Accident Severity Prediction in USA

IBM CAPSTONE PROJECT

Nandan Rajeev | September 2020

Contents

Introduction

With the continuously growing number of automobiles on the road, the road safety issues have been on the rise. Globally, approximately 1.35 million people die in road crashes each year and on average 3,700 people lose their lives every day on the roads. In the USA alone, more than 38,000 people die every year in crashes on roadways. The U.S. traffic fatality rate is 12.4 deaths per 100,000 inhabitants. Road crashes are the leading cause of death in the U.S. for people aged 1-54.

What can be done to reduce the risk of being in a road crash? Unfortunately, we can't control the weather and get rid of poor driving conditions. Can we autonomously control all the cars on the road so that the driver mistakes can be minimized? Maybe in the future we can but right now, it's not a feasible solution. Predicting accidents is difficult, but not impossible. Various factors come into play which can decide the severity of an accident. Such as car speed, weather, location of incident, road layout and so on. Can we make use of such features to predict the possibility of accidents and their severity? Yes! By studying and analyzing previous recorded data of accidents, we can construct predictor models which can help us predict the severity of accidents which will enable us to take adequate measures in reducing the risk or severity of the accident.

In this project, I will be using machine learning models to predict the severity of accidents by taking the various features as input which is the primary aim. We will also come across various additional insights into the data such as – Which state has the highest number of accidents? How severe are the accidents that occur?

Such models can be utilized to predict accidents and their severity in real-time which can lead to a considerable decrease in the dangers of road accidents.

This project has been completed for the attainment of the IBM Data Science Professional Certification.

Data

The data used in this project is a dataset consisting of road traffic accidents that occurred in USA between February 2016 and June 2020, covering all the states. The dataset consists of around 3.5 million entries. This data has been collected by Lyft, for analyzing the delays caused by accidents.

Our target variable here, will be the severity rating. Higher the severity of an accident, greater the traffic delay as the roads are often partially closed while the medical and road services are at work. By relating the traffic delay to the severity of the accident, the accidents are assigned a severity rating from 1 to 4 with 4 being the most severe.

The independent variables which will be used as the features here consist of entries such as – Visibility, Presence of speed bumps, Presence of roundabouts, and so on. By using these as the features, the model will be trained and tested to accurately predict the outcome. Different ML Algorithms will be applied and the best will be chosen based on their accuracies on an out-of-sample set.

The dataset will be split into 2 sets – Training Set and Test Set so we can achieve a better out of sample accuracy estimation.

The data will be visualized and prepared for the machine learning models. Since the dataset is really large, we will have to significantly reduce the dataset size to be able to compute on a 16GB RAM system. A sample set will be created to which the ML algorithms will be applied. After choosing the best model of the ones we consider, the complete dataset can be run through the chosen algorithm to get better accuracy. This will be discussed in detail.

The dataset has been obtained from Kaggle. The columns/attributes present are shown in the table below.

#	Attribute	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) 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

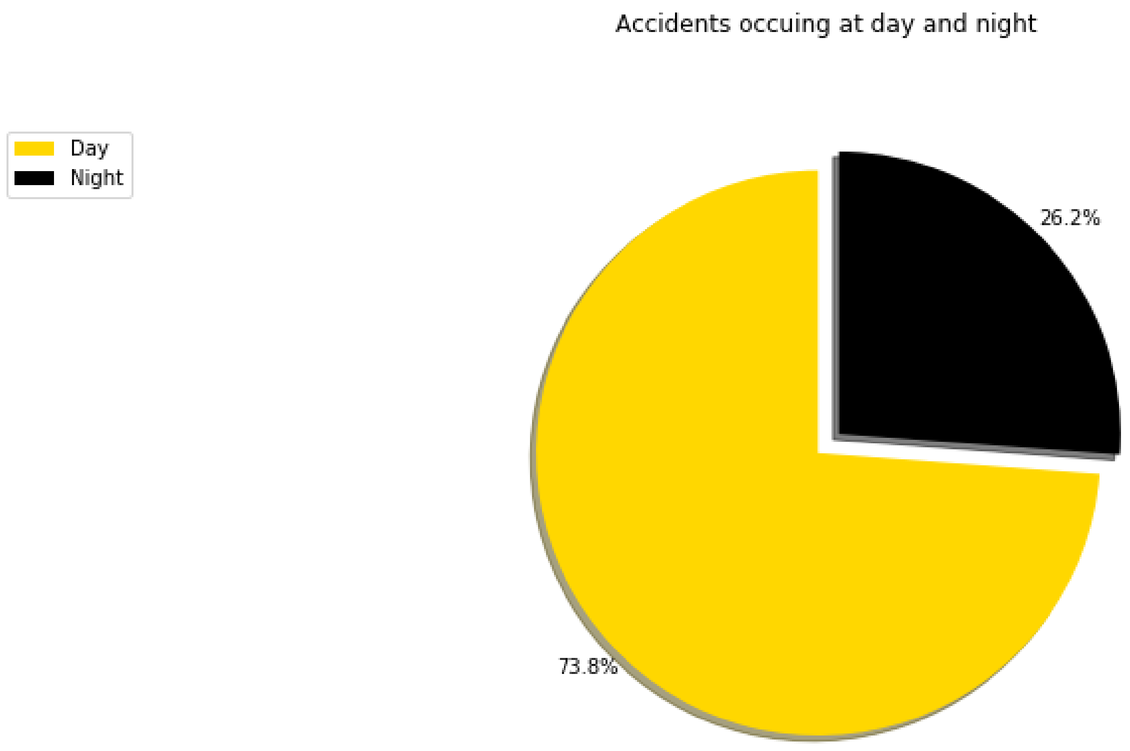
16	City	Shows the city in address field.	Yes
17	County	Shows the county in address field.	Yes
18	State	Shows the state in address field.	Yes
19	Zipcode	Shows the zipcode in address field.	Yes
20	Country	Shows the country in address field.	Yes
21	Timezone	Shows timezone based on the location of the accident (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 annotation which indicates presence of 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 in a nearby location.	No
36	Give_Way	A POI annotation which indicates presence of give_way in a nearby location.	No
37	Junction	A POI annotation which indicates presence of junction in a nearby location.	No
38	No_Exit	A POI annotation which indicates presence of no_exit in a nearby location.	No
39	Railway	A POI annotation which indicates presence of railway in a nearby location.	No
40	Roundabout	A POI annotation which indicates presence of roundabout in a nearby location.	No
41	Station	A POI annotation which indicates presence of station in a nearby location.	No
42	Stop	A POI annotation which indicates presence of stop in a nearby location.	No
43	Traffic_Calming	A POI annotation which indicates presence of traffic_calming in a nearby location.	No
44	Traffic_Signal	A POI annotation which indicates presence of traffic_signal in a nearby location.	No
45	Turning_Loop	A POI annotation which indicates presence of turning_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.	Yes
48	Nautical_Twilight	Shows the period of day (i.e. day or night) based on nautical twilight.	Yes
49	Astronomical_Twilight	Shows the period of day (i.e. day or night) based on astronomical twilight.	Yes

