ANALYSING US ACCIDENTS AND THEIR RELATED FACTORS

BY:

Kratika Sharma

ISYS 812 | Fall 2021 | San Francisco State University

ABSTRACT

Reducing the number of accidents is an important public safety challenge all over the world. Countries are implementing strong and good traffic rules and are working hard to give robust and safe roads. Therefore, accident analysis has always been a subject of research in recent years.

The aim of this project is to do analysis of the data of US accidents and inform government agencies and the public of the recent trend of accidents and highlight the factors which are contributing towards the severe and high number of accidents. So, measures can be taken by the public and government to reduce the number of accidents.

The analysis includes the number of accidents and severity of accidents by state, city, time of day, year, month, weather conditions, road conditions, time zone, visibility, temperature, etc. .

Complete analysis has been done on a Jupyter notebook using python. Different visualizations have been produced which can give insights to the people to take preventive measures to avoid road accidents.

Table of Contents

1. INTRODUCTION

In the world of today where data is being used to derive useful insights in all spheres, we wanted to use the power of data to analyze various aspects related to a lethal problem that the world, especially the United States of America is facing. Thus, the name of the dataset we have analyzed is **"US Accidents".** We noted that the country is one of the busiest countries in terms of road traffic with nearly 280 million vehicles in operation and more than 227.5 million drivers holding a valid driving license. Road accidents have become very common these days. In the USA, over 40k people die in road accidents each year, and 3 million get injured. Road crashes cost the country around $230 billion per year or an average of $ 822 per person.

1.1. ABOUT THE DATASET AND DATA COLLECTION:

As a part of our analysis, we have focused on the data of accidents in the United States of America and have considered and have taken into account the data related to all the 50 states. The dataset contains data of all the accidents in 50 states of the US. The data is collected using multiple API's that provide streaming traffic incident data. It has 47 variables and 1048576 observations. We have information about each observation which include the place of the accident, the time of the accident, the temperature at the place, the humidity of the place, the severity of the accident, the time of the day, the wind speed, the weather prediction for that day and many more.

Table 1: Description of columns of the data set

Column Name	Description
ID	This is a unique identifier of the accident record.

Severity	Shows the severity of the accident, with 1 being least severe and 4 being highest severity.
Start_Time	Shows start time of accident in local timezone
End_Time	Shows end time of accident in local timezone
Start_Lat	Shows latitude in GPS coordinate of the start time.
Start_Lng	Shows longitude in GPS coordinate of the start time.
End_Lat	Shows latitude in GPS coordinate of the end time.
End_Lng	Shows longitude in GPS coordinate of the end time.
Distance(mi)	The length of road extent affected by the accident.
Description	Natural language description of road accident.
Number	Shows the number in address field.
Street	Shows the street name in address field.
Side	Shows the relative side of strees(right/left).
City	Shows the city in address field.
County	Shows the county in addess field.
State	Shows the state in addess field.
Zipcode	Shows the zipcode in addess field.
Country	Shows the country in addess field.
Timezone	Shows the timezone (Central/Pacific,etc.).
Airport_Code	Record the airport based weather station.
Weather_Timestamp	Shows the timestamp of the weather observation record.
Temperature(F)	Shows the temperature.
Wind_Chill(F)	Shows the Wind Chill.
Humidity(%)	Shows the humidity.
Pressure(in)	Shows the Pressure.
Visibility(mi)	Shows the Visibility.
Wind_Direction	Shows the direction of wind.
Wind_Speed(mph)	Shows the speed of wind.
Precipitation(in)	Shows the precipitation(in) amount, if ther eis any.
Weather_Condition	Shows weather condition like rain, storm , etc.
Amenity	A POI annotation which indicates presence of amenity in a nearby location.
Bump	A POI annotation which indicates presence of bump in a nearby location.
Crossing	A POI annotation which indicates presence of Crossing in a nearby location.
Give_Way	A POI annotation which indicates presence of give_way in a nearby location.
Junction	A POI annotation which indicates presence of junction in a nearby location.
No_Exit	A POI annotation which indicates presence of no_exit in a nearby location.
Railway	A POI annotation which indicates presence of railway in a nearby location.
Roundabout	A POI annotation which indicates presence of roundabout in a nearby location.
Station	A POI annotation which indicates presence of station in a nearby location.
Stop	A POI annotation which indicates presence of stop in a nearby location.
Traffic_Calming	A POI annotation which indicates presence of traffic_calming in a nearby location.

