Pattern Recognition and Machine Learning

Major Project (Accident Detection)

Group Members:

1)Akriti Gupta (B21AI005)

2)Sagnik Goswami (B21AI034)

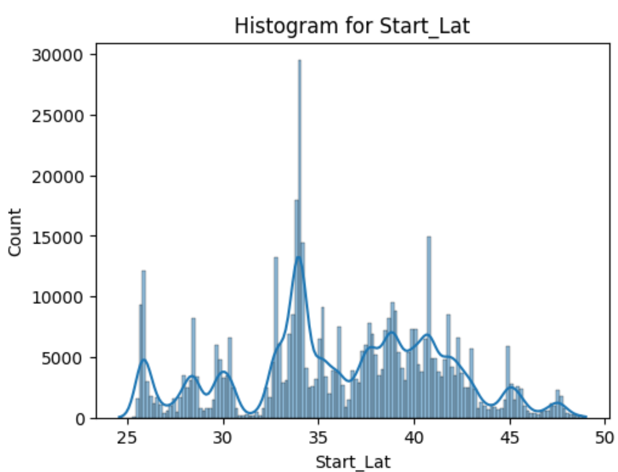
3)Rathva Yuvraj (B21CS064)

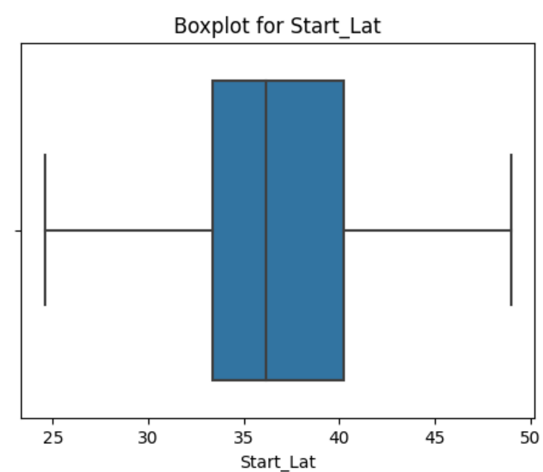
For Severity Prediction

Univariate Analysis:

1) Numerical Columns:

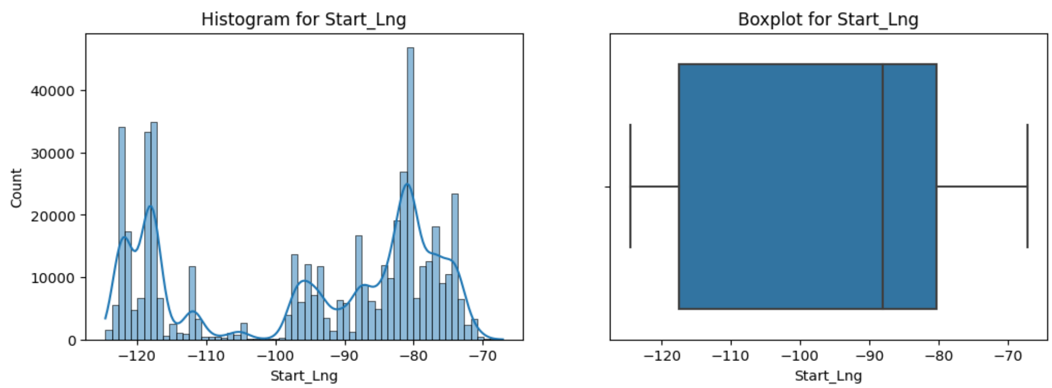
Start_Lat has skew=-0.14624093235873503



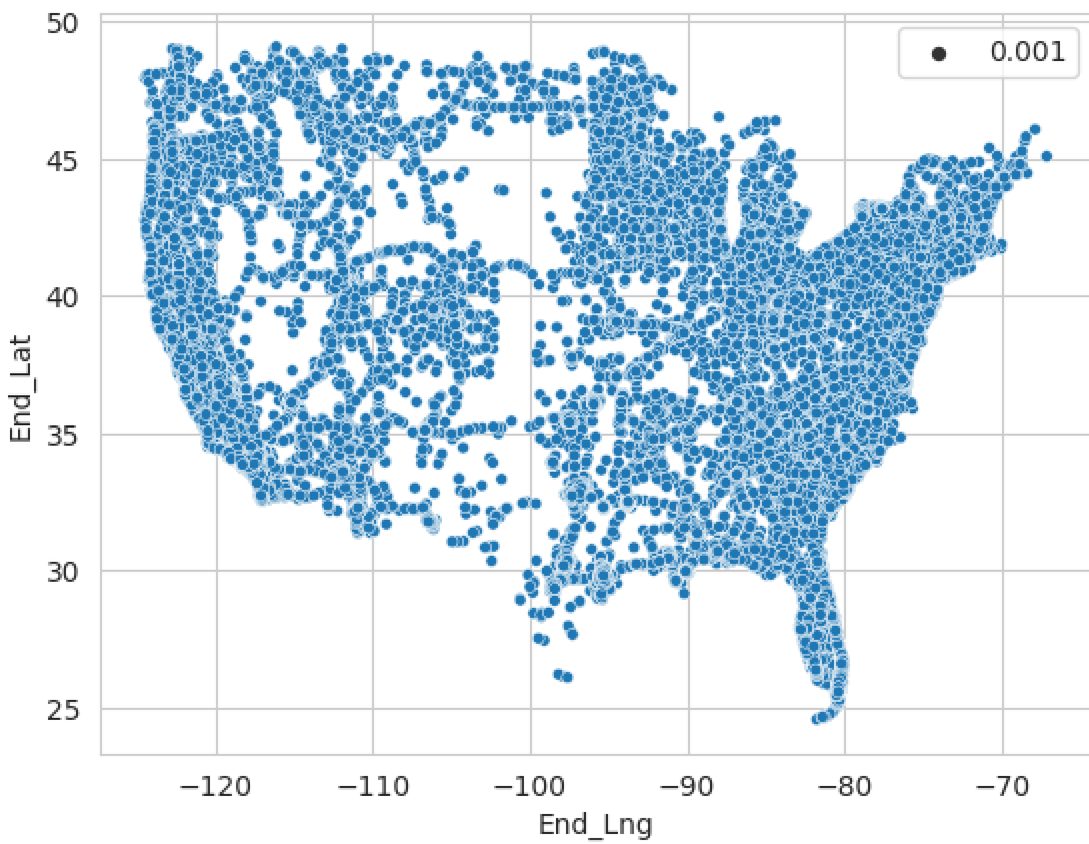


So, Start_Lat has a very low skew and no outliers. The distribution is multimodal in nature. Maximum Accidents were in Latitude in range of 33 to 35

Start_Lng has skew=-0.4126805805754043

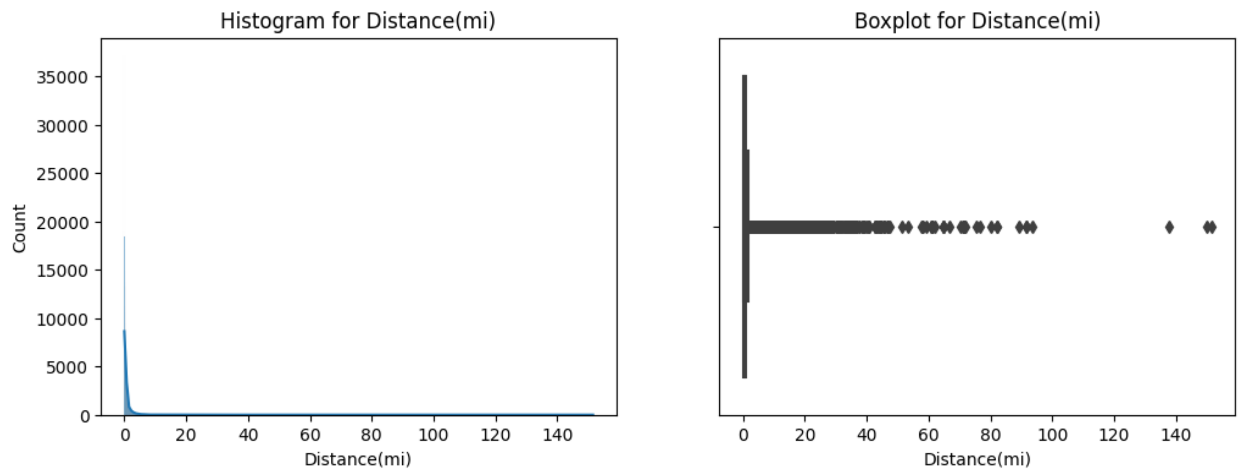


So Start_Lng is even more distributed than latitude. Major Accidents take place for high negative latitude and highly positive latitudes.



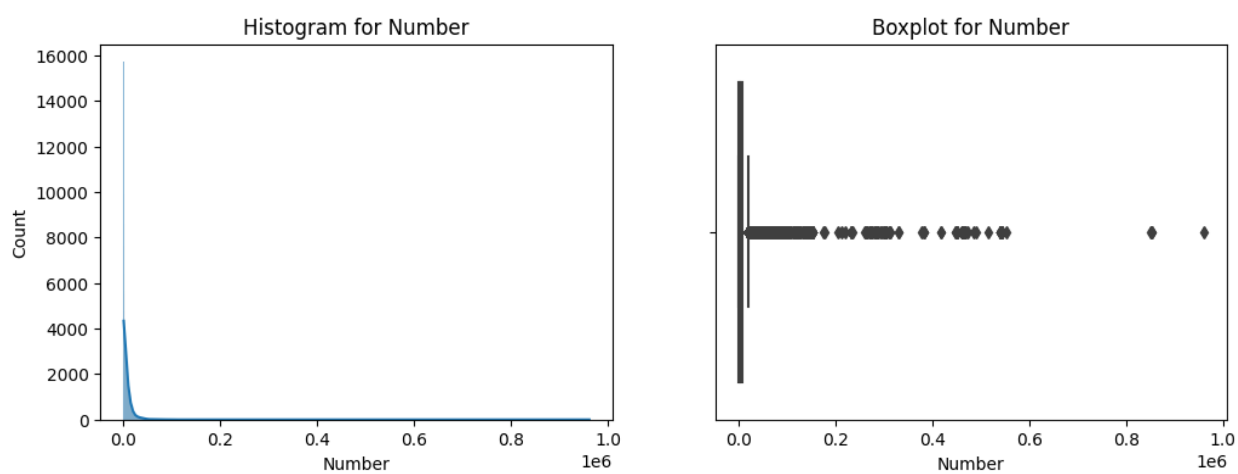
So, the accidents mostly takes place on the coastal regions validating for the distribution of Start and Ending Latitude

Distance(mi) has skew=15.567616917145584



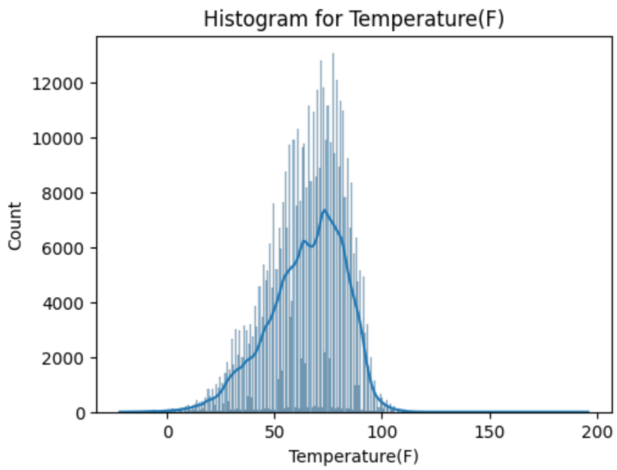
Distance(mi) has a very high positive skew, thus there were many accidents which caused a large length of road to be affected by the accident. Mostly the length of roads affected by the accident is near to 1 miles

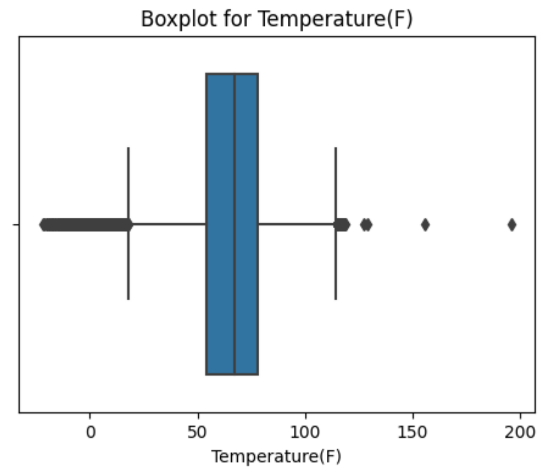
Number has skew=19.5050873565759



So, the street number is also high showing there were many accidents in street having very high number.

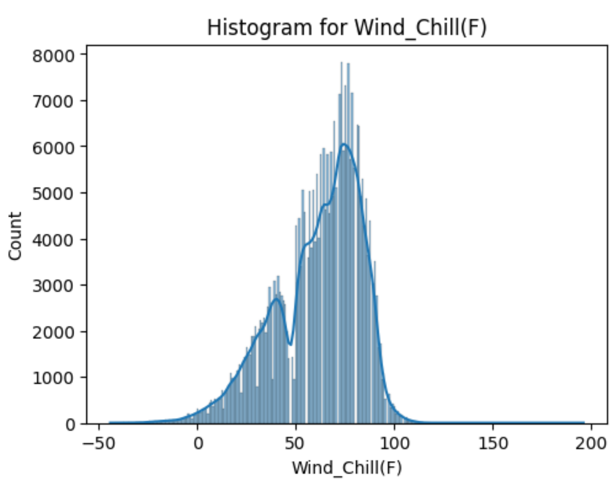
Temperature(F) has skew=-0.5974626186036305

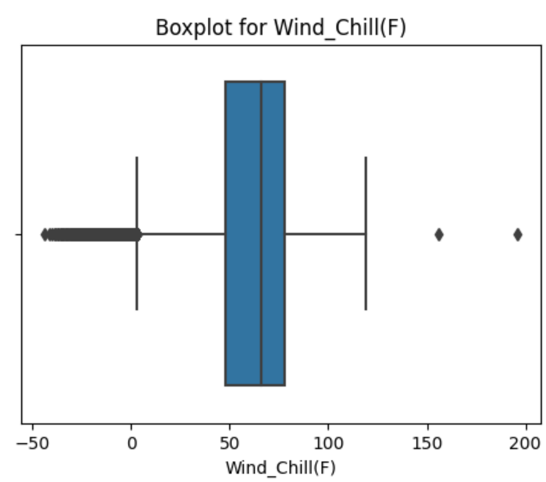




So, temperature has normal skewness and there were some accidents in which temperature is very low and some with temperature very high and those being treated as outliers.

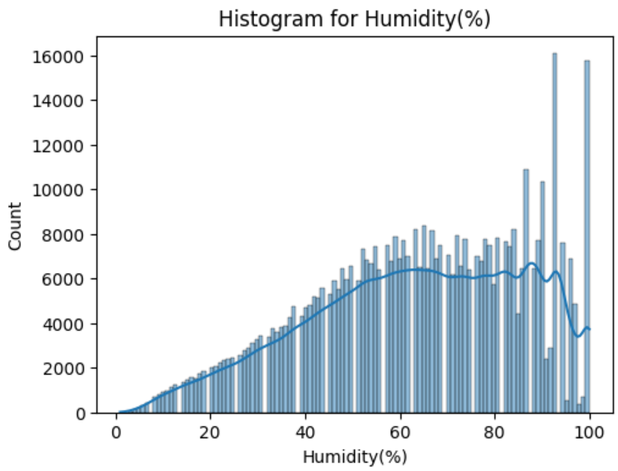
Wind_Chill(F) has skew=-0.7472096081034233

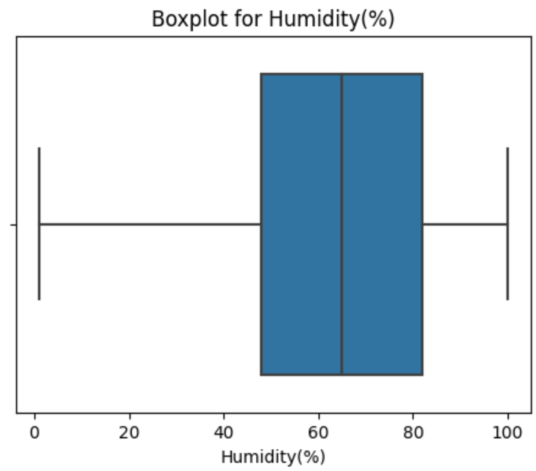




Wind Chill has a negative skew showing that there were many accidents taking place when wind chill was very low.

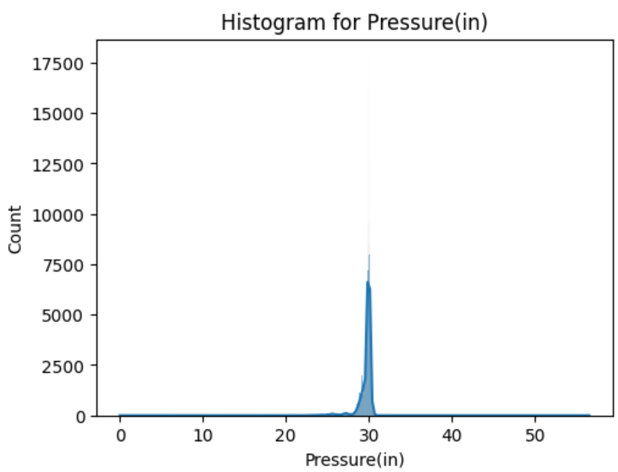
Humidity(%) has skew=-0.3434714887300856

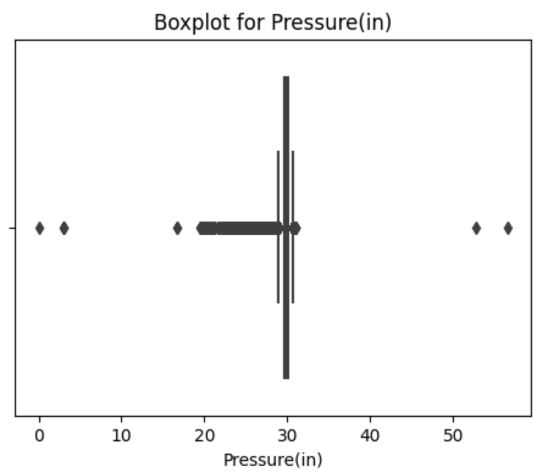




The humidity is left skewed showing that there were some accidents having very low values of Humidity.

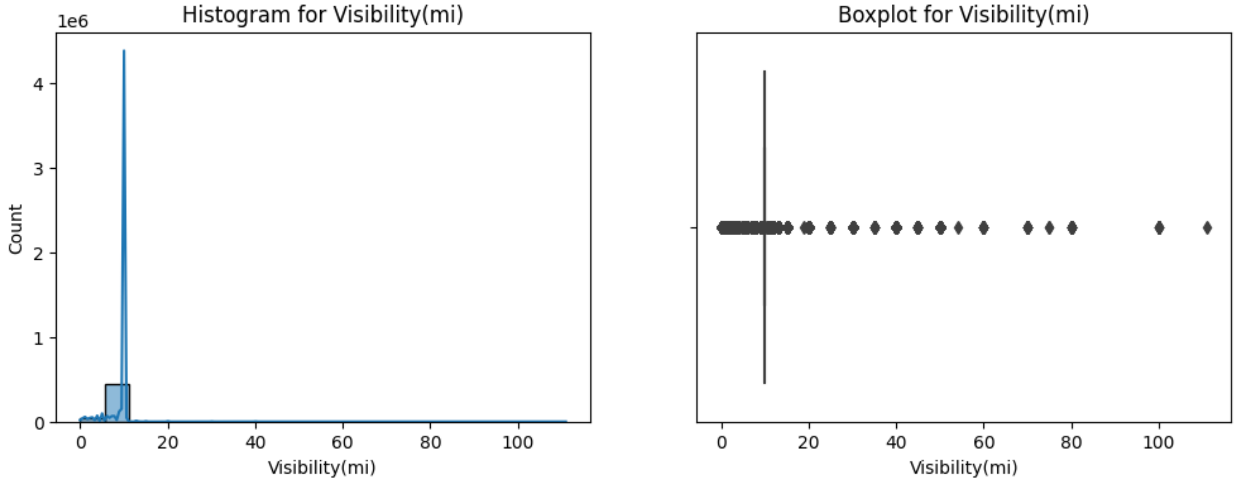
Pressure(in) has skew=-4.225031880602016



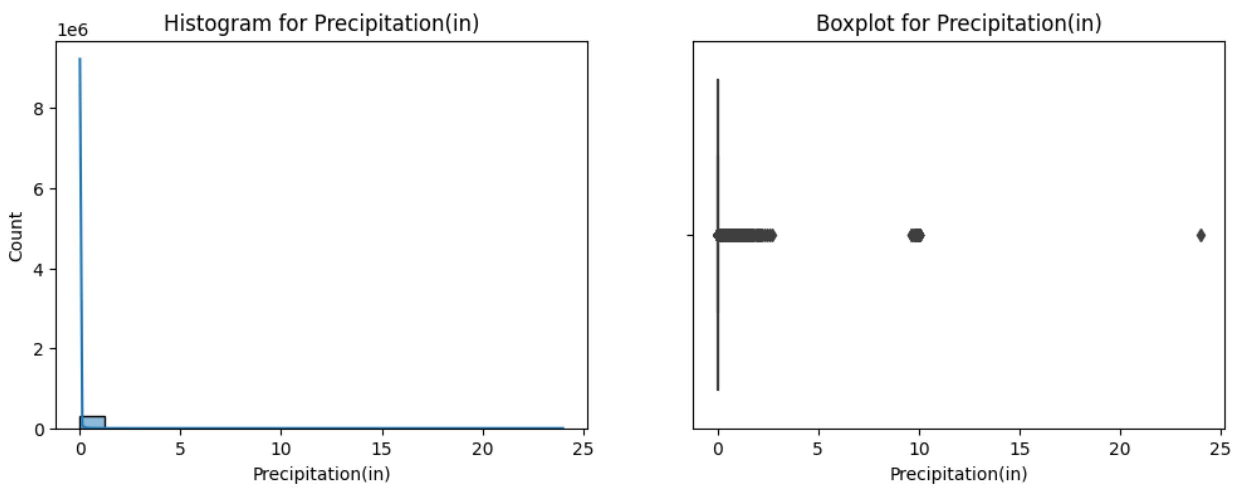


This shows that there were some accidents in which the pressure in the accident region is very different from normal pressure. It has a high negative skew showing that there were many accidents in which the pressure was very low. Some accidents also have high pressure(about 50 in)

Visibility(mi) has skew=3.7506177627765527

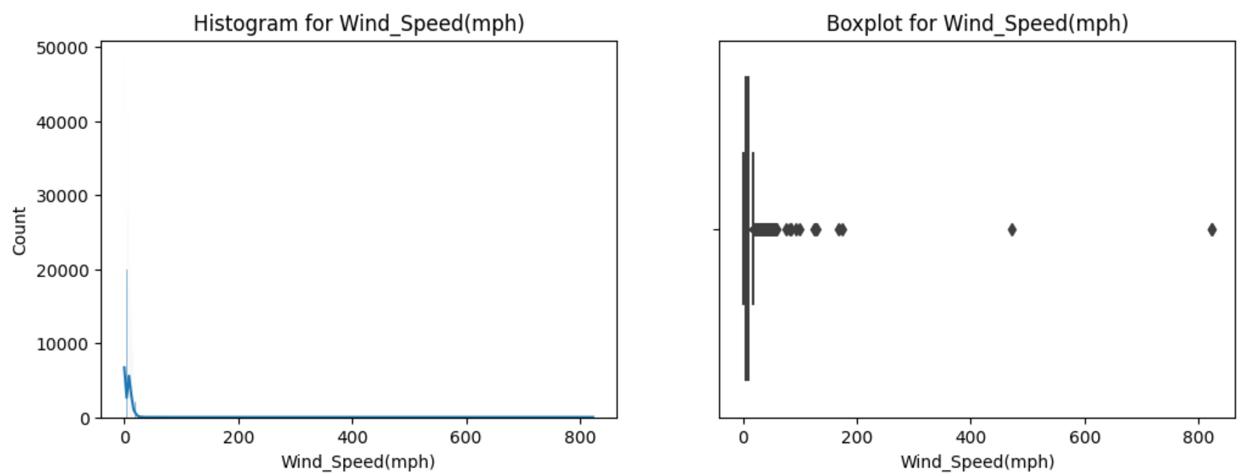


Most accidents occur when there is low visibility. But there were many accidents having high visibility showing that even if visibility was low still some users crashed to have accidents.Precipitation(in) has skew=58.073348480914646



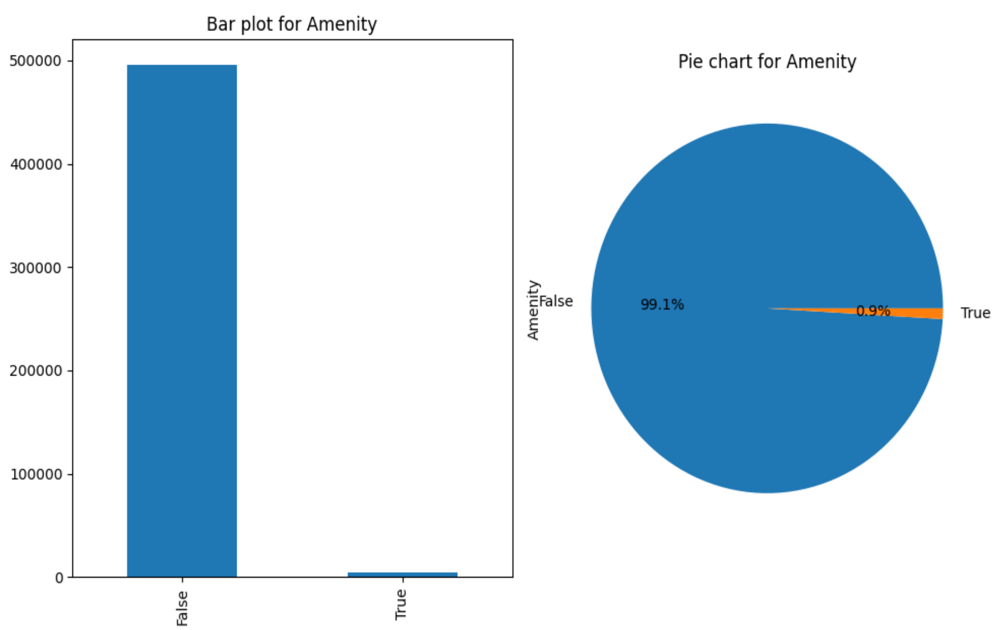
Precipitation has high positive skew showing that there were many accidents which occured with high value of precipitation. But mostly accidents occurred when precipitation was in the range 0 to 2 in.

Wind_Speed(mph) has skew=16.67005793558227

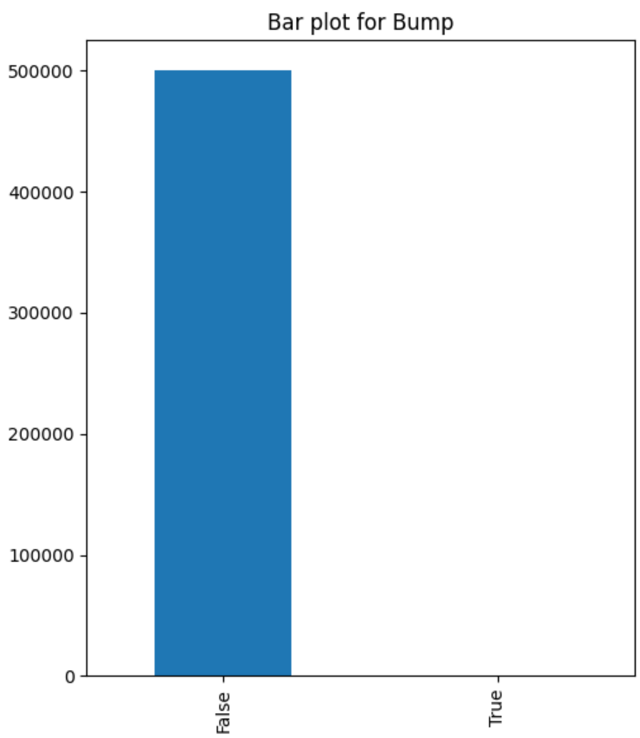


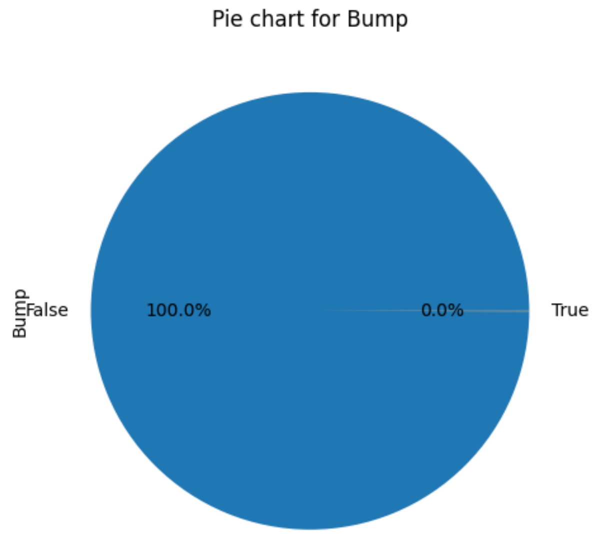
So, there were many accidents in which wind speed was very high. Many accidents occurred when wind speed was above 50 as shown in the box plot. But most accidents occurred when wind speed was in range 0 to 10 mph.

2) Categorical Columns:

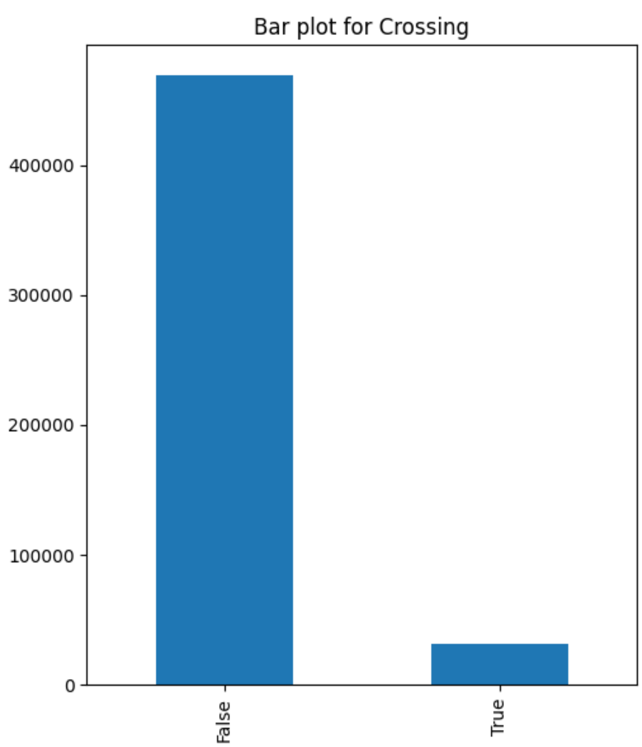


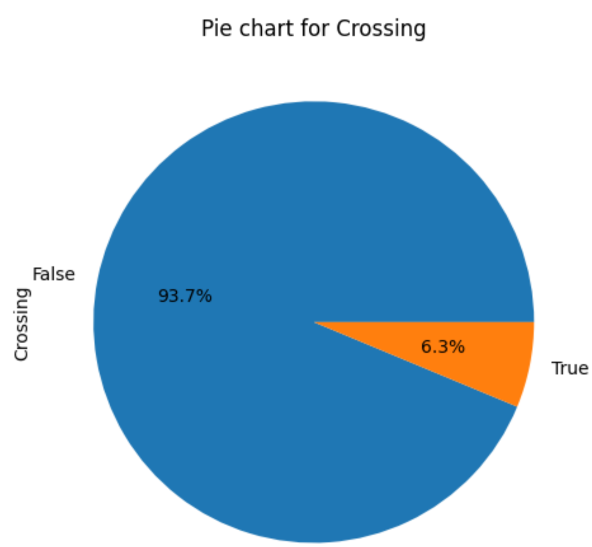
So, we can clearly see that there were very few accidents involving Amenity.



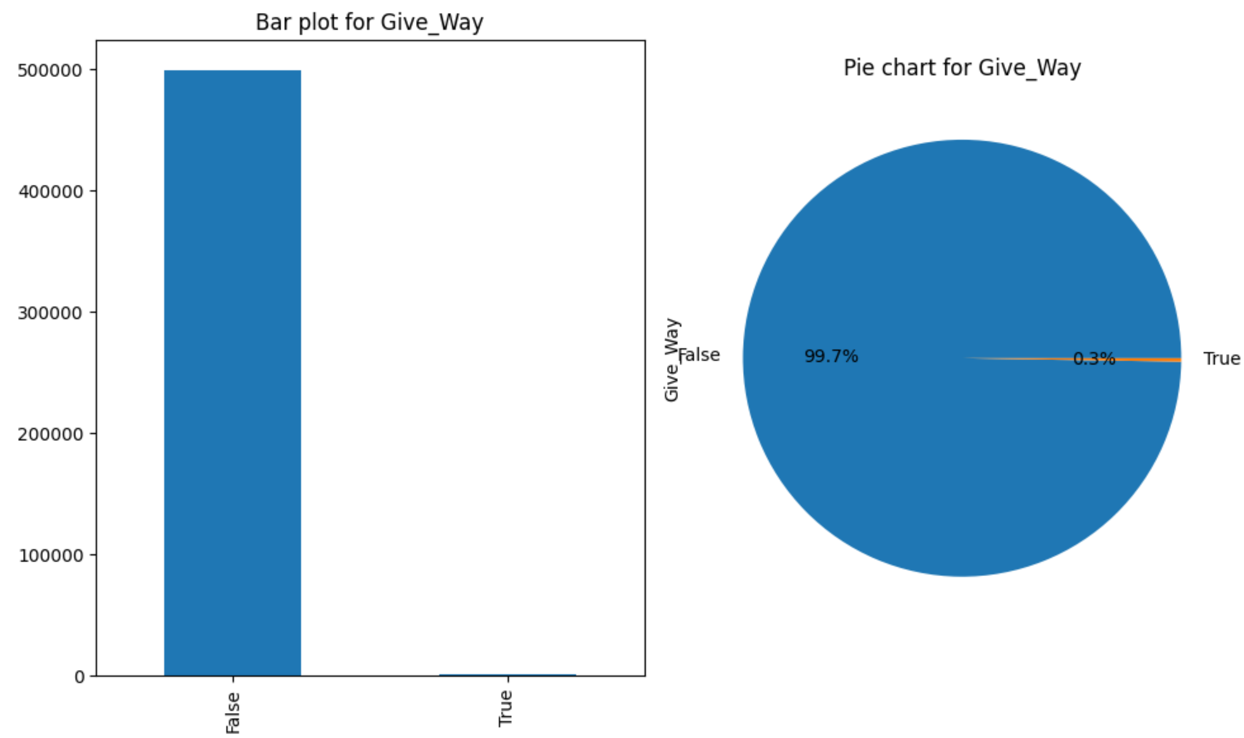


So, all the accidents had Bump so we drop the column as it has 0 variance.

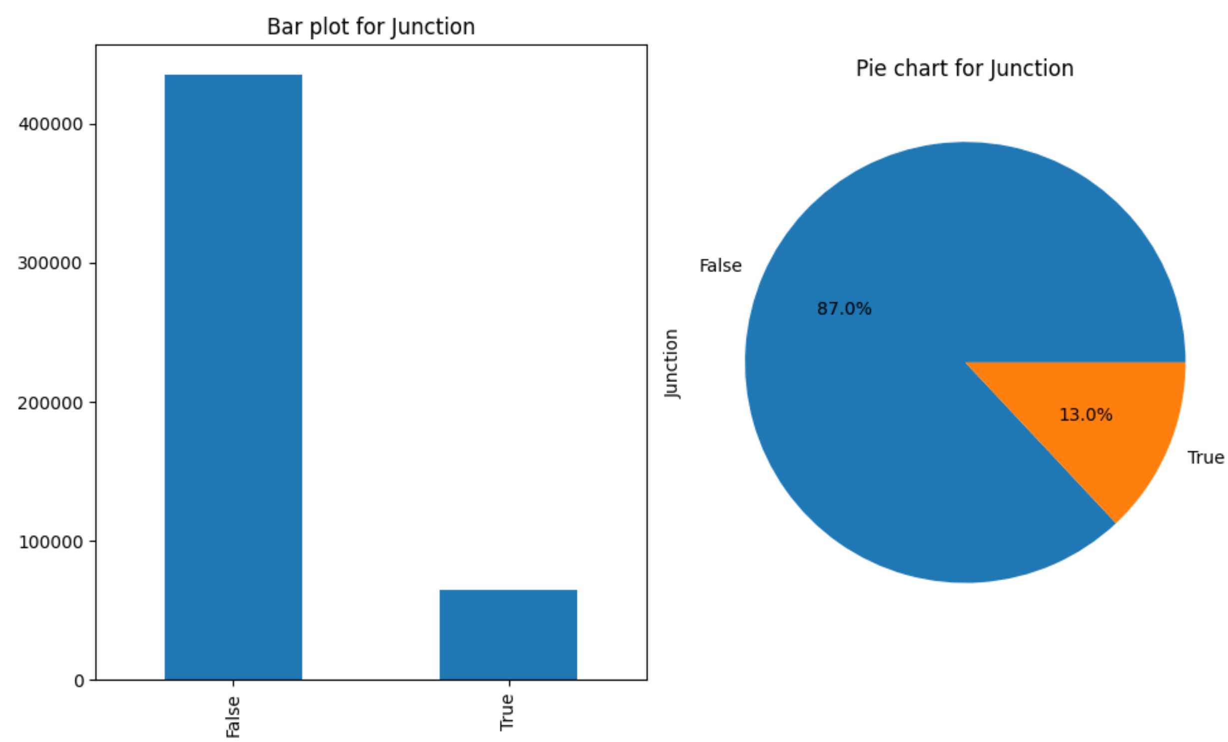




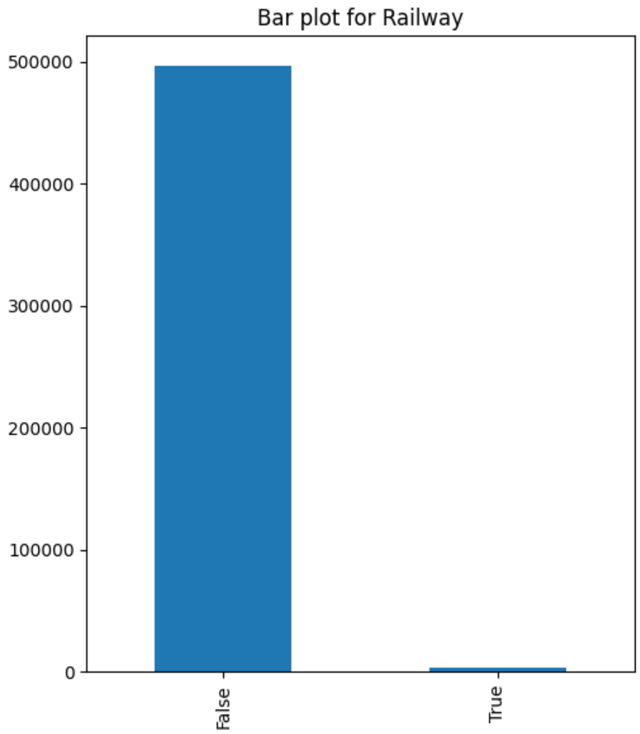
So, maximum accidents didn't have any crossing, but there were about 6% accidents which had Crossing.