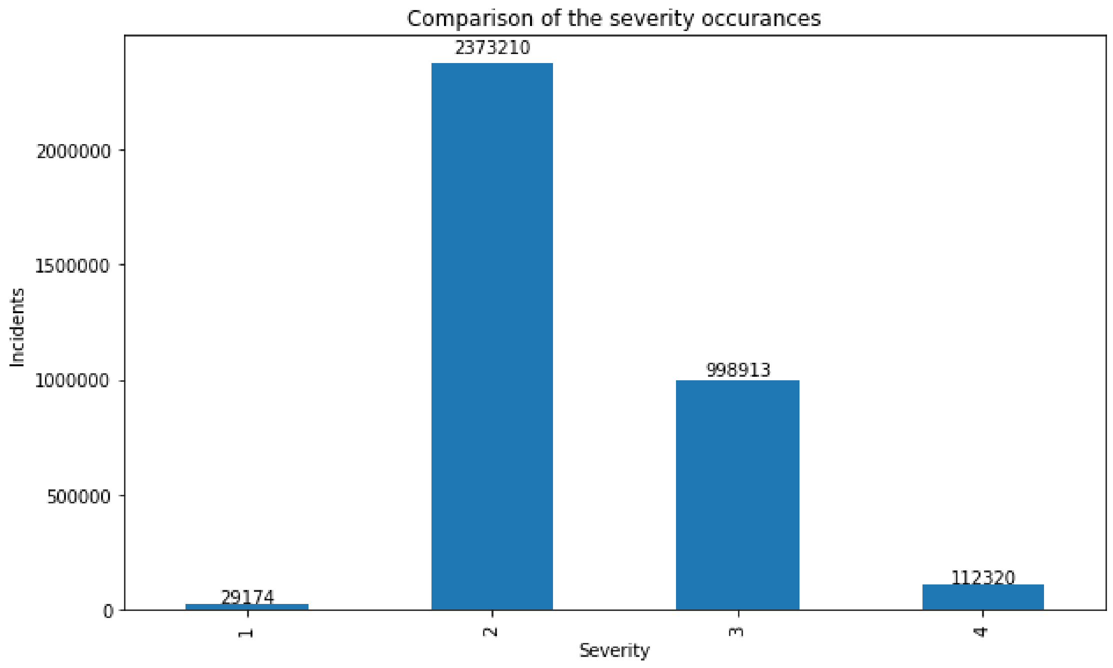
Data Visualization

We need to visualize the data to understand their relationships better. By understanding how factors influence the outcome and interact with each other, we can get interesting insights which can help in the data preparation phase as well as provide us with extra information regarding the topic.