Traffic_Signal	A POI annotation which indicates presence of traffic_signal in a nearby location.
Turning_Loop	A POI annotation which indicates presence of turning_loop in a nearby location.
Sunrise_Sunset	Shows the period of day based on sunset or sunrise.
Civil_Twilight	Shows the period of day based on civil_twilight.
Nautical_Twilight	Shows the period of day based on nautical_twilight.
Astronomical_Twilight	Shows the period of day based on astronomical twilight.

We have picked this data as it has almost all type of data such as time, date, numerical values, strings, floating type integers, Boolean values etc. and we wanted to experiment and learn data analysis on datasets which are much like those in the real world. The size of the dataset is also huge, and it might give us an opportunity to dive deep into the data and understand the basic concepts of data analysis in a much better way.

1.2. Missing Values in the Columns

There were multiple missing values in various columns of the dataset. The following table shows the number and percentages of missing values in each column in descending order.

Table 2: Number of missing values in each column

Column	Missing values
Number	1046095
Precipitation(in)	510549
Wind_Chill(F)	449316
Wind_Speed(mph)	128862
Humidity(%)	45509
Visibility(mi)	44211
Weather_Condition	44007
Temperature(F)	43033
Wind_Direction	41858
Pressure(in)	36274
Weather_Timestamp	30264

Airport_Code	4248
Timezone	2302
Zipcode	935
City	83
Nautical_Twilight	83
Astronomical_Twilight	83
Civil_Twilight	83
Sunrise_Sunset	83
Amenity	0
Bump	0
Severity	0
Start_Time	0
End_Time	0
Start_Lat	0
Start_Lng	0
End_Lat	0
End_Lng	0
Distance(mi)	0
Description	0
Street	0
Side	0
County	0
State	0
Turning_Loop	0
Country	0
Traffic_Signal	0
Traffic_Calming	0
Stop	0
Station	0
Roundabout	0
Railway	0
No_Exit	0
Junction	0
Give_Way	0
Crossing	0
ID	0

Table 3: Percentages of missing values in each column

Column	Missing values
Number	0.690007
Precipitation(in)	0.33676
Wind_Chill(F)	0.29637
Wind_Speed(mph)	0.084998
Humidity(%)	0.030018
Visibility(mi)	0.029162
Weather_Condition	0.029027
Temperature(F)	0.028385
Wind_Direction	0.02761
Pressure(in)	0.023926
Weather_Timestamp	0.019962
Airport_Code	0.002802
Timezone	0.001518
Zipcode	0.000617
Nautical_Twilight	0.000055
Sunrise_Sunset	0.000055
Civil_Twilight	0.000055
City	0.000055
Astronomical_Twilight	0.000055

2. PROBLEM STATEMENT AND DRIVING QUESTION

The purpose of our analysis is to explore some of the factors that contribute to road accidents in the United States. The number of fatal and disabling road accidents increases daily and is a real challenge for all to prevent. Awareness creation, strict implementation of traffic rules, and scientific engineering measures are the need of the hour to avoid this catastrophe.

To do the same, it is of utmost importance that we as people understand the trends and patterns of the accidents taking place in the country, and to do the same, analysis of the available data has to be carried out. Throughout our project, we have kept in mind the need of giving

concrete answers to important questions and finding a correlation between factors that would help the concerned authorities to prevent these accidents.

Our aim is to identify the parameters that affect the most for predicting the severity of an accident and to identify the main features that can be controlled to reduce the number of accidents or to decrease the level of severity of accidents.

3. DATA CLEANING

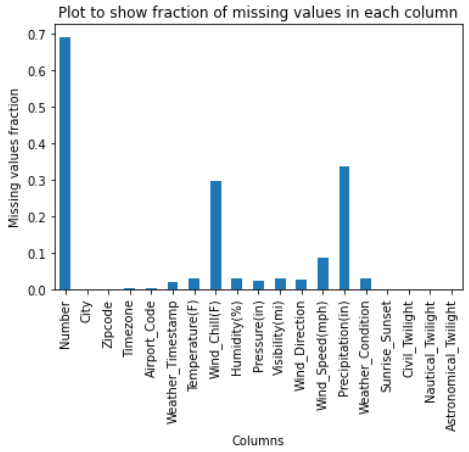
We have used different functions in Python and its associated libraries to clean our data. The data consists of 47 columns and 1,048,576 rows. Below are the steps we undertook to clean our data:

- Finding Null Values – Since the data is quite large, we suspected the presence of null/missing values and checked the same using the "isna" function.
- Data can have missing values for several reasons such as observations that were not recorded and data corruption. From the results of the function, we observed that out of 47 columns, 19 columns had null values.
- Out of these 19 columns, some were numeric. There are multiple ways to remove these null values: backfilling, forward filling, checking for outliers and then deciding to change it with mean or median, etc. According to ways of data manipulation, these values can be replaced by the median or mean of the observations/maximum or minimum value or zero value. But the approach of replacing these values for each column can be different depending on the nature of the data. We looked at the box plots of these

and according to the outliers present we replaced null values in numeric columns with median or mean.

- Some columns were of string data type. We applied our professional judgment and statistical theory to replace the values for each of the columns as explained in the following slides.
- We decided to drop certain columns which were not needed.
- We created columns: Month, Year, Day, Hour, Weekday from Start_time column
- We created Time_Diff column which is the difference between Start_time and End_time. This contains the duration of the accident.

Plot to show the fraction of missing values in each column



3.1. Dealing with missing values in numeric columns

Numeric columns that include Wind Speed, Humidity, Visibility, Temperature and Pressure, are important factors that may contribute towards high number of accidents or severe accidents. We need to analyze the impact of these factors on the severity and number

of accidents. But since these columns consists of null values, we need to clean these missing values from the data. We can clean these columns by taking below steps:

- First check the number of null values in a particular column.
- Compute the mean, median, max, min, and mode of the column.
- Create a box plot of the column and checked for outliers.
- If outliers were found, replace null values with median. If no outlier were found, replace null values with the mean.

There are multiple data points in our dataset that may act as outliers. These outliers will have an important impact on the mean of the data and hence, in these cases, it is not suggested to use means to replace the missing values in the data. So, for symmetric distribution of data we can use mean but for skewed dataset it is suggested to use median to replace missing values.

3.2. Dealing with missing values in string columns

Our dataset consists of missing values in multiple string columns which we need for our analysis. There is not one single strategy to clean such columns. We looked at the data and tried various methods to replace missing values in such columns and implemented the best method we could find for each column. To clean string columns, different strategies were employed.

3.2.1. Columns: Wind_Direction and Weather_Condition

For columns Wind Direction and Weather Condition, below steps were taken:

- It was observed some values in these columns are repeatedly written in different ways (case). Like ‘CALM’ and ‘Calm’, ‘North’ and ‘N’, etc. in

column Wind_Direction. And ‘Thunderstorm’ and ‘T-Storm’, etc. in Weather_Condition.

- To maintain uniformity in the data, we used the 'replace' function and replaced ‘Calm’ with ‘CALM’, ‘North’ with ‘N’, etc.
- Further, missing values were calculated and looking at the number of non-null values, we divided the missing number among all the non-null values by using fraction of the count of non-null values, as per below formula:

data['Wind_Direction'].value_counts()/len(data['Wind_Direction'])*missing['Wind_Direction']).round(0)

- We kept dividing these null values, till all the null values were replaced by non-null values.

3.2.2. Columns: City and Timezone

The null values in the columns like City and Time zone were cleared by using below steps:

City:

- We tried to see the correlation between the column 'City' and every other column which has zero null values
- After multiple hit and trials, we found that column 'County' is a best fit for consideration
- Each County has multiple City in it
- We created a Data Frame which has all the null 'City' and corresponding 'County'

- Since each 'County' has multiple 'City' in it, so, we found the 'City' with maximum number of accidents in each 'County' present in above created Data Frame
- Finally, we replaced the null values in column 'City' with the 'City' with maximum number of accidents in each 'County' present in above created Data Frame

Time zone:

- We tried to see the correlation between the column 'Timezone' and every other column which has zero null values
- After multiple hit and trials, we found that column 'County' and 'City' were the best fit for consideration
- Few County has multiple Time zones in it. Similarly, few City has multiple Time zone in it
- Upon checking, it was found that there are some Cities in the data set which has no Time zone associated with them
- But this was not the case with County. Every County has at least one associated Time zone in it
- We created a Data Frame which has all the null 'Timezone' and corresponding 'County'
- Since multiple 'County' has multiple 'Timezone' in it, so, we found the 'Timezone' with maximum number of accidents in each 'County' present in above created Data Frame

