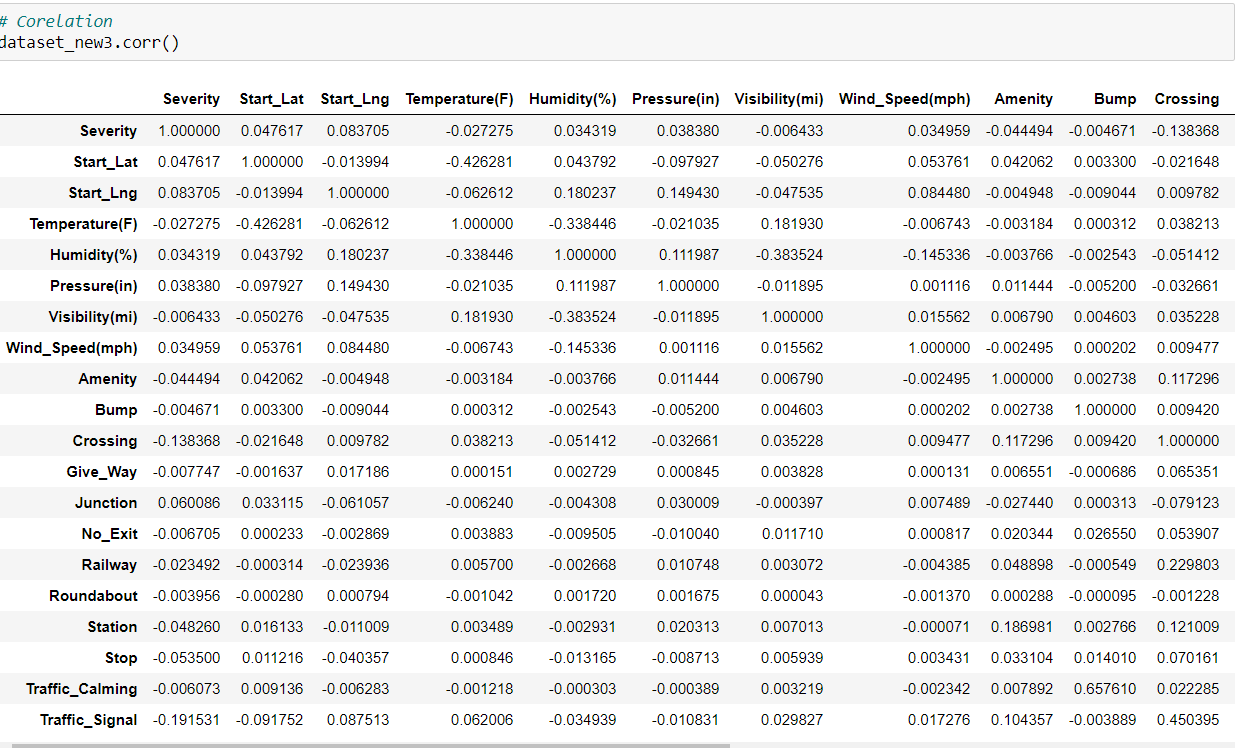
* Dropping Features with Maximum Missing Data

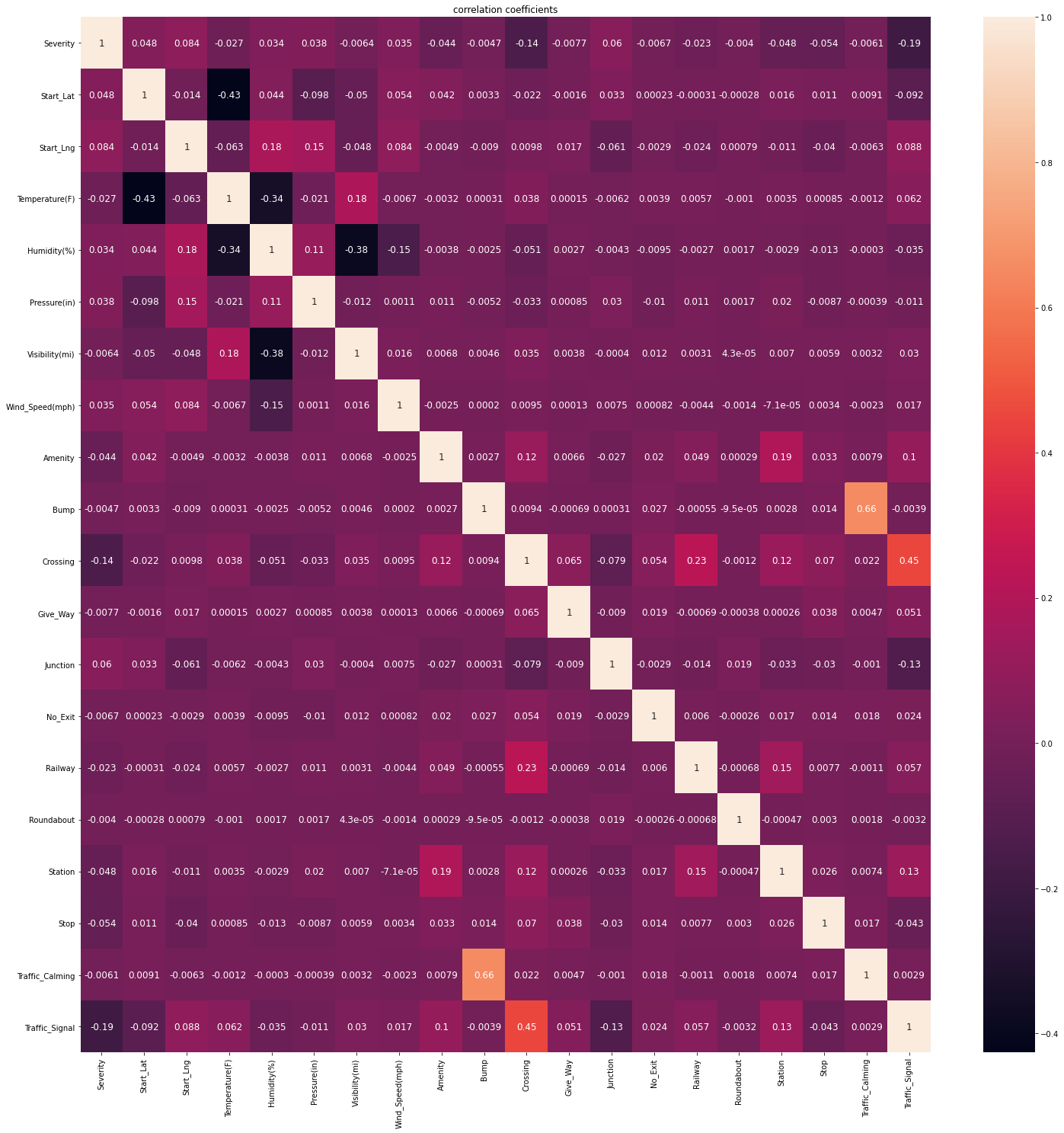


It was found that the Feature Wind\_Chill(F) has 1868249, and Precipitation(in) has 2025874 missing data. As these numbers are big, these two Features are dropped from the dataset.