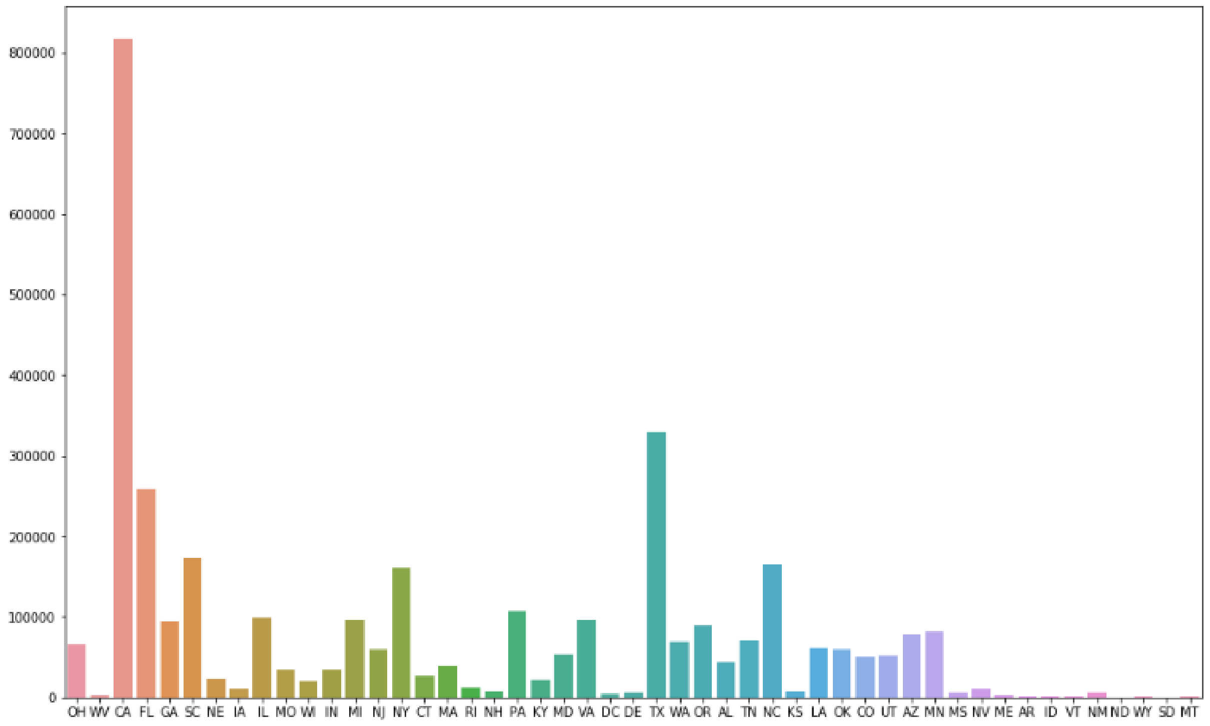
- By plotting the below pie chart, we see that accidents are more common in the day. This suggests that the nigh visibility issues are not much of a factor but the higher day traffic seems to result in more accidents.



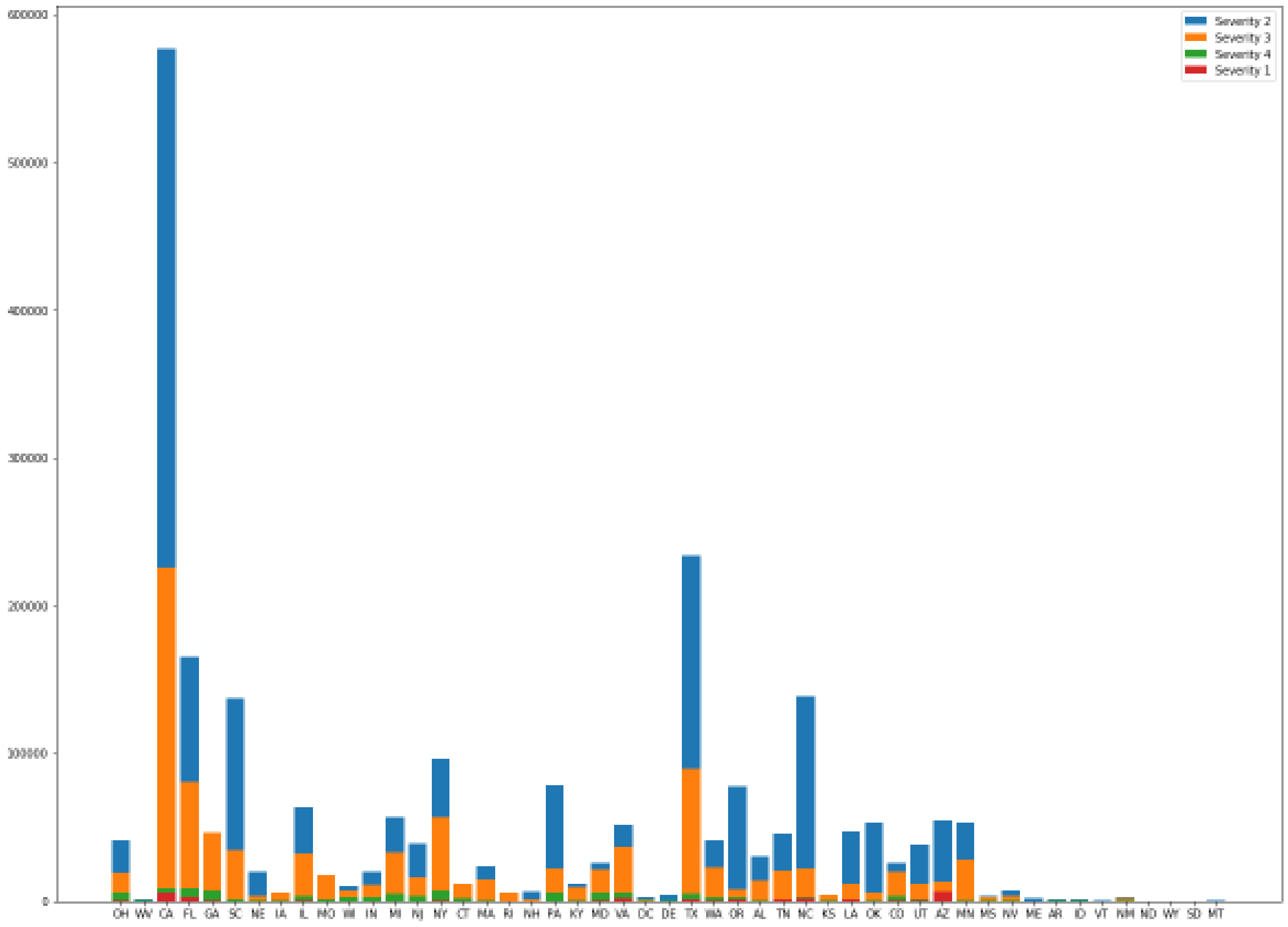
- Comparing the severity occurrences, we examine which severity is most common. Severity 2 seems to be the most common followed by Severity 3. Severity 1 and 4 are quite rare.