- Finally, we replaced the null values in column 'Timezone' with the 'Timezone' with maximum number of accidents in each 'County' present in above created Data Frame

3.3. Dropping Certain Columns

After understanding the columns and their definitions we decided to delete certain columns:

- The columns Number, Precipitation, and Wind_Chill had most missing values (40-60% of the data). So, it was wise to remove those columns altogether.
- Since this dataset was of United States only, so it made sense to drop column 'Country' as that possesses only one value which is United States
- Airport Code: Retaining the Airport code of the region where the accidents occurred wasn't very useful in our analysis. Since a city/county wise analysis is being carried out, the use of airport code would have been redundant and non-necessary.
- Zip Code: As we already have City and County. Further filling zip code null values in relation to city and county was not possible as every county and city had multiple zip codes. And City already had null values. Further our analysis does not require Zipcode as we were mostly linking to City or County level
- Astronomical_Twilight, Nautical_Twilight, and Civil_Twilight was not necessary for our analysis. So, we decided to drop those columns too
- We were already fetching hour of day from Start_Date column so, we did not need Sunset_Sunrise for analysis

- Final check on duplicate values was also done. Fortunately, there were no duplicate values found in the data

3.4. Introducing new columns

For our analysis, we needed to plot some graphs which could give us hourly, weekly, yearly, and monthly plots of the number of accidents and severity of accidents. To do so, we fetched the day, hour, week, year, and month of the accident from the Start_Time column. We further created another column named Time_Diff which is the difference between Start_time and End_time. This contains the duration of the accident.