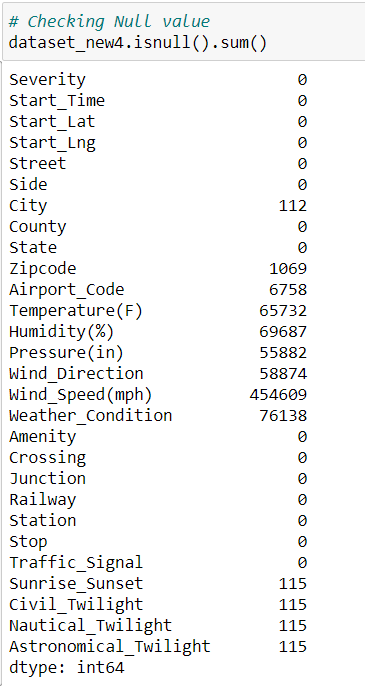
* Dropping Features having least Correlation with the Target







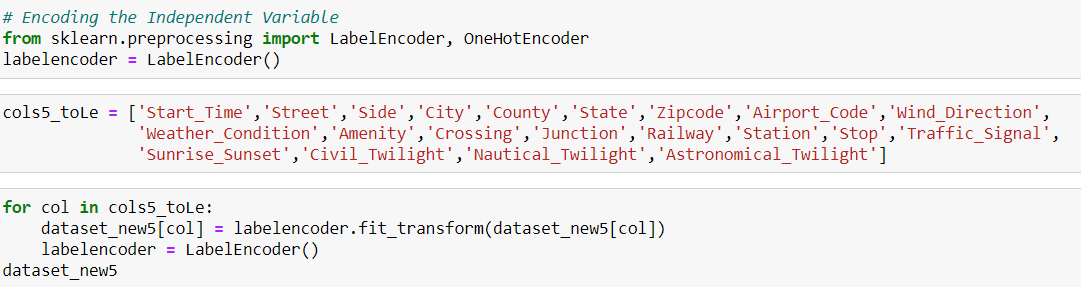
* Dropping records with Null value

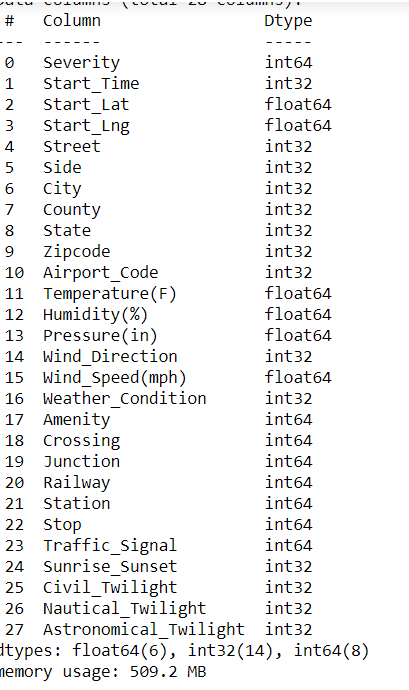


Drop Records having null values

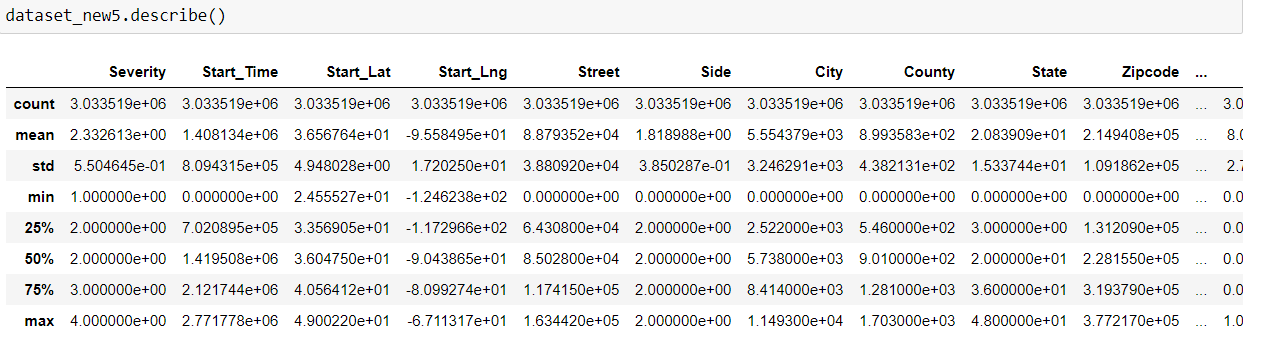


* Encoding the Independent Variable