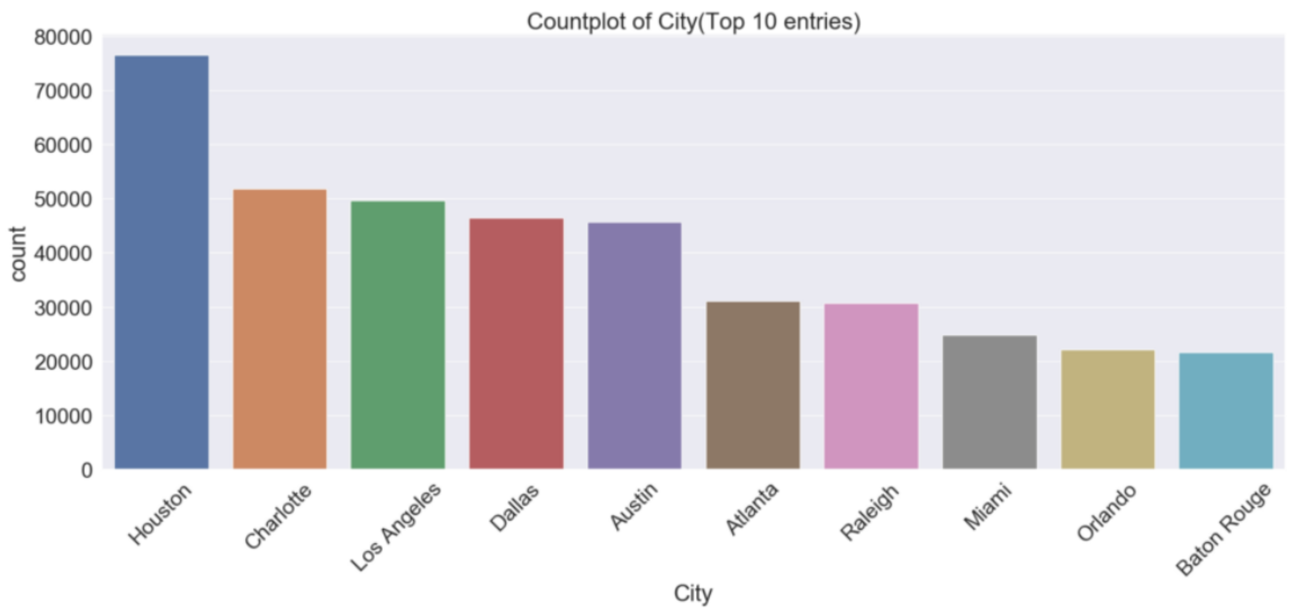
- On plotting the state wise accidents report, we see that California sees the greatest number of accidents.



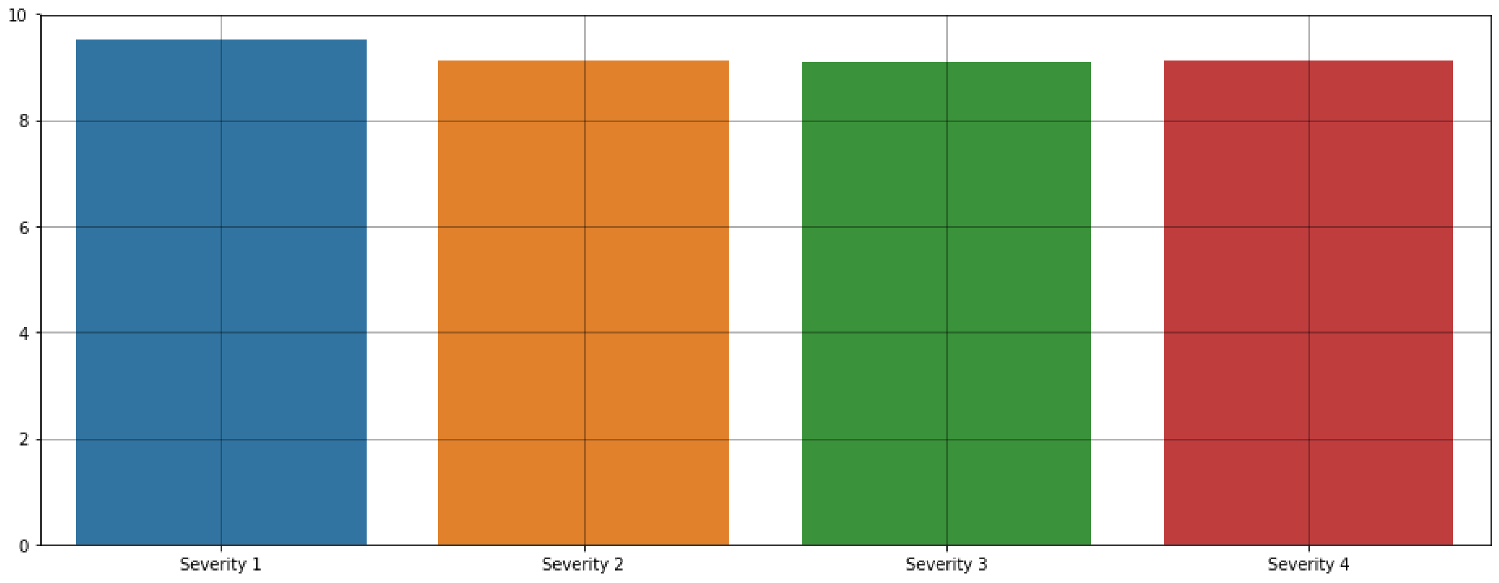
- We'll examine the state-wise chart deeper to see if the severity level shows more information.