Figure: Number of missing values in each column after data cleaning

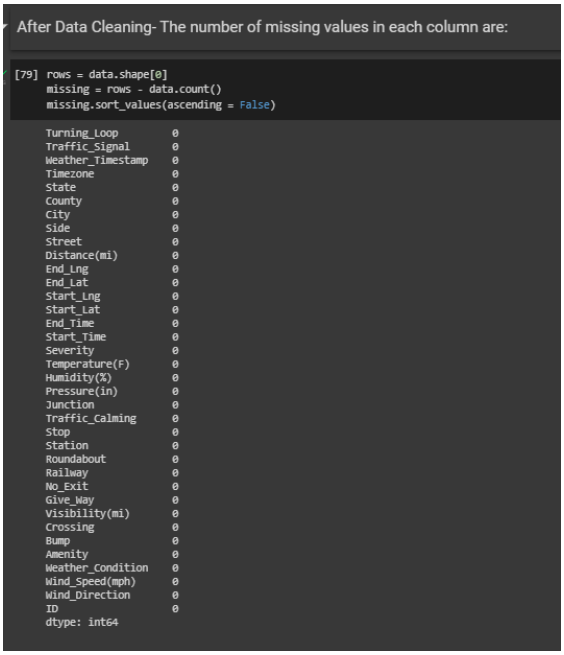


Figure: Data set after data cleaning

	ID	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance(mi)	Street	Side	City	County	State	Timezone	Weather_Timestamp	Temperature(F)	Humidity(%)	Pressure(in)	Visibility(mi)	Wi
0	A-2716600	3	2016-02-08 00:37:08	2016-02-08 06:37:08	40.10891	-83.09286	40.11206	-83.03187	3.230	Outerbelt E	R	Dublin	Franklin	OH	US/Eastern	2016-02-08 00:53:00	42.1	58.0	29.76	10.0	
1	A-2716601	2	2016-02-08 05:56:20	2016-02-08 11:56:20	39.86542	-84.06280	39.86501	-84.04873	0.747	I-70 E	R	Dayton	Montgomery	OH	US/Eastern	2016-02-08 05:58:00	36.9	91.0	29.68	10.0	
2	A-2716602	2	2016-02-08 06:15:39	2016-02-08 12:15:39	39.10266	-84.52468	39.10209	-84.52396	0.055	I-75 S	R	Cincinnati	Hamilton	OH	US/Eastern	2016-02-08 05:53:00	36.0	97.0	29.70	10.0	
3	A-2716603	2	2016-02-08 06:15:39	2016-02-08 12:15:39	39.10148	-84.52341	39.09841	-84.52241	0.219	US-50 E	R	Cincinnati	Hamilton	OH	US/Eastern	2016-02-08 05:53:00	36.0	97.0	29.70	10.0	
4	A-2716604	2	2016-02-08 06:51:45	2016-02-08 12:51:45	41.06213	-81.53784	41.06217	-81.53547	0.123	I-77 N	R	Akron	Summit	OH	US/Eastern	2016-02-08 06:54:00	39.0	55.0	29.65	10.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
1516059	A-4239402	2	2019-08-23 18:03:25	2019-08-23 18:32:01	34.00248	-117.37936	33.99888	-117.37094	0.543	Pomona Fwy E	R	Riverside	Riverside	CA	US/Pacific	2019-08-23 17:53:00	86.0	40.0	28.92	10.0	
1516060	A-4239403	2	2019-08-23 19:11:30	2019-08-23 19:38:23	32.76696	-117.14806	32.76555	-117.15363	0.338	I-8 W	R	San Diego	San Diego	CA	US/Pacific	2019-08-23 18:53:00	70.0	73.0	29.39	10.0	
1516061	A-4239404	2	2019-08-23 19:00:21	2019-08-23 19:28:49	33.77545	-117.84779	33.77740	-117.85727	0.561	Garden Grove Fwy	R	Orange	Orange	CA	US/Pacific	2019-08-23 18:53:00	73.0	64.0	29.74	10.0	
1516062	A-4239405	2	2019-08-23 19:00:21	2019-08-23 19:29:42	33.99246	-118.40302	33.98311	-118.39565	0.772	San Diego Fwy S	R	Culver City	Los Angeles	CA	US/Pacific	2019-08-23 18:51:00	71.0	81.0	29.62	10.0	
1516063	A-4239406	2	2019-08-23 18:52:06	2019-08-23 19:21:31	34.13393	-117.23092	34.13736	-117.23934	0.537	CA-210 W	R	Highland	San Bernardino	CA	US/Pacific	2019-08-23 20:50:00	79.0	47.0	28.63	7.0	

Wind_Direction	Wind_Speed(mph)	Weather_Condition	Amenity	Bump	Crossing	Give_Way	Junction	No_Exit	Railway	Roundabout	Station	Stop	Traffic_Calming	Traffic_Signal	Turning_Loop	Day	Hour	Year	Weekday	Time_Diff
SW	10.4	Light Rain	False	False	False	False	False	False	False	False	False	False	False	False	False	8	0	2016	0	360.0
CALM	7.0	Light Rain	False	False	False	False	False	False	False	False	False	False	False	False	False	8	5	2016	0	360.0
CALM	7.0	Overcast	False	False	False	False	True	False	False	False	False	False	False	False	False	8	6	2016	0	360.0
CALM	7.0	Overcast	False	False	False	False	True	False	False	False	False	False	False	False	False	8	6	2016	0	360.0
CALM	7.0	Overcast	False	False	False	False	False	False	False	False	False	False	False	False	False	8	6	2016	0	360.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
W	13.0	Fair	False	False	False	False	False	False	False	False	False	False	False	False	False	23	18	2019	4	28.0
SW	6.0	Fair	False	False	False	False	False	False	False	False	False	False	False	False	False	23	19	2019	4	26.0
SSW	10.0	Partly Cloudy	False	False	False	False	True	False	False	False	False	False	False	False	False	23	19	2019	4	28.0
SW	8.0	Fair	False	False	False	False	False	False	False	False	False	False	False	False	False	23	19	2019	4	29.0
SW	7.0	Fair	False	False	False	False	False	False	False	False	False	False	False	False	False	23	18	2019	4	29.0

4. EXPLORATORY DATA ANALYSIS

There are different questions we formulate to analyze our dataset and showcase how each factor is contributing towards the number of accidents and its severity.

4.1. Analyzing various factors with number of accidents and their severity

4.1.1. Analyzing State and City with number of accidents

Below are the various observations we concluded from the graphs shown from Fig1. – Fig.