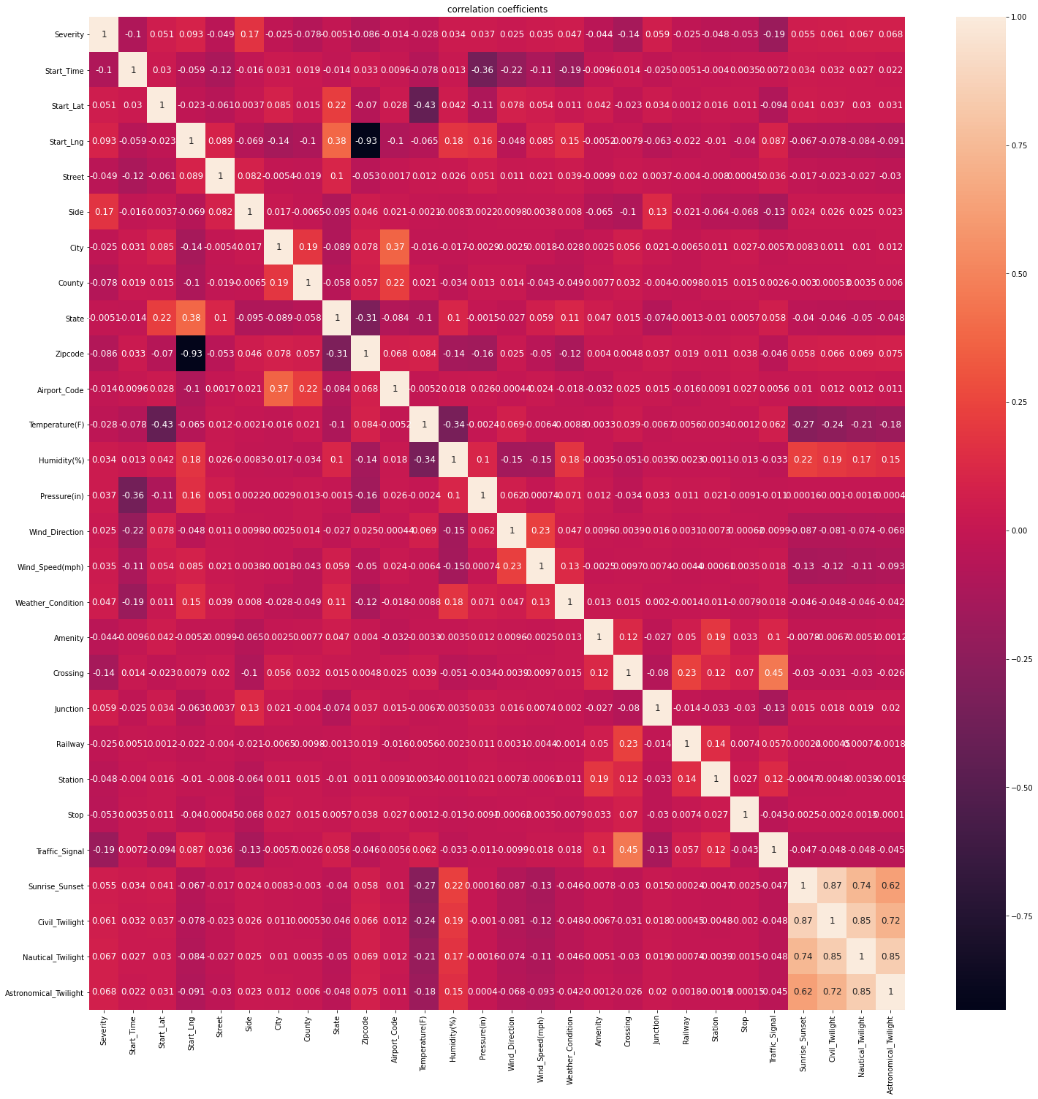




Description of the Dataset



Correlation after Data preprocessing and data cleaning

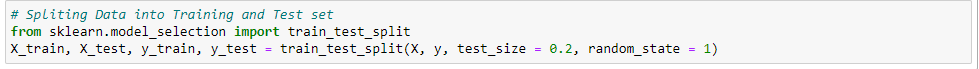


**Predict the accident’s Severity**

**Training and Testing Data**

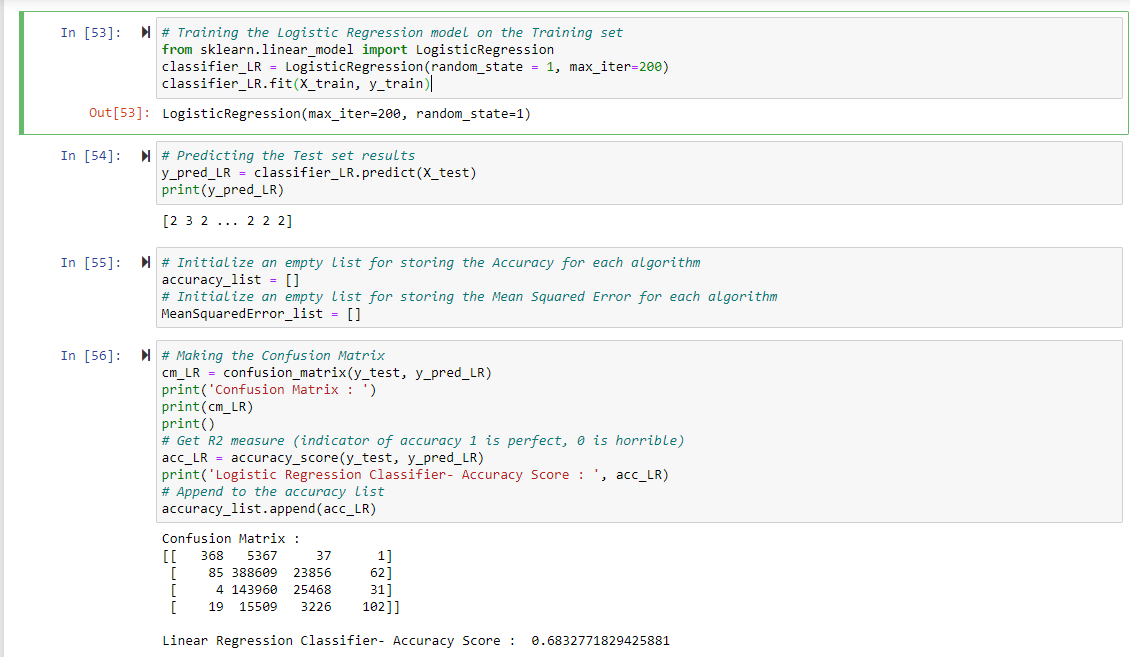
Data is separated in Training set as well as Testing set in ratio 10:2 to 10:3