- Examining the accidents by city, we can see which cities see the most accidents.



- Visibility which is influenced by weather conditions such as fog, rain etc. makes driving conditions worse. However, we see that the incidents severities aren't largely affected by this factor.



Data Preparation

The giant dataset that we're dealing with needs to go through multiple filtering stages.

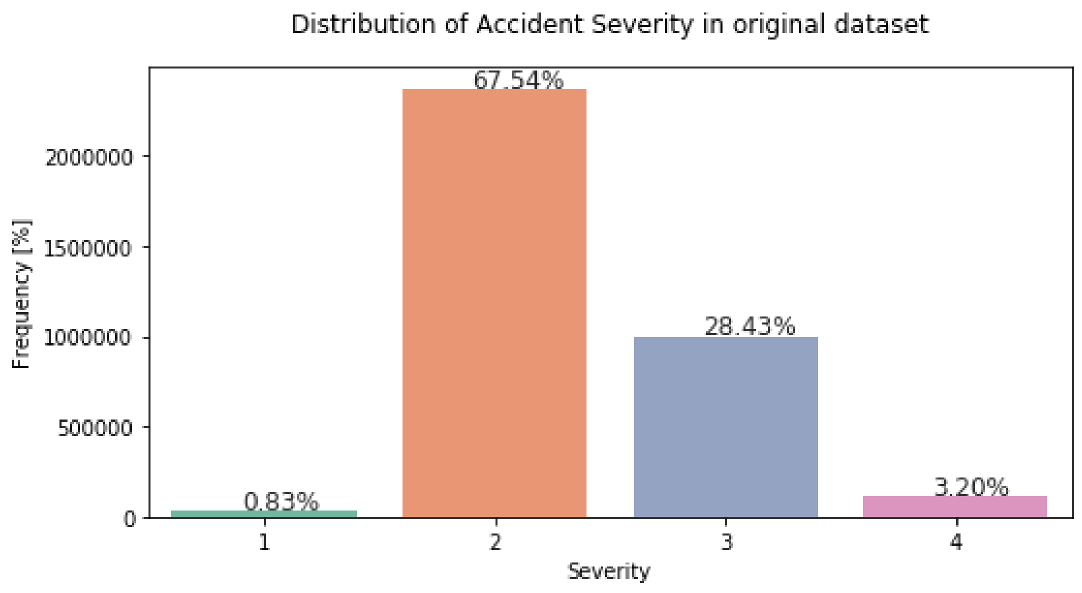
This stage consisted of –

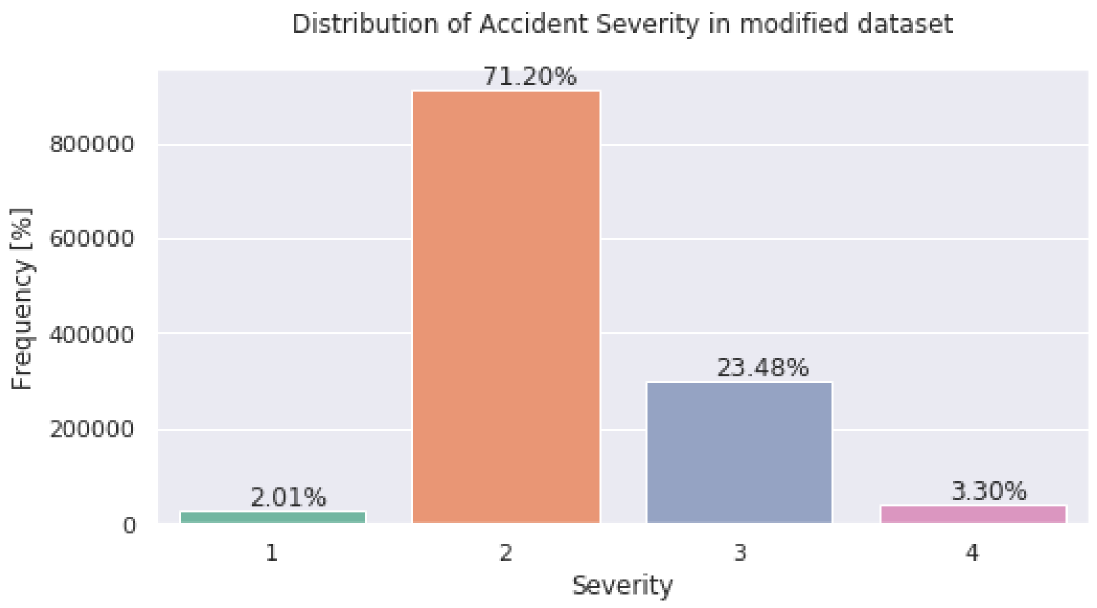
- **Dropping irrelevant columns**

  After listing and seeing the various columns present, a lot of the columns seemed to be unnecessary. So, columns such as – Airport Code, Twilight Zone, etc. were dropped.