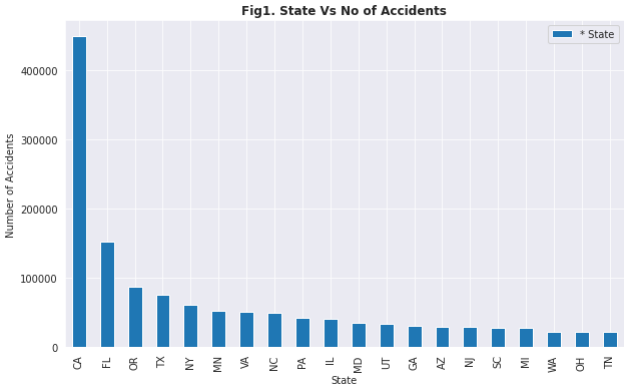
a) The highest number of accidents were recorded in the state of California.

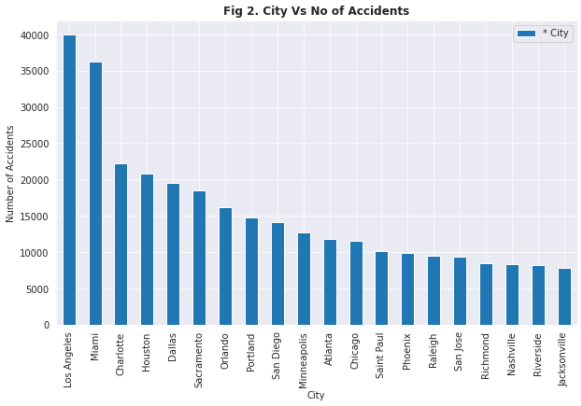
b) There is a lot of presence of cities from California (LA, Sacramento, San Diego, Riverside, Jacksonville), followed by Texas (Houston, Dallas, Austin). This is the same result suggested by the States plot which has the top contributing states: California, Florida, Texas, and North Carolina.

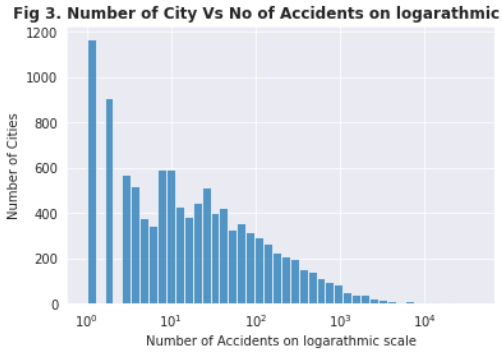
c) 2.35% of the total number of cities have recorded an accident number greater than 1000.

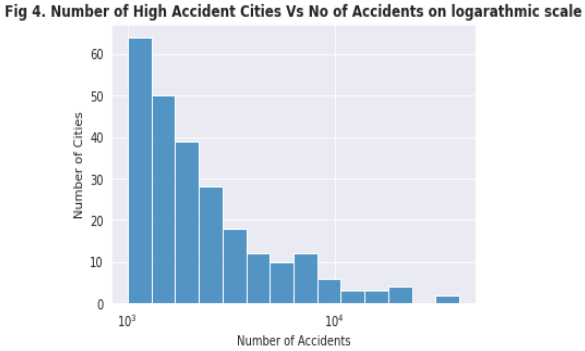
d) Less than 70 cities reported more than 1000 accidents during the period.

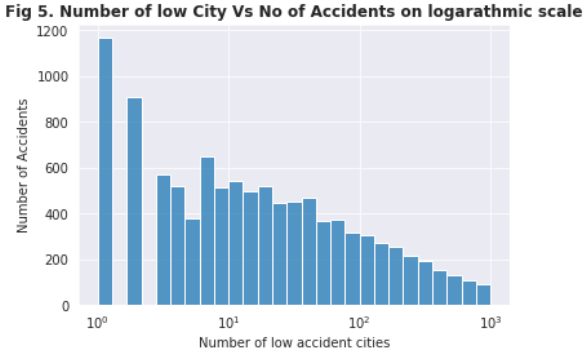
e) Over 1000 cities have reported just one accident in four years.