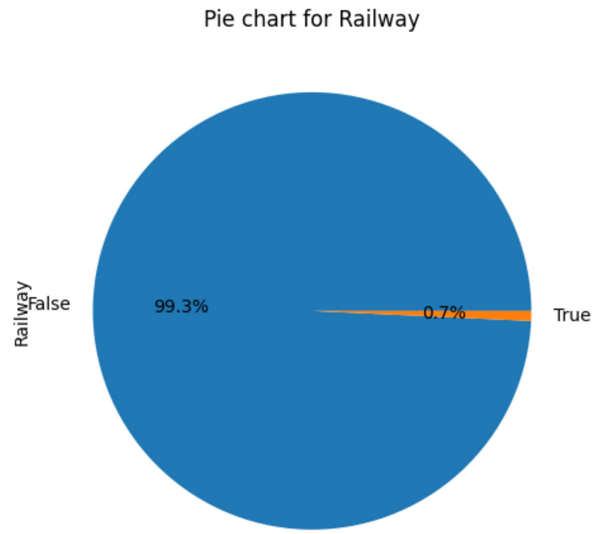


So, most accidents doesn’t have Give_Way

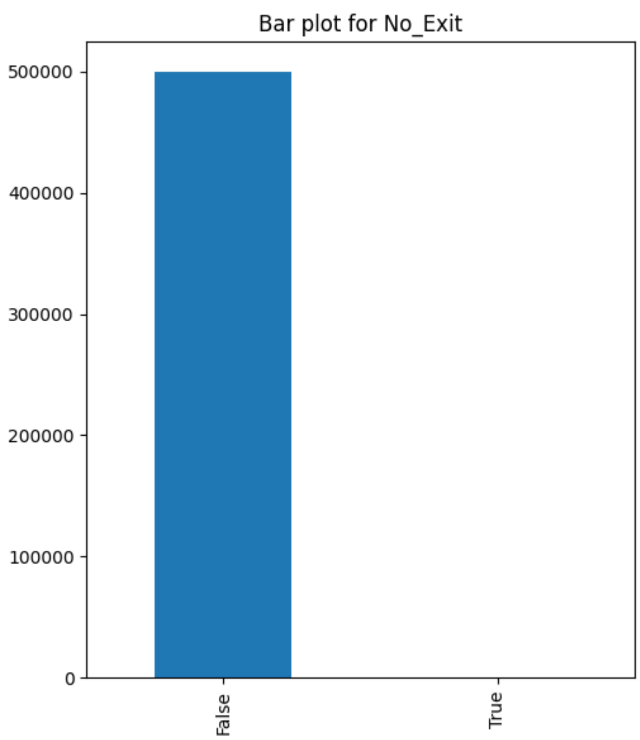


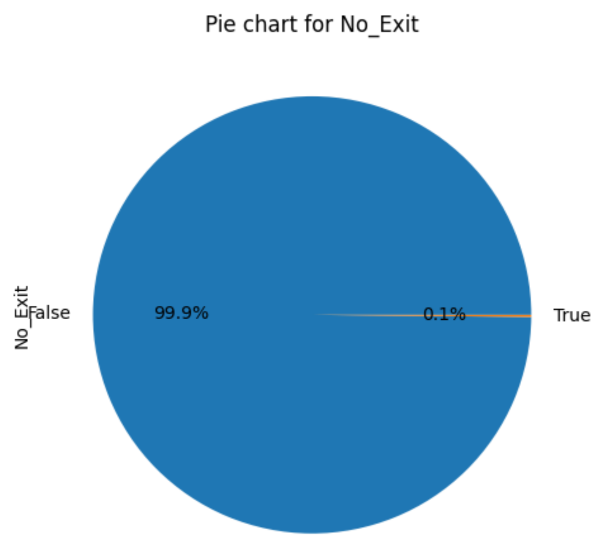
So there were many accidents having Junction but most of them don't have any junction.



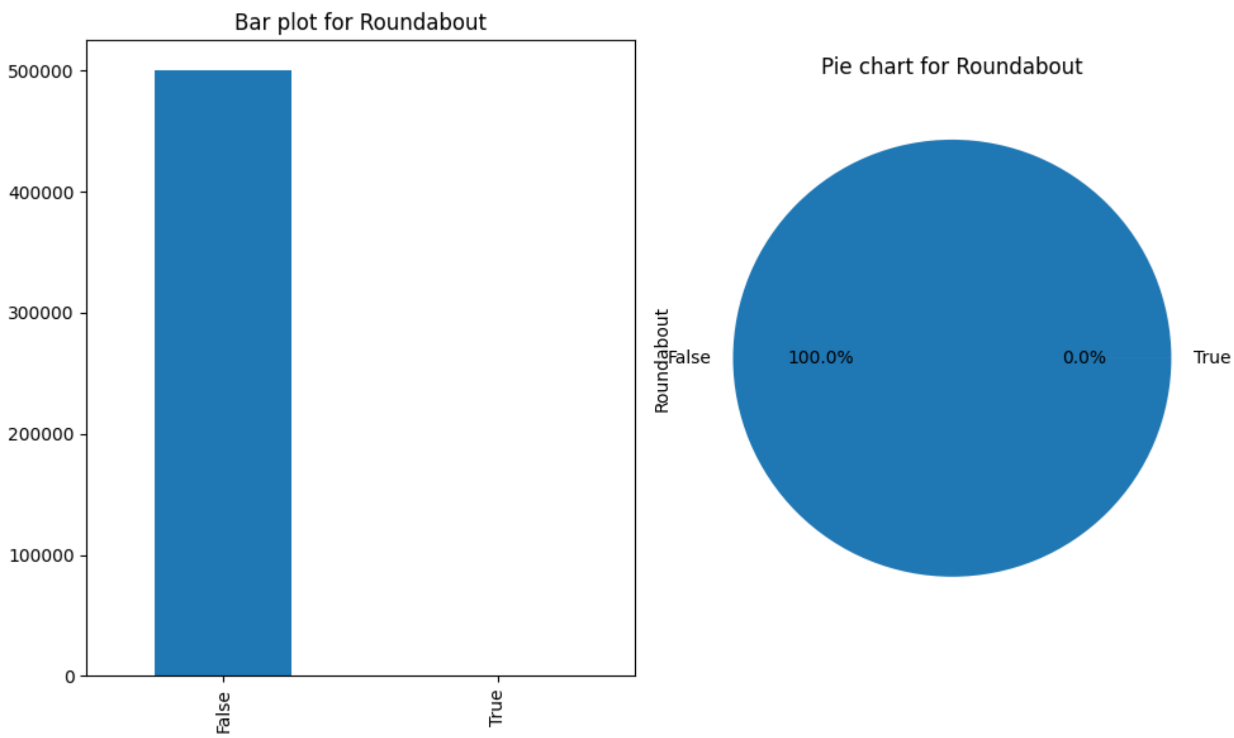


So, there were no railways in most of the accidents.

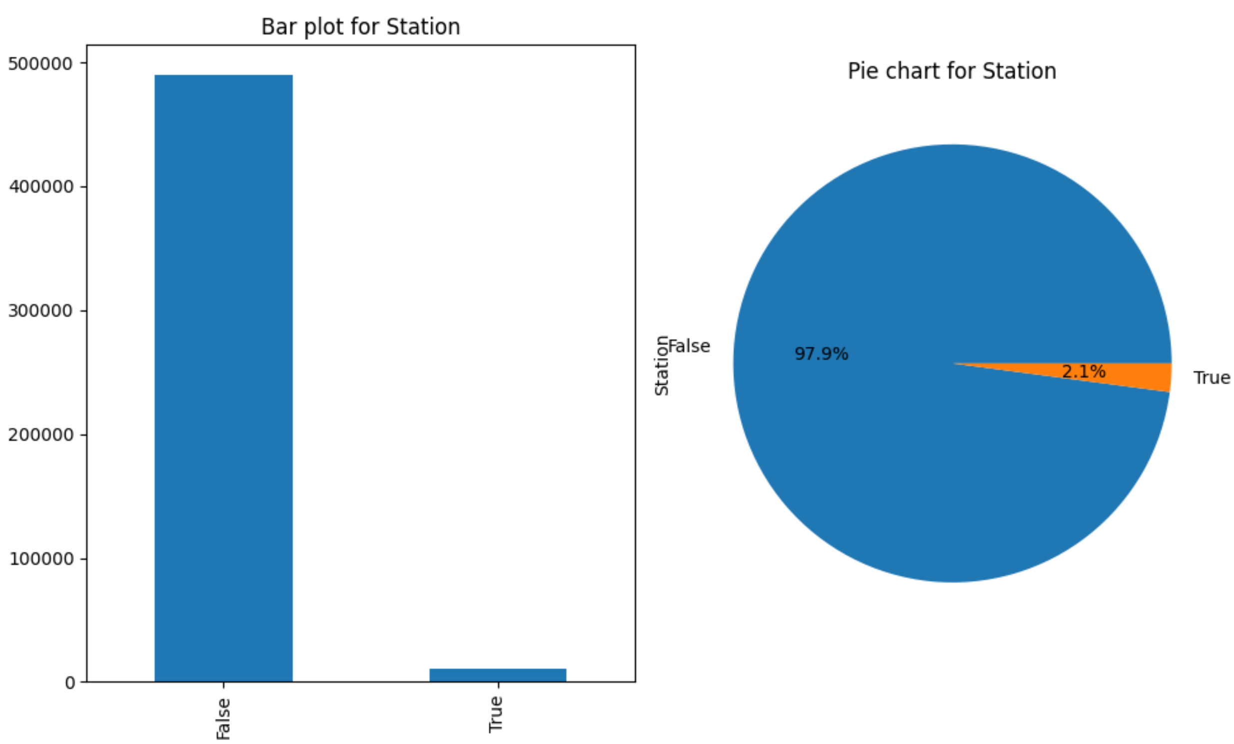




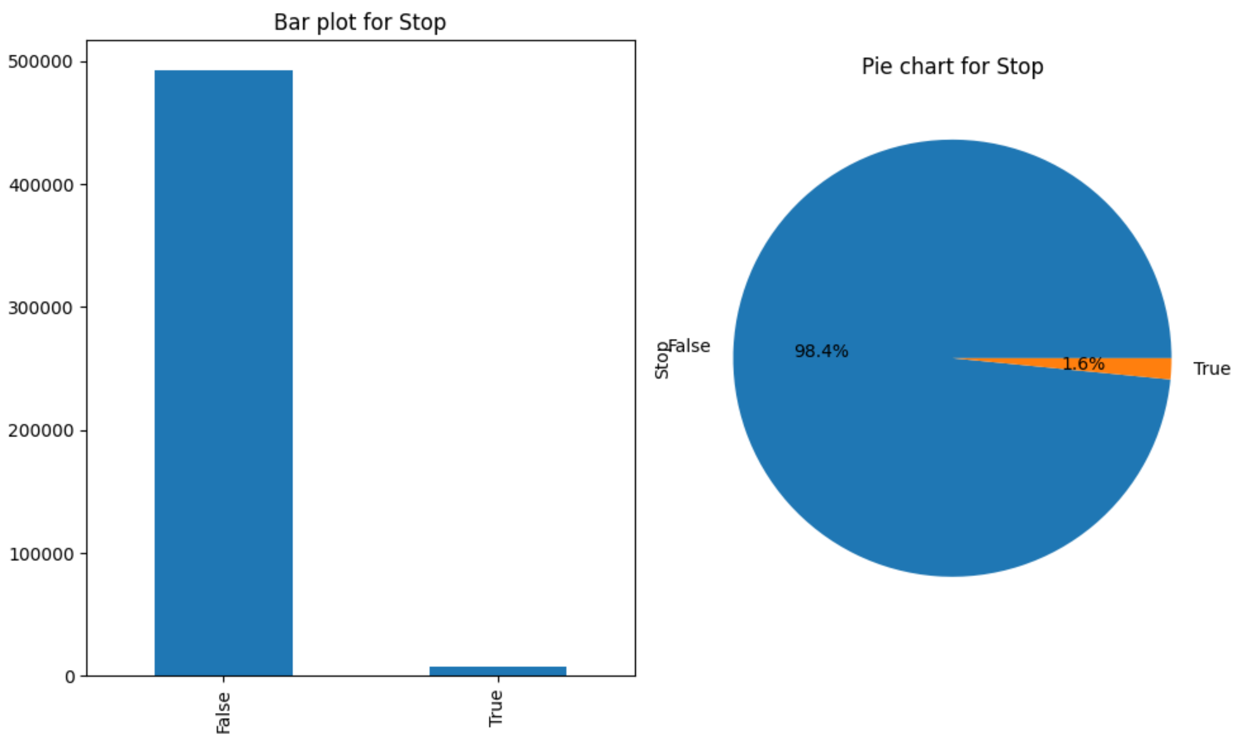
So, most of the accidents did not have any No_Exit



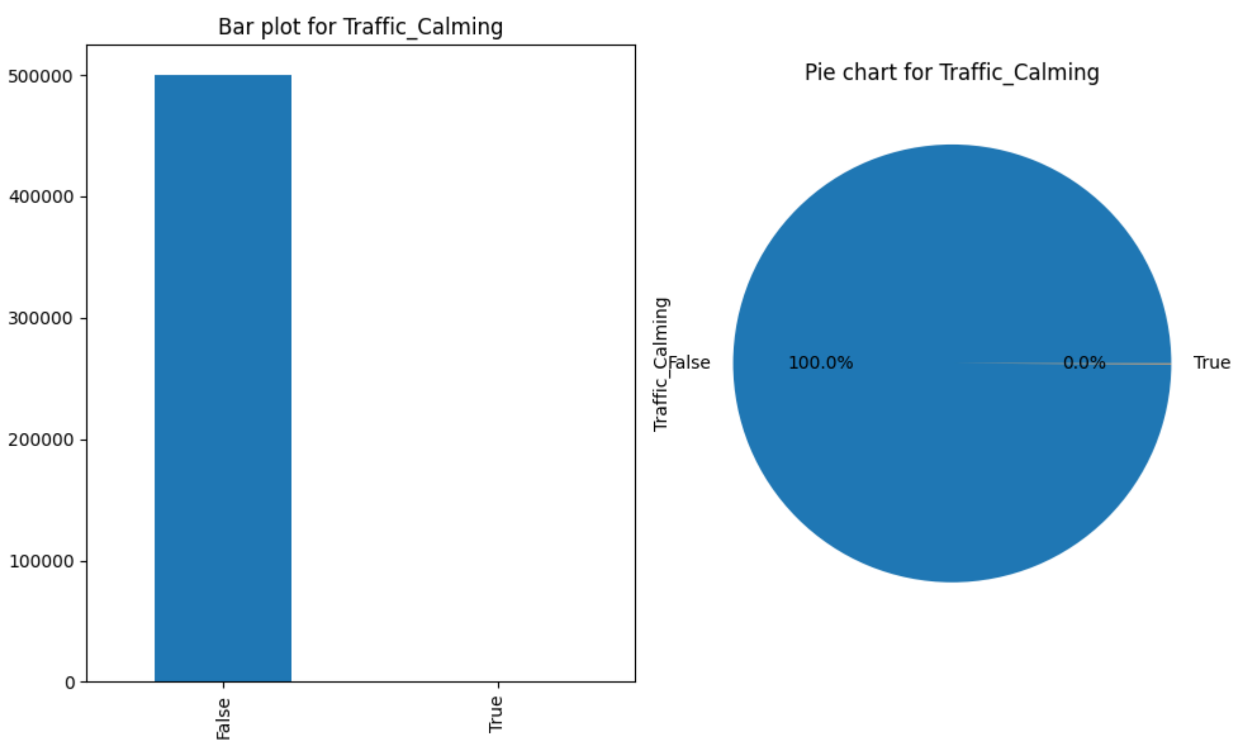
So, we dropped Roundabout as all accidents did not have any Roundabout.



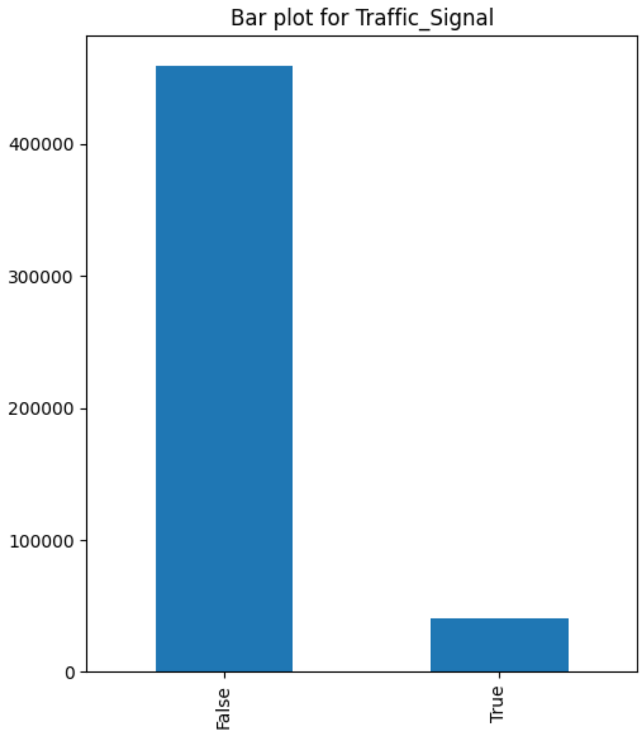
Again, there were no Station in most of the accidents

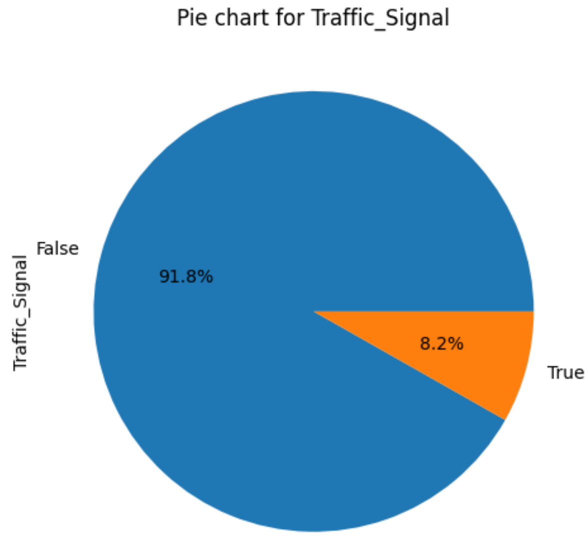


There were very few accidents having Stop.

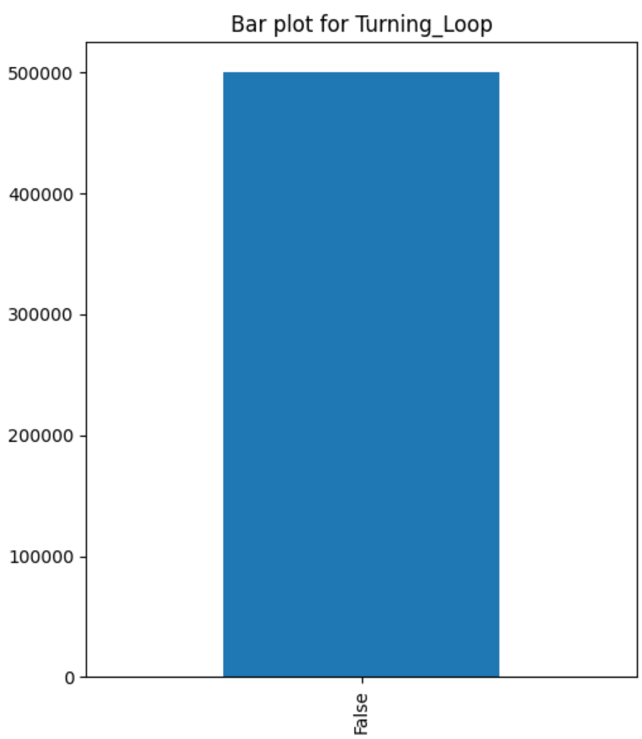


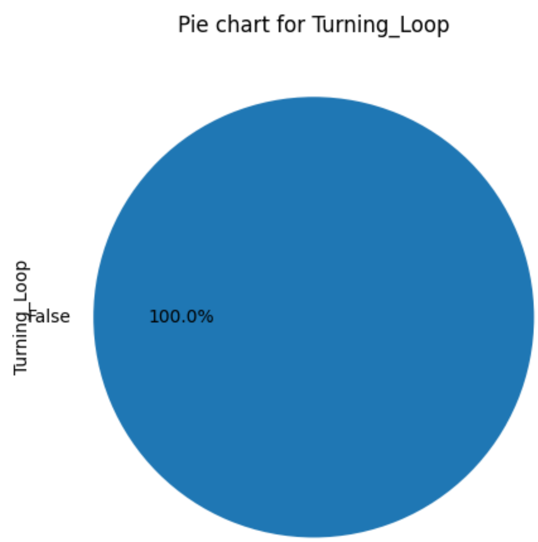
So, Traffic_Calming is dropped as there is no variance in the traffic column.





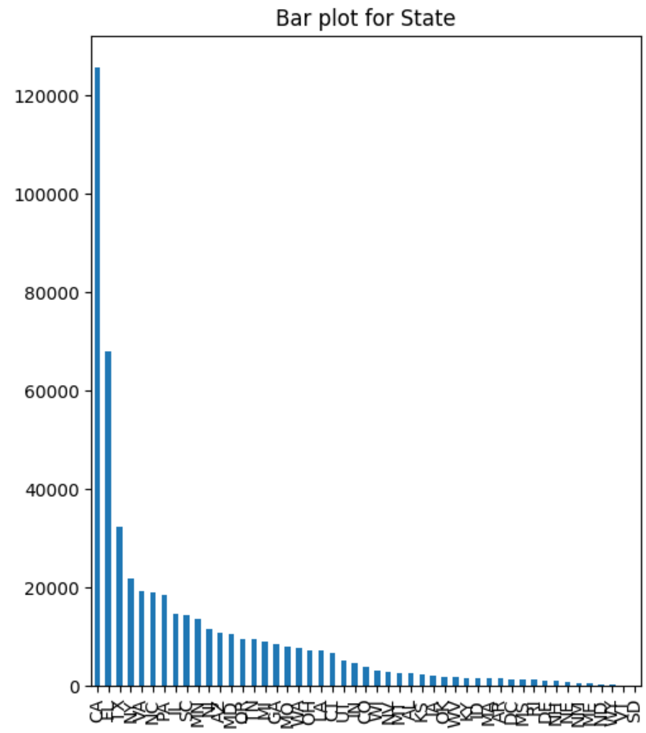
There were few accidents(about 8%) near traffic signals.

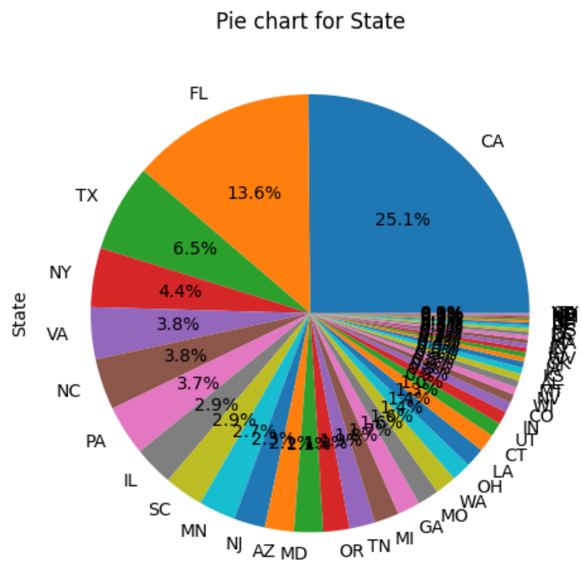




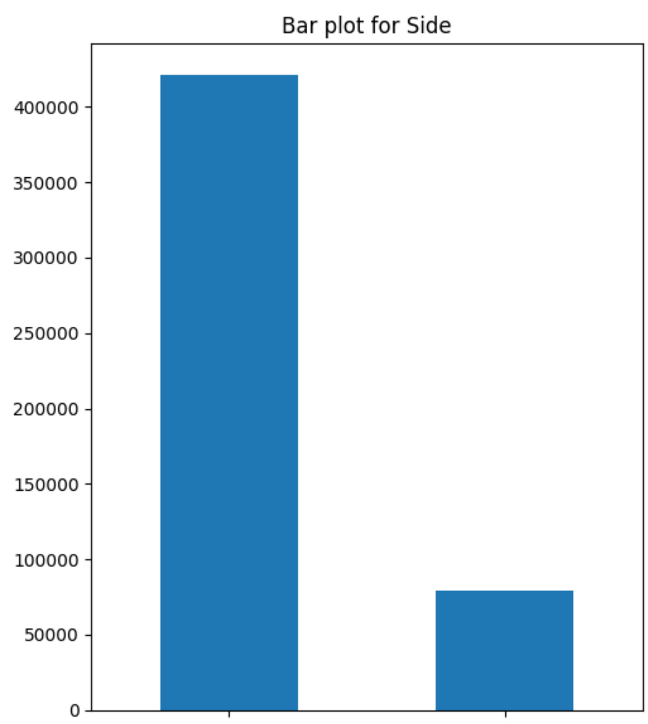
We thus drop Turning_Loop. We also drop Country as Country is US as this is US accident dataset.

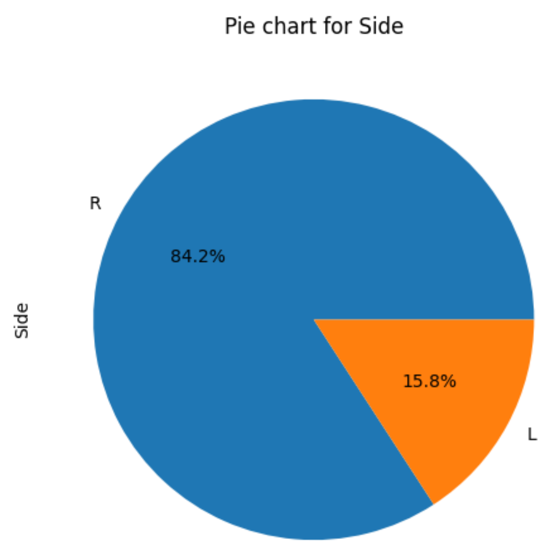
Conclusion: US accidents don’t take place much near Bump/Turning Loop/Stop but more accidents take place near Traffic signals and Junctions.



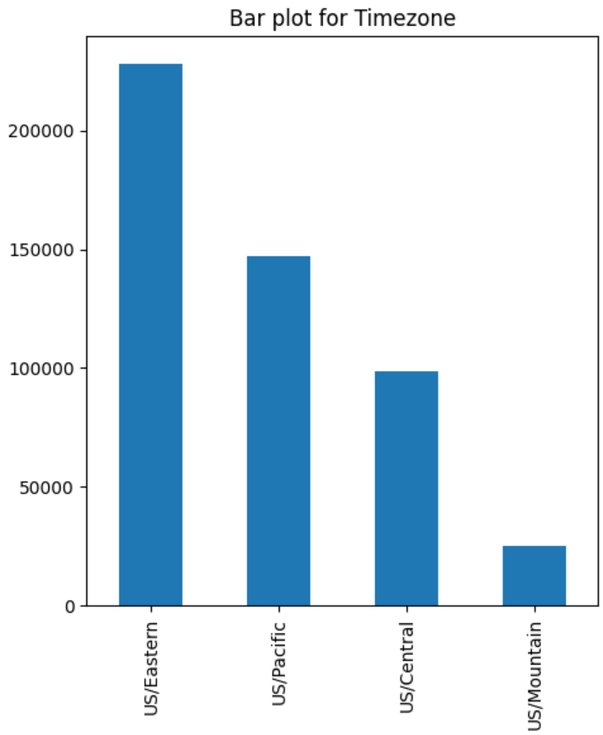


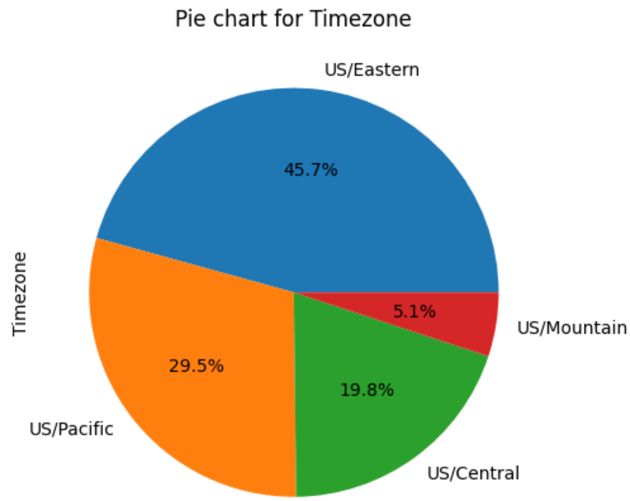
So, CA state of US have very high accidents(about 1/4th) followed by FL and TX.



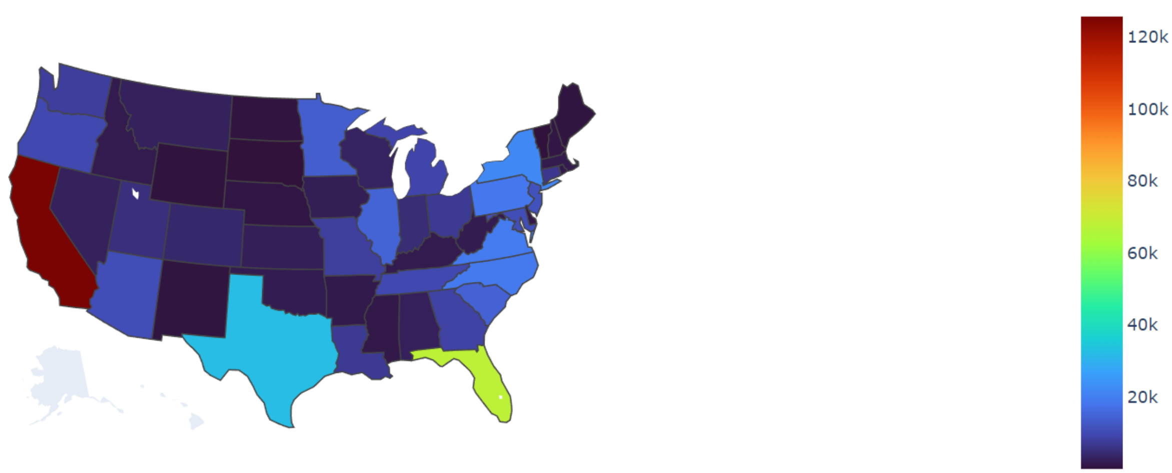


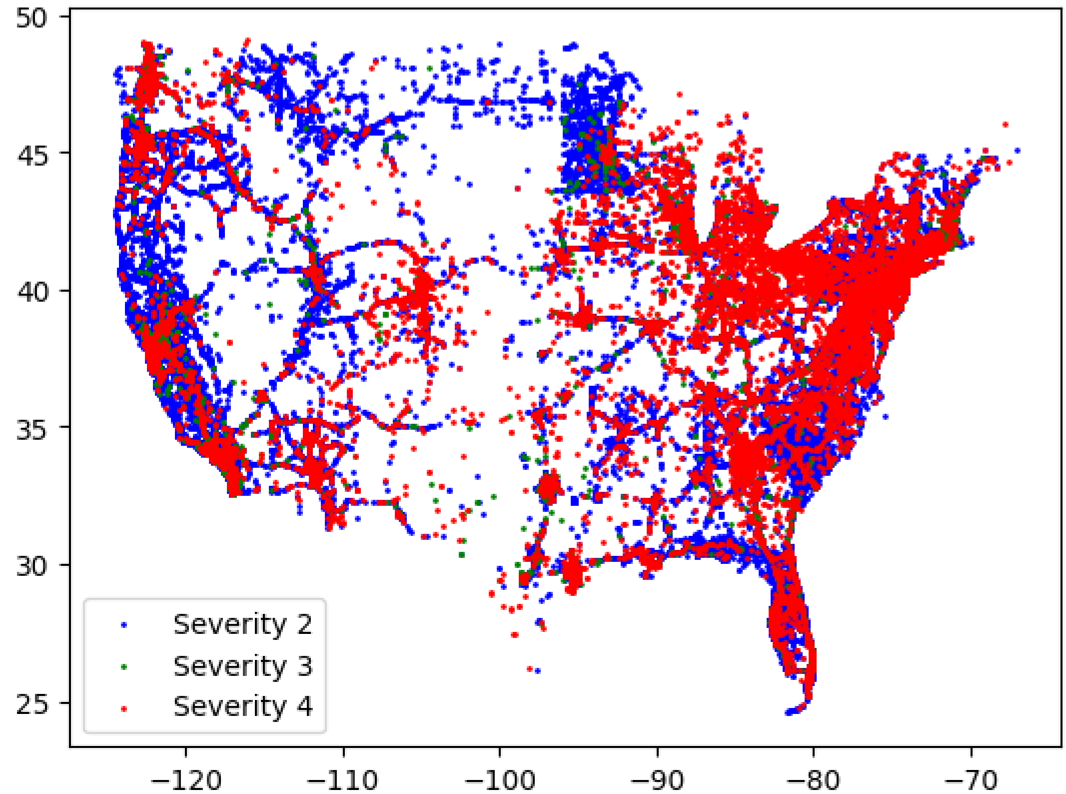
As US have Right side driving rule so most of the accidents(about 85%) takes place in the Right side of the road.



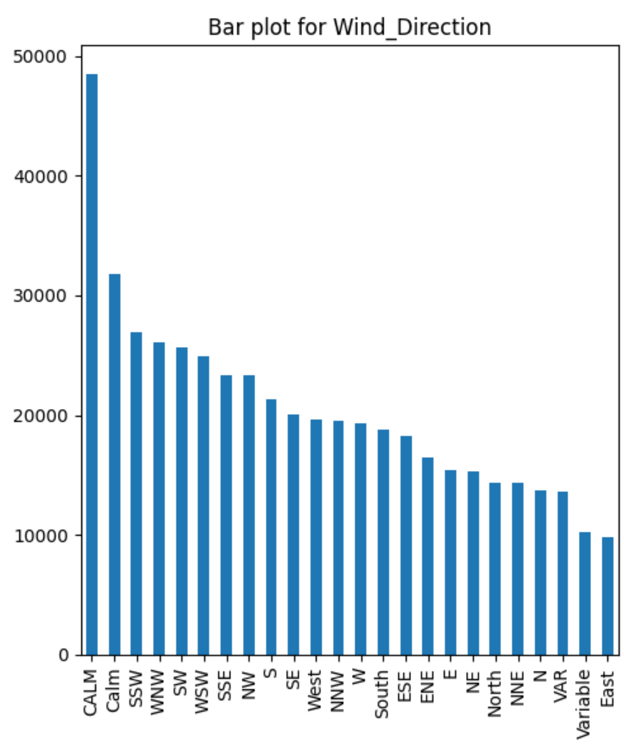


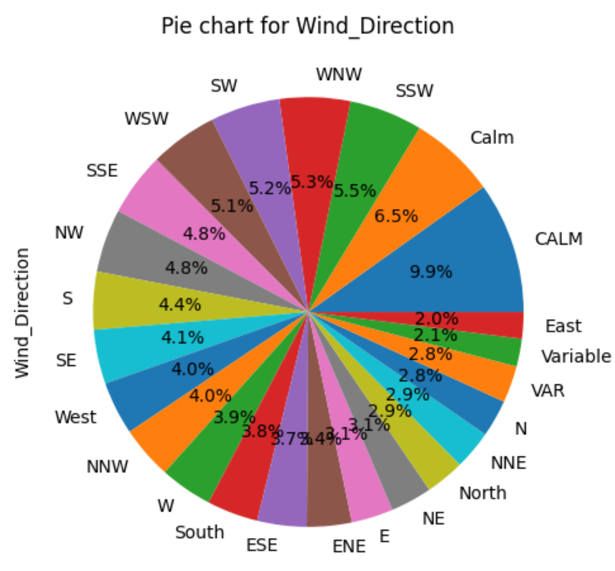
Thus, mostly accidents takes place in US/Eastern time zone region(about 50%) followed by US Pacific



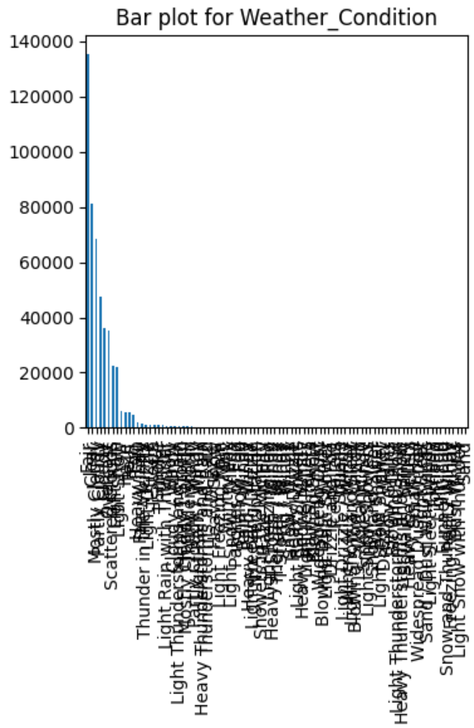


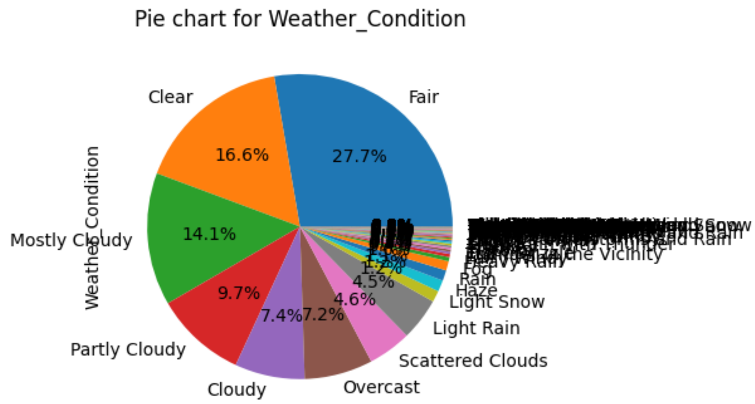
So, the coastal areas have the most high severity accidents(red ones)
This validates upon our time zone data as well.