- **Dealing with missing data**

  Missing data can be crucial as they help provide a larger dataset for training and testing. However, in our case where we need to minimize the dataset due to computational resource limitations it's better to get rid of the entries which have missing values which are crucial. This will reduce the dataset size significantly as the dataset seems to have a large number of rows with some NaN values. But in doing so we also get a purer dataset with accurate readings as compared to those that could have been interpolated.
  Before proceeding with the new dataset with the dropped entries, it helps to examine the difference in the frequency distribution in the dataset before and after dropping rows. This is to ensure that the frequency distribution is similar. If one or two severities in particular lose out on more data than the others due to this step, we will have to change our approach.

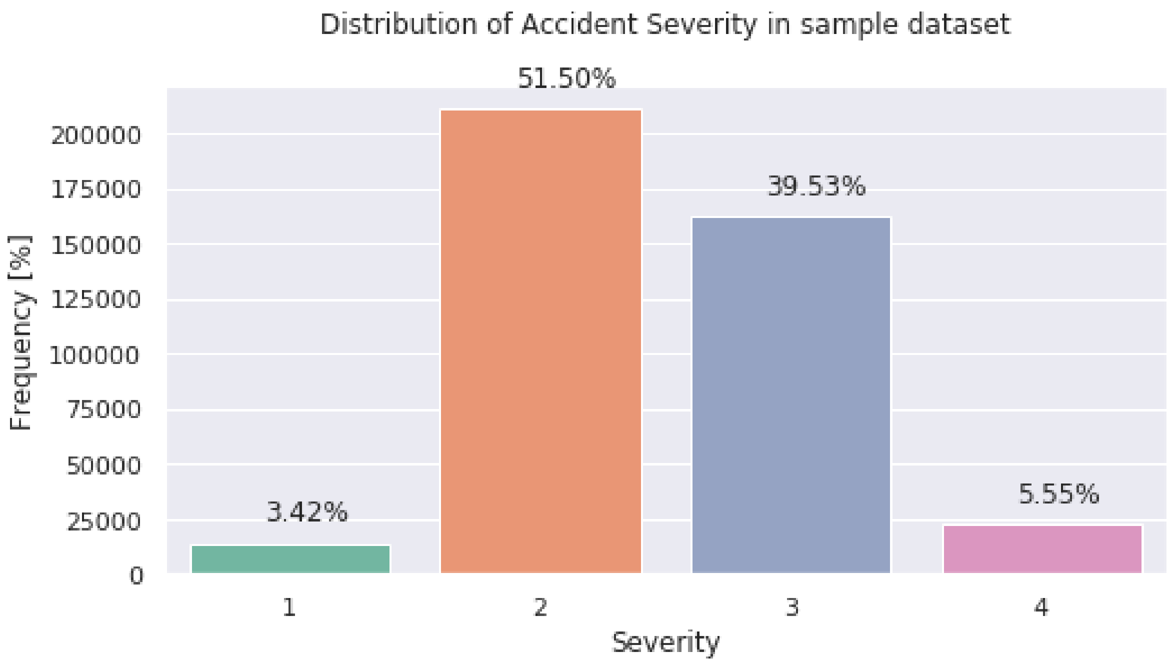




As we can see, there isn't a significant change in the weightage of the severities. So, we can proceed with this method of data cleaning/reduction.

- **Sample Creation**

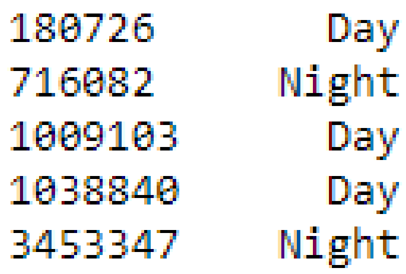
As stated before, we need to select a sample from this data to make the computation process easier. This step can be skipped if there is access to very high computing power.
So, from the severity occurrence percentage values we plotted above, we will select our sample such that -
* Severity 2 doesn't overpower the dataset
* The lower occurring severities have an improved density in the dataset.



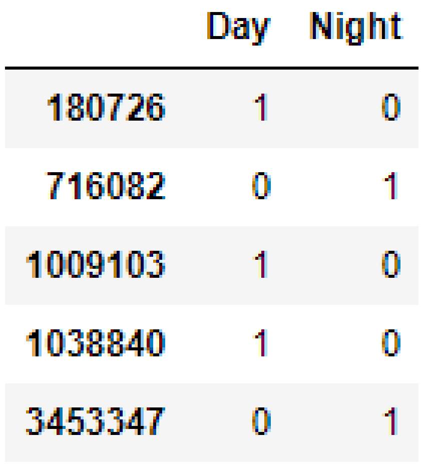
The weightages are modified to better suit the modelling process as mentioned above.

- **Data Transformation**

Many of the columns in the dataframe were of object type. Eg – The Sunrise_Sunset column had entries 'Day' and 'Night'. The ML models used cannot interpret these objects as they are not in the form of integers or Booleans. With the process of Dummy Variable Creation, dummy columns are created with each different entry in the column, and filled with 1's and 0's as shown below –



```
180726       Day
716082     Night
1009103      Day
1038840      Day
3453347    Night
```



	Day	Night
180726	1	0
716082	0	1
1009103	1	0
1038840	1	0
3453347	0	1

As shown, entries which have 'Day' under the Sunrise_Sunset column will have a 1 in the Day column and 0 in the Night column. The original column will be dropped. This style of dummy creation is also carried out for a few other columns which were of object type.

The Weather_Condition column could have also been transformed using this method but the column had over 100 different entries which were also duplicates in many cases eg – Some entries mentioned 'T-Storm" while others said "Thunderstorm". So, this column was dropped.