And then classification models are applied on it

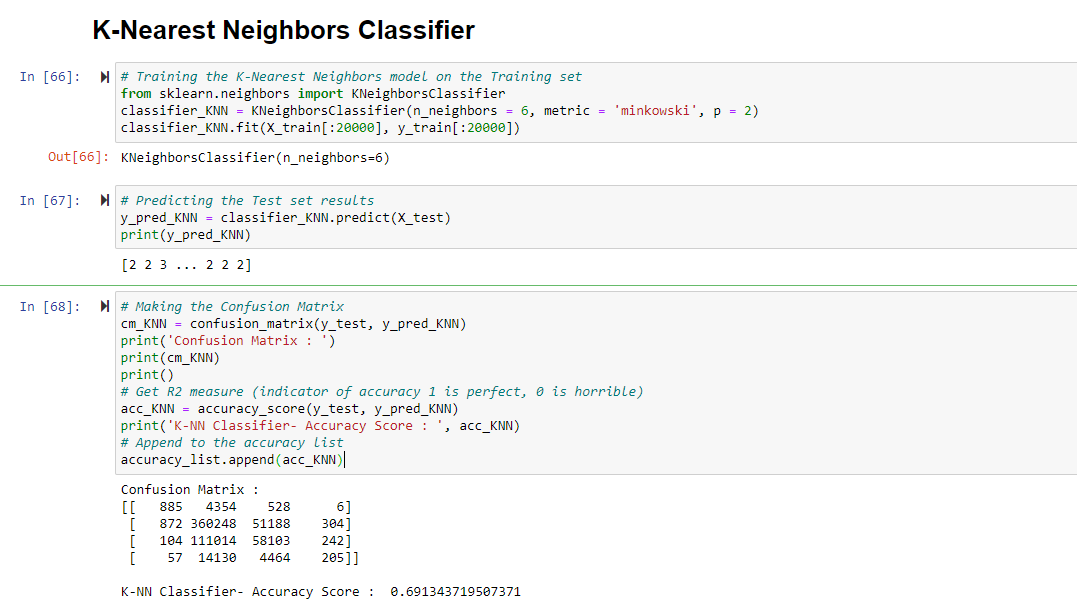


Model with dataset of Entire USA

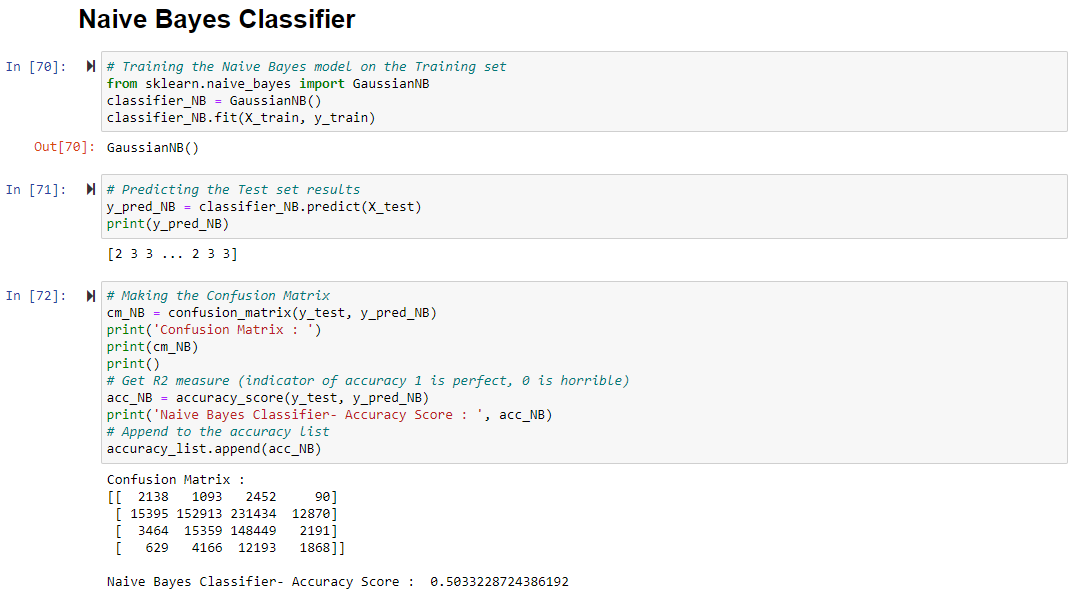
Logistic Regression Classification-



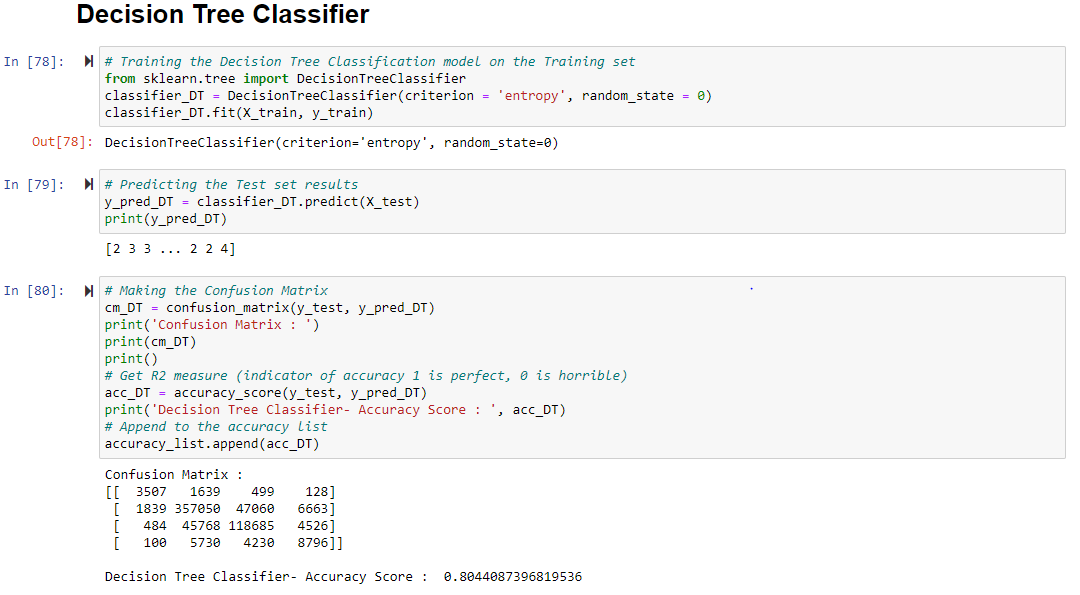
K-Nearest Neighbors Classification



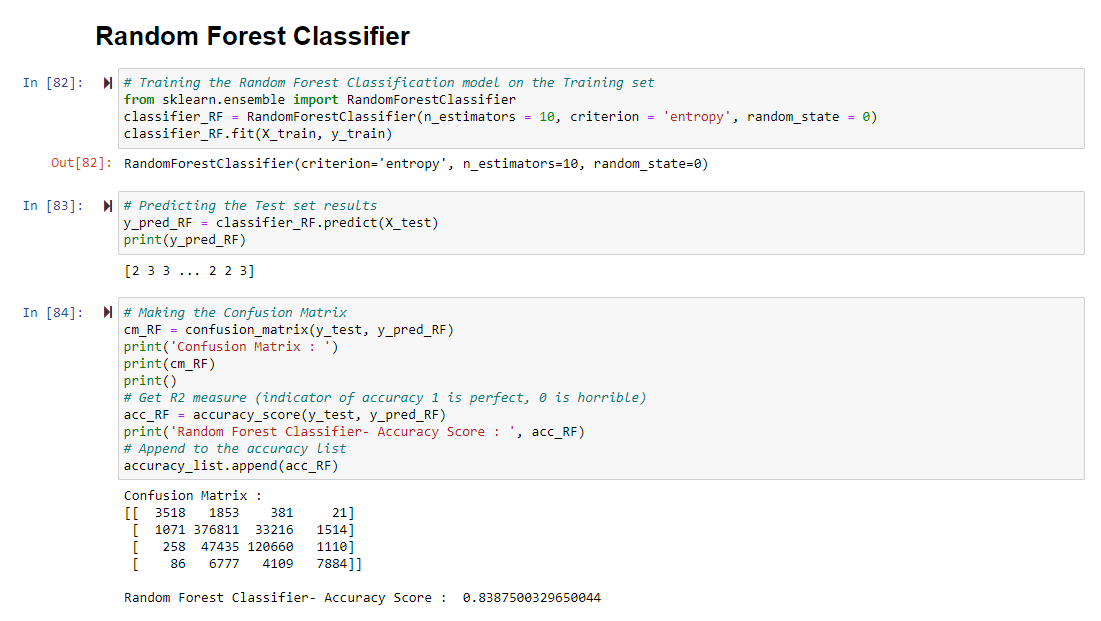
Naïve Bayes Classification



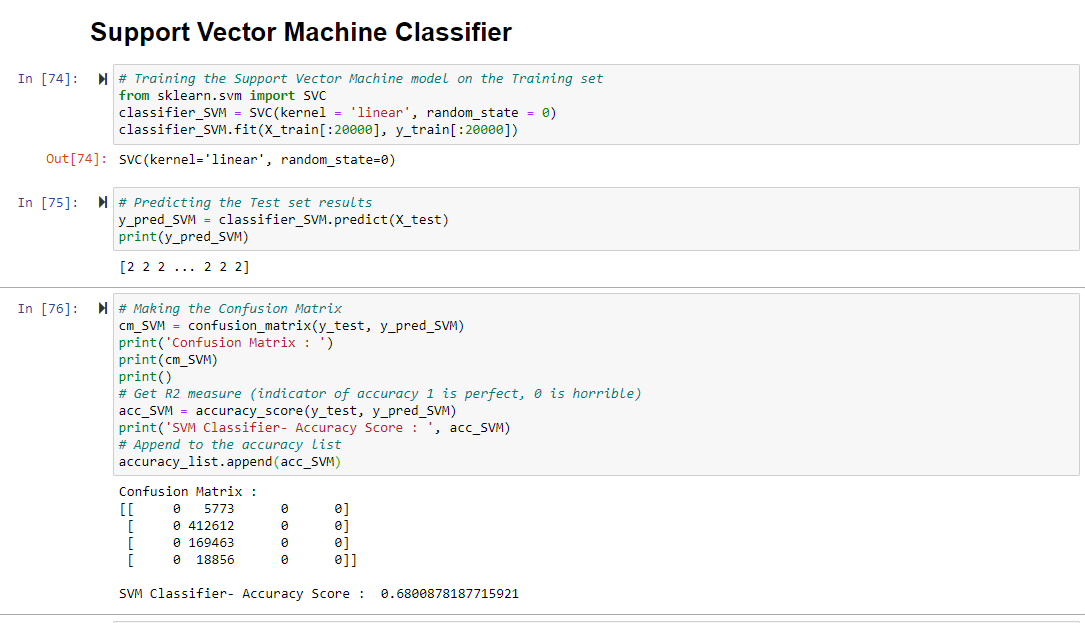
Decision Tree Classification



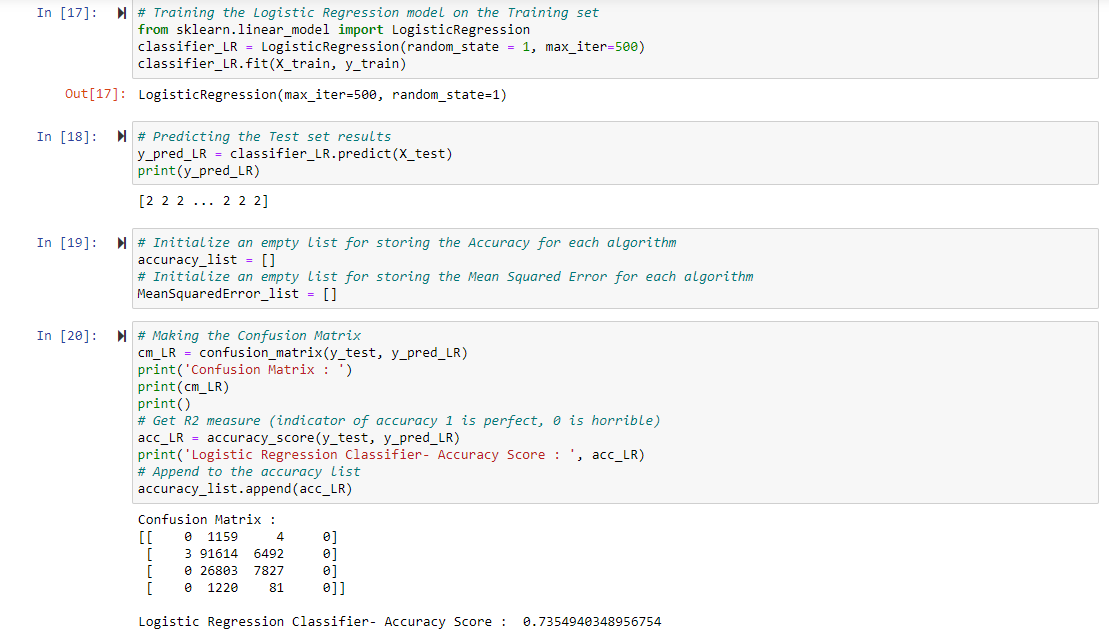
Random Forest Classification



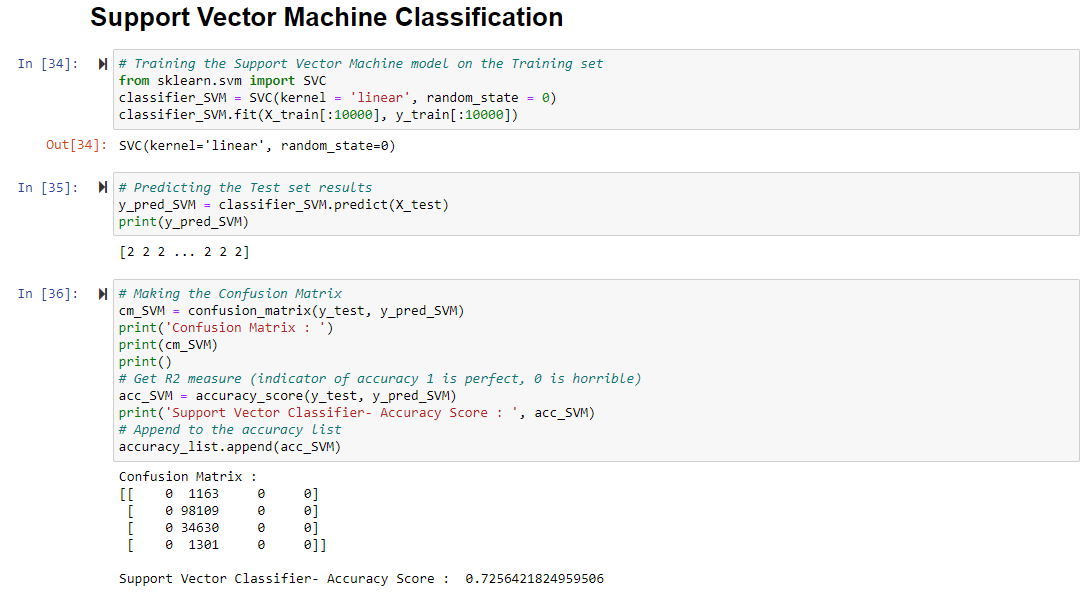
Support Vector Machine Classification