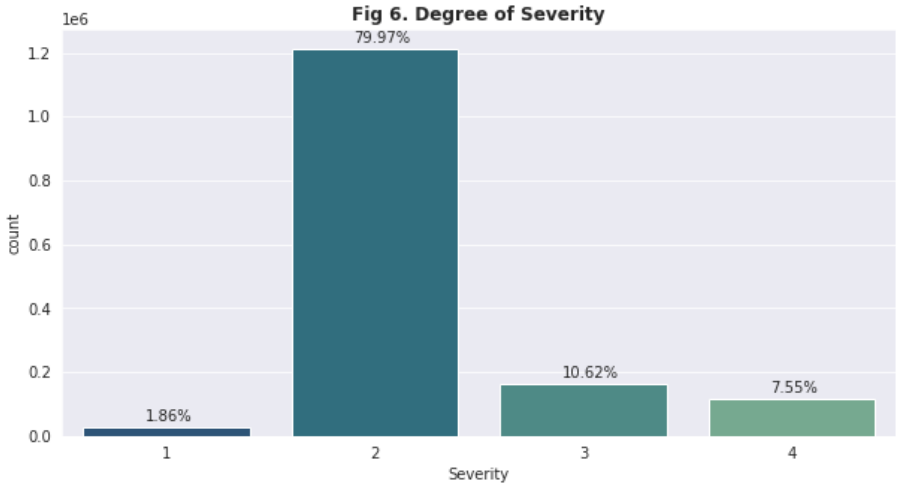






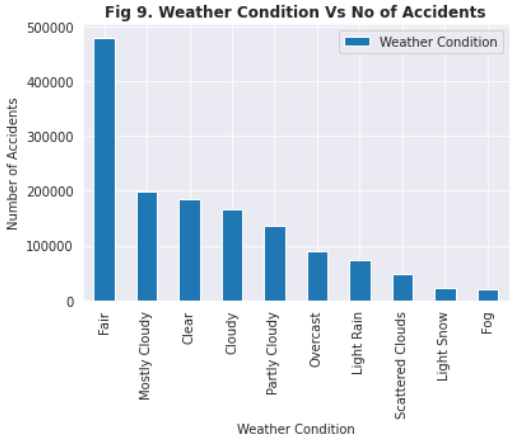
4.1.2. Analyzing level of severity that has been recorded the most

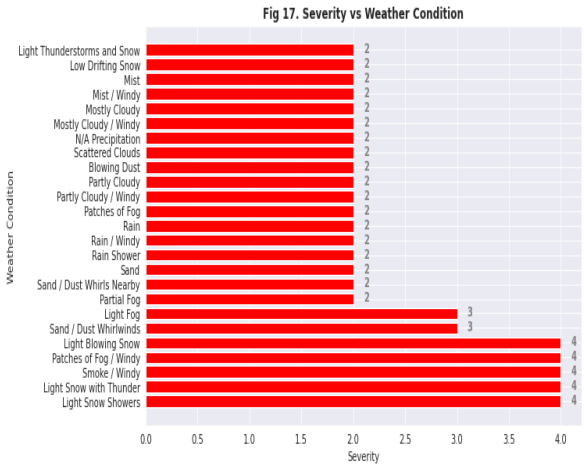
Below graph suggests that most of the accidents had severity of 2(average) followed by severity 3(above average). There are very few accidents with very low severity(0 and 1).



4.1.3. Analyzing weather condition and number and severity of accident

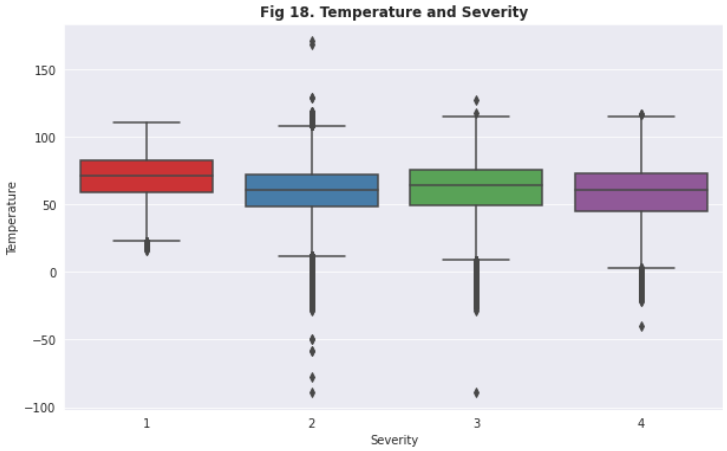
The graphs suggest that the weather condition for most of the accidents was clear, followed by overcast and cloudy. Overcast and cloudy are reasonable factors for accidents unlike clear. But the weather conditions like Snow, smoke and windy has seen high severe accidents. This is logically correct given such extreme weather conditions are likely to have caused severe accidents.