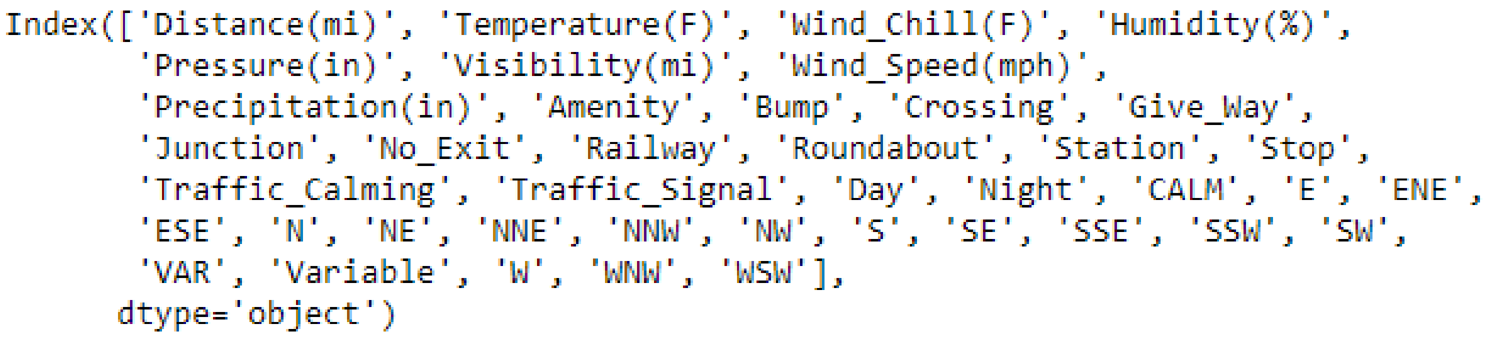
- **Data Preparation for Modeling**

  Now that the data is in a clean dataframe consisting of all the required data types for modelling, it will be prepared for machine learning by:

  1. Defining feature/independent variables set and dependent variable set
  2. Data standardizing is carried out to normalize the data so the algorithms can perform better.
  3. Splitting the data into training and testing sets. I have used a 70-30 split.

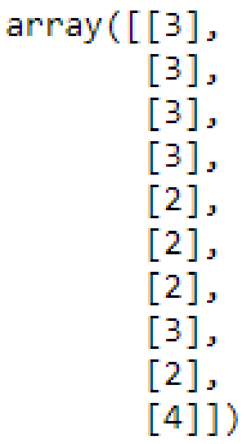
- **Features and Target List considered for predicting the outcome**

  The following parameters are considered for training the models:



```
Index(['Distance(mi)', 'Temperature(F)', 'Wind_Chill(F)', 'Humidity(%)',
       'Pressure(in)', 'Visibility(mi)', 'Wind_Speed(mph)',
       'Precipitation(in)', 'Amenity', 'Bump', 'Crossing', 'Give_Way',
       'Junction', 'No_Exit', 'Railway', 'Roundabout', 'Station', 'Stop',
       'Traffic_Calming', 'Traffic_Signal', 'Day', 'Night', 'CALM', 'E', 'ENE',
       'ESE', 'N', 'NE', 'NNE', 'NNW', 'NW', 'S', 'SE', 'SSE', 'SSW', 'SW',
       'VAR', 'Variable', 'W', 'WNW', 'WSW'],
      dtype='object')
```

The target set consists of the severity ratings:



```
array([[3],
       [3],
       [3],
       [3],
       [2],
       [2],
       [2],
       [3],
       [2],
       [4]])
```

Prediction Models

I have taken into consideration 4 different machine learning algorithms. These are discussed below –

- **Support Vector Machines**

  A Support Vector Machine is a supervised algorithm that can classify cases by finding a separator. SVM works by first mapping data to a high dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. Then, a separator is estimated for the data. The data should be transformed in such a way that a separator could be drawn as a hyperplane.

  The support vector machine's accuracy depends on the number of iterations it carries out. Here, the default value of 1000 is used so as to minimize the computation time.

- **Logistic Regression**

  Logistic regression is a statistical and machine learning technique for classifying records of a dataset based on the values of the input fields.
  In logistic regression, we use one or more independent variables to predict an outcome which we call the dependent variable representing the severity of the accident. Logistic regression is analogous to linear regression but tries to predict a categorical or discrete target field instead of a numeric one. In linear regression, we might try to predict a continuous value of variables such as the price of a house, blood pressure of a patient, or fuel consumption of a car. But in logistic regression,

we predict a variable which is binary such as yes/no, true/false, successful or not successful, pregnant/not pregnant, and so on, all of which can be coded as zero or one. In logistic regression independent variables should be continuous. If categorical, they should be dummy or indicator coded. This is why we have created dummy values for the object type columns.

- **Decision Tree**

  Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.

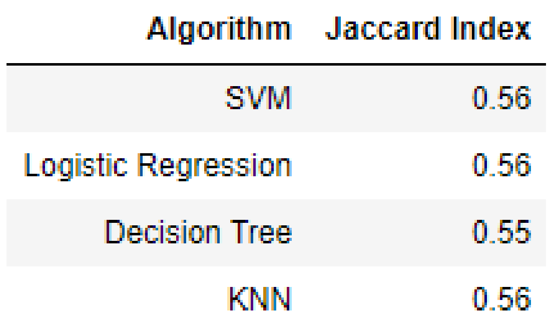
- **K Nearest Neighbors**

  The K-Nearest Neighbors algorithm is a classification algorithm that takes a bunch of labeled points and uses them to learn how to label other points. This algorithm classifies cases based on their similarity to other cases. In K-Nearest Neighbors, data points that are near each other are said to be neighbors. K-Nearest Neighbors is based on this paradigm. Similar cases with the same class labels are near each other. Thus, the distance between two cases is a measure of their dissimilarity.
  Here, I assume the k values as 534. This number is chosen by the general thumb rule of choosing k which is to take the square root of the number of samples. This might not be the best k value but it will be close. To find the best k value, we can do so by writing a FOR loop and fighting the value of k where the accuracy is highest. However, this is very time consuming on a low resource computer and so the k value has been assumed.