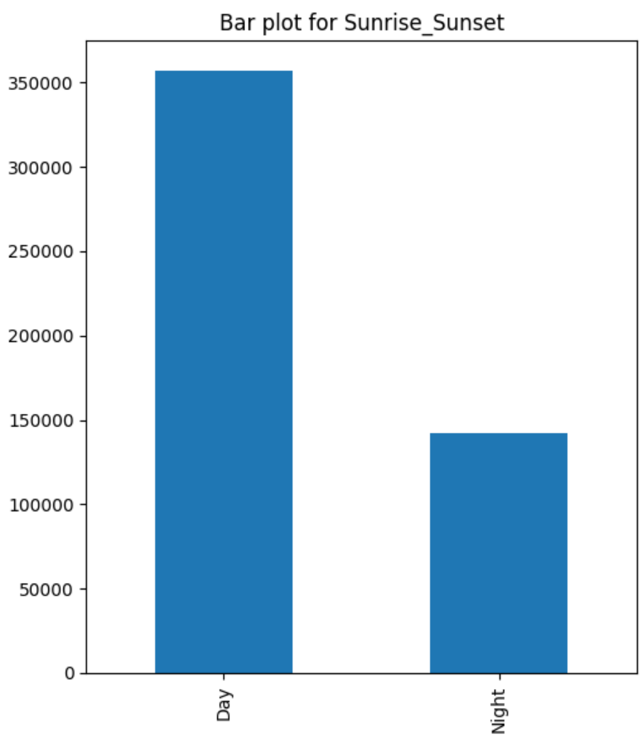


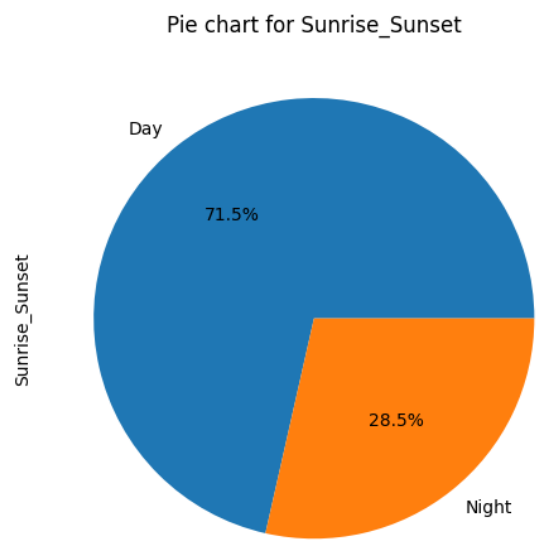
Mostly the accidents occurs when wind is absent or calm(about 15%)

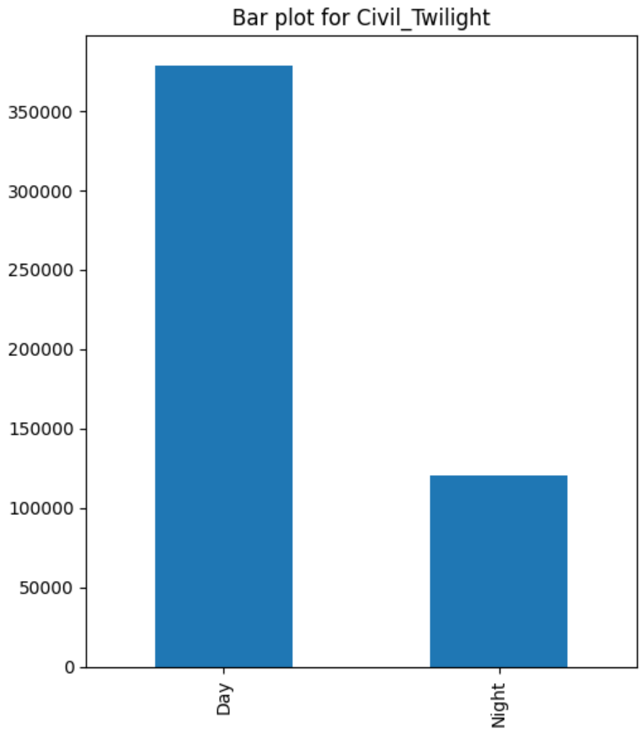


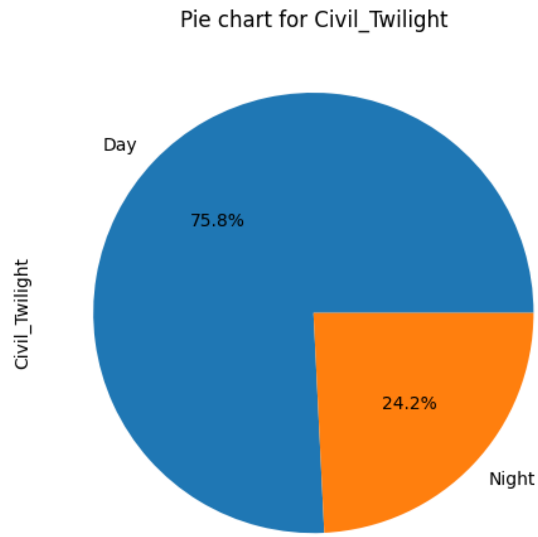


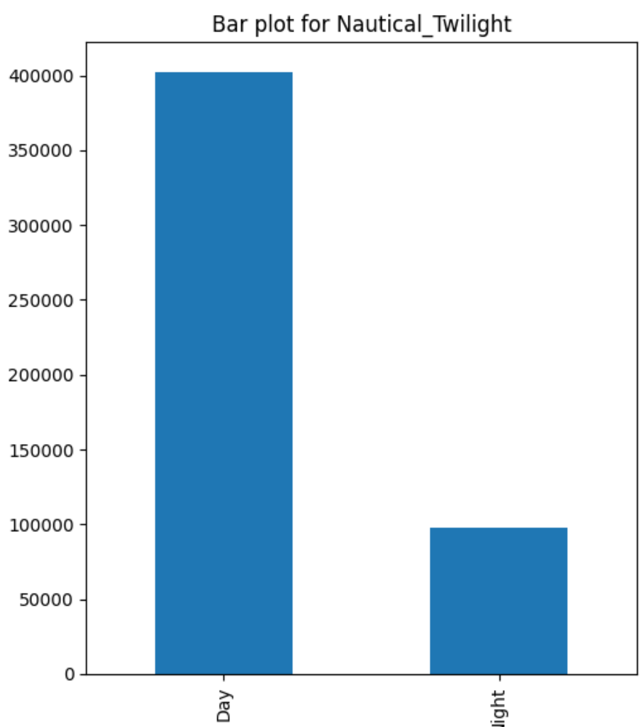
Thus when the weather is fair or clear then accidents take place. Very few accidents take place when there is rain in the US.

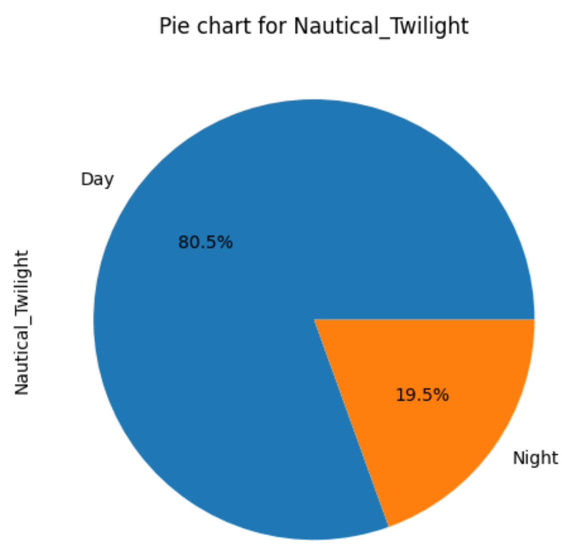


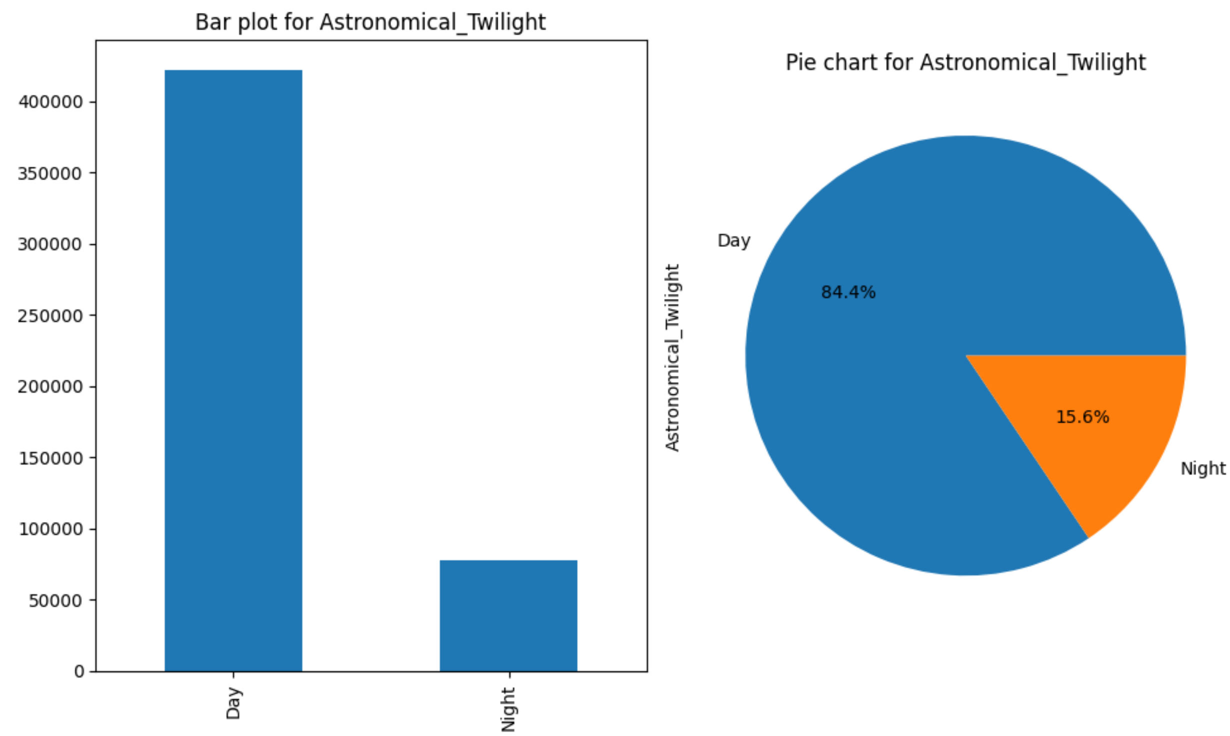



Bar plot for Civil_Twilight
350000
300000
250000
200000
150000
100000
50000
0
Day
Night

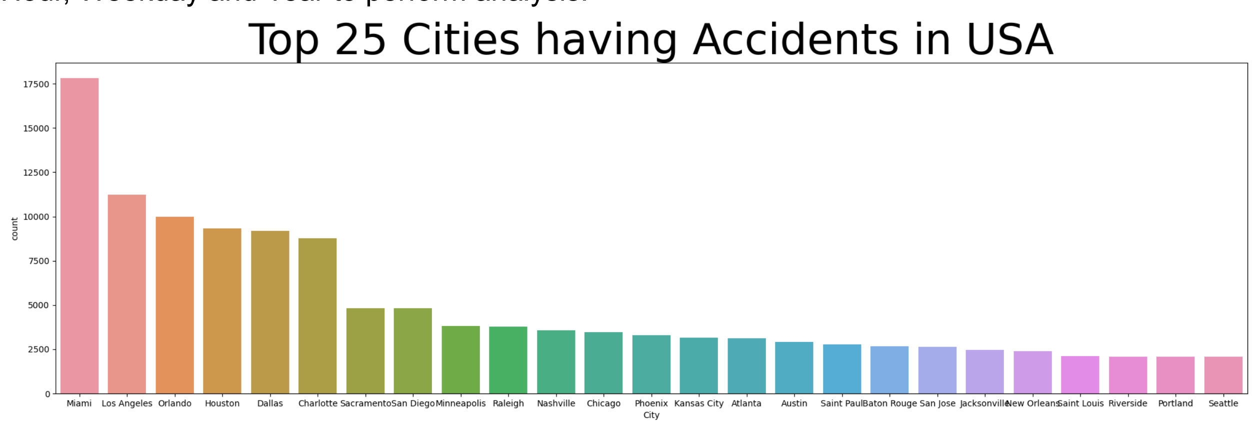

Pie chart for Civil_Twilight
Civil_Twilight
Day
75.8%
24.2%
Night


Bar plot for Nautical_Twilight
400000
350000
300000
250000
200000
150000
100000
50000
0
Day
Night


Pie chart for Nautical_Twilight
Nautical_Twilight
Day
80.5%
19.5%
Night

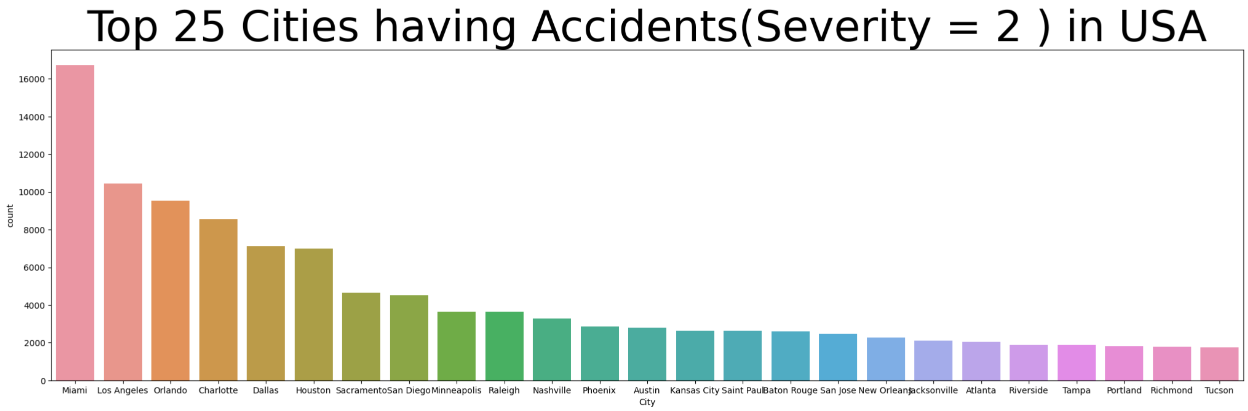


Now, we convert the End_Time of accident column into datetime and then extract Month, Hour, Weekday and Year to perform analysis.

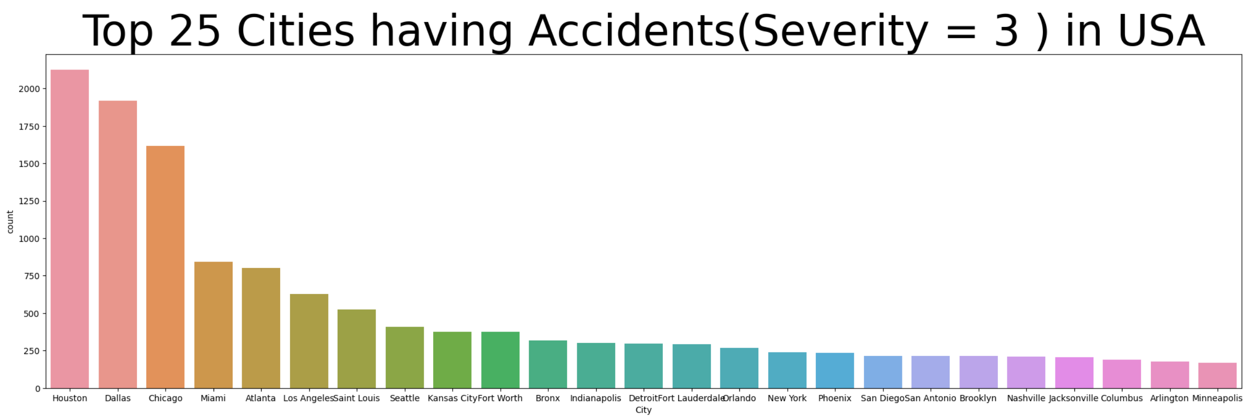


So, Miami has the highest accidents followed by Los Angeles and Orlando.

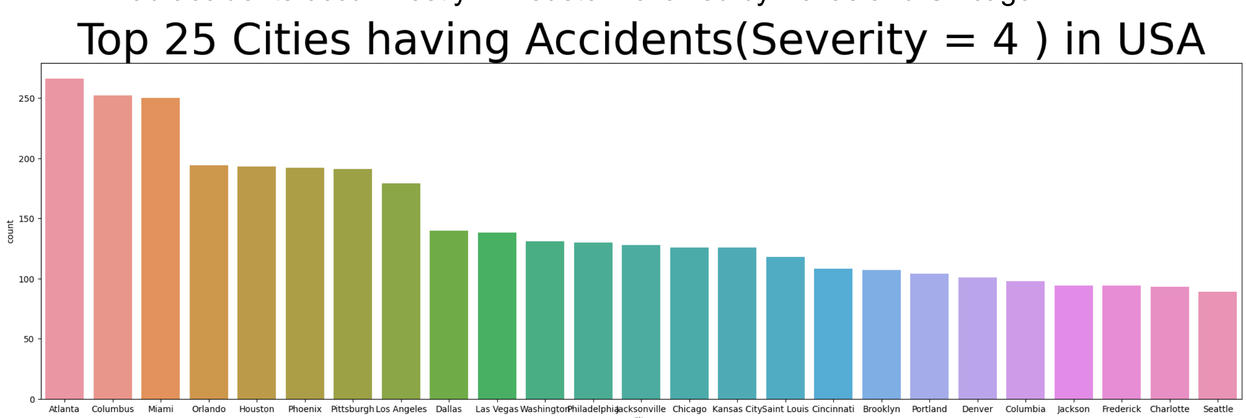
Severity wise analysis:



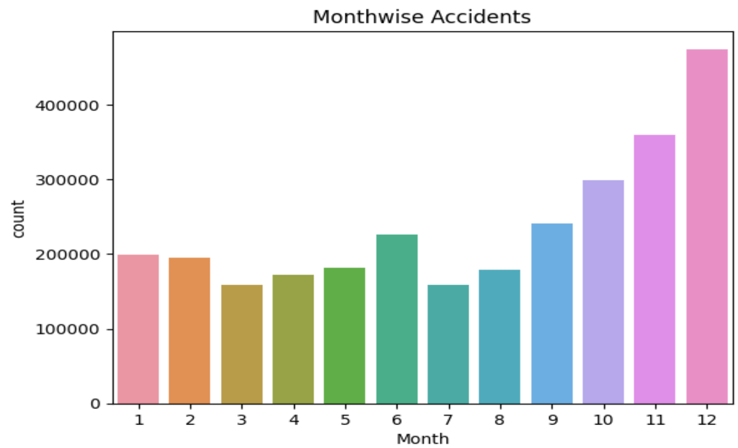
Miami still has the highest moderate level accidents



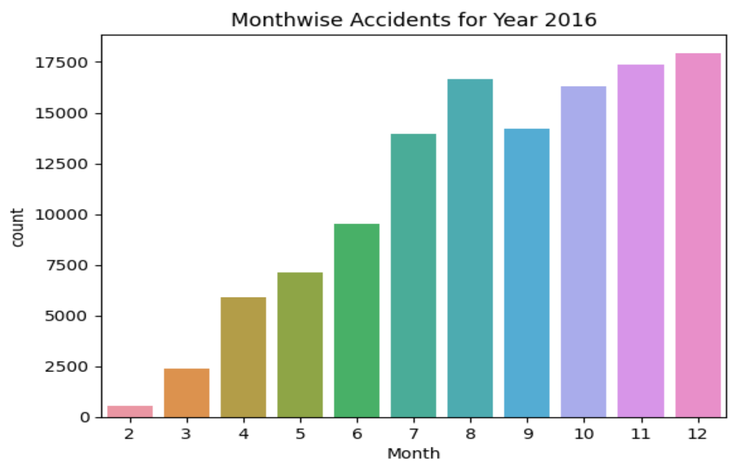
Bad accidents occur mostly in Houston followed by Dallas and Chicago



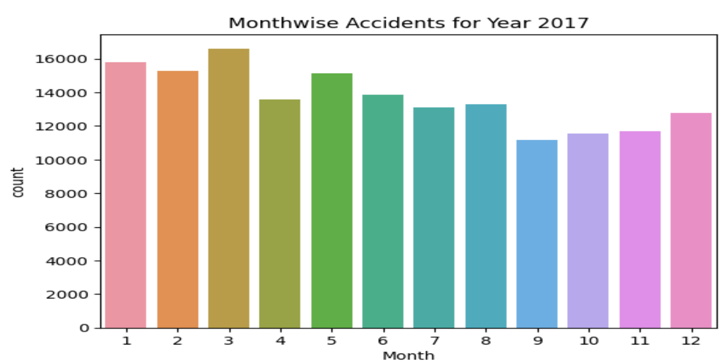
So, atlanta and columbus has the most very severe accidents



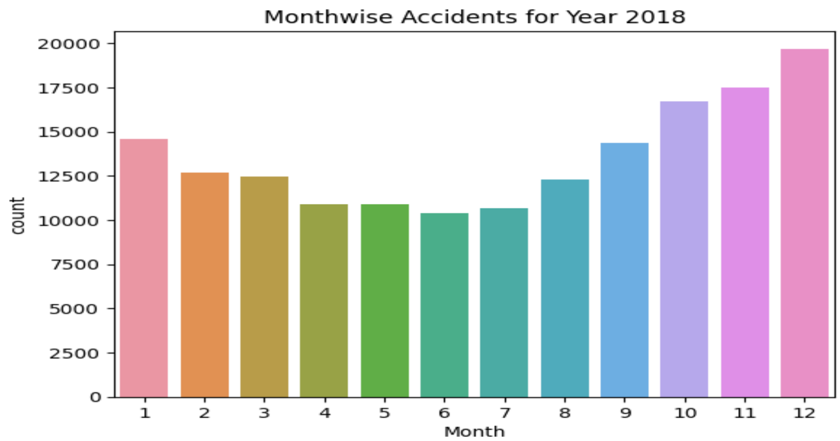
So, december has maximum accidents and July has the minimum accident



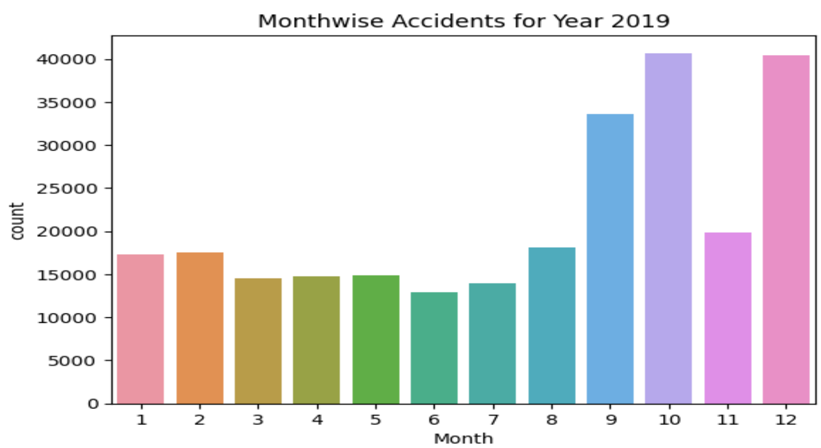
So, in 2016 also December had maximum accidents but August and July also had many accident



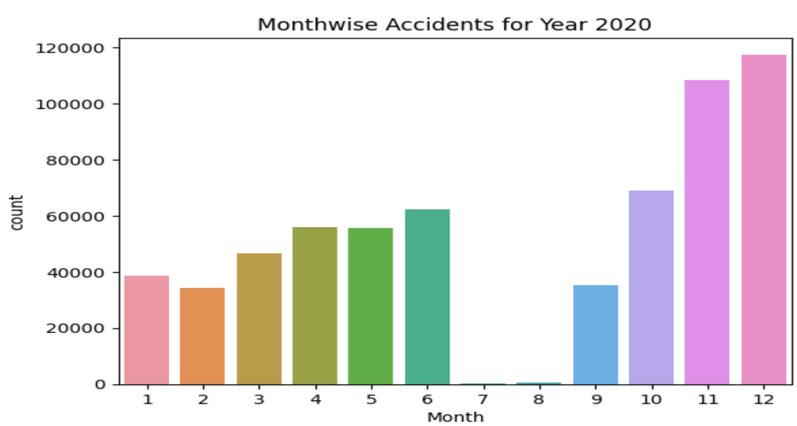
So, in 2017 March had the maximum accidents and almost all the months had maximum accidents.



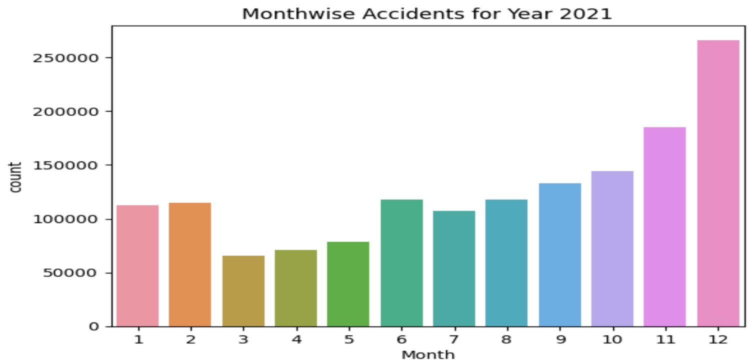
Again, december had maximum accidents

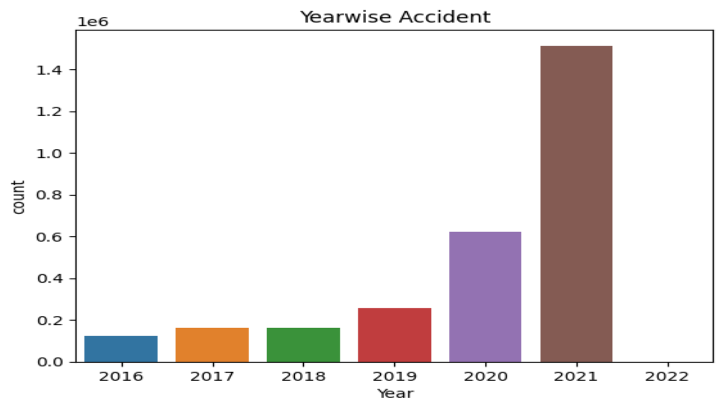


So, July had very few accidents along with June and august. October had the maximum accidents in 2019

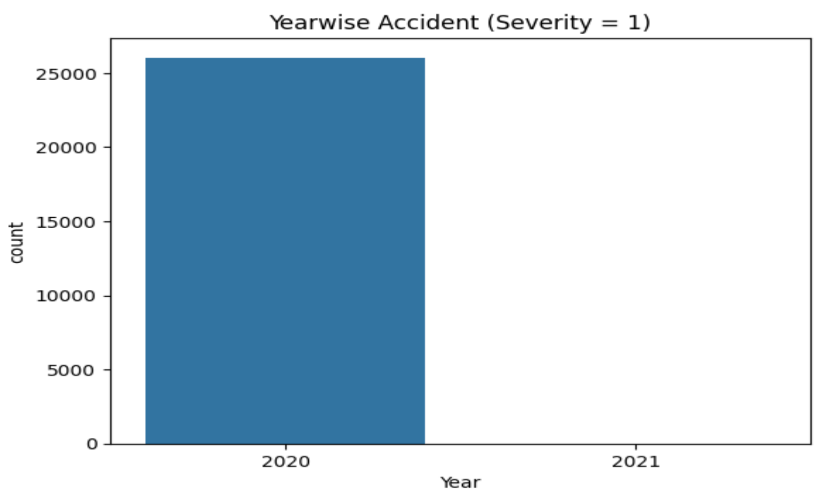


So, due to lockdown, accidents were very few near July and August.

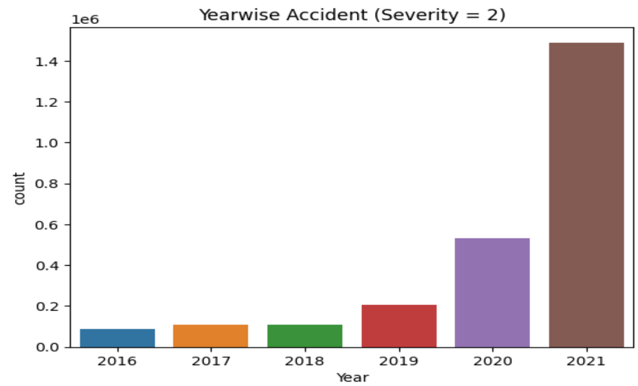




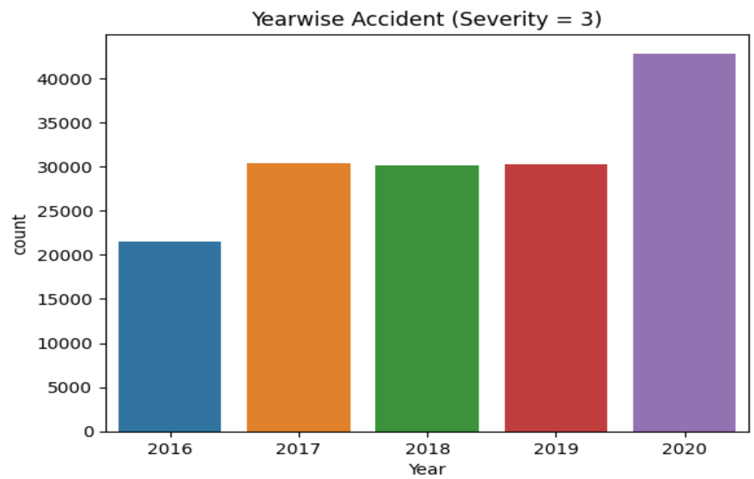
So, in the US, accidents increase exponentially per year.



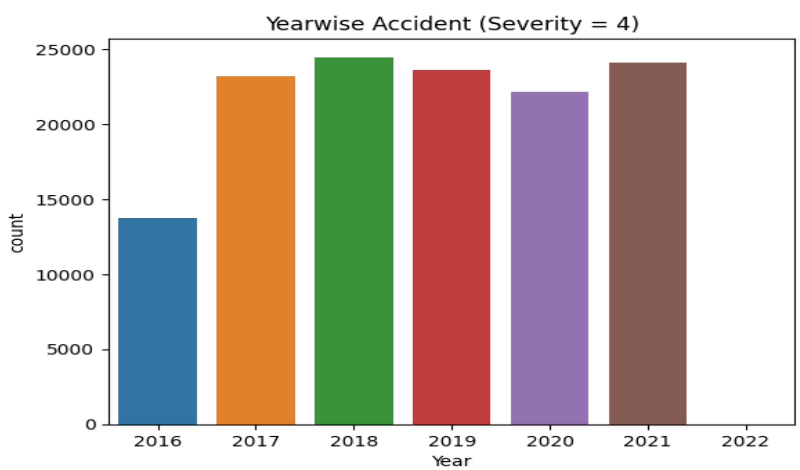
So, only the year 2020 and 2021 had Severity had very low severity accidents this can be due to advanced car safety measure in the recent years



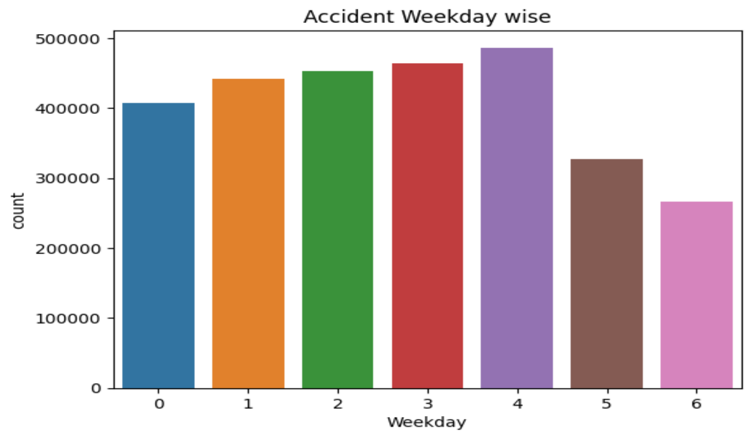
So, severity 2 accidents increase exponentially every year



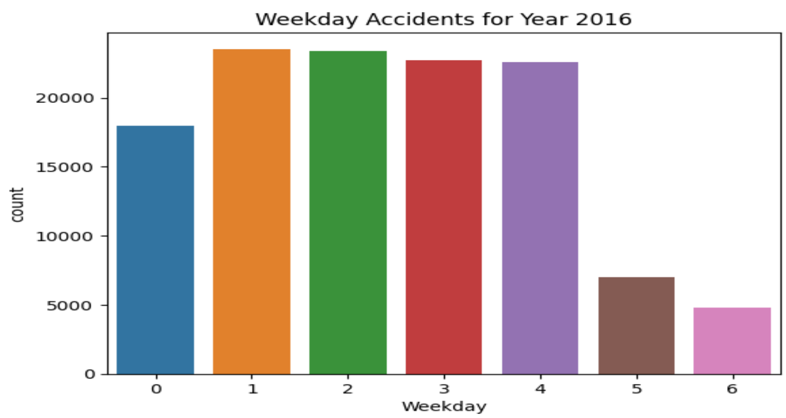
So, there are no high severe accidents in 2021 and high severity accidents does not increase exponentially thanks to safety equipments



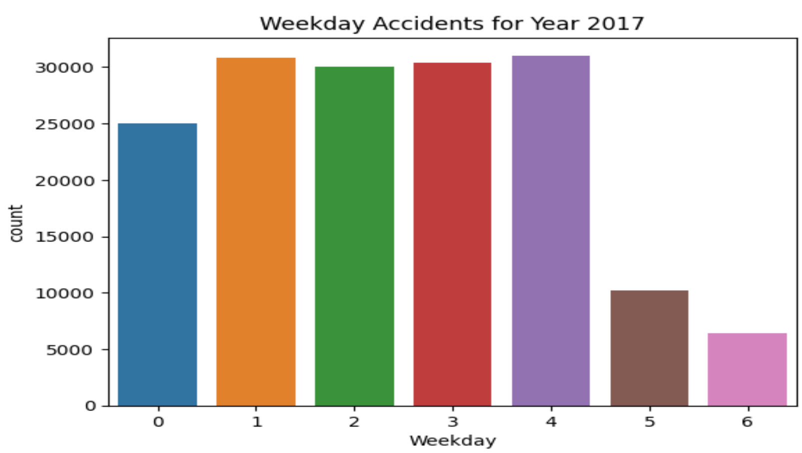
Again, very fatal accidents are not increasing exponentially every year. So exponential increase in accidents is due to exponentially increasing moderate severity accidents

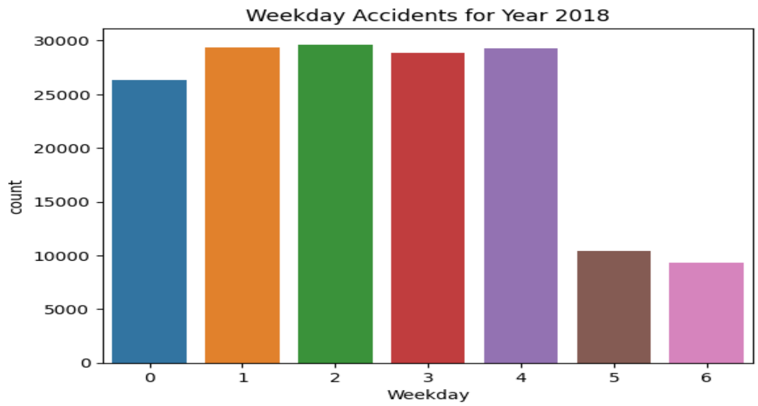


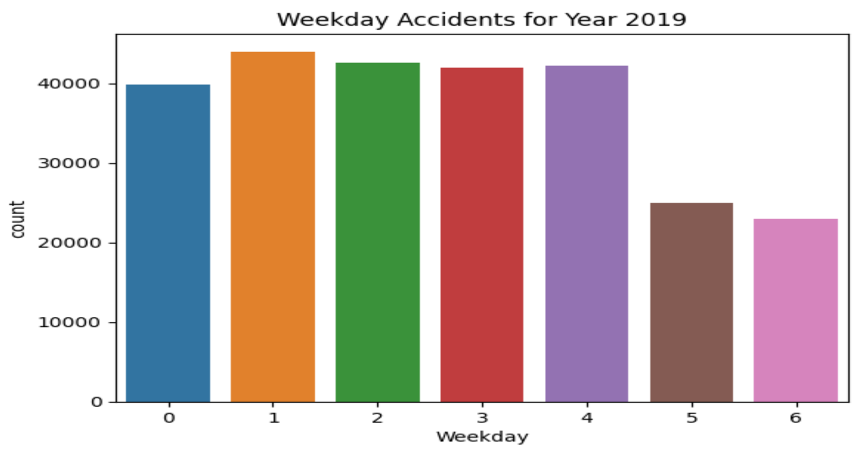
So, Weekends have least accidents in the US

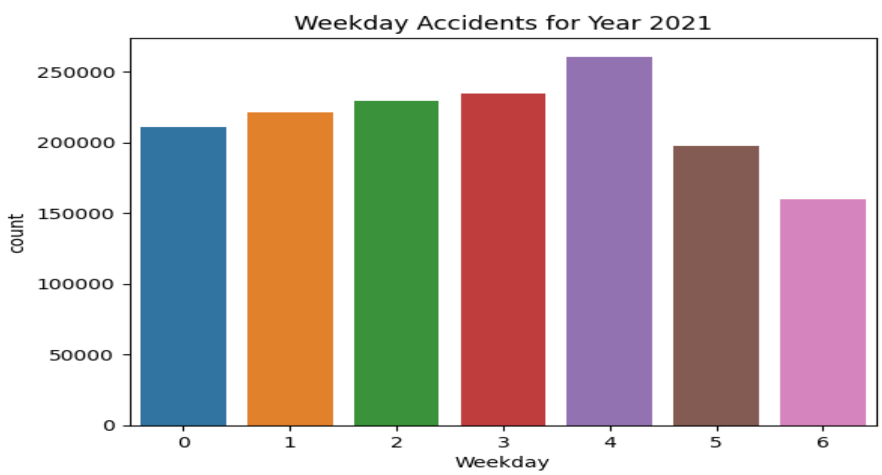


Weekends had least accidents in 2016

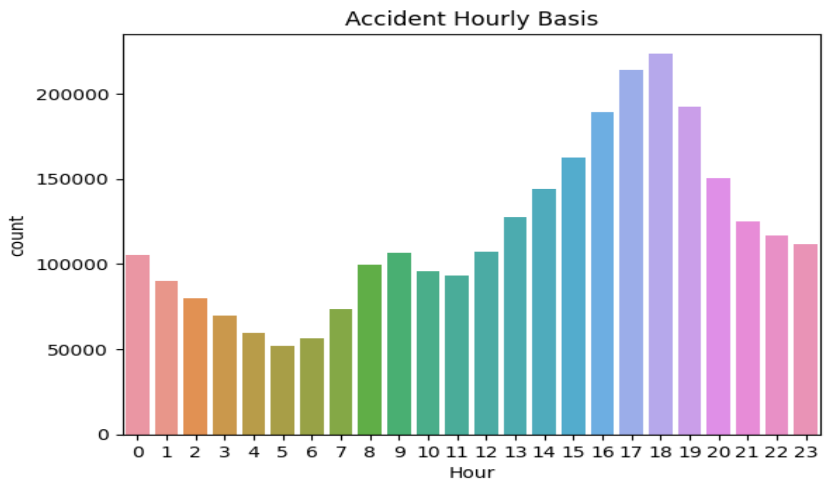




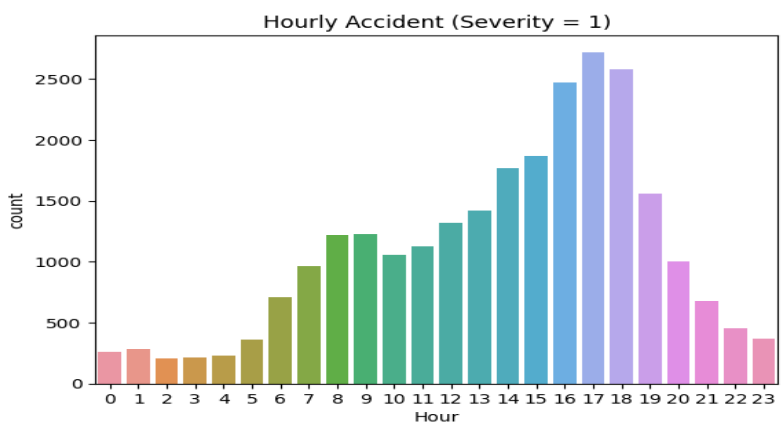




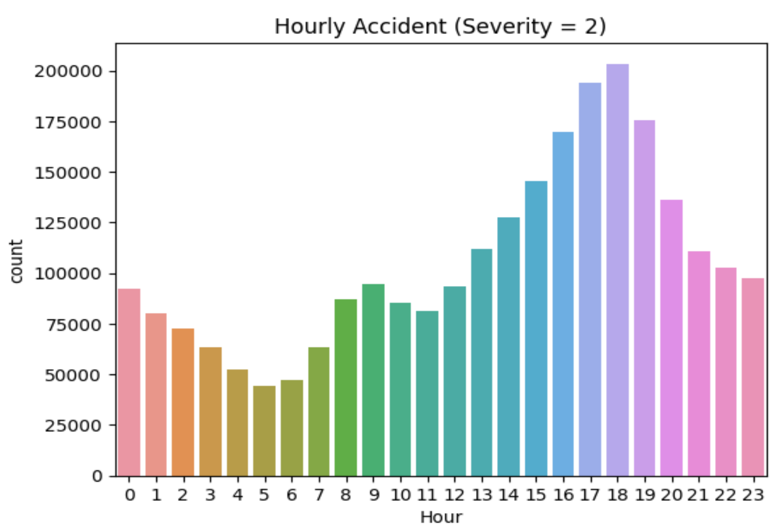
So, all the years had a similar pattern in accidents with weekends having the lowest accidents and Friday and Thursday having the maximum accidents.