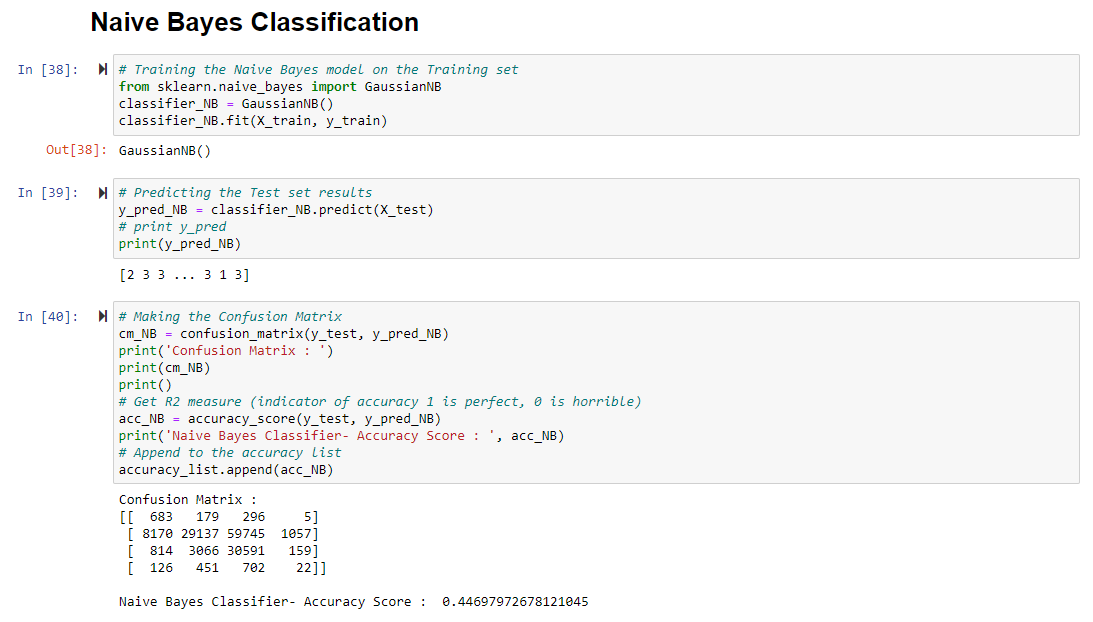


**Predicting the Target Variable on the Data of State of California**

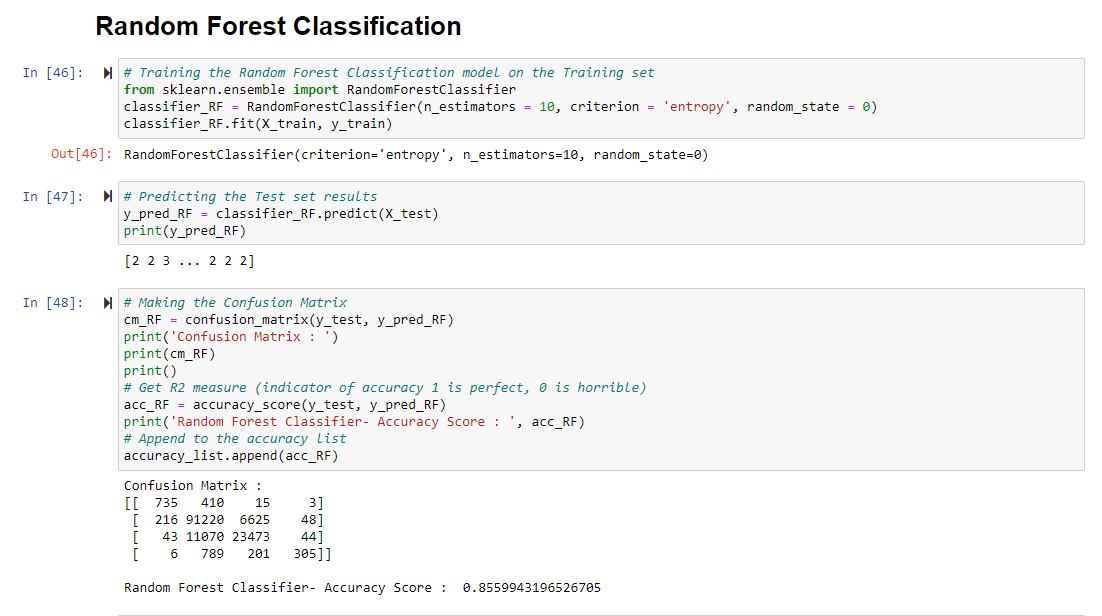






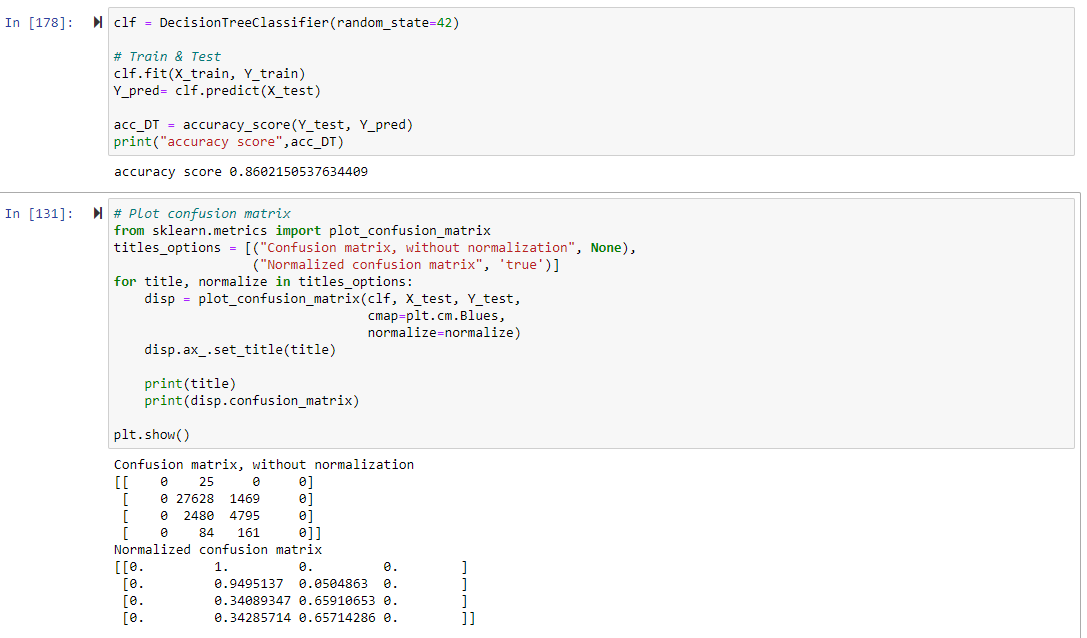


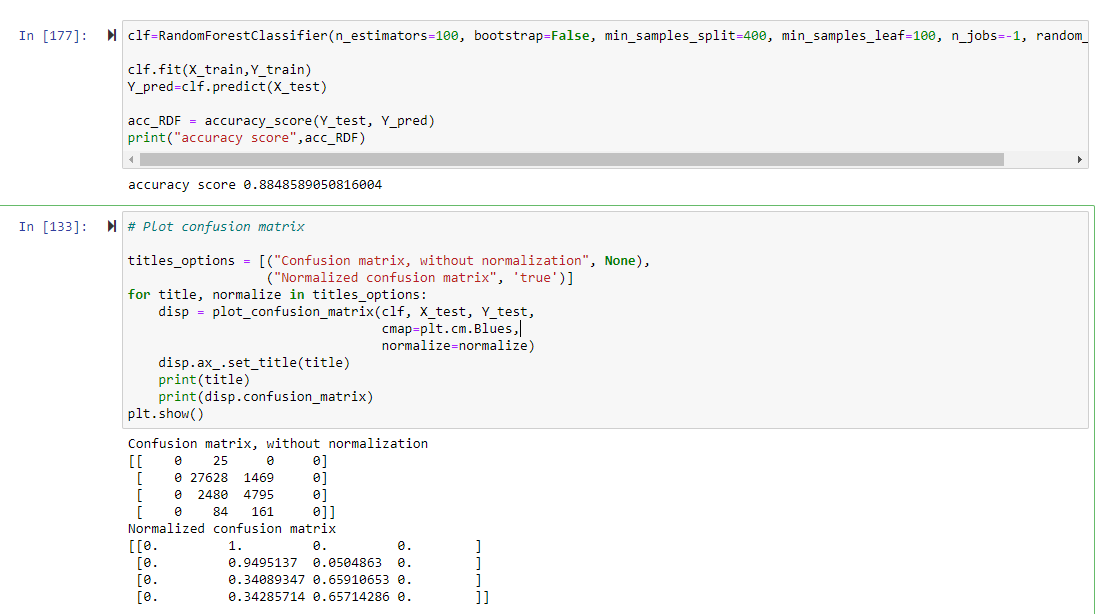


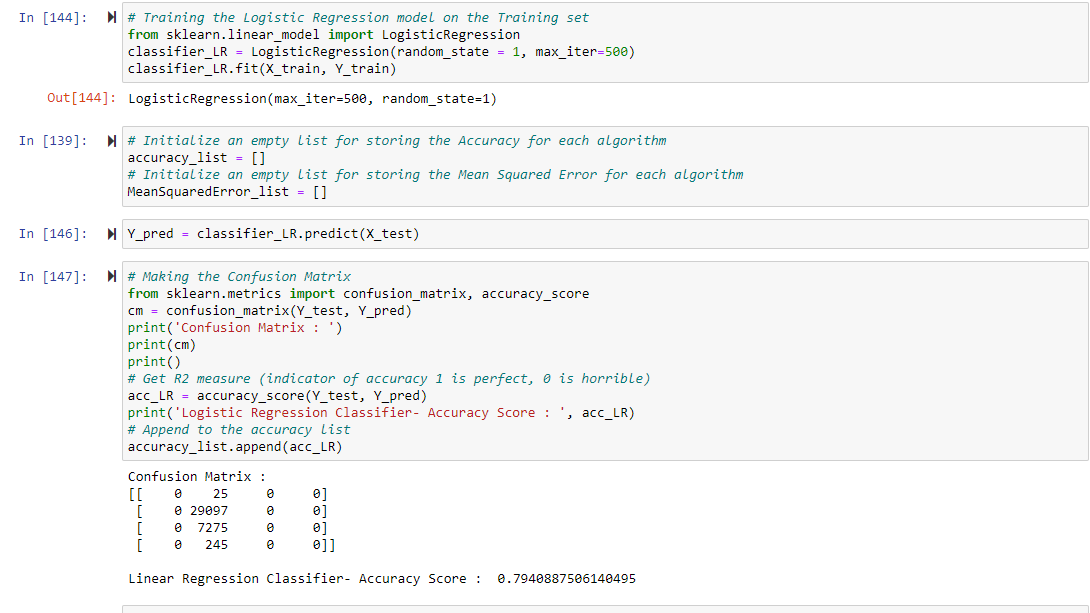


**Predicting the Target Variable on the data of state of South Carolina**

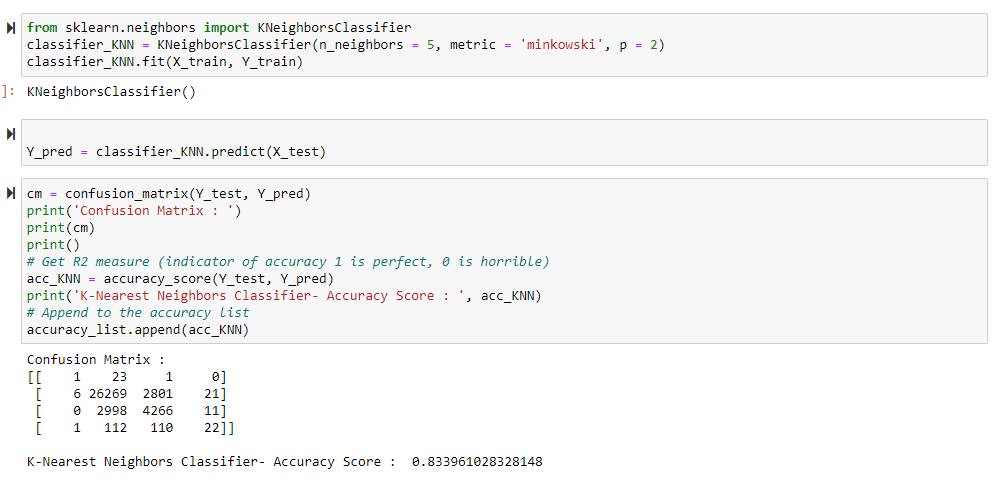
**Decision Tree**

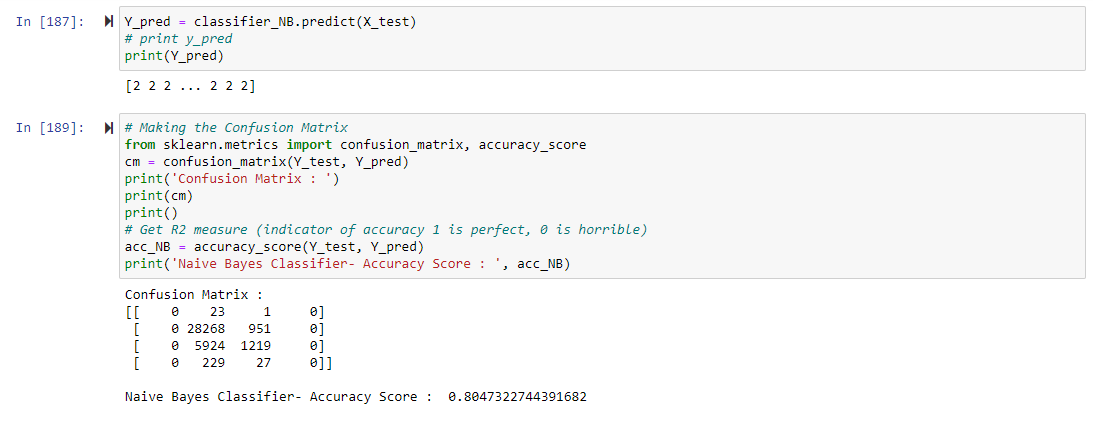






**KNN**





Naïve Bayes

SVC

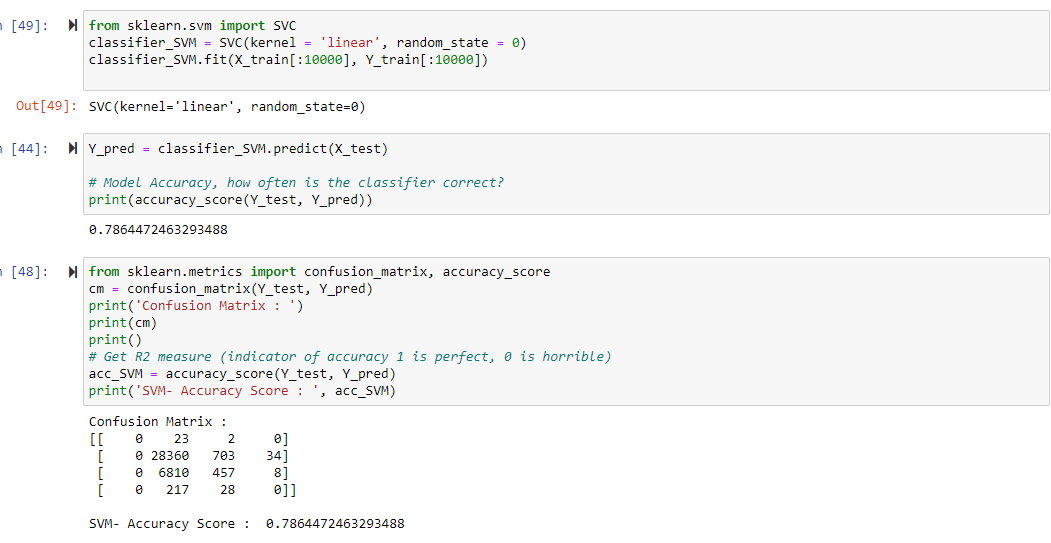