Model Evaluation

Now that we have trained the different models, we will compare the accuracies so that we can choose the best.

We will compare them by finding their Jaccard Indices which is a depiction of how similar the predicted values are to the actual values.



Algorithm	Jaccard Index
SVM	0.56
Logistic Regression	0.56
Decision Tree	0.55
KNN	0.56

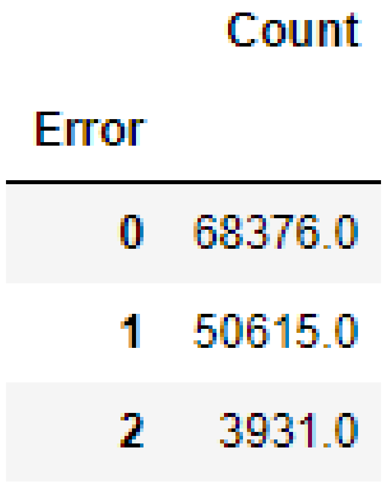
We can see that all the models have a similar accuracy.

It is also worth mentioning how different the runtimes are for the different model training and predicting.

Logistic Regression and SVM took the least amount of time. Decision Tree model took slightly longer. The KNN method had by far the longest runtime (Would be even longer if we were calculating the optimal k value).

The accuracies seem to be poor here. Let's get a better understanding of what this means –

The severities of the accidents can often be misinterpreted due to minor differences in the conditions leading to it. We can take a look at how far off the severity predictions are in the cases of incorrect prediction.



	Count
Error	
0	68376.0
1	50615.0
2	3931.0

By analyzing the predictions, we see that a lot of the incorrect predictions have been off by just one severity level. Few have been missed by 2 levels and none have varied by 3 levels. The accuracy will be nearly 97% if considering the predictions which were off by just one level. This is a good thing because it means that the model has done a good job in classification but there are some cases where the values are contradicting with the neighboring severity level and hence predicting the wrong value. From this, we can conclude that the accuracy can be improved by providing the model with more training data and modifying some iteration parameters. How this can be done is discussed next.

Discussion: Accuracy Improvement

Now that we have examined the accuracies of all the models and found that the best algorithms to use are -

- Logistic Regression Model
- Support Vector Machine (With a higher number of iterations)

We can now use this model to evaluate the dataset with greater accuracy. When I ran the more complex and larger dataset consisting of more dummy variables (3.5 million entries x 169 parameters) than those considered above through the machine learning models, the computation time was really long and the kernel often crashed. So firstly, it is important to have access to a supercomputer to be able to train models using the full dataset.

We can take the following measures for getting more accurate predictions –

- Consider the full dataset. We can fix a very large part of the missing data by interpolating the weather conditions. For this to be accurate, we need to sort the dataset by geo-location and time of occurrence. This will help create the weather parameters by considering the conditions of the areas in its vicinity at a time close to the incident.
- Support Vector Machine - We can increase the maximum iterations to get a more improved solution. By default, we consider 1000 iterations. Increasing this will increase the computational time as well.
- K Nearest Neighbours Method - The optimal value of K has not been found here. To find the most suitable k value, we will need to run a FOR loop for a large range (like 10,400). This requires extremely high computational resources but will result with a better accuracy.
- Improving the data collection quality. There are many areas where the data recording process can be improved. If a more accurate and precise dataset was available (for eg- the numerous similar weather conditions which were phrased differently) the prediction capabilities will also be better.

Conclusion

We have gained some valuable insights on the accidents that occur in USA. We got a better understanding of some factors and how they influence the outcome of such incidents. Some additional information such as knowing the amount and frequency of accidents that occur state-wise and city-wise also suggests that certain places need better regulation of road traffic.

The dataset which has been cleaned, transformed and sampled has been put through the machine learning process and prediction models have been trained. From the accuracy comparison, we see that the Logistic Regression, SVM and KNN methods have the same accuracy. The Logistic Regression and SVM algorithms are the better choice of these as these model takes significantly lesser time and resources to train and predict. Now that we have chosen the best models for this project, these models can be trained on a more advanced computational environment using the complete dataset with interpolated missing values to get an even more accurate prediction model.

Using such models, the accident severity can be predicted in real time which can help emergency services in reacting more promptly as it can alert them before someone reports the incident.

References

1. Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, and Rajiv Ramnath. "A Countrywide Traffic Accident Dataset.", arXiv preprint arXiv:1906.05409 (2019).

2. Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. "Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights." In proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2019.

3. Zhang Y. (2012) Support Vector Machine Classification Algorithm and Its Application. In: Liu C., Wang L., Yang A. (eds) Information Computing and Applications. ICICA 2012. Communications in Computer and Information Science, vol 308. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34041-3_27

4. Sadiku, Matthew & Shadare, Adebowale & Musa, Sarhan & Akujuobi, Cajetan & Perry, Roy. (2016). DATA VISUALIZATION. International Journal of Engineering Research and Advanced Technology (IJERAT). 12. 2454-6135.