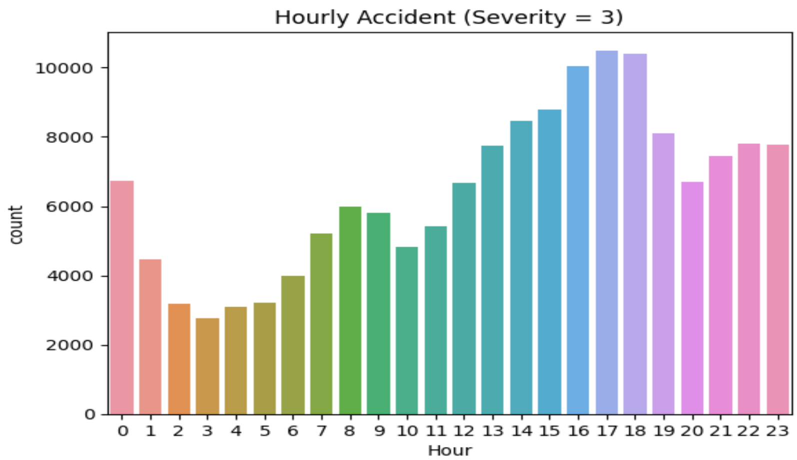


6 pm had the maximum accident as that was the hours when work got over and people went back home using the road. 5 am is usually a very non busy hour and thus it has the lowest accident rate.

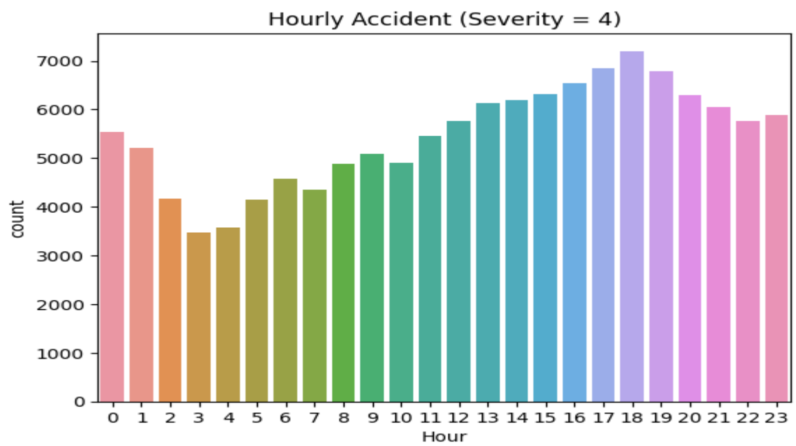


5 pm and 6 pm had the maximum amount of accidents.

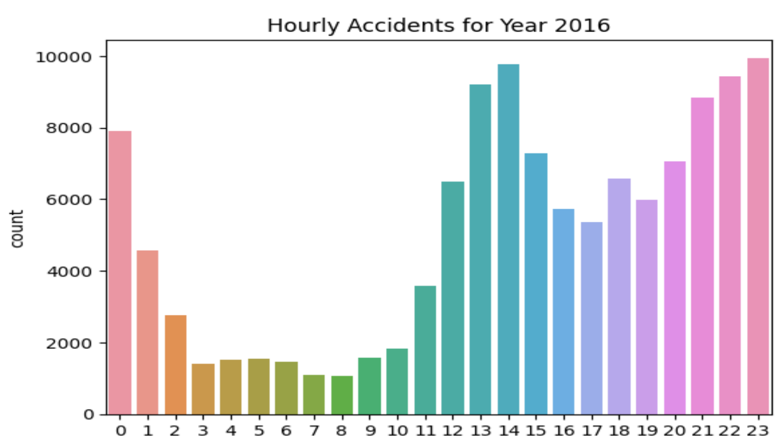




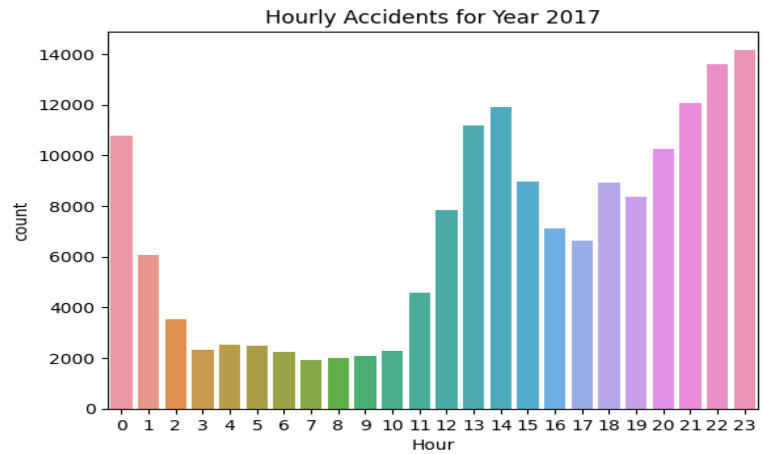
12 am accidents with high severity is also very high due to people drinking and driving the maximum during that time

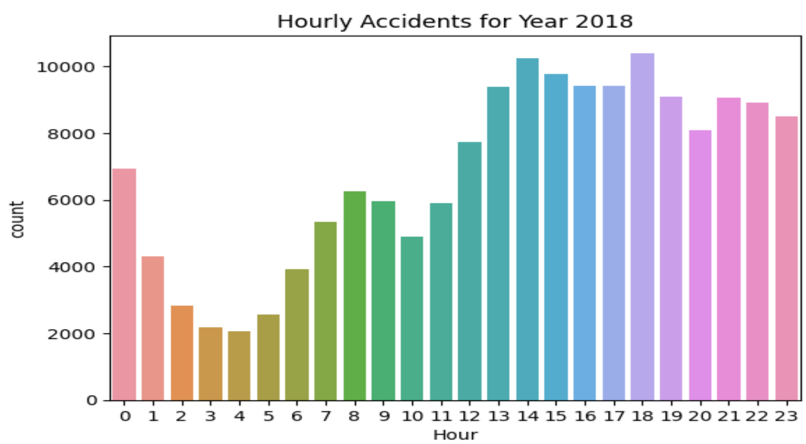


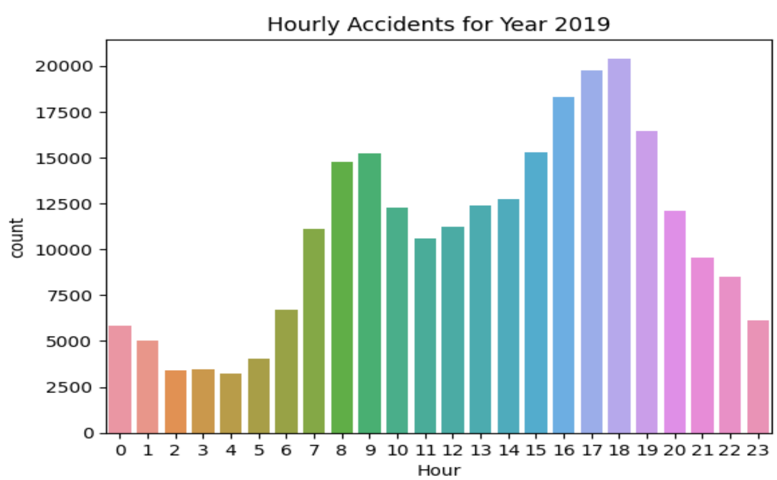
Very severe accidents occur almost every year.

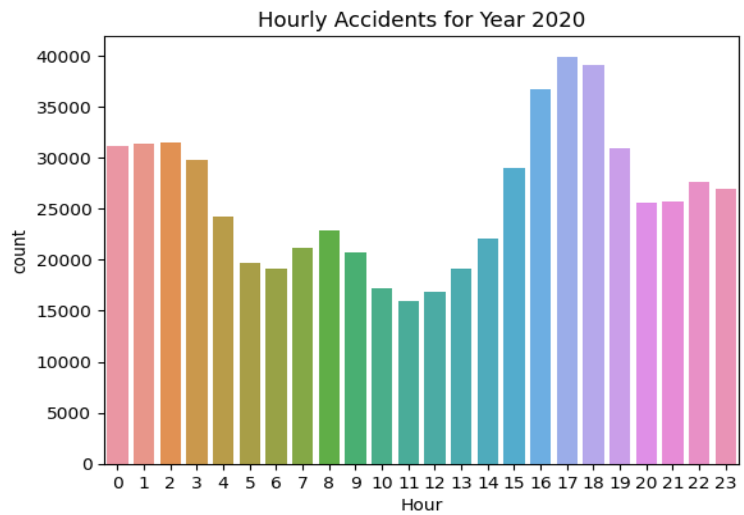


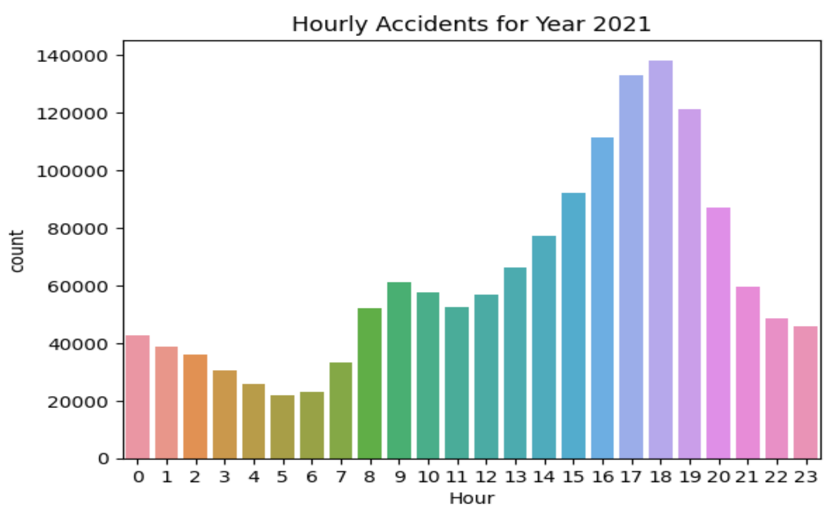
2016 had the maximum accidents near 11 pm,2 pm and 12 am.


Hourly Accidents for Year 2017
count
Hour

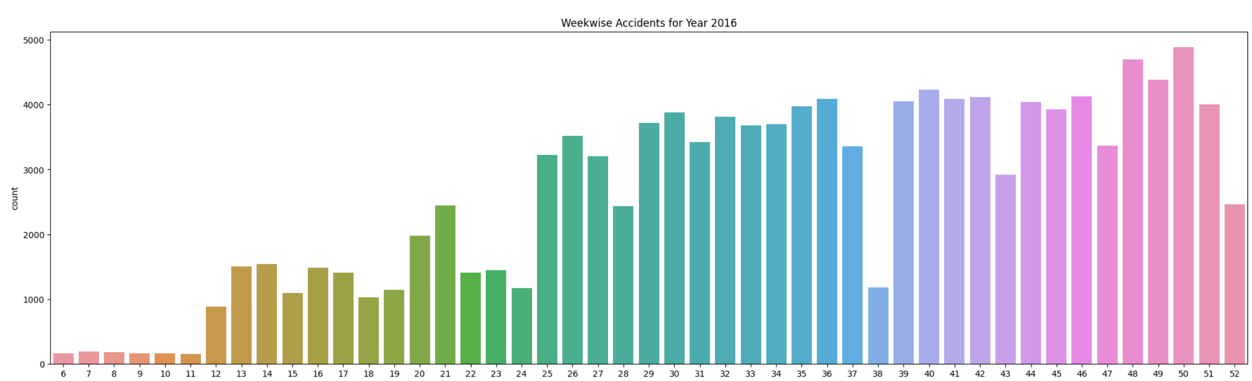

Hourly Accidents for Year 2018
count
Hour


Hourly Accidents for Year 2019
count
Hour

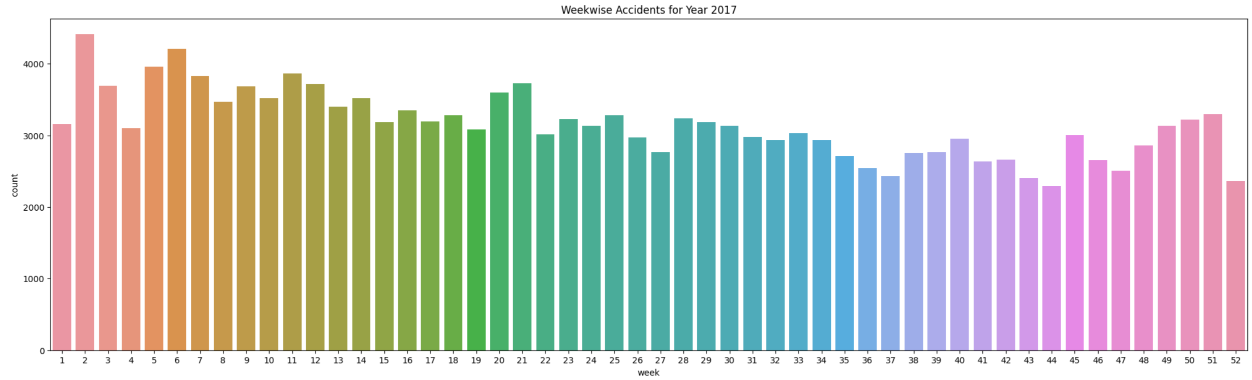




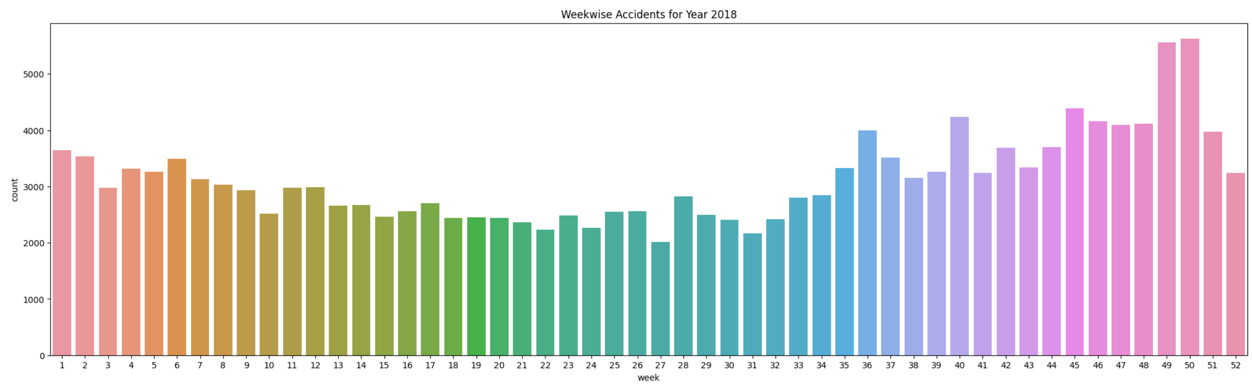
2021 having very high number of accidents had most accident near 6pm and thus that influenced the data as well for having high accidents near 6 pm.



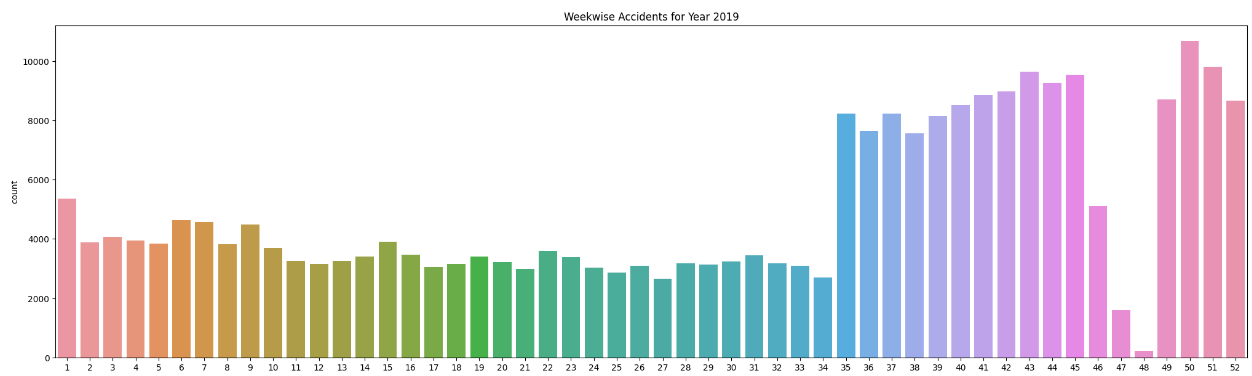
There were very few accidents in the starting weeks of 2016



2017 had an almost uniform rate of accidents throughout the year.

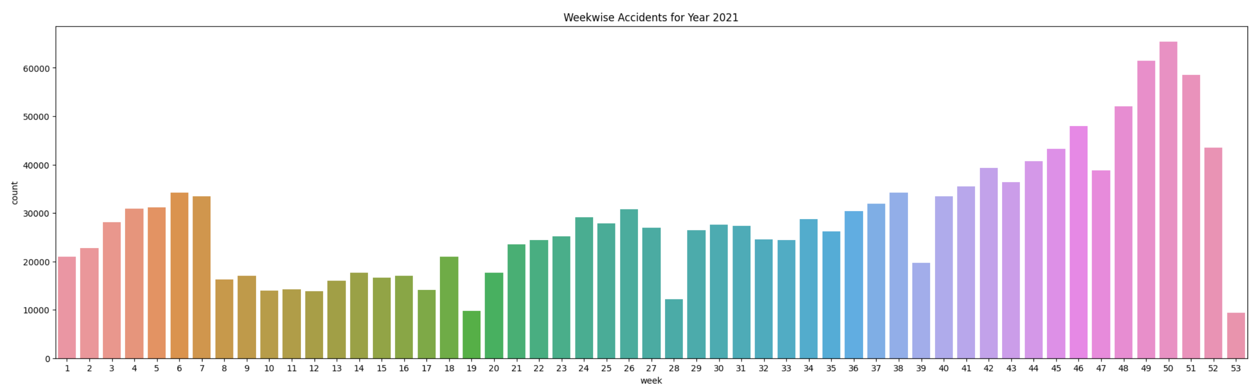


2018 had suddenly very high number of accidents in the 1st and 2nd week of december



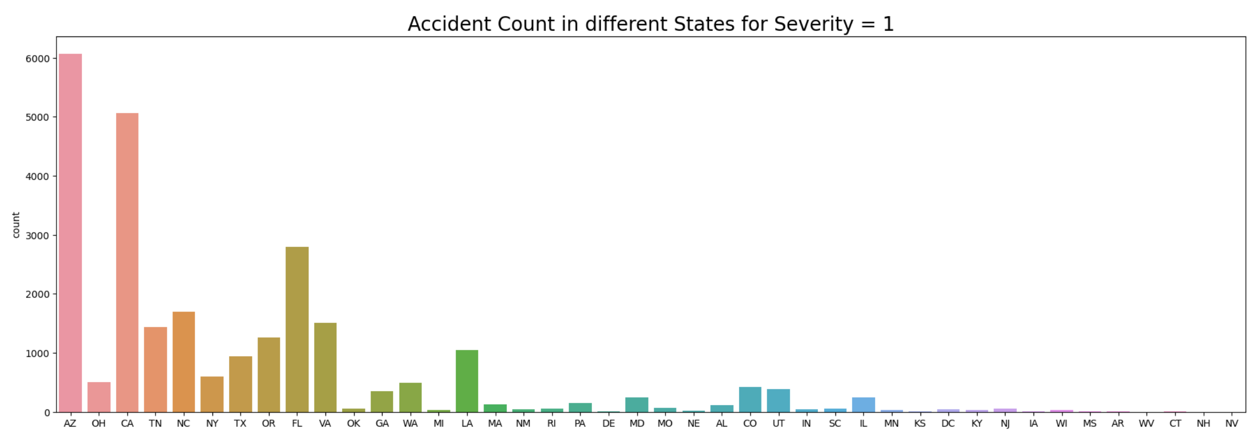
There was a very low number of accidents in the last week of November in 2019, but the next week suddenly had high accidents.

Due to COVID lockdown there were negligible amount of accidents in july august in 2020

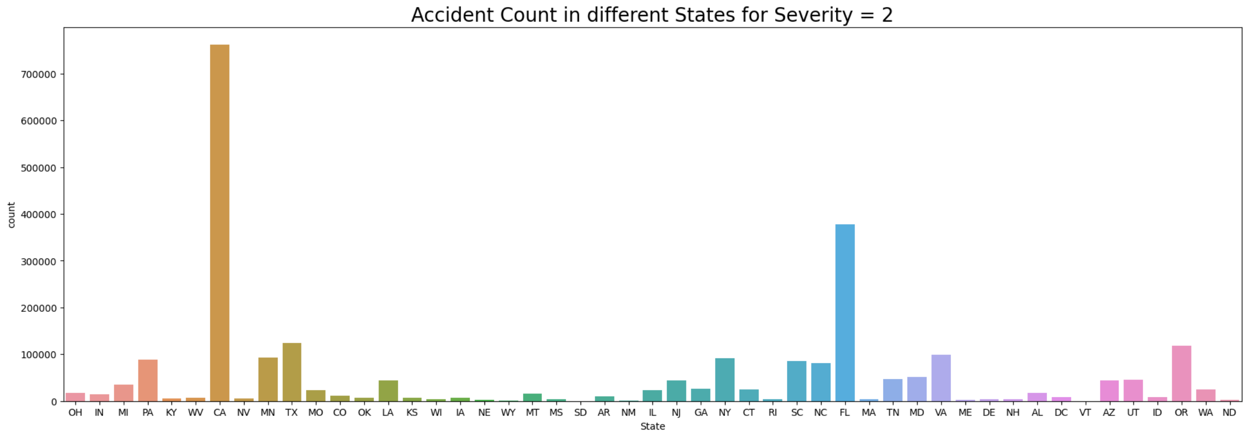


Again december's 1st week had the maximum number of accidents.

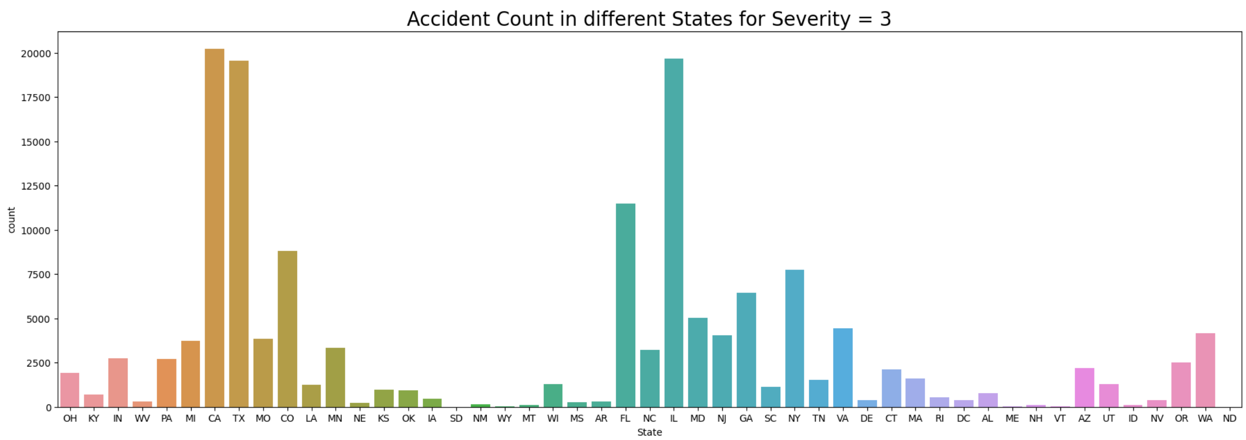
So, December's first week has the maximum number of accidents.



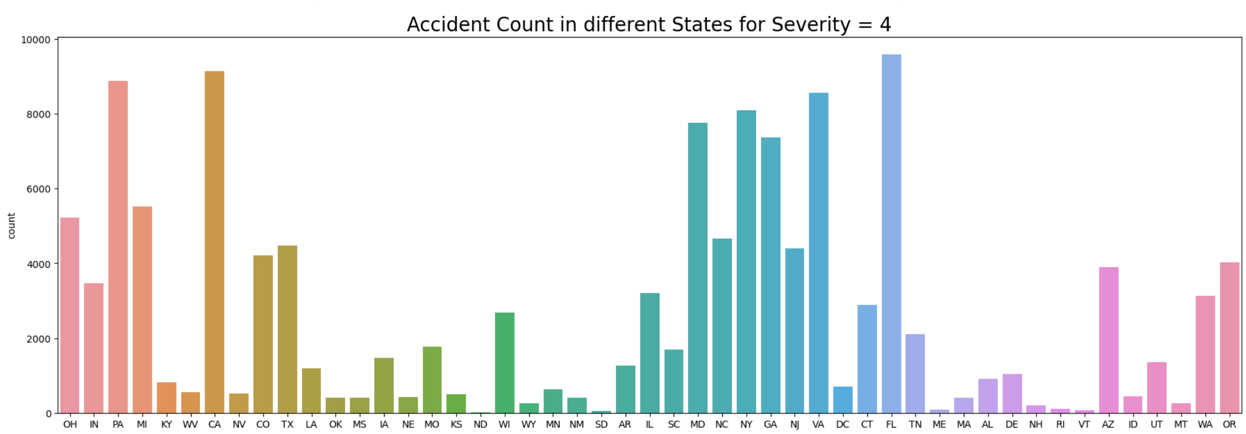
So, AZ state has the maximum number of accidents which are of very low severity



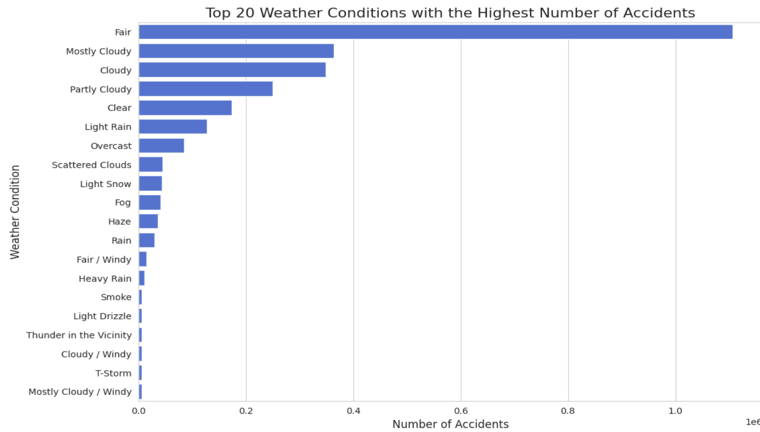
CA state has maximum number of moderate severity accident



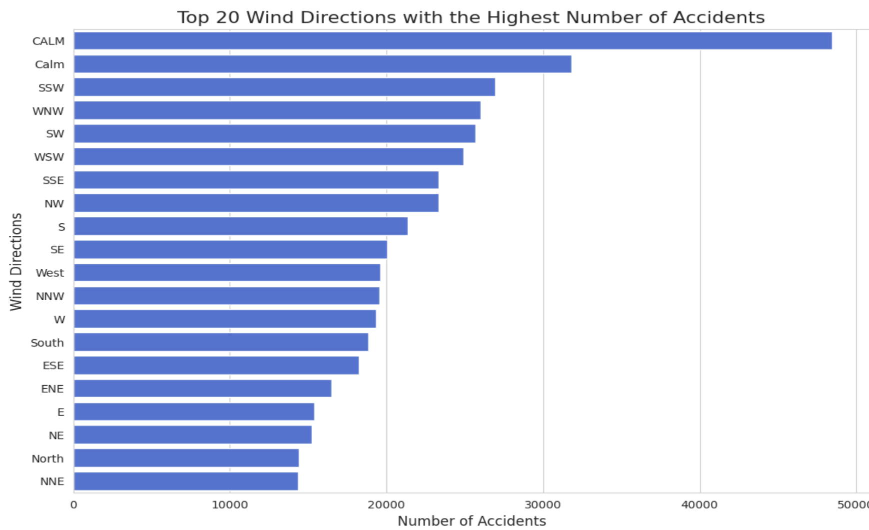
CA and TX again has maximum number of accidents along with IL



TL and CA thus is the hub for more severe accidents(severity 3 and 4)



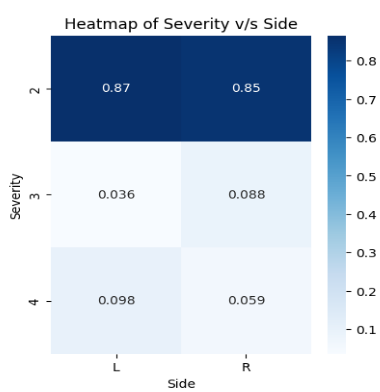
So, maximum accidents take place when weather condition is fair or mostly cloudy

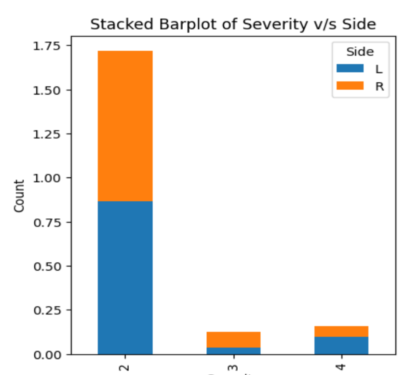


Calm weather gives us the maximum number of accidents.

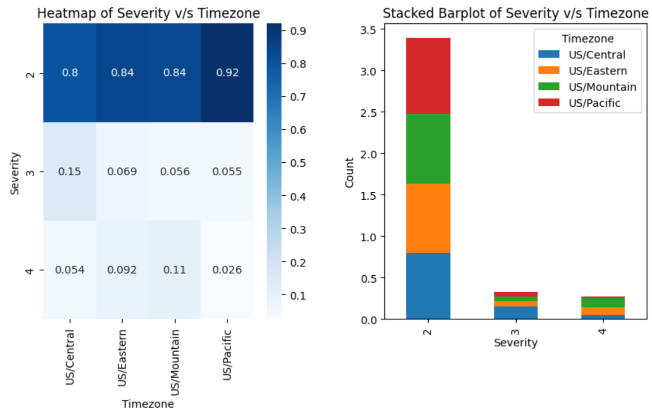
<u>Bivariate Analysis:</u>

1) Categorical Columns vs Categorical Columns:

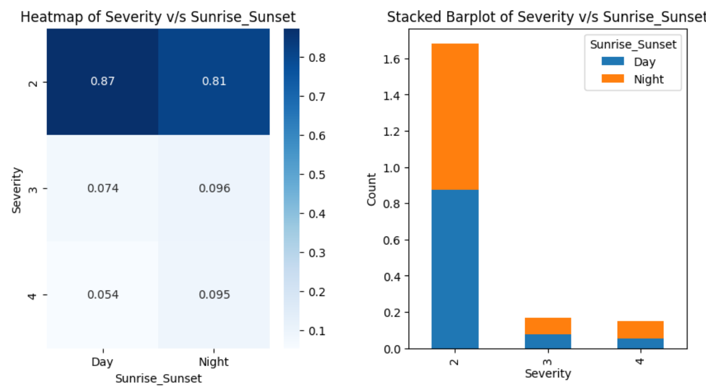


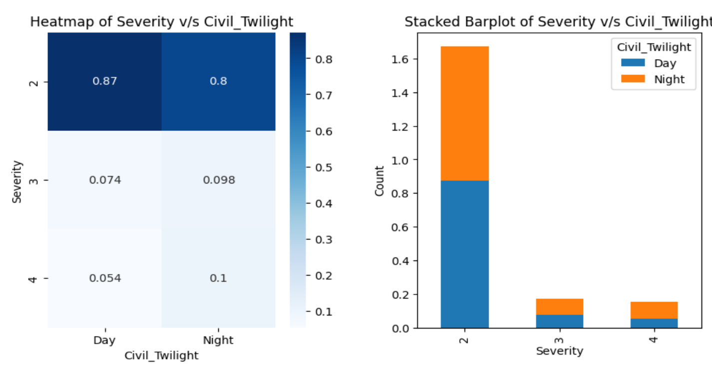


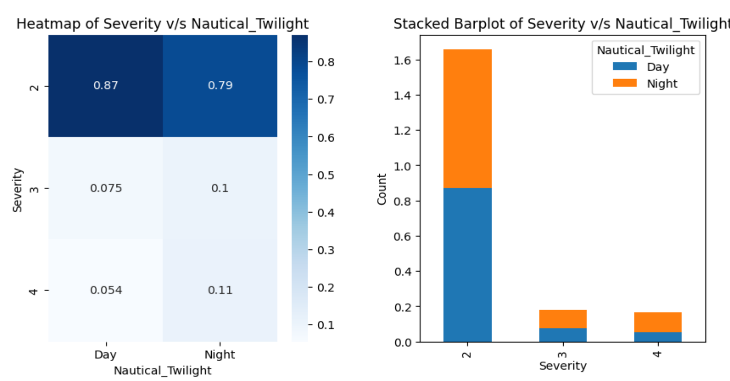
So, we have the most accidents as Severity 2 with L and R as both nearly 50%. Severity 3 had more accidents appearing on Right side of the road than on the left. Severity 4 had more accidents on Left side of the road.