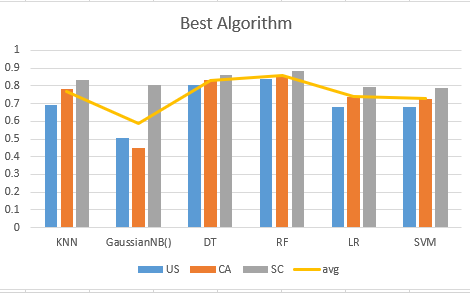


Table of all outputs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| selection | US | CA | SC | avg |
| KNN | 0.6913 | 0.7826 | 0.8339 | 0.7693 |
| GaussianNB() | 0.5033 | 0.4469 | 0.8047 | 0.585 |
| DT | 0.8044 | 0.8325 | 0.8602 | 0.8324 |
| RF | 0.8387 | 0.8559 | 0.8848 | 0.8598 |
| LR | 0.6832 | 0.7354 | 0.794 | 0.7375 |
| SVM | 0.68 | 0.7256 | 0.7864 | 0.7307 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |



**Conclusion**

With Accuracy Score of all the used Algorithms, the Random Forest Algorithm Classifier produced a significant score of accuracy for prediction of the Target Variable (Severity).

**References**

* Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, and Rajiv Ramnath. [“A Countrywide Traffic Accident Dataset.”](https://arxiv.org/abs/1906.05409), arXiv preprint arXiv:1906.05409 (2019).
* Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. [“Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights.”](https://arxiv.org/abs/1909.09638) In proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2019.
* <https://www.kaggle.com/sobhanmoosavi/us-accidents>
* <https://stackoverflow.com/>
* <https://cran.r-project.org/web/packages/dlookr/vignettes/EDA.html>
* Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, and Rajiv Ramnath. [“A Countrywide Traffic Accident Dataset.”](https://arxiv.org/abs/1906.05409), arXiv preprint arXiv:1906.05409 (2019).
* Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. [“Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights.”](https://arxiv.org/abs/1909.09638) In proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2019.
* <https://scikit-learn.org/stable/>
* <https://matplotlib.org/>