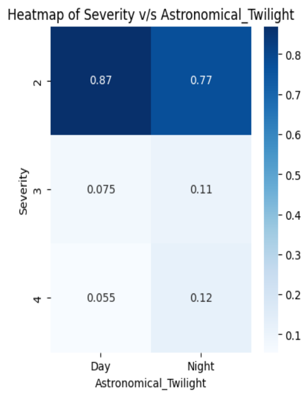


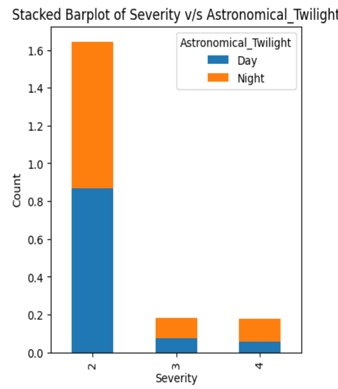
For severity as 3 and mostly were of timezone US/Central and severity 4 has US/Mountain

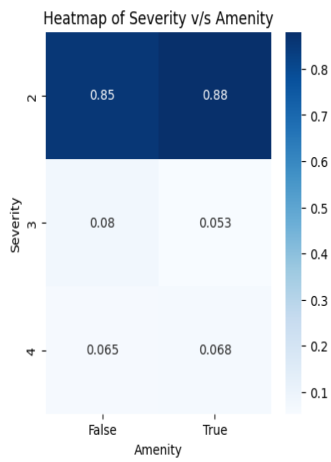


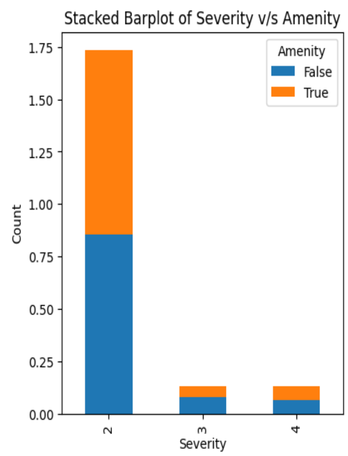


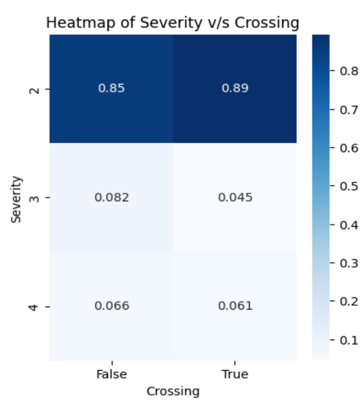


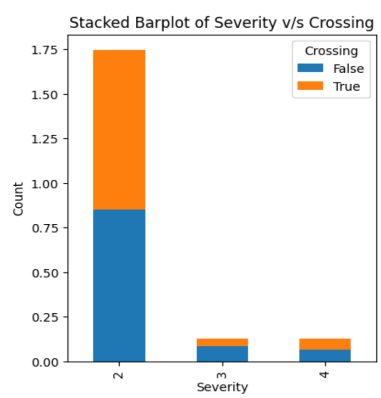

Heatmap of Severity v/s Astronomical_Twilight
0.87
0.77
0.075
0.11
0.055
0.12
Severity
Day
Night
Astronomical_Twilight

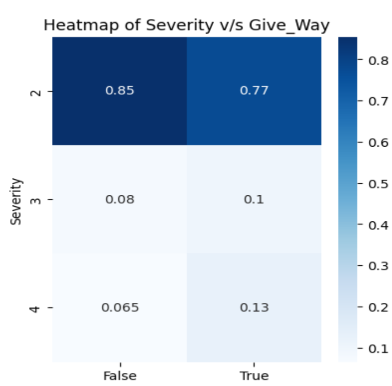

Stacked Barplot of Severity v/s Astronomical_Twilight
Astronomical_Twilight
Day
Night
Count
Severity

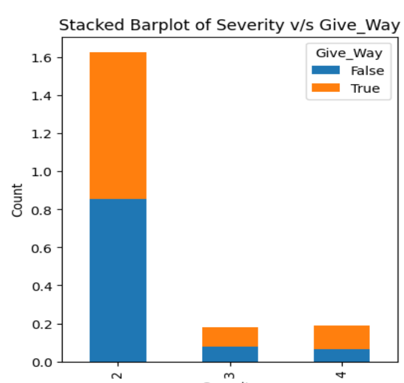

Heatmap of Severity v/s Amenity
0.85
0.88
0.08
0.053
0.065
0.068
Severity
False
True
Amenity

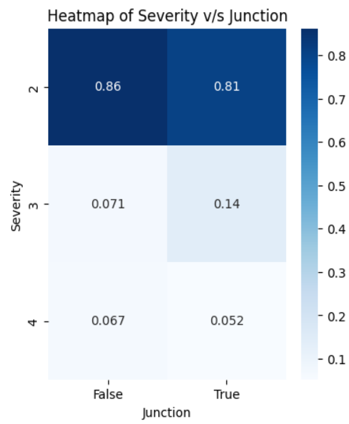

Stacked Barplot of Severity v/s Amenity
Amenity
False
True
Count
Severity

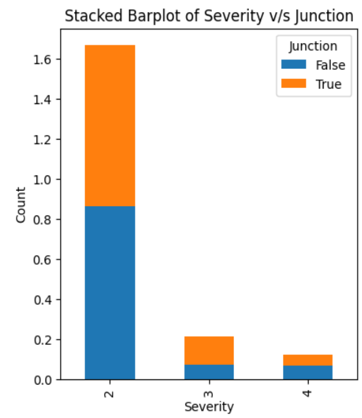

Heatmap of Severity v/s Crossing
0.85
0.89
0.082
0.045
0.066
0.061
Severity
False
True
Crossing

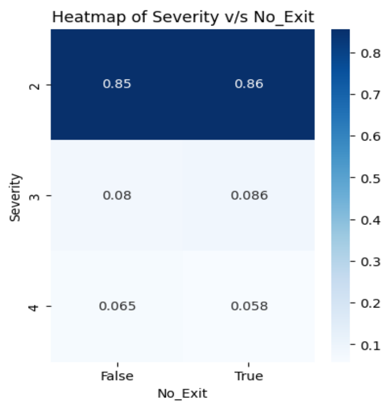

Stacked Barplot of Severity v/s Crossing
Crossing
False
True
Count
Severity

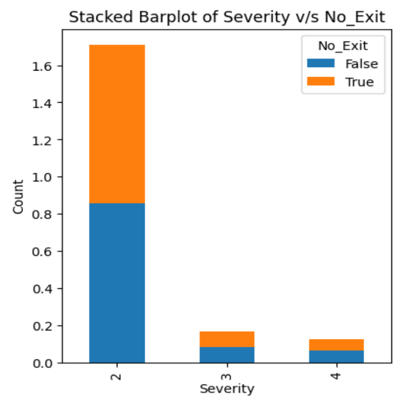

Heatmap of Severity v/s Give_Way
0.85
0.77
0.08
0.1
0.065
0.13
Severity
False
True
Give_Way

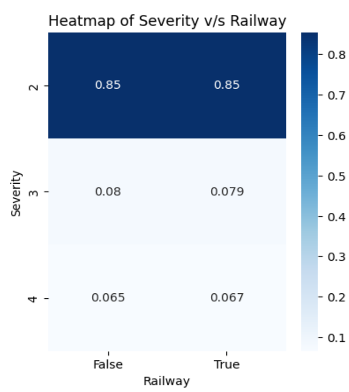

Stacked Barplot of Severity v/s Give_Way
Give_Way
False
True
Count
Severity

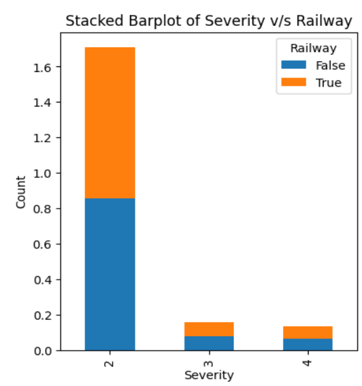

Heatmap of Severity v/s Junction
Severity
Junction
False
True
2
3
4
0.86
0.81
0.071
0.14
0.067
0.052

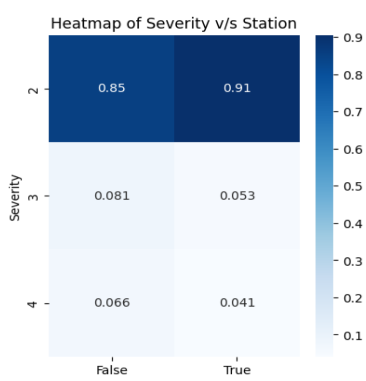

Stacked Barplot of Severity v/s Junction
Junction
False
True
Count
Severity

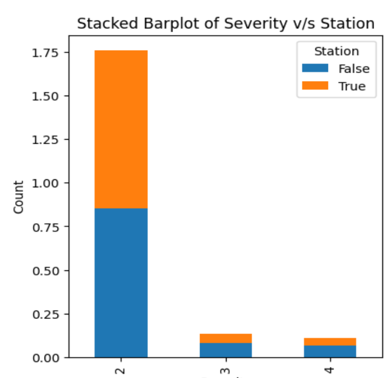

Heatmap of Severity v/s No_Exit
Severity
No_Exit
False
True
2
3
4
0.85
0.86
0.08
0.086
0.065
0.058

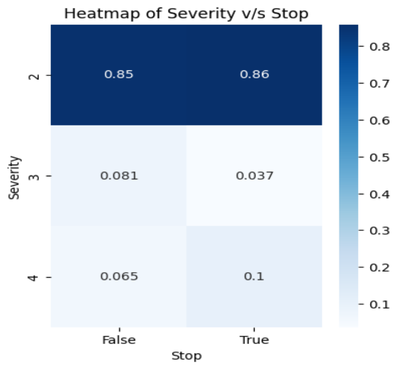

Stacked Barplot of Severity v/s No_Exit
No_Exit
False
True
Count
Severity

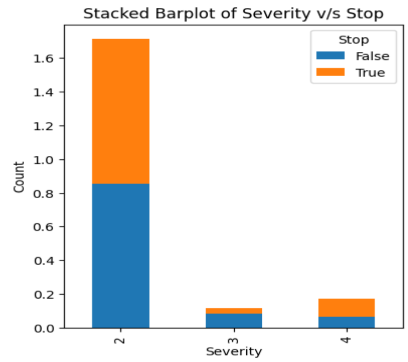

Heatmap of Severity v/s Railway
Severity
Railway
False
True
2
3
4
0.85
0.85
0.08
0.079
0.065
0.067

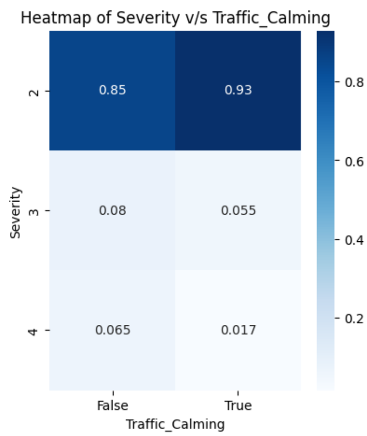

Stacked Barplot of Severity v/s Railway
Railway
False
True
Count
Severity

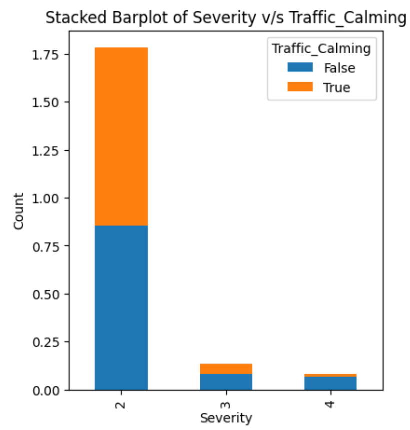

Heatmap of Severity v/s Station
Severity
Station
False
True
2
3
4
0.85
0.91
0.081
0.053
0.066
0.041

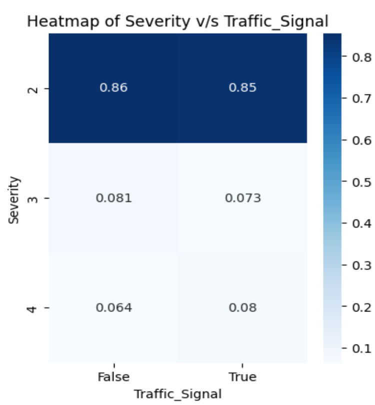

Stacked Barplot of Severity v/s Station
Station
False
True
Count
Severity

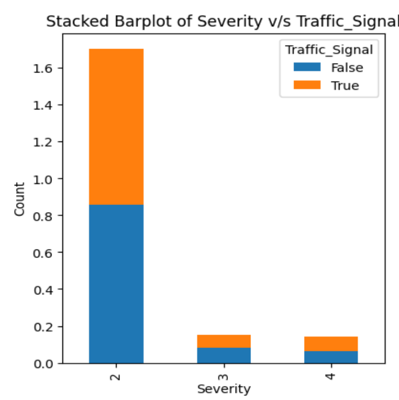

Heatmap of Severity v/s Stop
Severity
2
3
4
0.85
0.86
0.081
0.037
0.065
0.1
False
True
Stop
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8


Stacked Barplot of Severity v/s Stop
Stop
False
True
Count
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
2
3
4
Severity

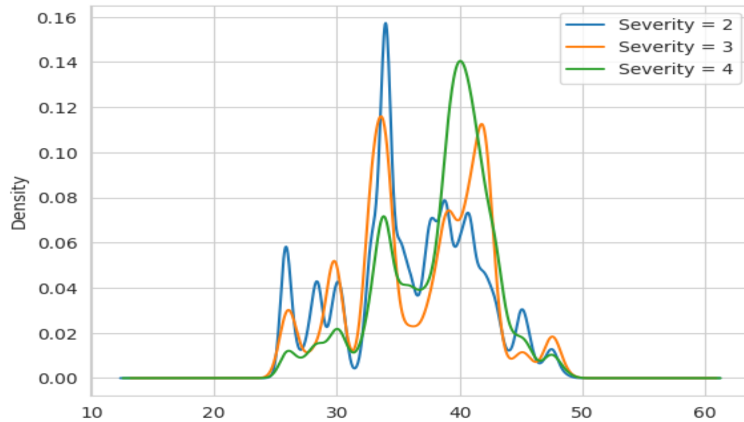

Heatmap of Severity v/s Traffic_Calming
Severity
2
3
4
0.85
0.93
0.08
0.055
0.065
0.017
False
True
Traffic_Calming
0.2
0.4
0.6
0.8


Stacked Barplot of Severity v/s Traffic_Calming
Traffic_Calming
False
True
Count
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2
3
4
Severity

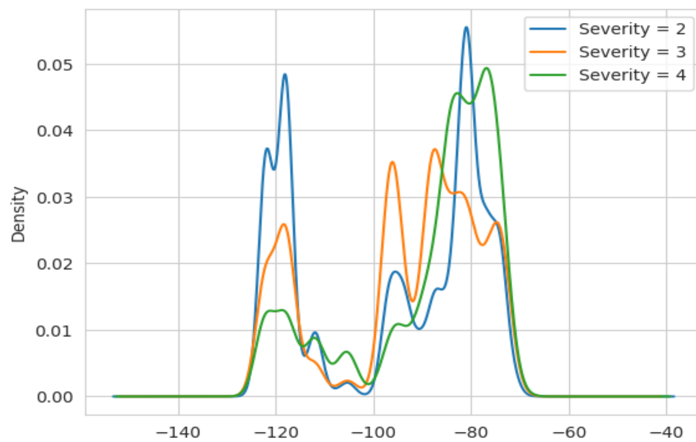

Heatmap of Severity v/s Traffic_Signal
Severity
2
3
4
0.86
0.85
0.081
0.073
0.064
0.08
False
True
Traffic_Signal
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8


Stacked Barplot of Severity v/s Traffic_Signal
Traffic_Signal
False
True
Count
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
2
3
4
Severity

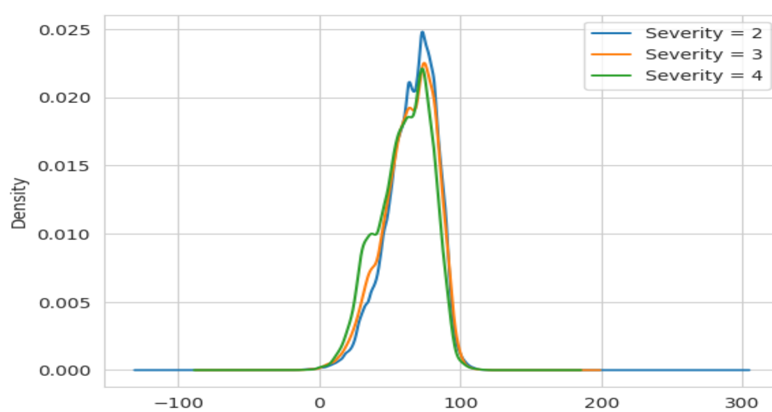
2) Numerical Columns vs Categorical Columns:



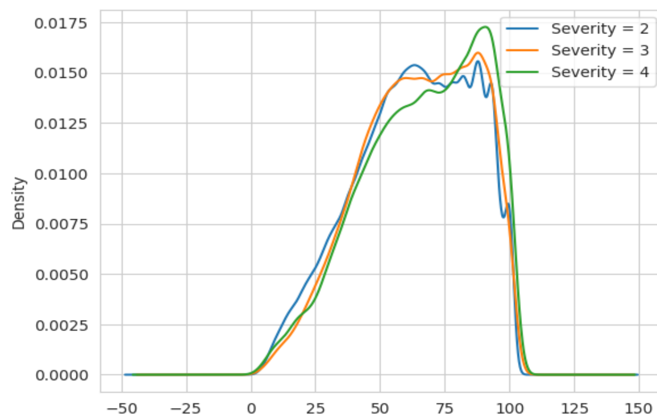
So, severity 4 accidents occur mostly near 40 End_Lat and severity 3 mostly near 35 and Severity 2 near 32



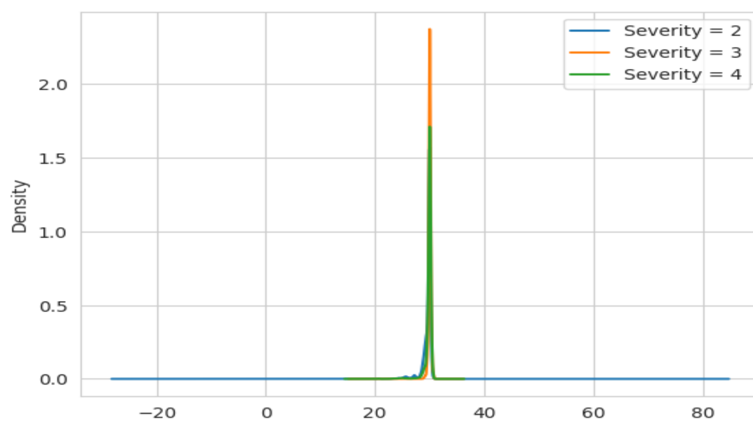
So, severity 4 accidents occur mostly near -78 End_Lng and severity 3 mostly near -100 and Severity 2 near -80

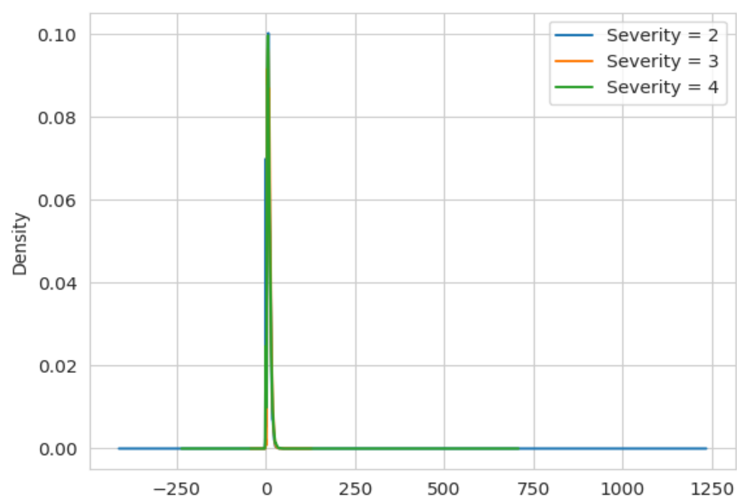


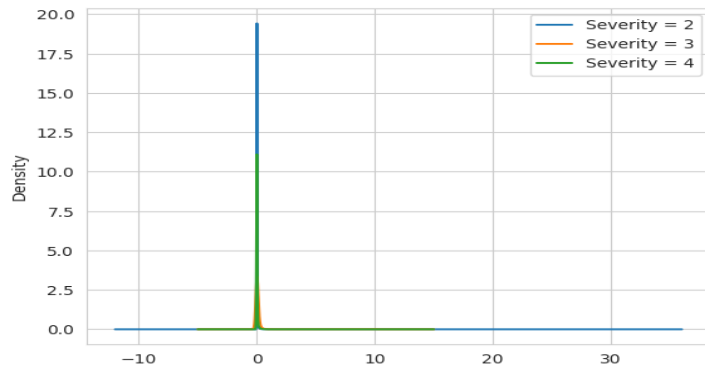
Temperature has almost similar distribution for all severities ie. severity doesn't depend that much on temperature

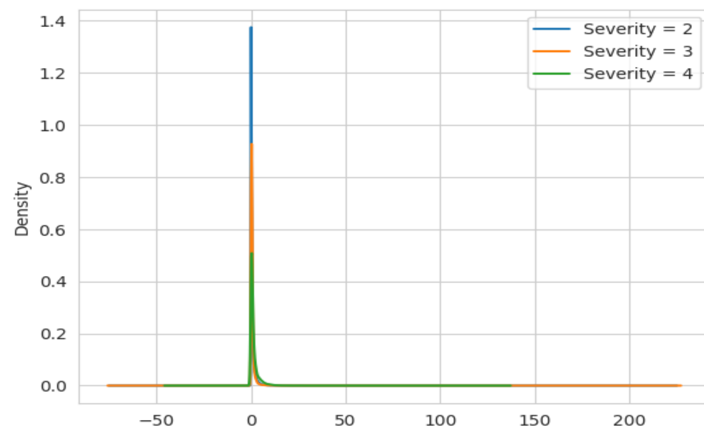


Similar to tempearture





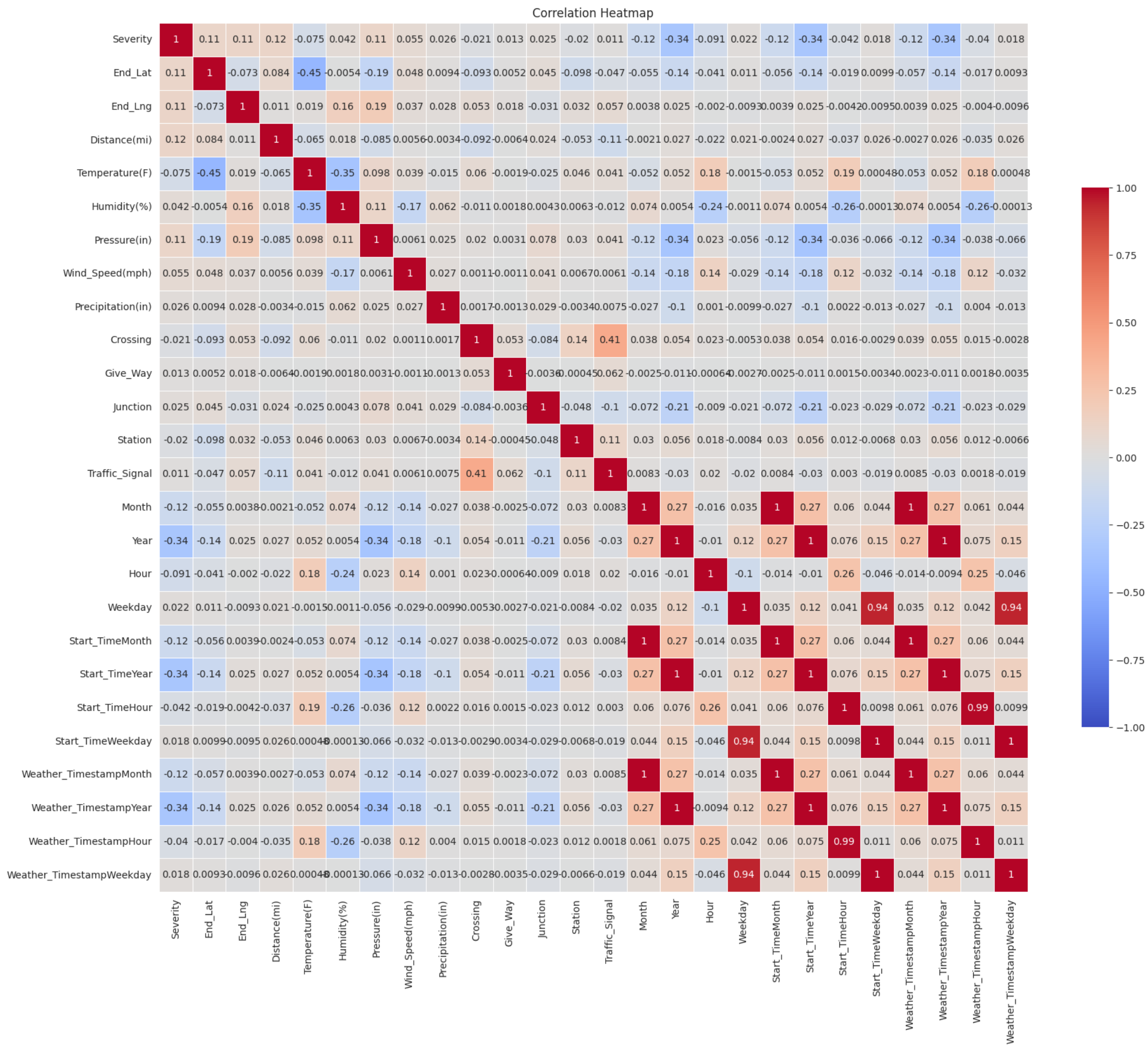




Thus, Pressure,Wind Speed,Precipitation,Distance had similar distribution for all the severities

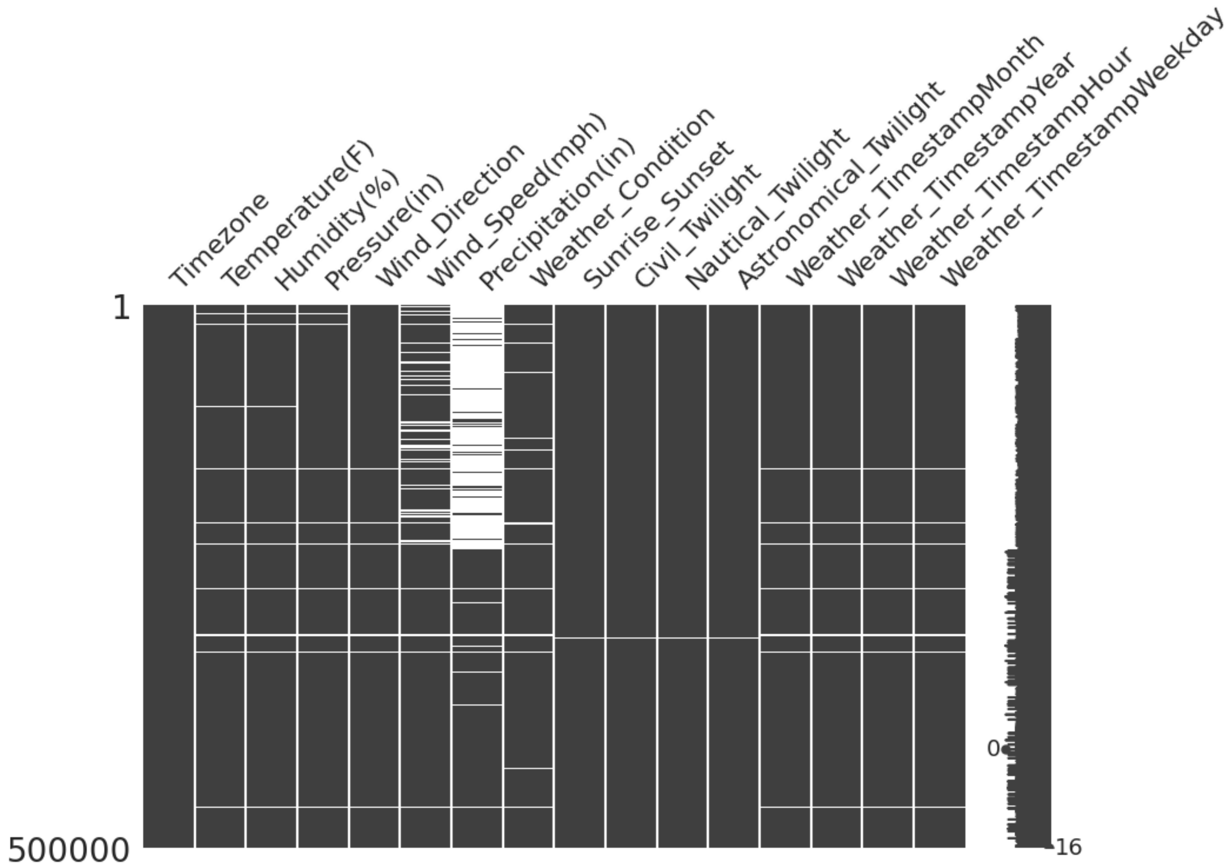
Now, we delete the object type columns having many distinct values as that would simply increase the complexity of our model and increase our training time.

Plotting the correlation heatmap:



So, we delete those input columns which are highly correlated to each other and those which are very less correlated to the output column

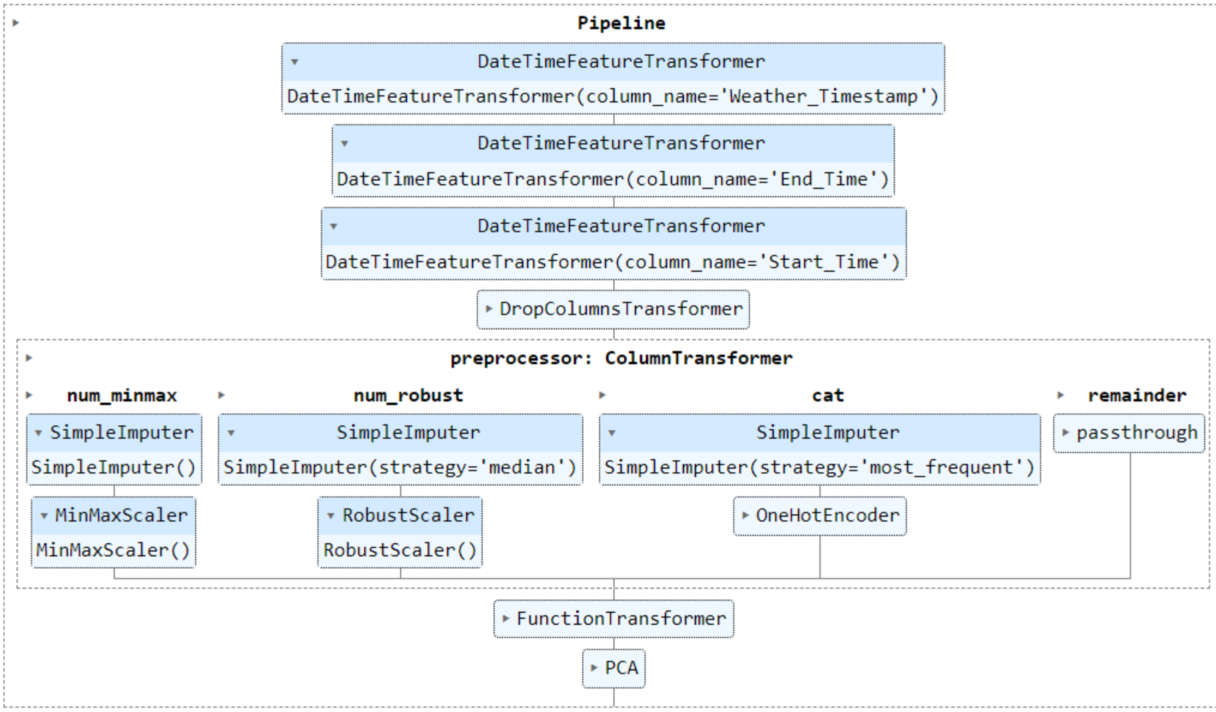
For NaN values:



So, we observe that NaN values are distributed at random for most of the columns. For them, we apply Simple Imputer with strategy as mean and for others, we apply strategy as median. For Precipitation, we have the majority of data having value 0 so we fill the nan values with 0.

Now, we encode the categorical columns using One Hot Encoder as the columns are nominal in nature instead of ordinal. Then, we apply Robust Scaler for columns having outliers else we apply Min Max Scaler for other columns as all others have some fixed range of values beyond which the values can’t exceed. For columns having right skew, we apply log transformation and for those having left skew, we apply square transformation.

Now, we summarize everything down into sklearn pipeline which appears as shown:



ie . We apply PCA at the end as it gives better results over other feature extraction techniques.

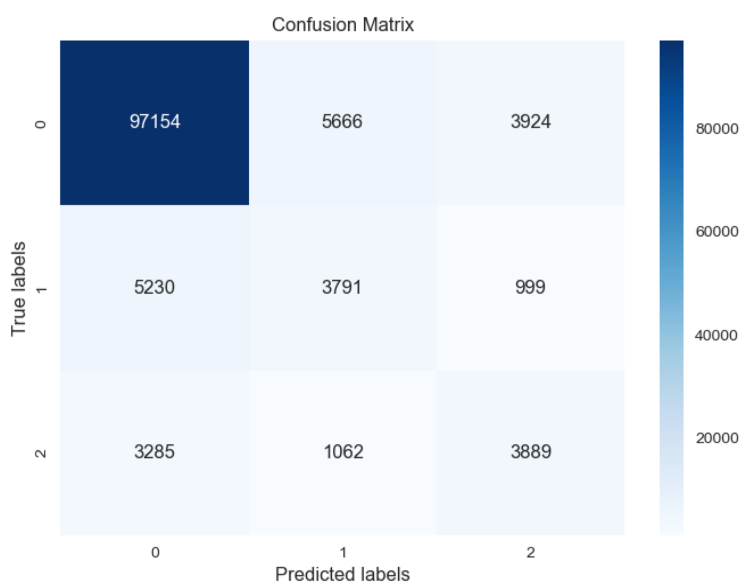
Now, we experiment with different machine learning algorithms at the beginning followed by deep learning algorithms

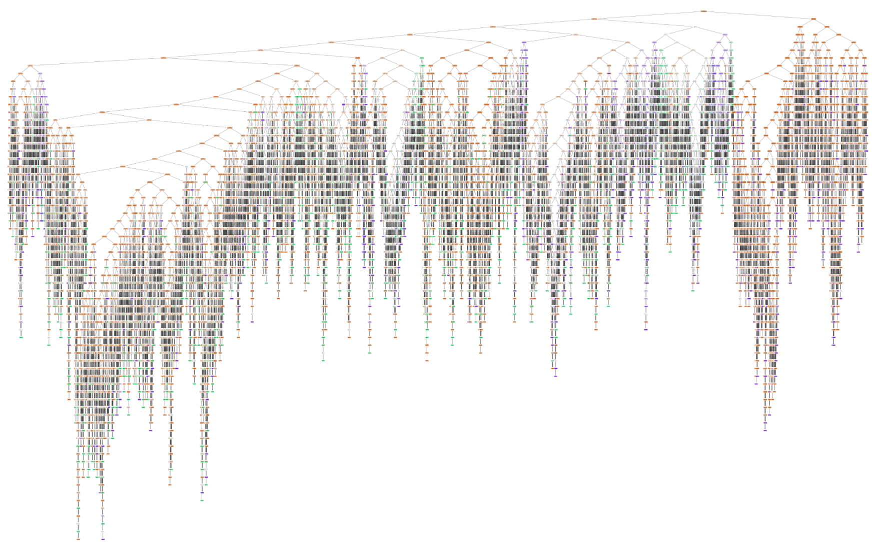
<u>Decision Tree</u>:

Training accuracy: 0.9998266666666666

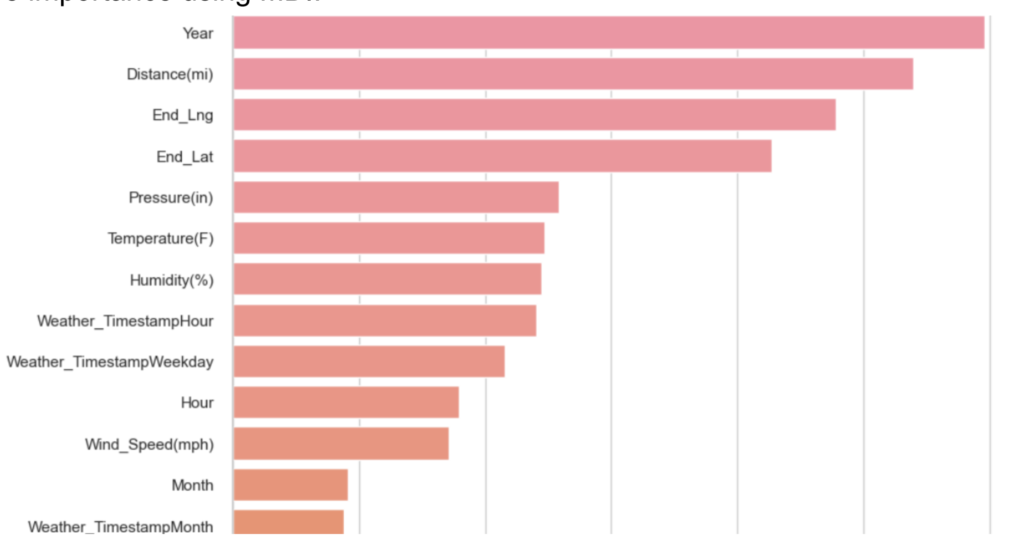
Testing accuracy: 0.838672

Accuracy: 0.838672





Feature importance using MDI:



So, year from End Time is the most important feature for predicting Severity followed by Distance and location factors.

Now, we apply grid search cv to tune hyper-parameters of the decision tree and get the best testing accuracy as 85.67.

Classification Report:

	precision	recall	f1-score	support
2	0.86	1.00	0.92	106744
3	0.72	0.09	0.17	10020
4	0.33	0.00	0.00	8236
accuracy			0.86	125000
macro avg	0.64	0.36	0.36	125000
weighted avg	0.81	0.86	0.80	125000

Naive Bayes(Gaussian):

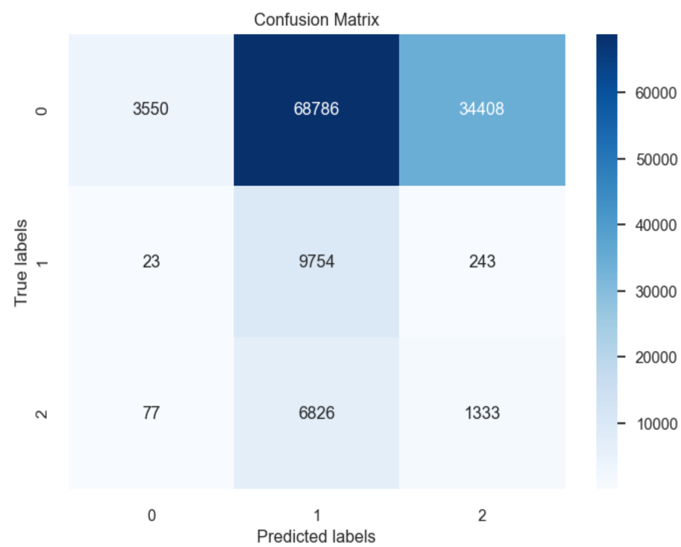
Training accuracy: 0.11736666666666666

Testing accuracy: 0.117096

Balanced accuracy: 0.3895202150695556

ROC score (one-vs-one): 0.6920219212102686

ROC score (one-vs-rest): 0.6576626492406757



Classification Report:

	precision	recall	f1-score	support
2	0.97	0.03	0.06	106744
3	0.11	0.97	0.20	10020
4	0.04	0.16	0.06	8236
accuracy			0.12	125000
macro avg	0.37	0.39	0.11	125000
weighted avg	0.84	0.12	0.08	125000

Now, we tune the hyper-parameters:

Best parameter: {'var_smoothing': 10}

Best score: 0.8549466666666667

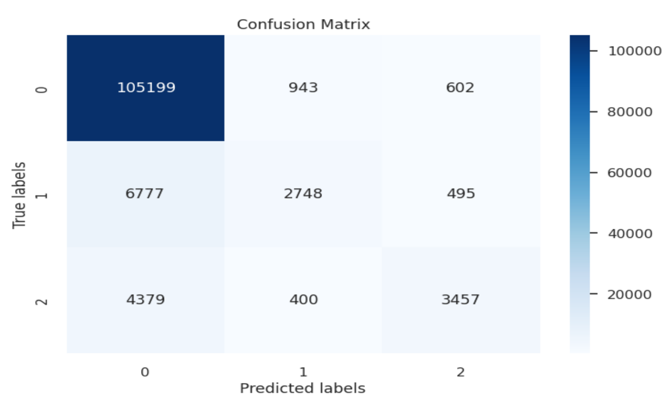
Classification Report:

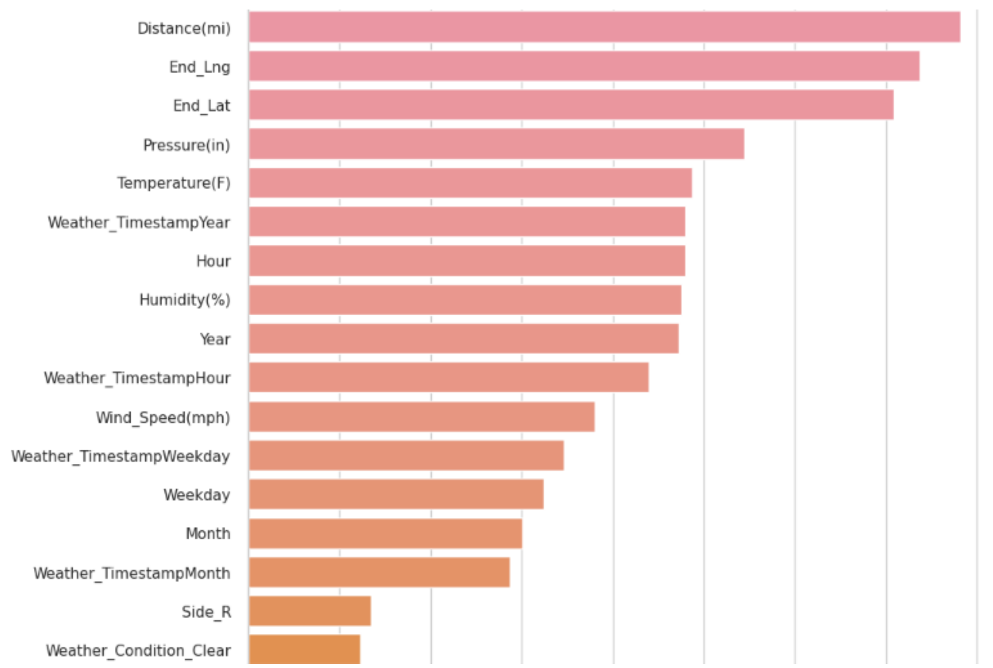
	precision	recall	f1-score	support
2	0.85	1.00	0.92	106744
3	0.17	0.00	0.00	10020
4	0.50	0.00	0.01	8236
accuracy			0.85	125000
macro avg	0.51	0.33	0.31	125000
weighted avg	0.78	0.85	0.79	125000

<u>Random Forest:</u>

Training accuracy: 0.9997966666666667
Testing accuracy: 0.891232
Balanced accuracy: 0.5598400696873385
ROC score (one-vs-one): 0.8779368690143694
ROC score (one-vs-rest): 0.9133019806632238
Classification Report:

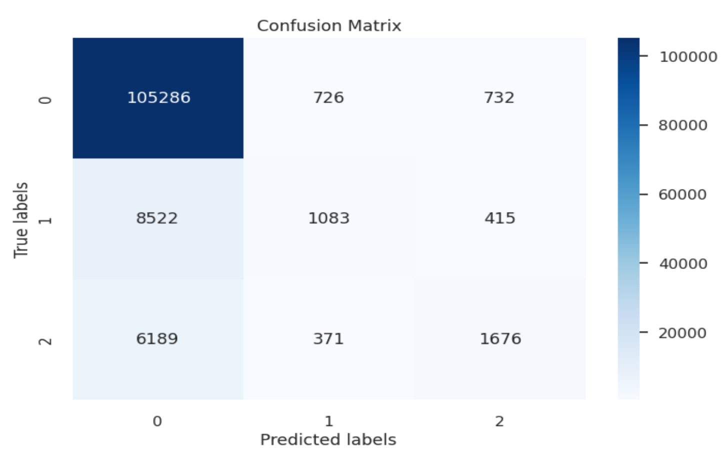
	precision	recall	f1-score	support
2	0.90	0.99	0.94	106744
3	0.67	0.27	0.39	10020
4	0.76	0.42	0.54	8236
accuracy			0.89	125000
macro avg	0.78	0.56	0.62	125000
weighted avg	0.88	0.89	0.87	125000





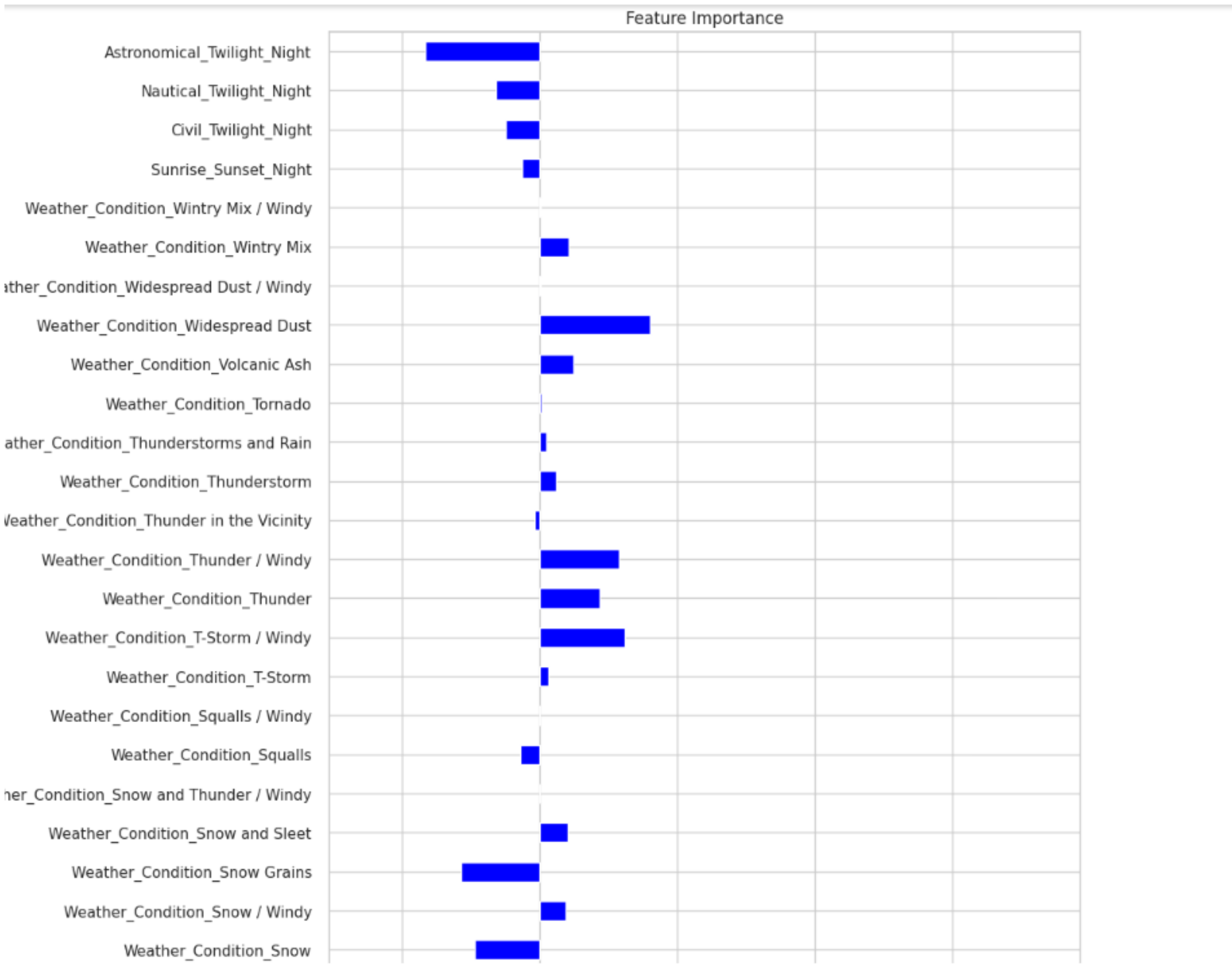
So, Distance and location is the most important feature according to random forest feature importance.

<u>Logistic regression:</u>
Training accuracy: 0.8656433333333333
Testing accuracy: 0.86436
Balanced accuracy: 0.43264060937579324
ROC score (one-vs-one): 0.8220674190595432
ROC score (one-vs-rest): 0.8689506662517182



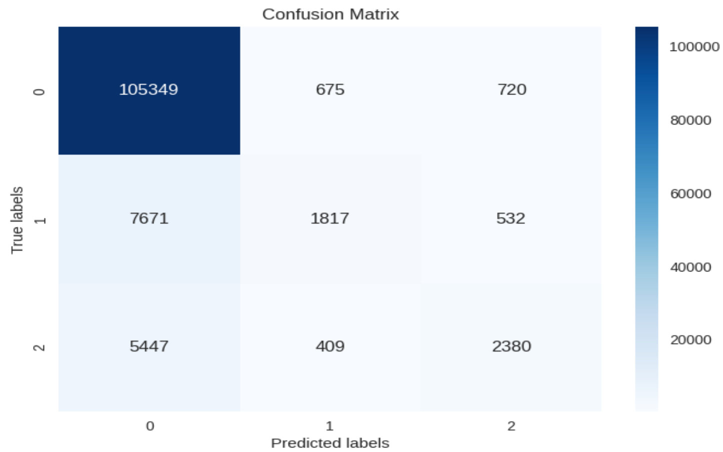
Classification Report:

	precision	recall	f1-score	support
2	0.88	0.99	0.93	106744
3	0.50	0.11	0.18	10020
4	0.59	0.20	0.30	8236
accuracy			0.86	125000
macro avg	0.66	0.43	0.47	125000
weighted avg	0.83	0.86	0.83	125000



<u>Gradient Boosting:</u>

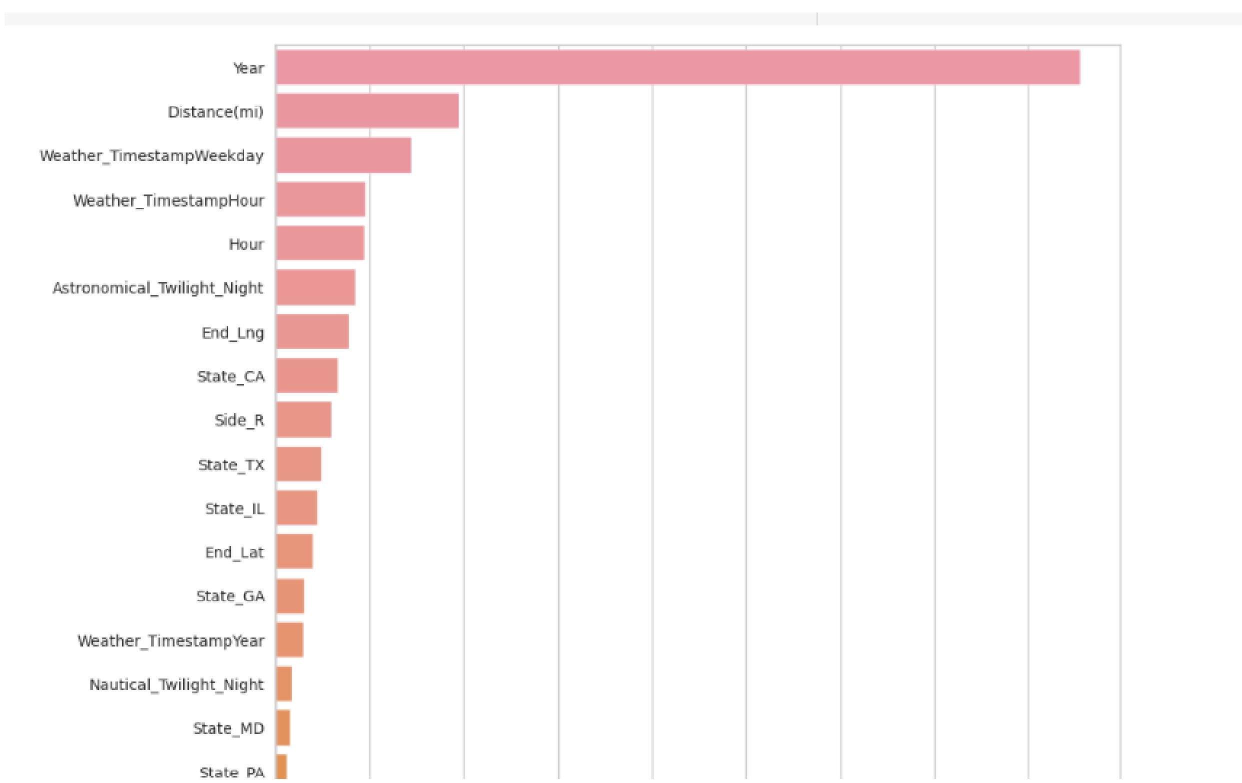
Training accuracy: 0.87842
Testing accuracy: 0.876368
Balanced accuracy: 0.48574796860507635
One-vs-one macro accuracy: 0.8435378872087244
One-vs-rest macro accuracy: 0.8941212664426826



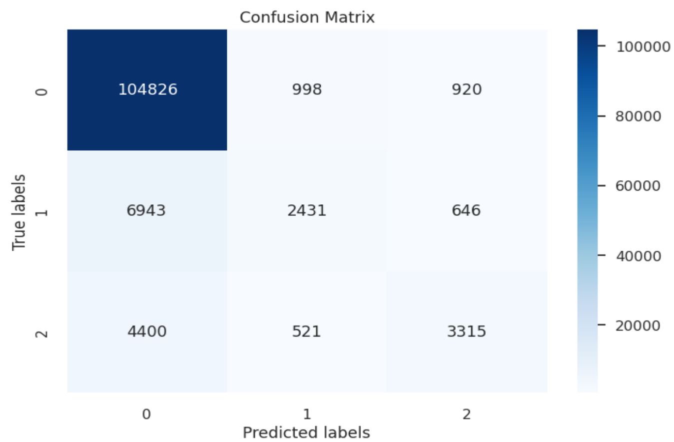
Classification Report:

	precision	recall	f1-score	support
2	0.89	0.99	0.94	106744
3	0.63	0.18	0.28	10020
4	0.66	0.29	0.40	8236
accuracy			0.88	125000
macro avg	0.72	0.49	0.54	125000
weighted avg	0.85	0.88	0.85	125000

Feature Importance:

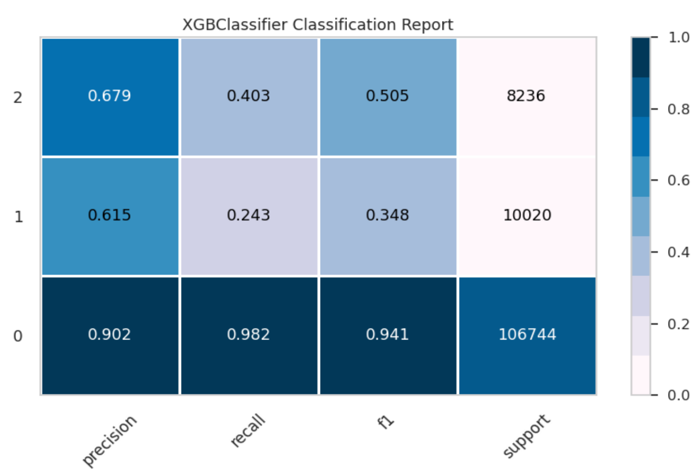


XGBoost:
Training accuracy: 0.8956033333333333
Testing accuracy: 0.884576
Accuracy: 0.884576
Balanced accuracy: 0.5423825872008087
ROC one-vs-one macro accuracy: 0.8736074700258722
ROC one-vs-rest macro accuracy: 0.9127412686566672

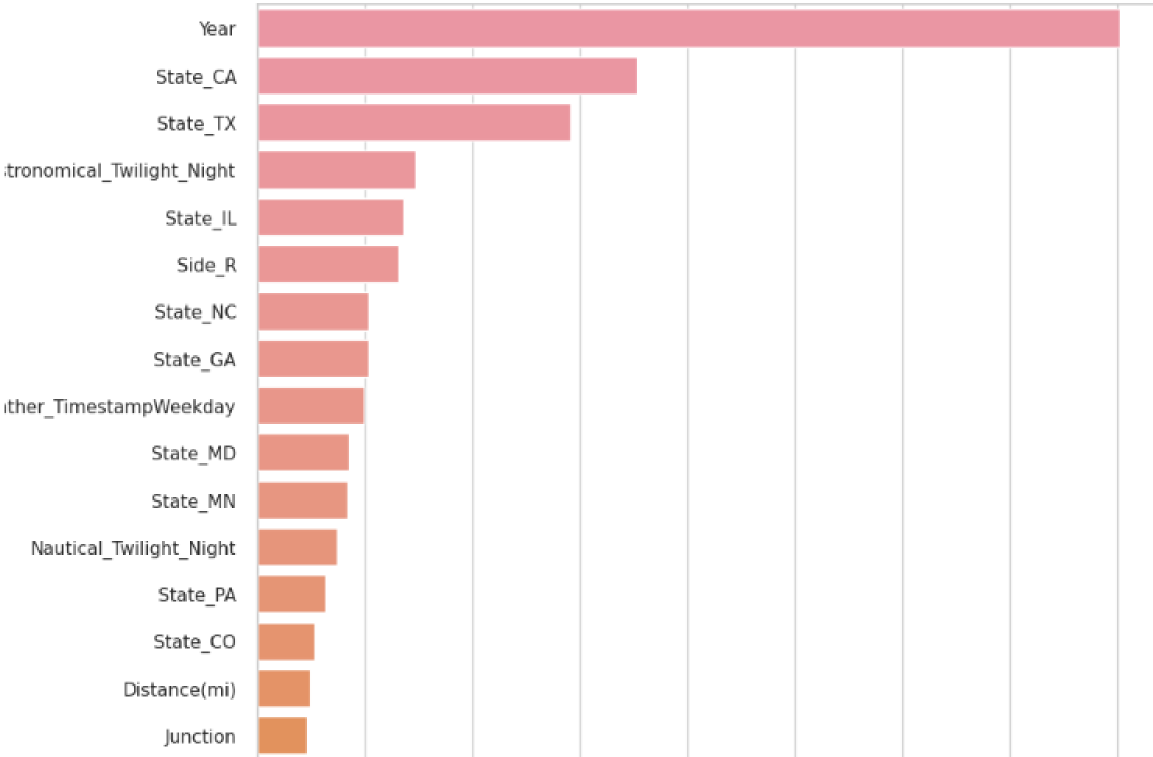


Classification Report:

	precision	recall	f1-score	support
0	0.90	0.98	0.94	106744
1	0.62	0.24	0.35	10020
2	0.68	0.40	0.51	8236
accuracy			0.88	125000
macro avg	0.73	0.54	0.60	125000
weighted avg	0.86	0.88	0.86	125000



Feature Importance:



KNN:

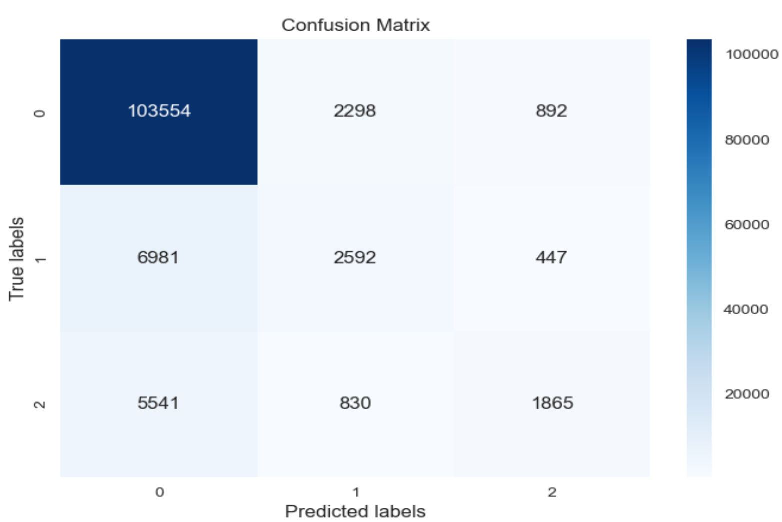
Training accuracy: 0.89669

Testing accuracy: 0.864088

Balanced accuracy: 0.48508097573572573

One-vs-one macro accuracy: 0.776852367316808

One-vs-rest macro accuracy: 0.8068038431167629

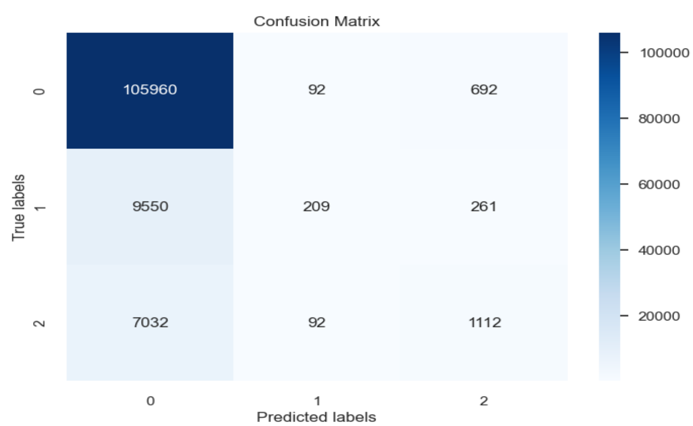


Classification Report:

	precision	recall	f1-score	support
2	0.89	0.97	0.93	106744
3	0.45	0.26	0.33	10020
4	0.58	0.23	0.33	8236
accuracy			0.86	125000
macro avg	0.64	0.49	0.53	125000
weighted avg	0.84	0.86	0.84	125000

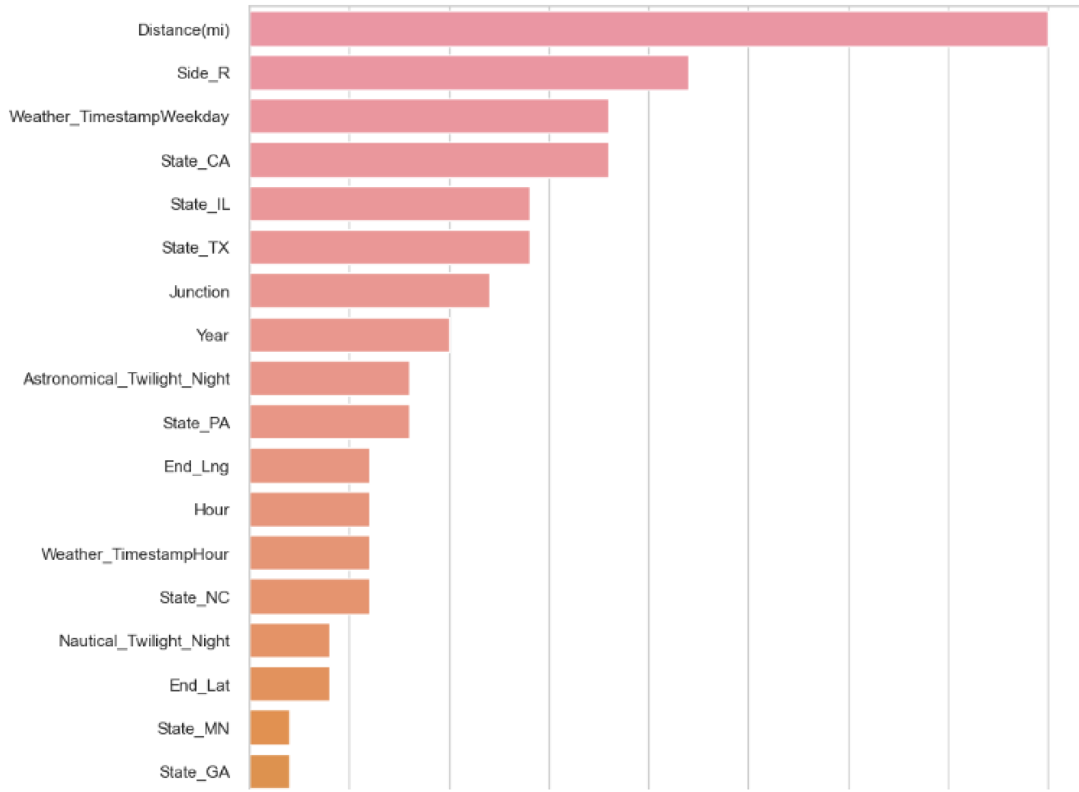
<u>Adaboost:</u>

Training accuracy: 0.85971
Testing accuracy: 0.858248
Balanced accuracy: 0.3895202150695556
ROC score (one-vs-one): 0.7635332899408267
ROC score (one-vs-rest): 0.7155236619476799

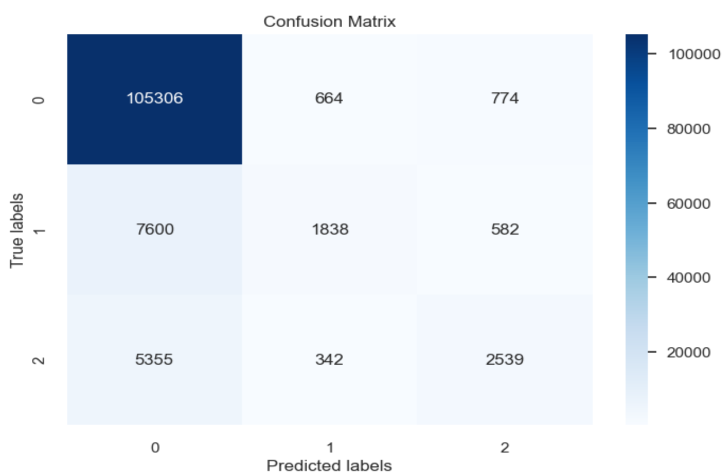


Classification Report:

	precision	recall	f1-score	support
2	0.86	0.99	0.92	106744
3	0.53	0.02	0.04	10020
4	0.54	0.14	0.22	8236
accuracy			0.86	125000
macro avg	0.64	0.38	0.39	125000
weighted avg	0.82	0.86	0.81	125000

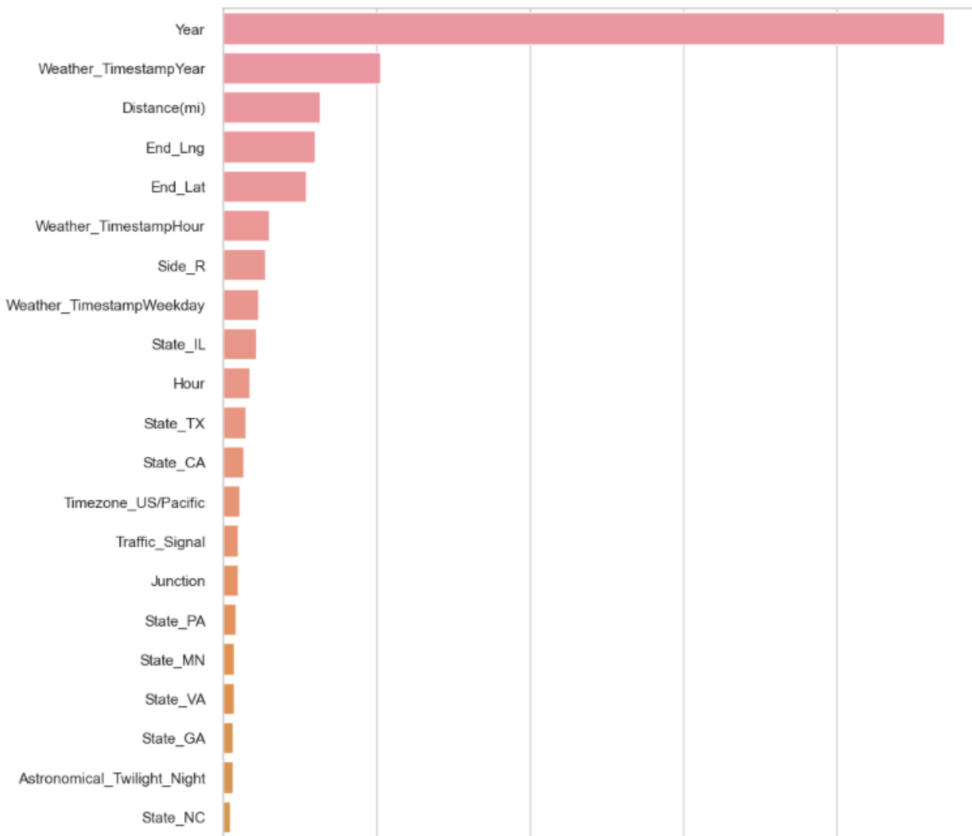


Catboost:
Training accuracy: 0.8792766666666667
Testing accuracy: 0.877464
Balanced accuracy: 0.4927474564511228
ROC score (one-vs-one): 0.8454912416995999
ROC score (one-vs-rest): 0.8962885045445804



Classification Report:

	precision	recall	f1-score	support
2	0.89	0.99	0.94	106744
3	0.65	0.18	0.29	10020
4	0.65	0.31	0.42	8236
accuracy			0.88	125000
macro avg	0.73	0.49	0.55	125000
weighted avg	0.86	0.88	0.85	125000



Now, we use Artificial Neural Network to train our model using keras tuner as hyper parameter tuner. We vary the following hyper-parameters:

1) Number of layers in range 2 to 10.
2) Number of nodes per layer in range 32 to 512 with step size as 32.
3) Activation Function as ReLU, tanh and sigmoid.
4) We also add L1 and L2 regularization.
5) We add dropout layers with fractions varying from 0.1 to 0.9.
6) Now finally we check if batch normalization at any layer gives better results.
7) We use softmax as output layer activation function and loss as categorical cross entropy.
8) Now, we choose optimizer amongst adam,rmsprop and stochastic gradient descent.

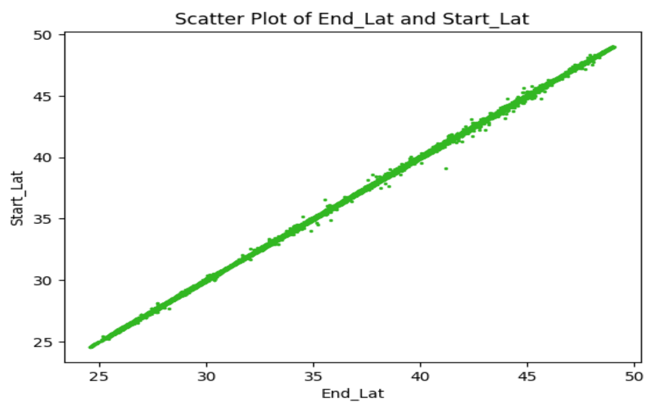
Using all these different hyper-parameters, we find out the best model.

This model gives validation accuracy as 87.35%, training accuracy as 87.34% and testing as 87.24%. This model doesn't overfit and thus gives a very general result, almost all the machine learning models did overfit. So, we use this model as our final model due to its good generality and testing accuracy so that it can work on different new data points.

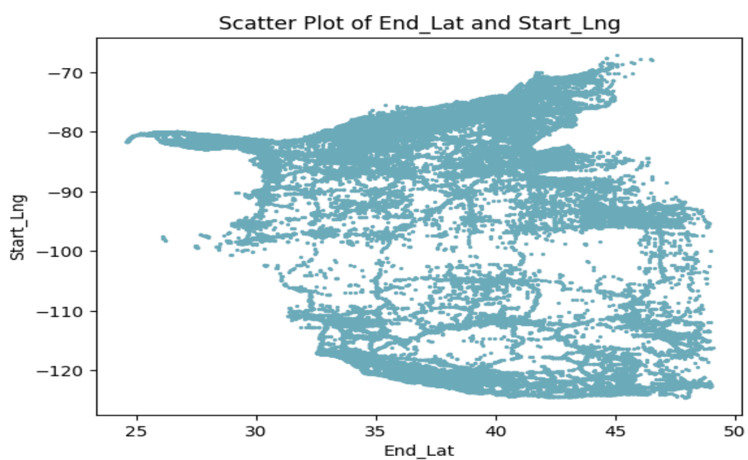
For Location Prediction

We consider 2 output continuous variable columns as End_Lat and End_Lng and thus perform

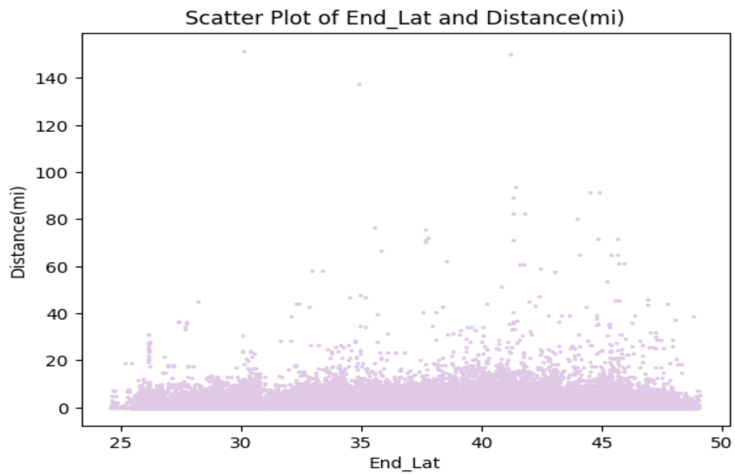
1) Numerical Columns vs Categorical Columns:



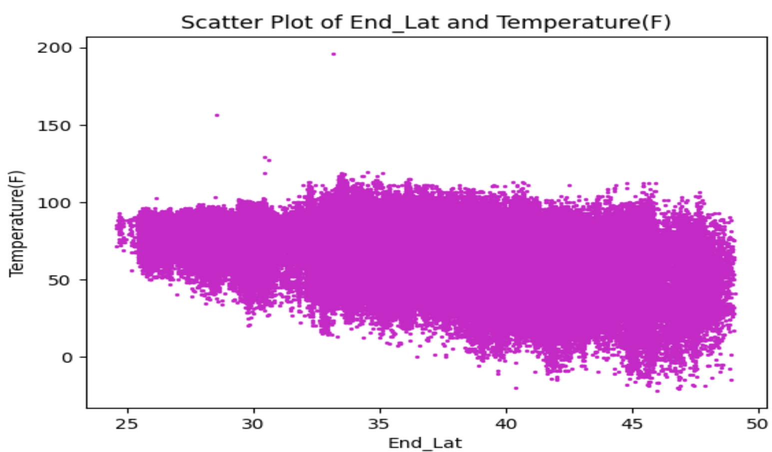
Thus, they are highly correlated to each other



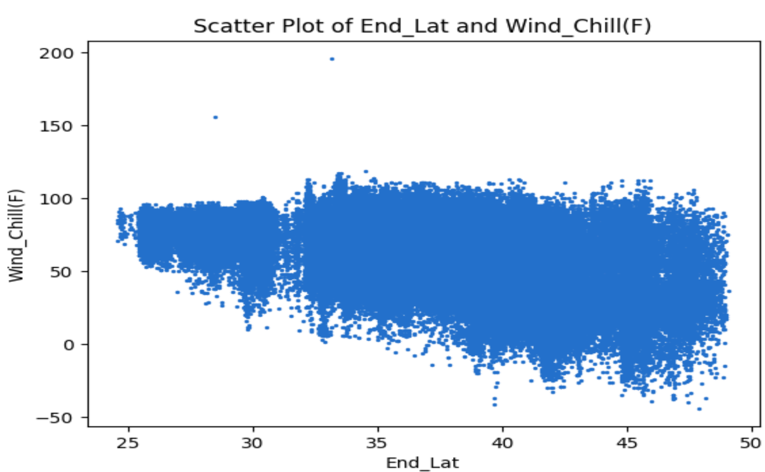
This is the flipped map of US where the accidents occurred



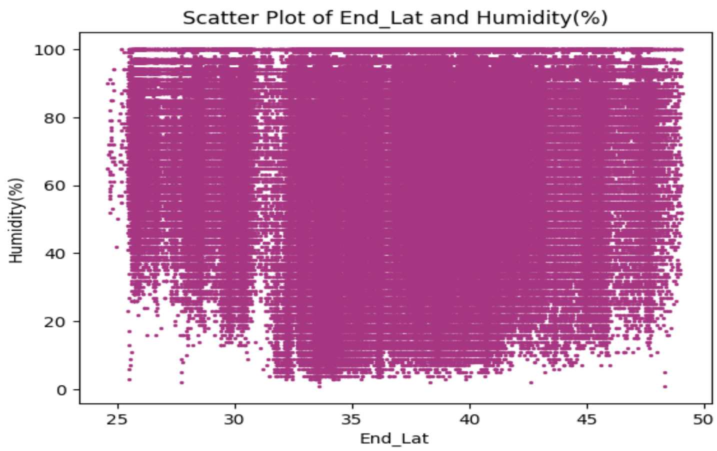
So, clearly the distance is very low for most of the Latitude or region but there are some outliers as clearly visible having much more value of distance.



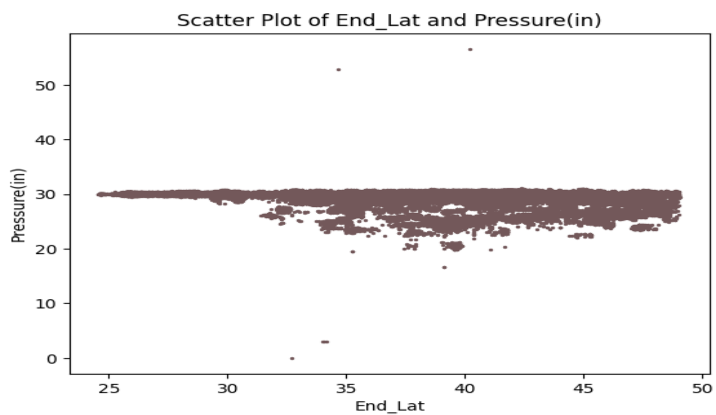
So, the temperature is spread with a range about 50 to 100 mostly, and more End_Lat have more variety of temperature



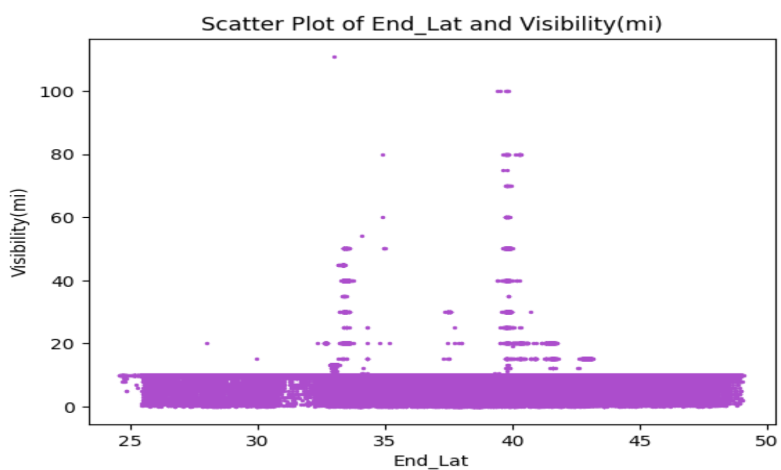
Very similar to temperature



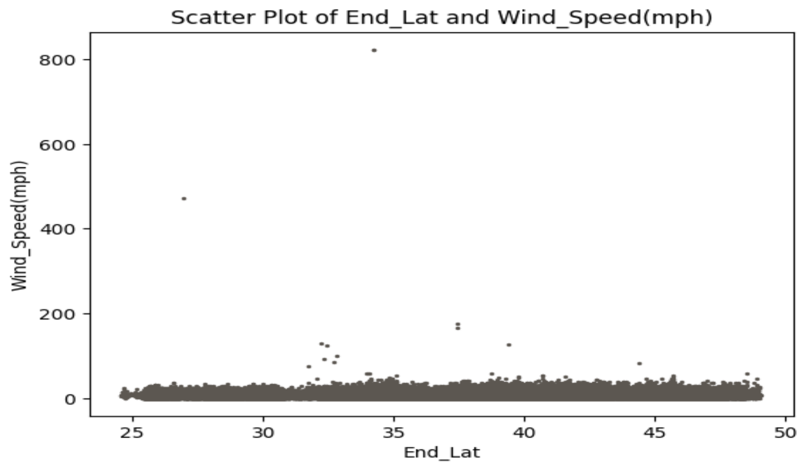
Humidity is spread throughout the region



Pressure is limited to a range near 30 in

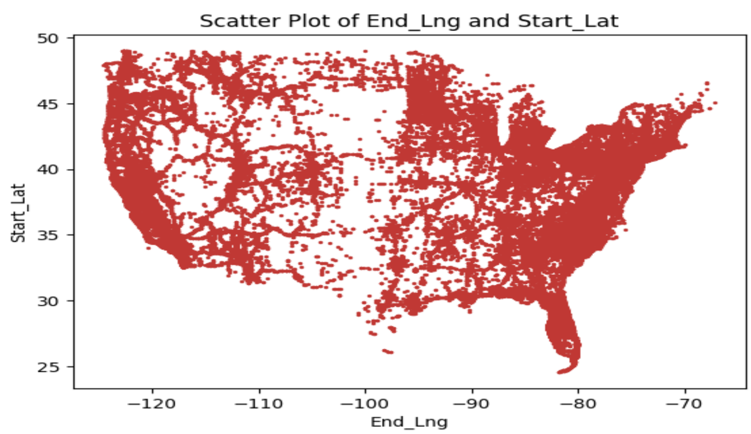


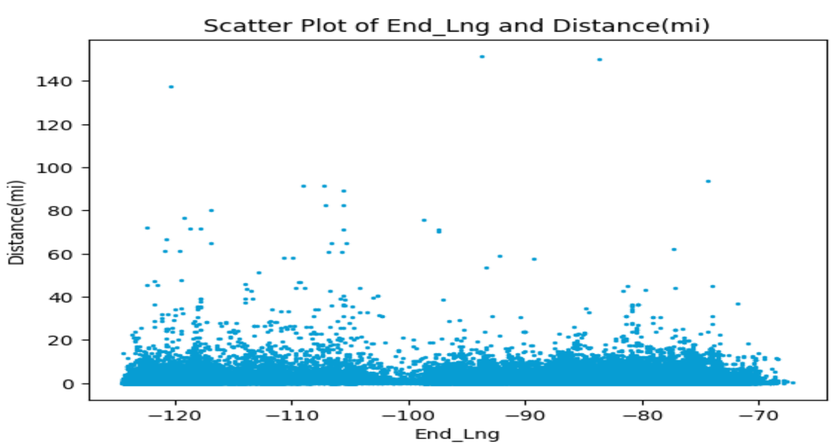
Accidents mostly occur at low visibility

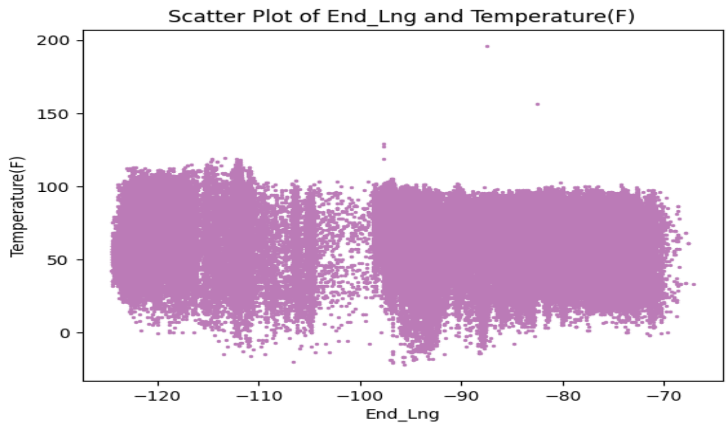


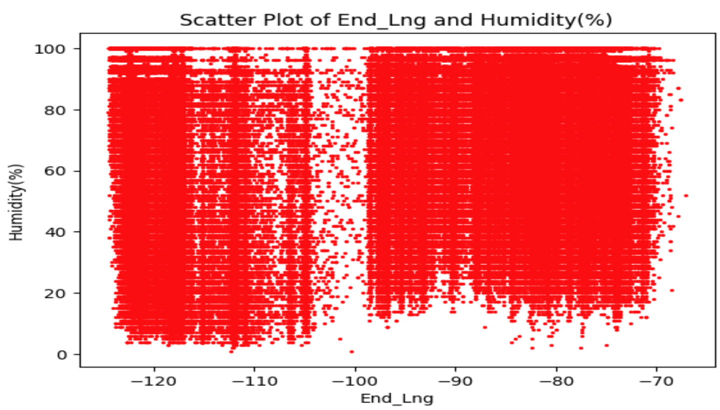
There were some instance of high wind speed near latitude of 30 to 35 where major accidents occurred else mostly the wind speed remains low

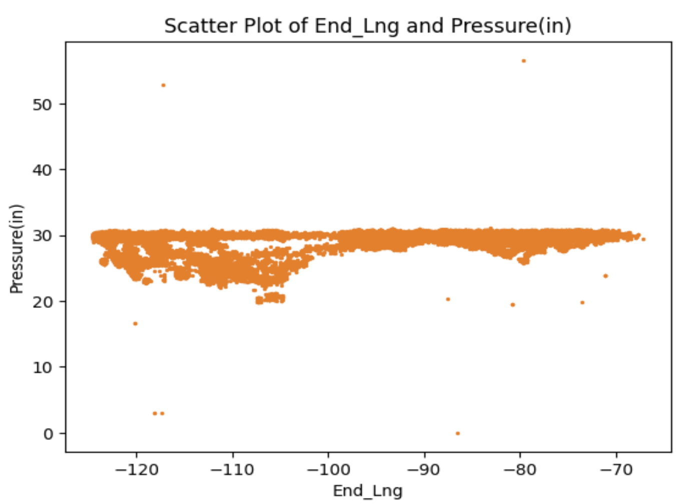
Now, we do similar scatter plots, but wrt other output column, End_Lng



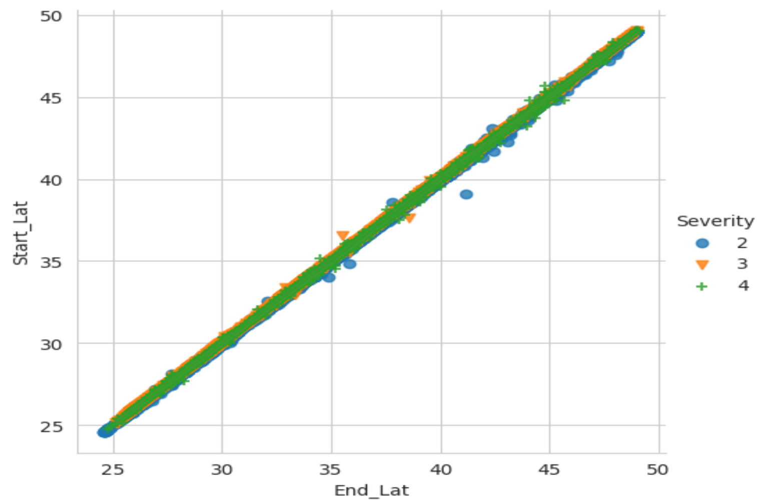




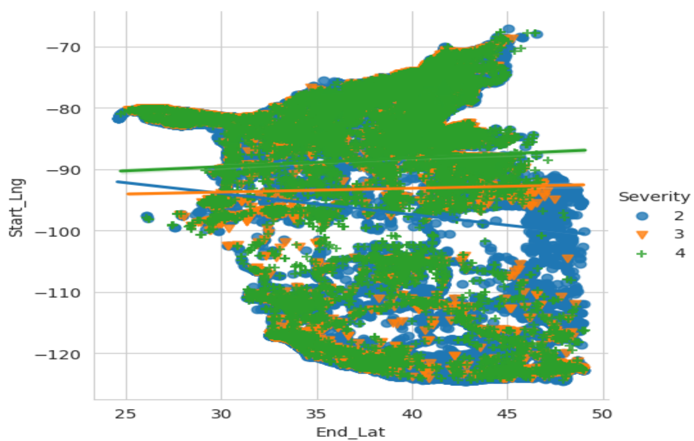




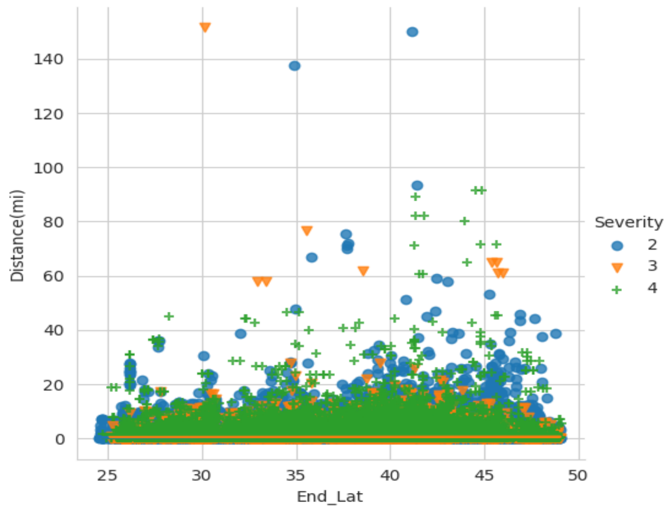
So, the plots are quite similar to the End_Lat only difference occurs in values of End_Lng. Now, we do regression plots wrt all numerical columns:



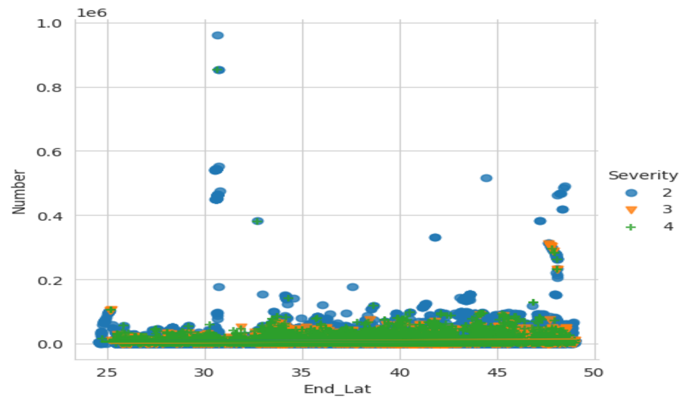
50
45
40
35
30
25
Start_Lat
25
30
35
40
45
50
End_Lat
Severity
2
3
4

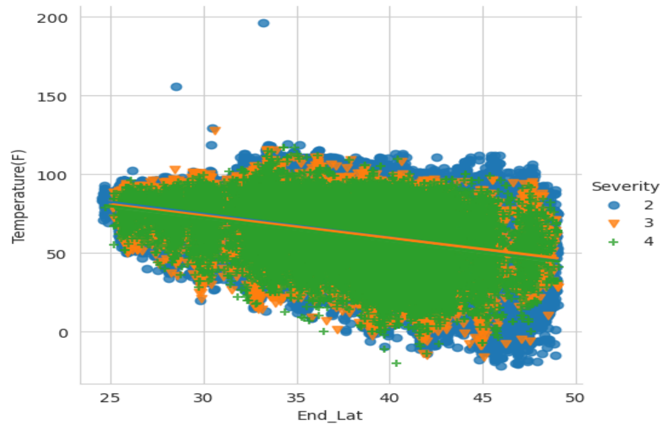


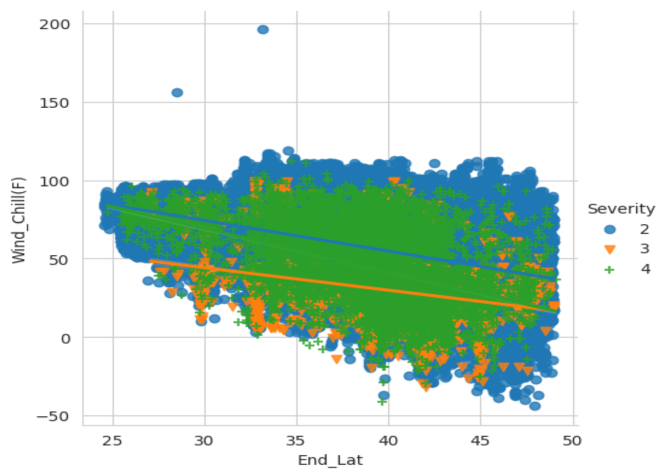
−70
−80
−90
−100
−110
−120
Start_Lng
25
30
35
40
45
50
End_Lat
Severity
2
3
4

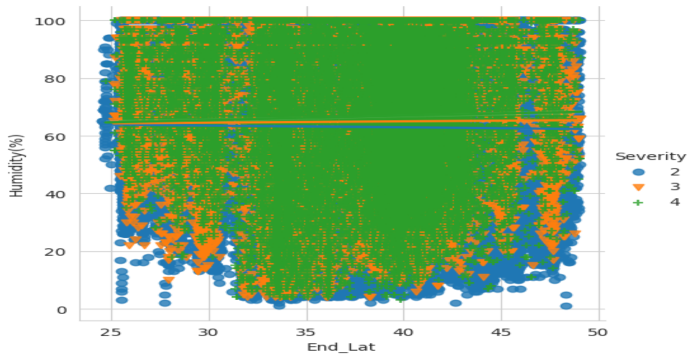


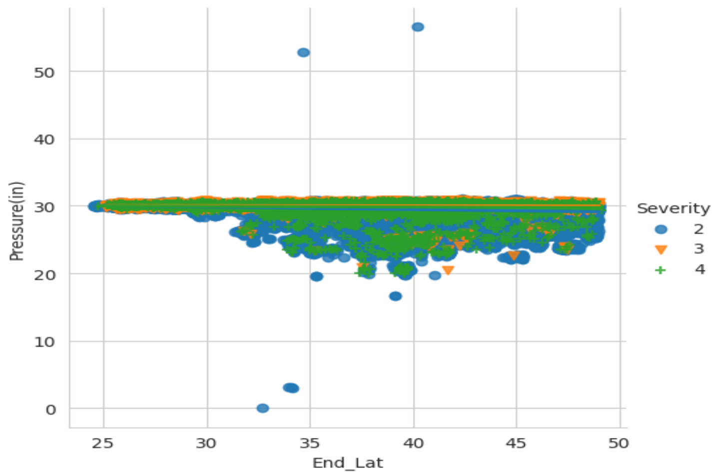
140
120
100
80
60
40
20
0
Distance(mi)
25
30
35
40
45
50
End_Lat
Severity
2
3
4


1e6
1.0
0.8
0.6
0.4
0.2
0.0
Number
25
30
35
40
45
50
End_Lat
Severity
2
3
4

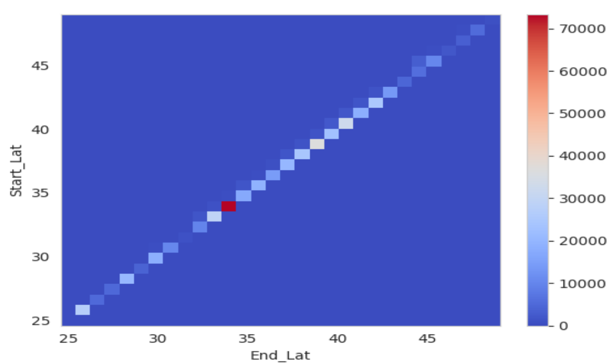

200
150
100
50
0
Temperature(F)
25
30
35
40
45
50
End_Lat
Severity
2
3
4

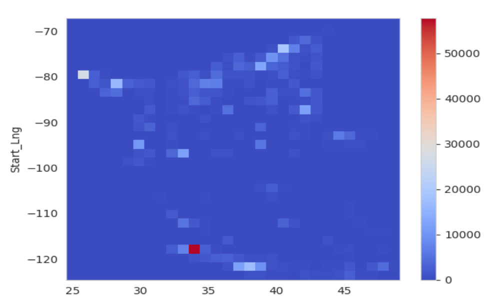

200
150
100
50
0
−50
Wind_Chill(F)
25
30
35
40
45
50
End_Lat
Severity
2
3
4

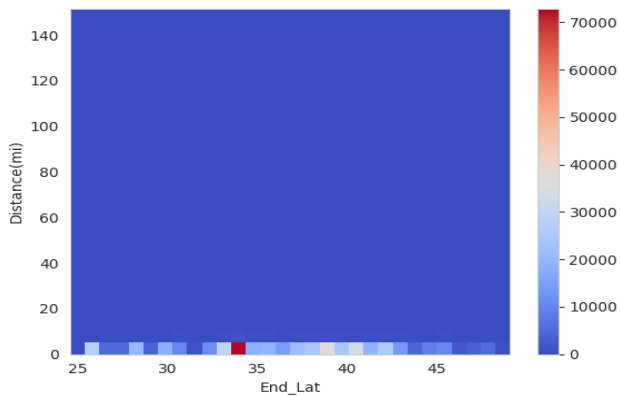




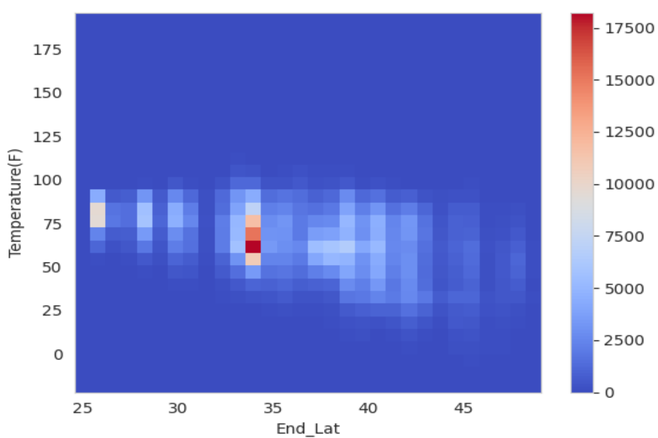
We also plot 2D histplot:



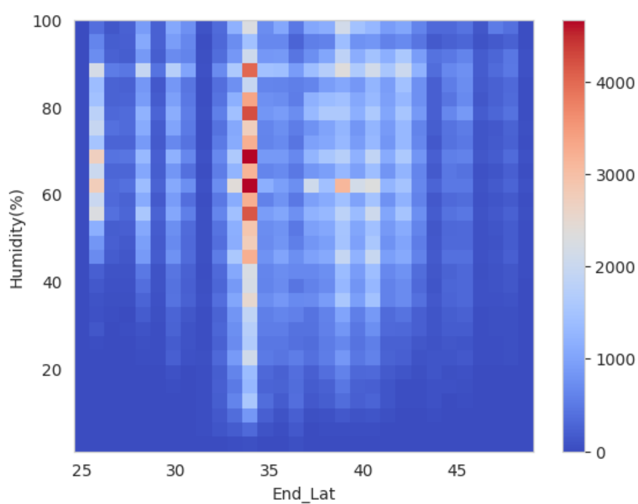




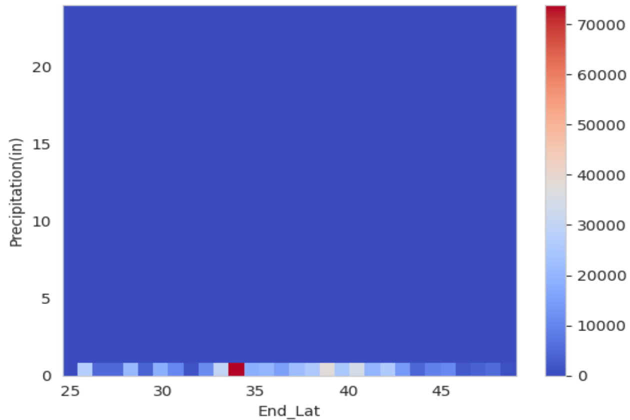
Distance(mi)
End_Lat



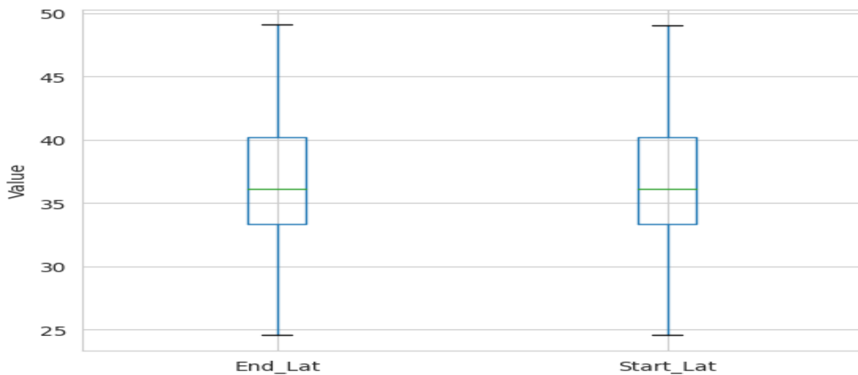
Temperature(F)
End_Lat

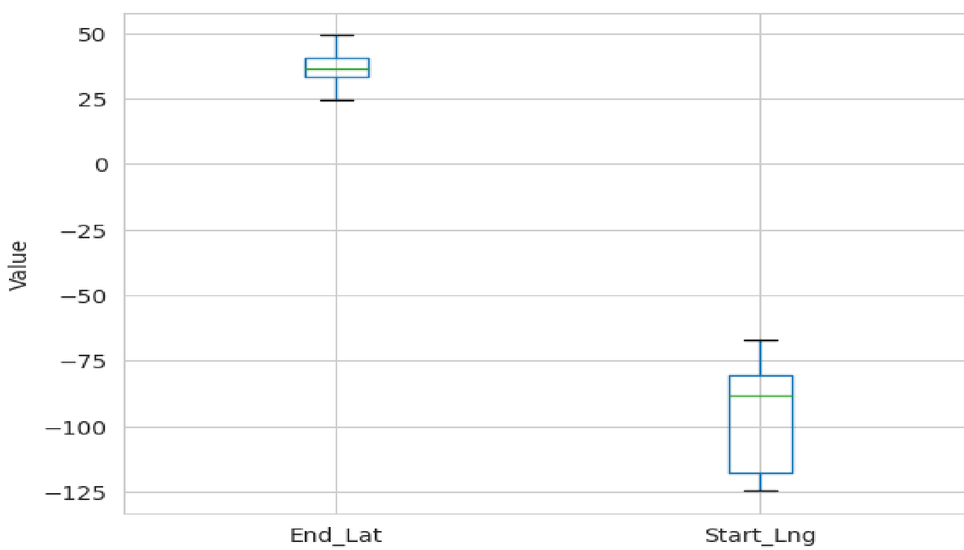


Humidity(%)
End_Lat

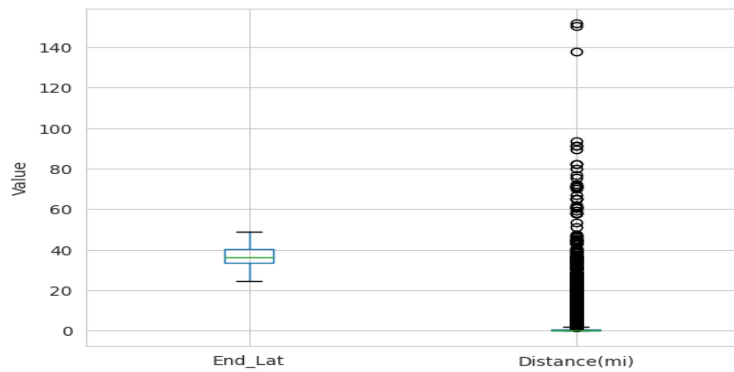


All these plots provide some great insights about how data relates to the location coordinates
Now, we plot box plots wrt all the continuous columns:

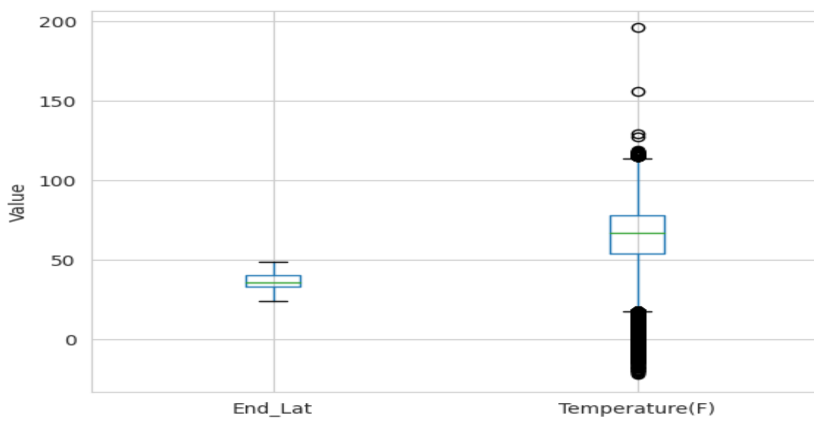




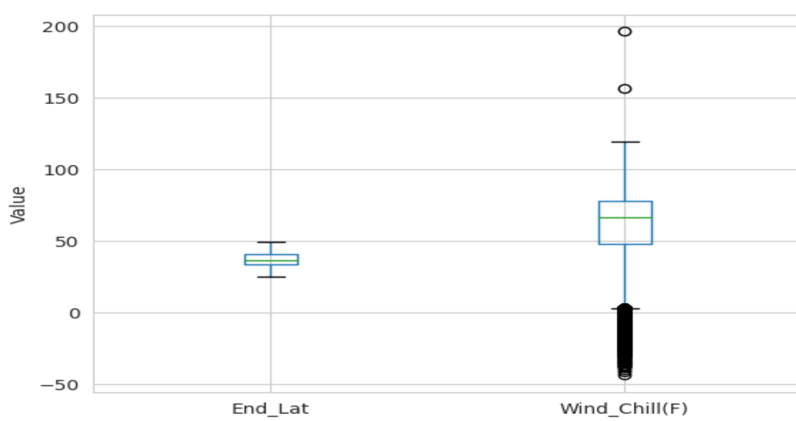
So, the End_Lat has a much higher value than Start_Lng



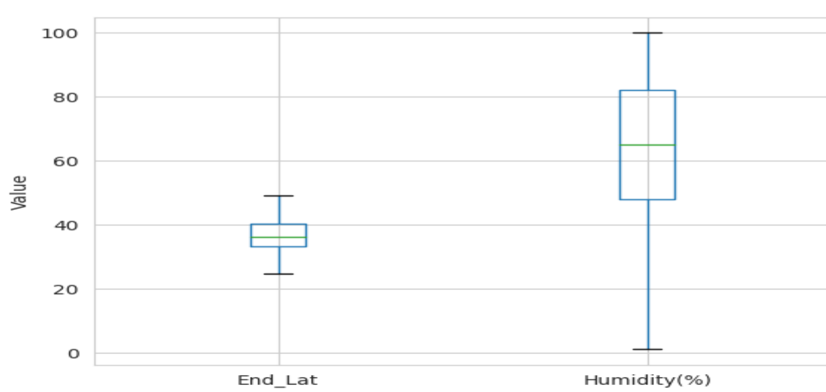
140
120
100
80
60
40
20
0
Value
End_Lat
Distance(mi)



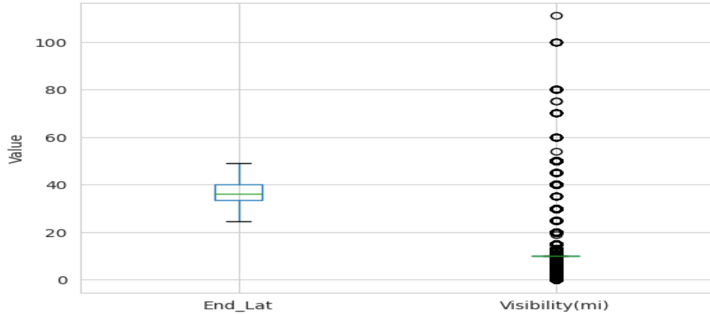
200
150
100
50
0
Value
End_Lat
Temperature(F)

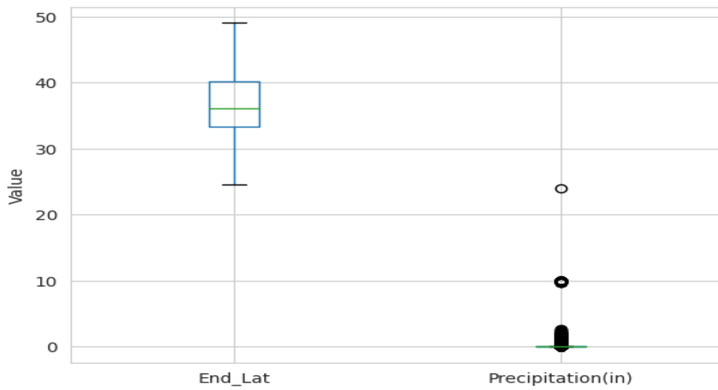


200
150
100
50
0
−50
Value
End_Lat
Wind_Chill(F)



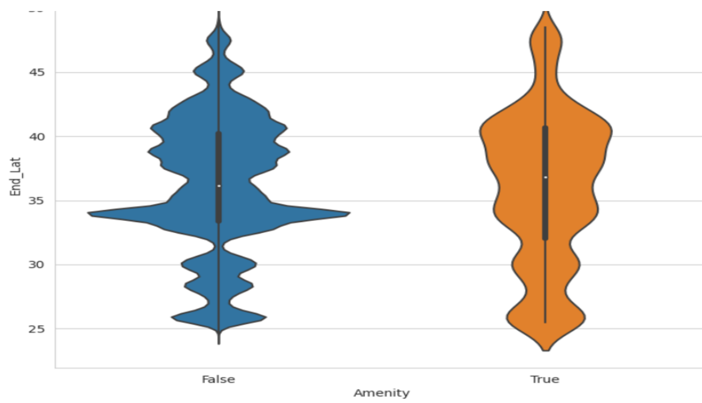
100
80
60
40
20
0
Value
End_Lat
Humidity(%)

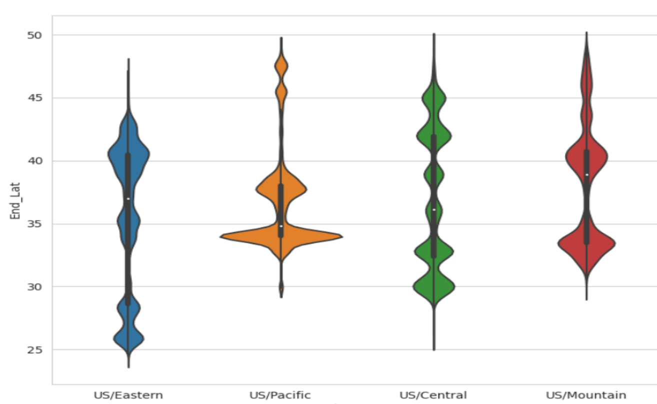


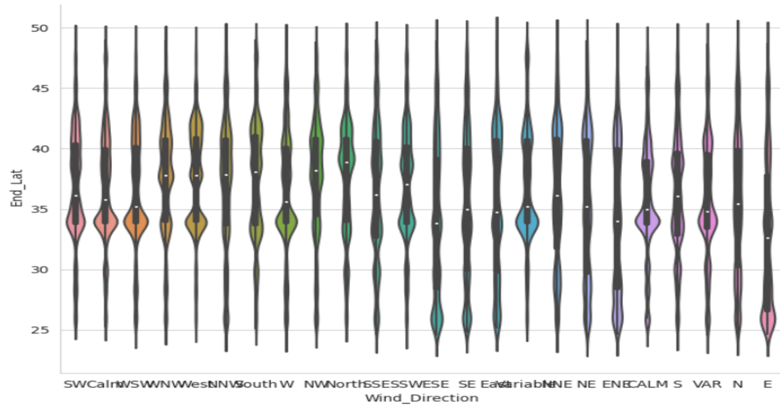


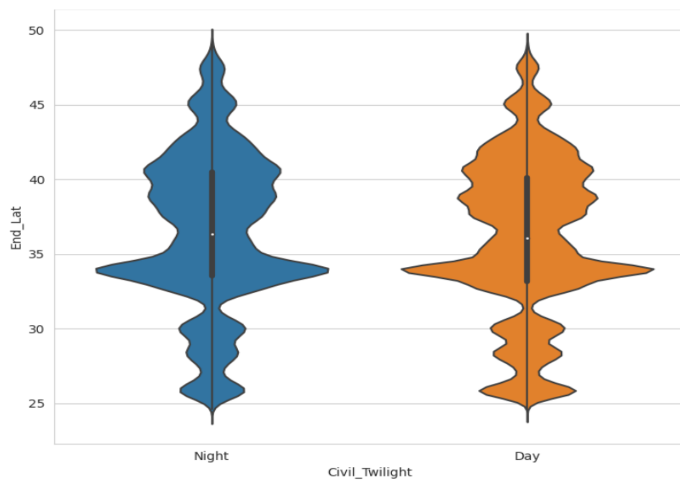
2) Categorical Columns vs Numerical Columns:

We plot violin plots as shown:



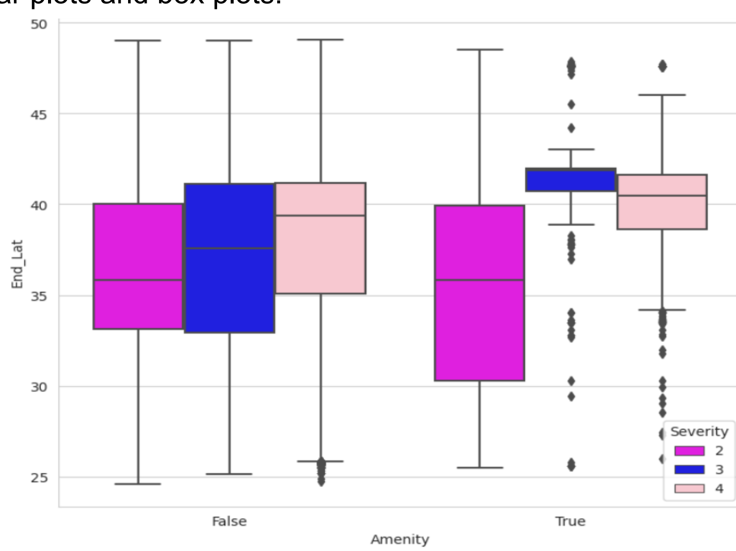


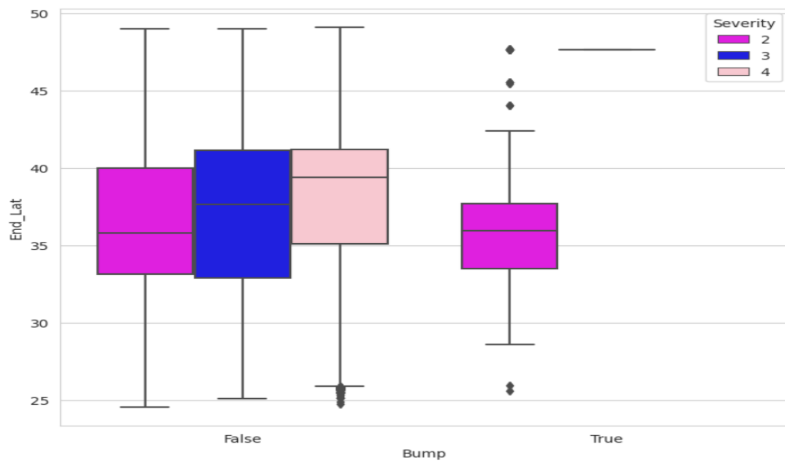


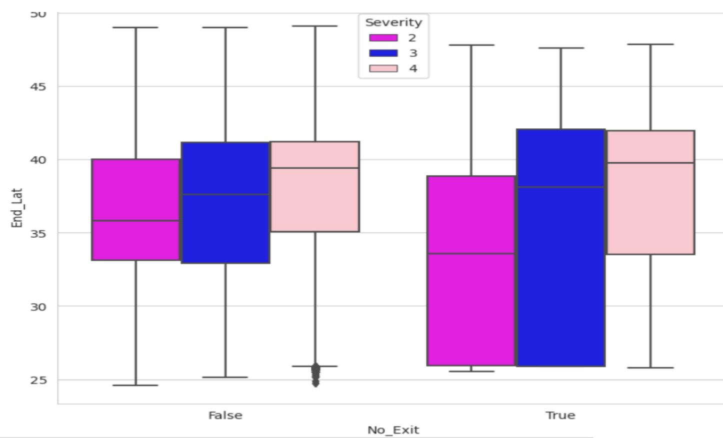


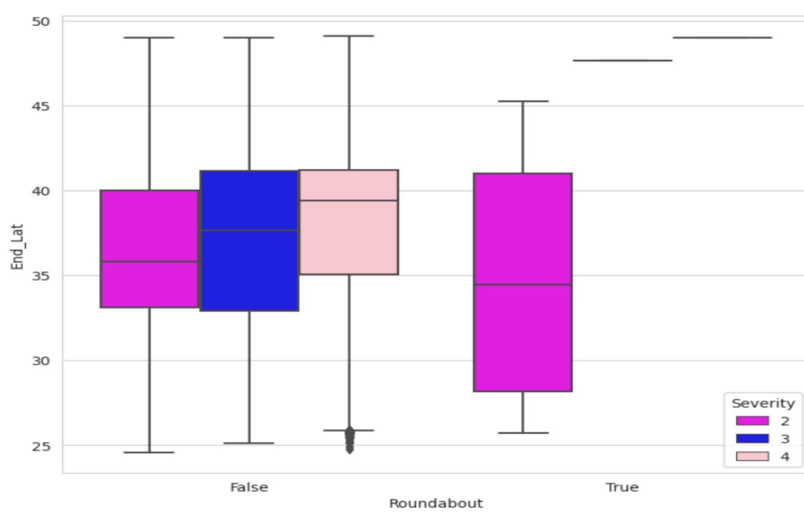
Similarly, we plot more such plots one can refer collab file

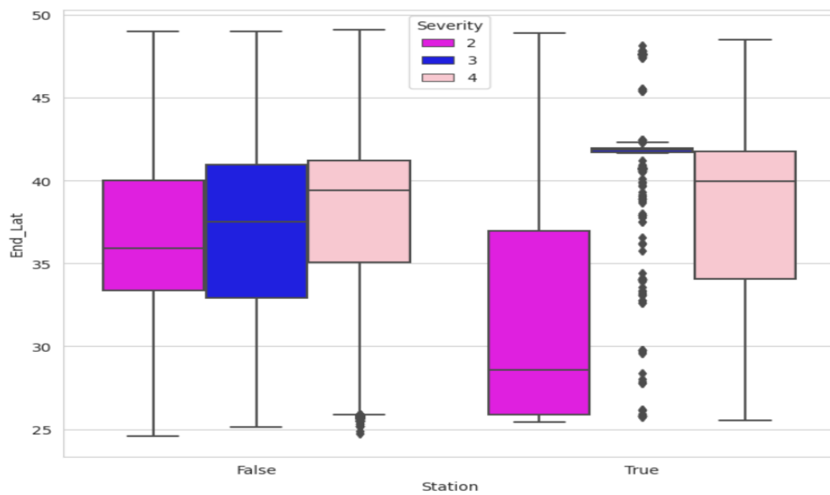
We also plot bar plots and box plots:



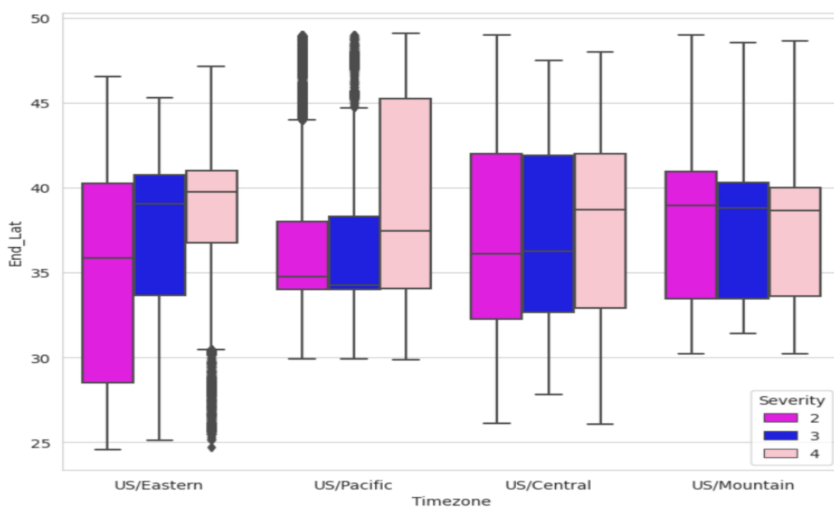

End_Lat
50
45
40
35
30
25
False
True
Bump
Severity
2
3
4


End_Lat
50
45
40
35
30
25
False
True
No_Exit
Severity
2
3
4


End_Lat
50
45
40
35
30
25
False
True
Roundabout
Severity
2
3
4



So, accidents having Station nearby have a lot of outliers in location for high severity accidents

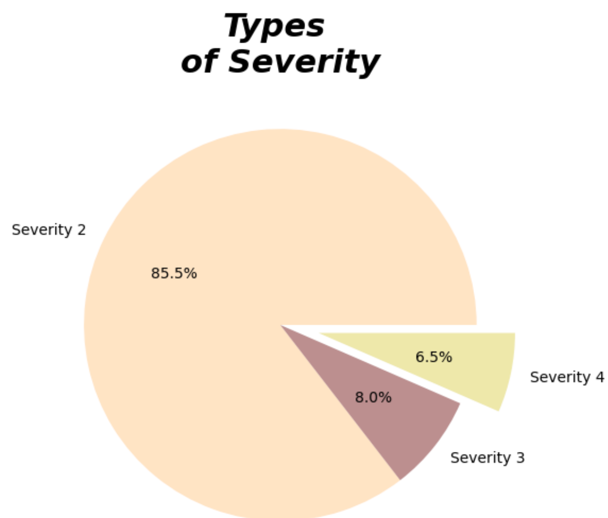


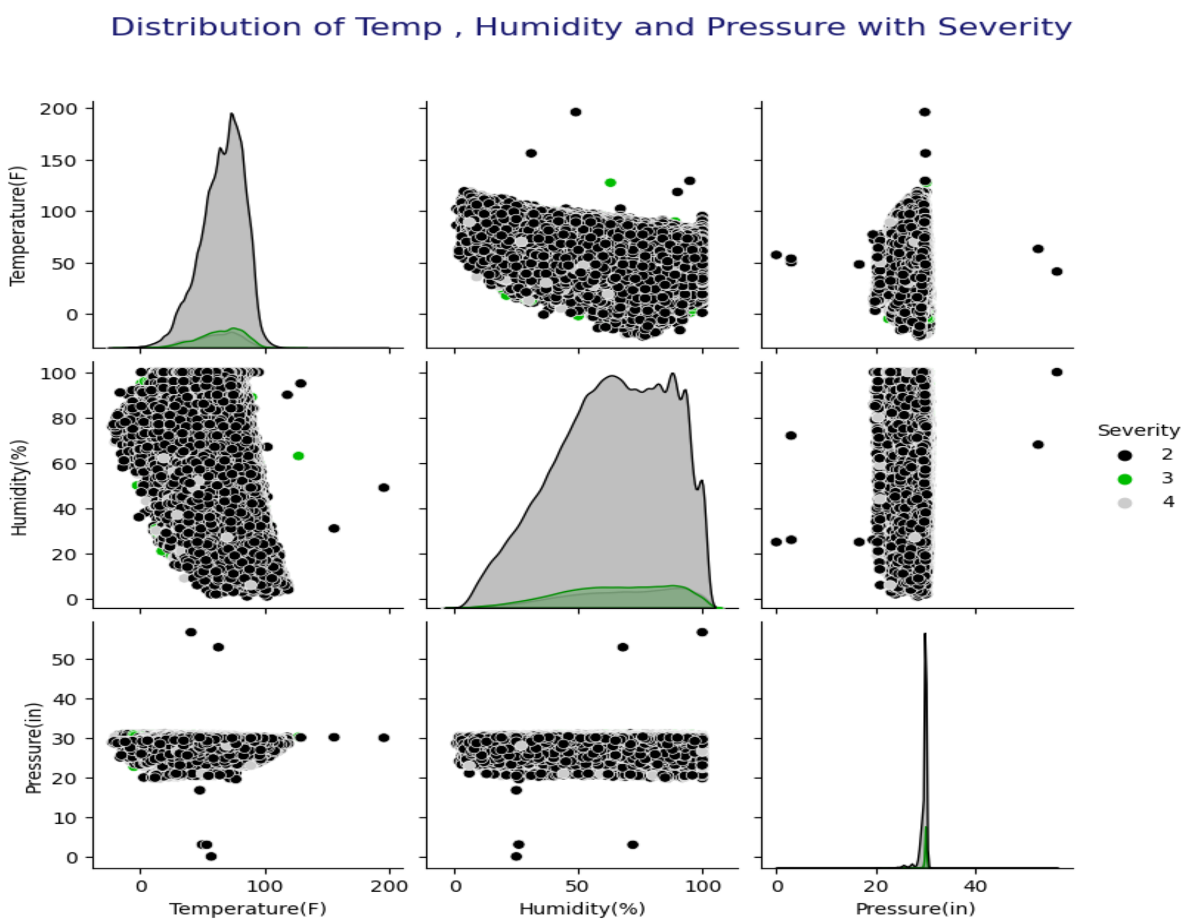
After this, we find the columns having high correlation with the output and delete them and then fill nan values using the similar pipeline structure used in the Severity model.

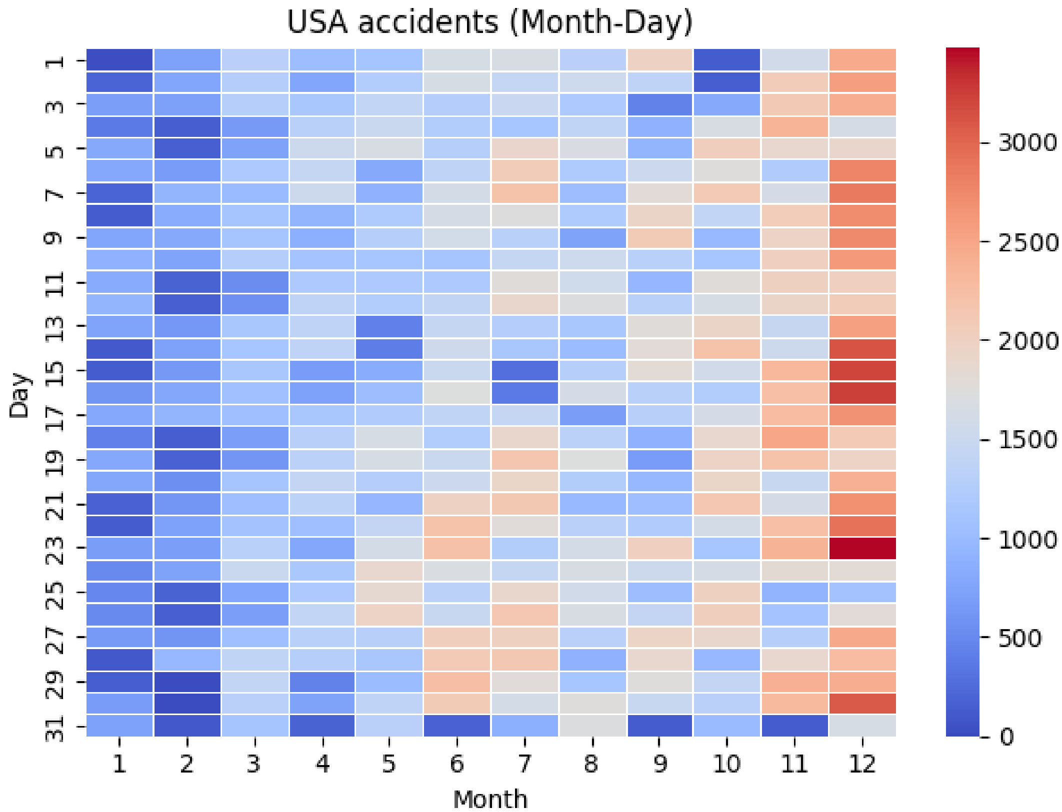
Then we use the similar keras tuner structure for finding the best neural network with the only difference being a regression problem so we use output layer activation as linear and loss as mean squared error.

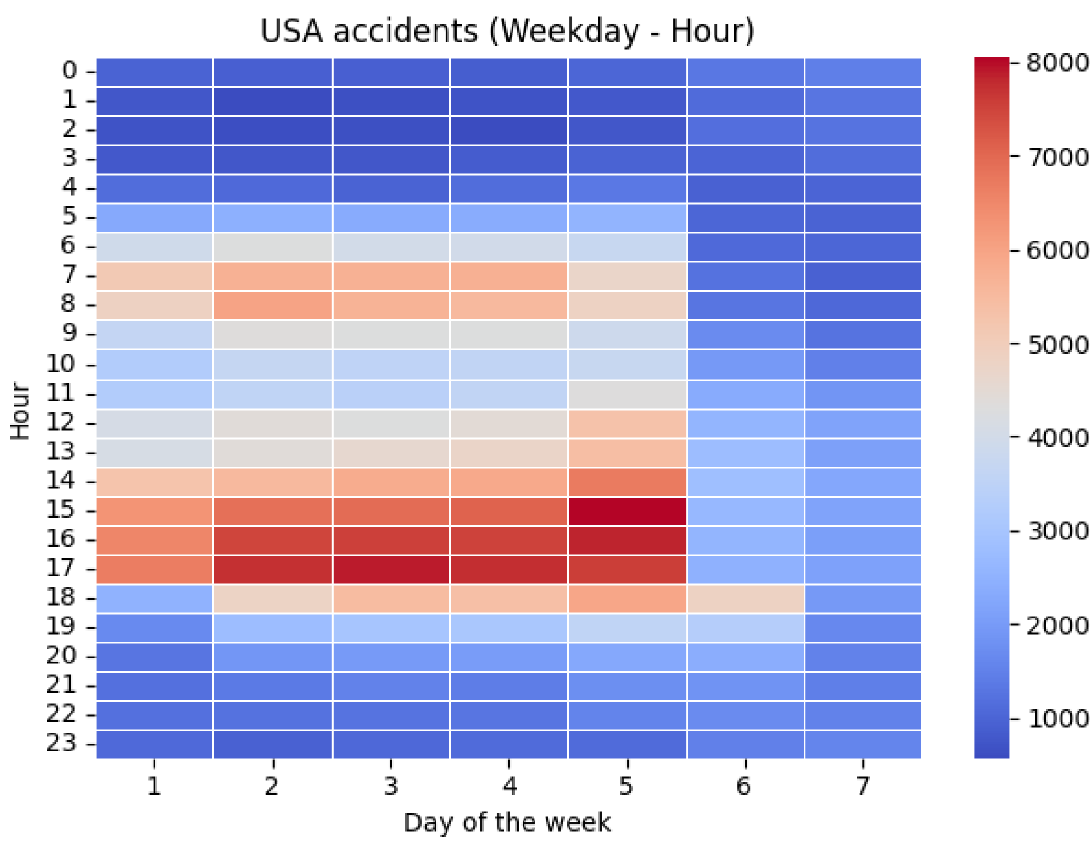
We get the validation loss as 0.08 and training loss as 0.076 so our model thus performs very very well on the data and the location prediction comes out to be very close to actual coordinates of the accident

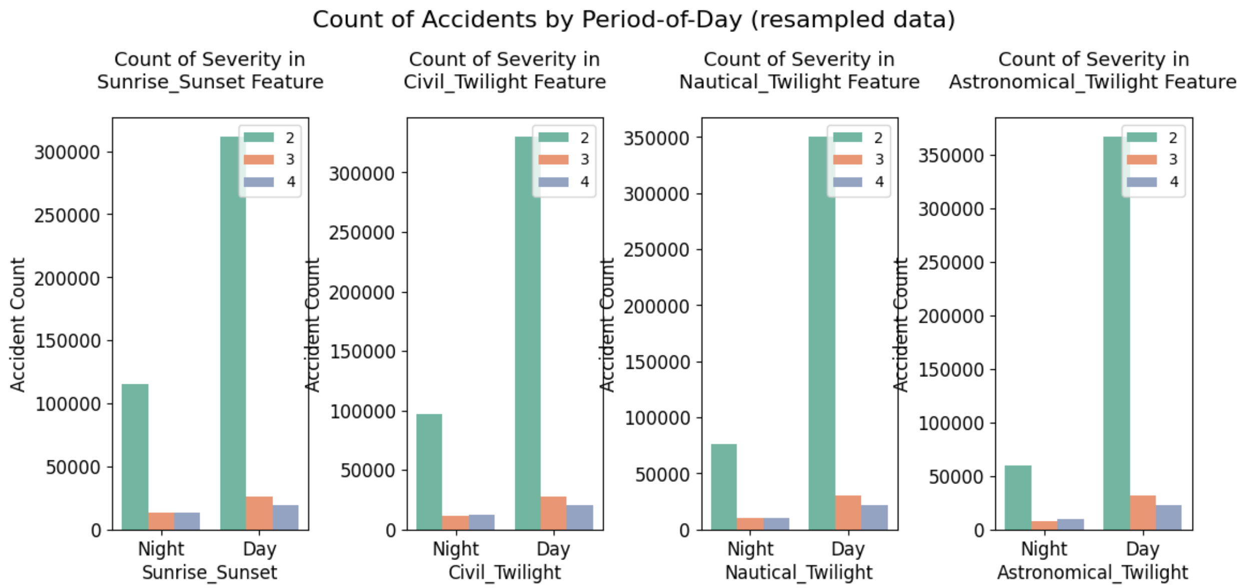
Some more Exploratory Data Analysis:

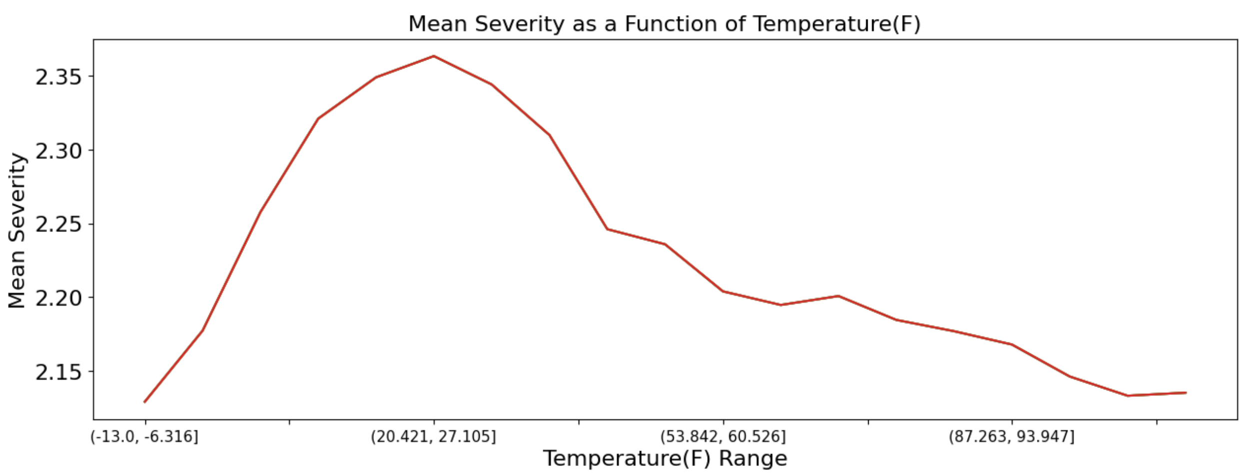


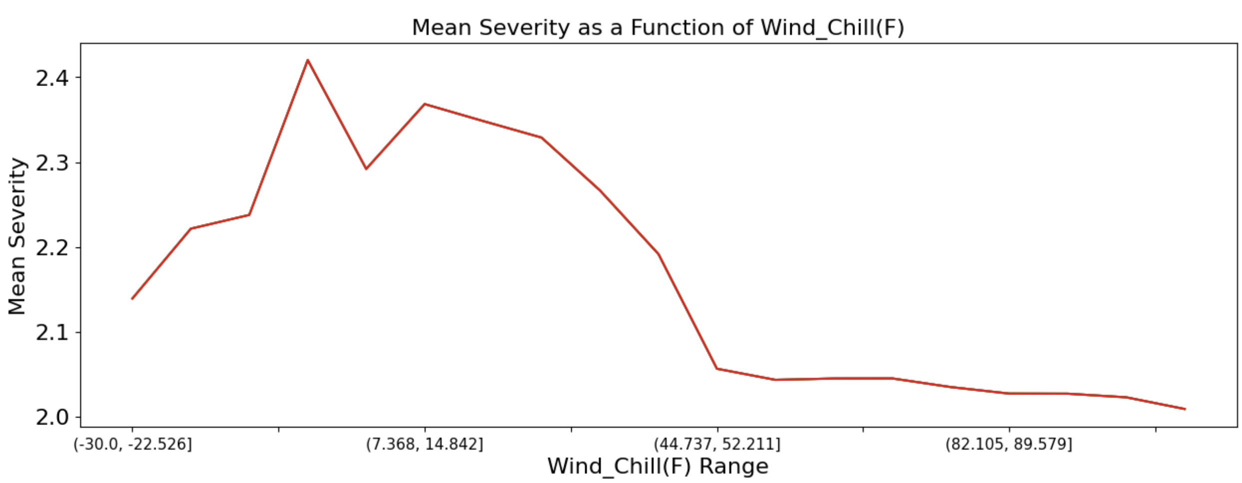


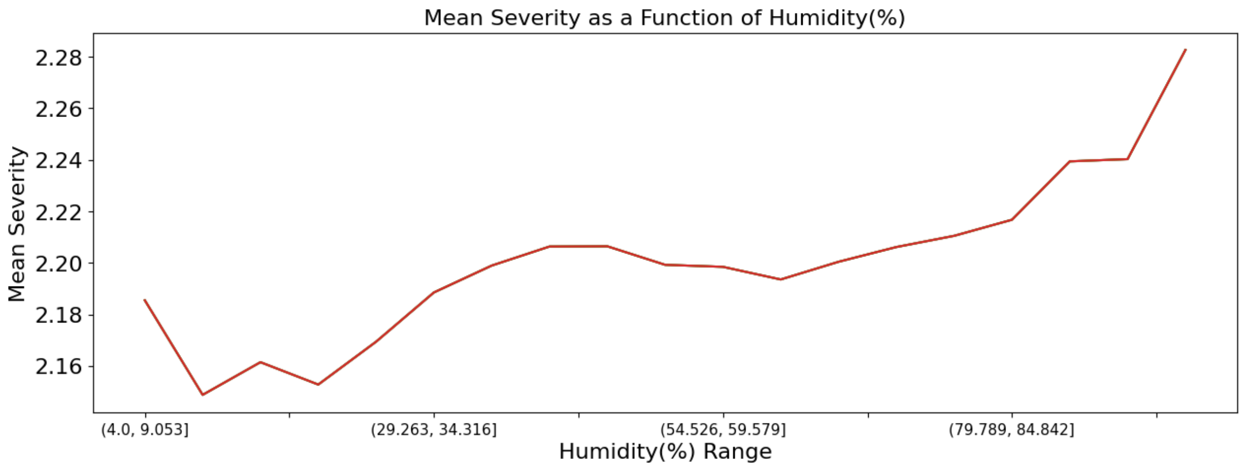

USA accidents (Month-Day)
Day
Month
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31
1 2 3 4 5 6 7 8 9 10 11 12
0
500
1000
1500
2000
2500
3000

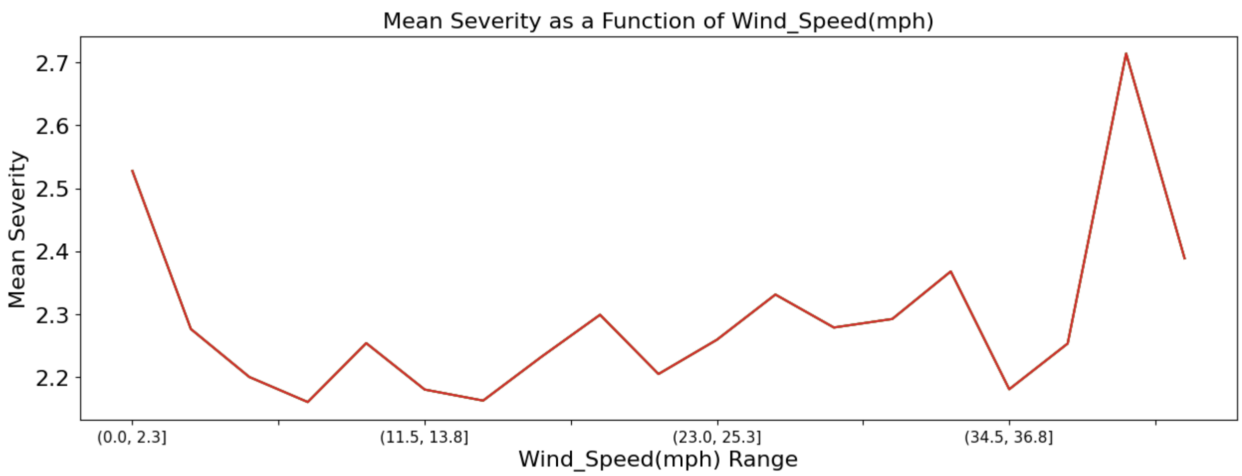

USA accidents (Weekday - Hour)
Hour
Day of the week
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
1 2 3 4 5 6 7
1000
2000
3000
4000
5000
6000
7000
8000

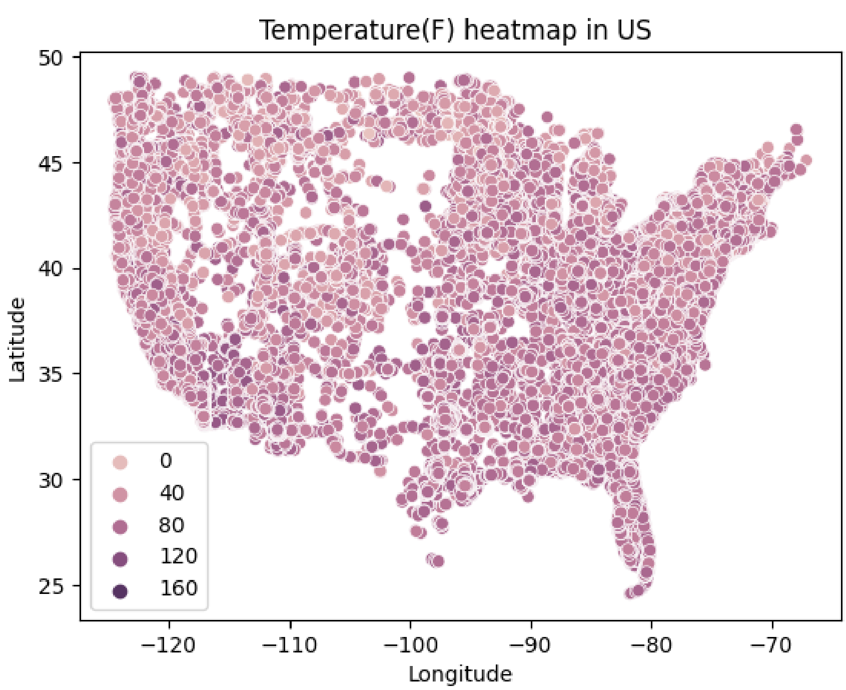

Count of Accidents by Period-of-Day (resampled data)
Count of Severity in Sunrise_Sunset Feature
Count of Severity in Civil_Twilight Feature
Count of Severity in Nautical_Twilight Feature
Count of Severity in Astronomical_Twilight Feature
Accident Count
2
3
4
Night
Day
Sunrise_Sunset
Civil_Twilight
Nautical_Twilight
Astronomical_Twilight

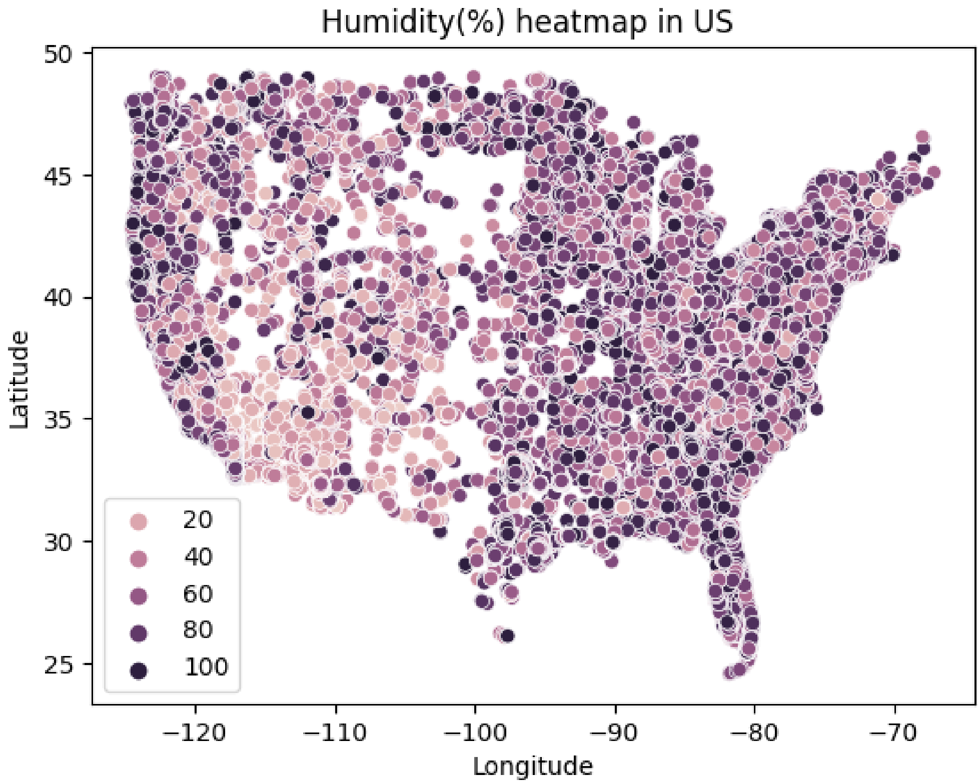

Mean Severity as a Function of Temperature(F)
Mean Severity
(-13.0, -6.316]
(20.421, 27.105]
(53.842, 60.526]
(87.263, 93.947]
Temperature(F) Range

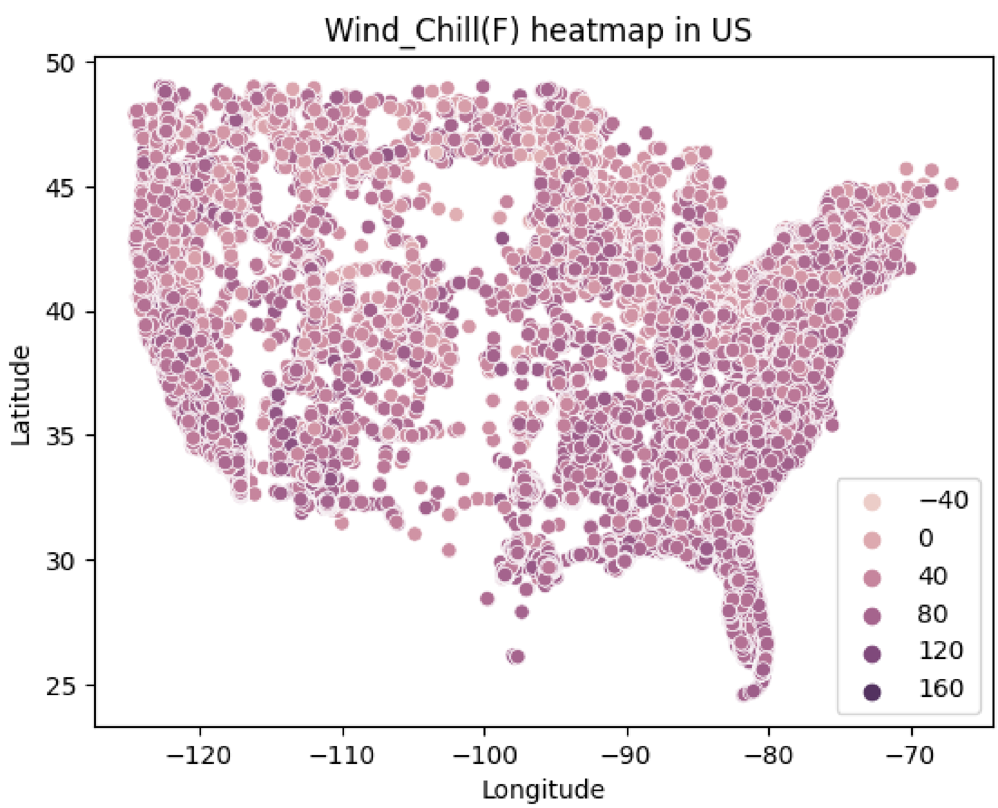

Mean Severity as a Function of Wind_Chill(F)
Mean Severity
(-30.0, -22.526]
(7.368, 14.842]
(44.737, 52.211]
(82.105, 89.579]
Wind_Chill(F) Range


Mean Severity as a Function of Humidity(%)
Mean Severity
2.28
2.26
2.24
2.22
2.20
2.18
2.16
(4.0, 9.053]
(29.263, 34.316]
(54.526, 59.579]
(79.789, 84.842]
Humidity(%) Range


Mean Severity as a Function of Wind_Speed(mph)
Mean Severity
2.7
2.6
2.5
2.4
2.3
2.2
(0.0, 2.3]
(11.5, 13.8]
(23.0, 25.3]
(34.5, 36.8]
Wind_Speed(mph) Range

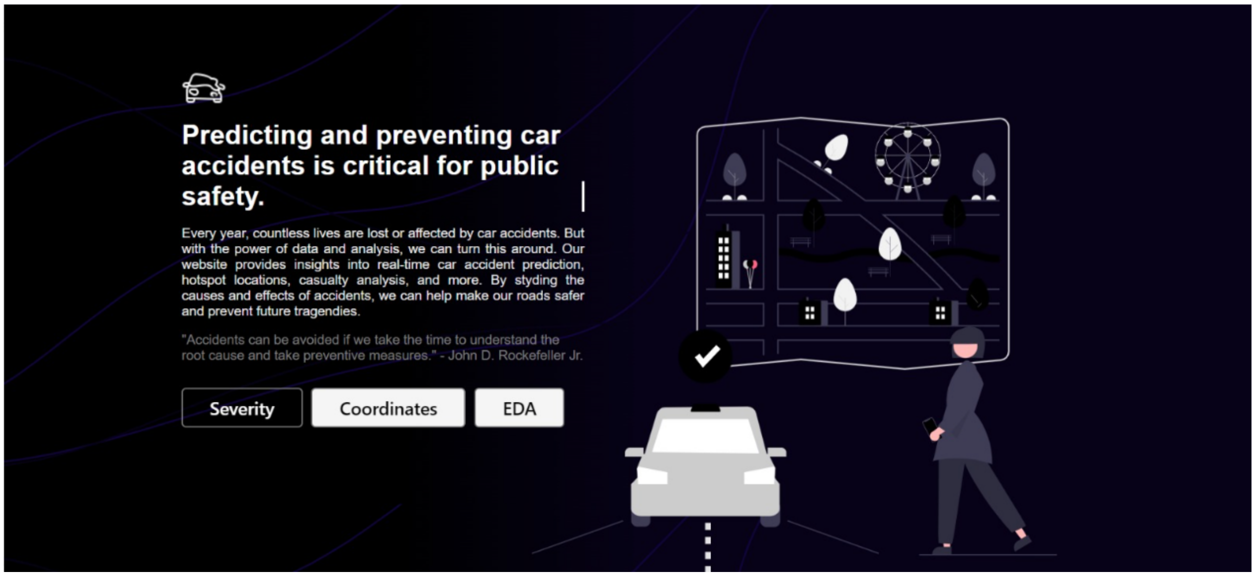

Temperature(F) heatmap in US
Latitude
50
45
40
35
30
25
0
40
80
120
160
−120
−110
−100
−90
−80
−70
Longitude


Humidity(%) heatmap in US
Latitude
Longitude
20
40
60
80
100

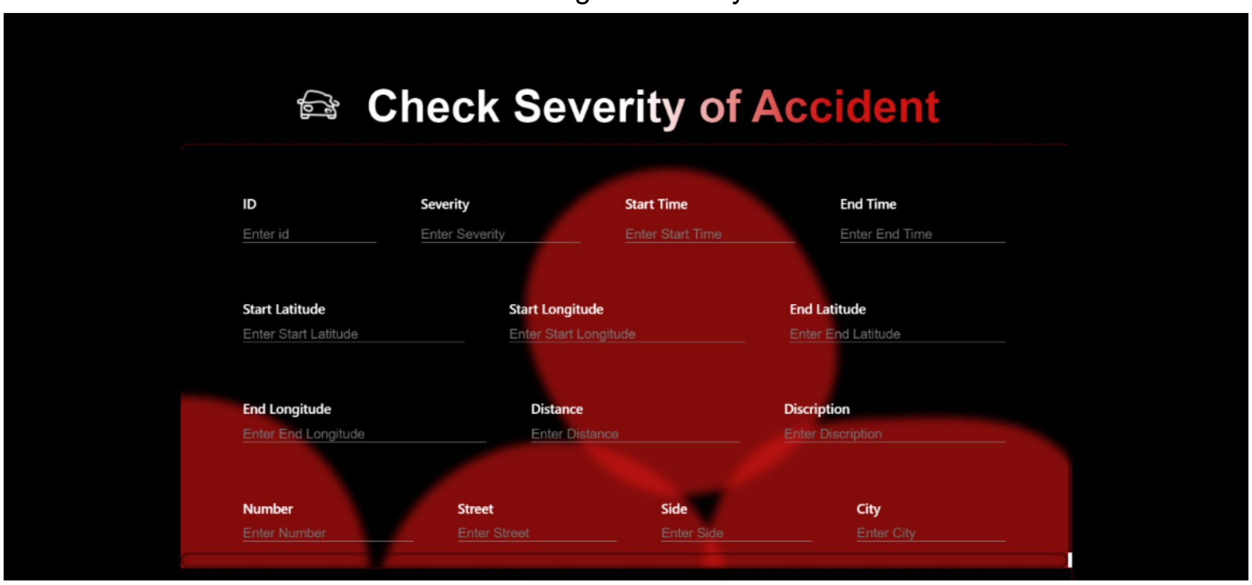

Wind_Chill(F) heatmap in US
Latitude
Longitude
−40
0
40
80
120
160

<u>Website Creation:</u>

We create a website (link : https://github.com/yuvrajrathva/ML-Major-Project)
The home page of our website:

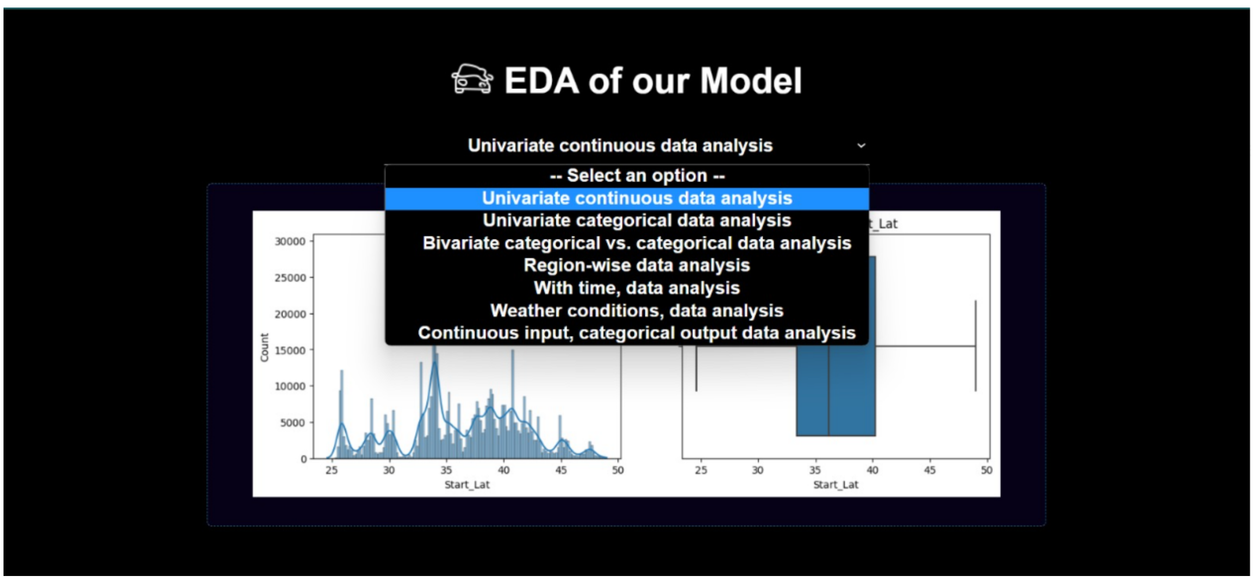


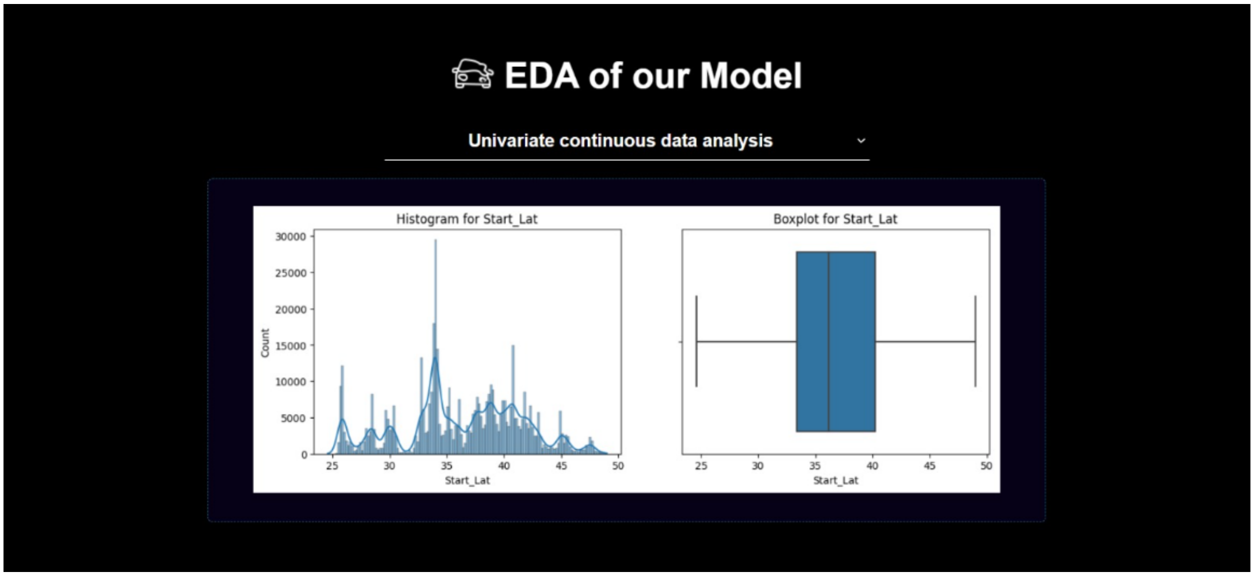
Clicking on Severity:



Clicking on Coordinates:

Now, clicking on the EDA





Contribution:

The following contributions were made by each team member for the project:

1. **Yuvraj Rathwa was responsible for creating the project website.**
2. **Sagnik and Akriti conducted exploratory analysis to gain insights into the dataset.**
3. **Sagnik and Yuvraj collaborated on data preprocessing, ensuring that the data was clean and ready for analysis.**
4. **Sagnik utilized Keras Tuner to tune the hyperparameters of the neural network, and was responsible for all deep learning tasks.**
5. **Akriti deployed all the machine learning models and and was responsible for its complete analysis.**
6. **Akriti and Yuvraj worked together on hyperparameter tuning to optimize the machine learning models for accuracy.**
7. **All team members collaborated on pipelining, ensuring that the various components of the project were integrated and functioned smoothly together.**