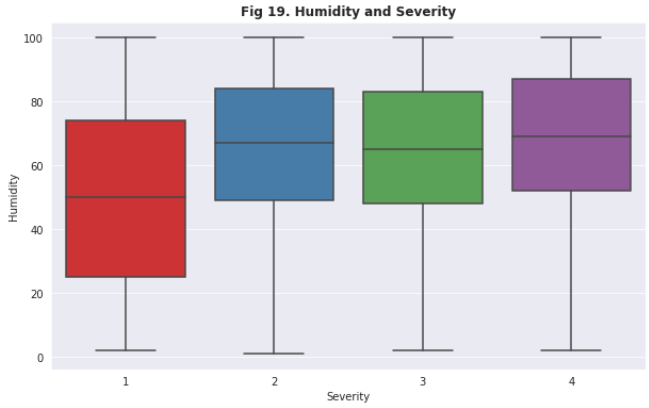
4.1.4. Analyzing temperature and severity of accident

From graphs plotted, we can see that there is almost no difference in the median temperature in Severity 1,2 and 3, while lower median temperature in severity 4, which might indicate that lower temperature might result in more severe accidents.



4.1.5. Analyzing humidity and severity of accident

From the graphs below, we can see that higher humidity might lead to more severe accidents.



4.1.6.. Analyzing yearly trend of number of accidents and changes in severity

Below figures suggest that the number of accidents has increased from 2016 to 2020. Every year, we have seen a drop in accidents in the months of August and September. This could

possibly be due to summer break and then the new school term around August-September. We further observed, Severity 2 accidents are increasing year over year at a rapid speed. Severity 3 accidents have seen a decrease in 2019 compared to 2018. Severity 1 and 4 are relatively flat year over year. More severity 1 accidents were seen in 2020 than other years.

Fig 14. No. of Accidents in Year

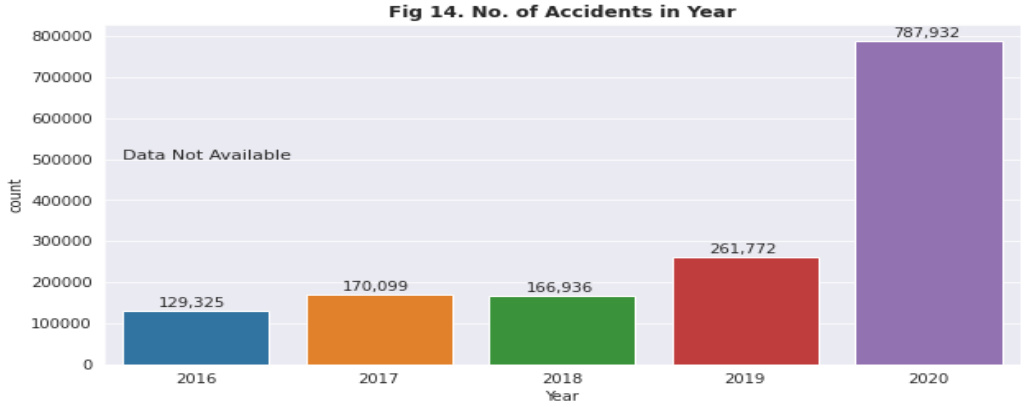


Fig 16. Plot of the data by Severity and by Year to show the trend of accidents by year and by accident severity

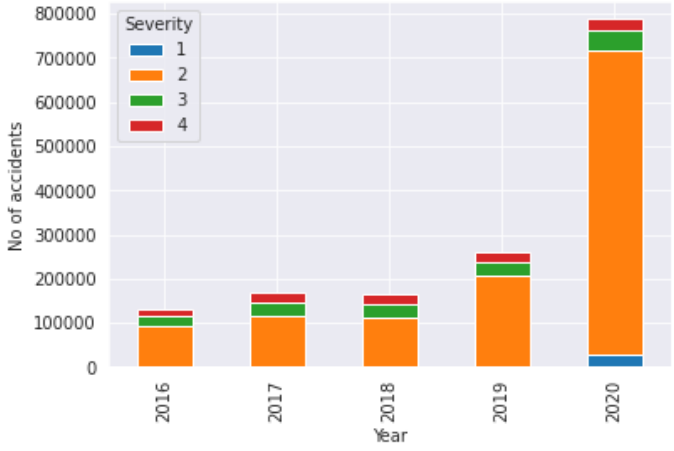
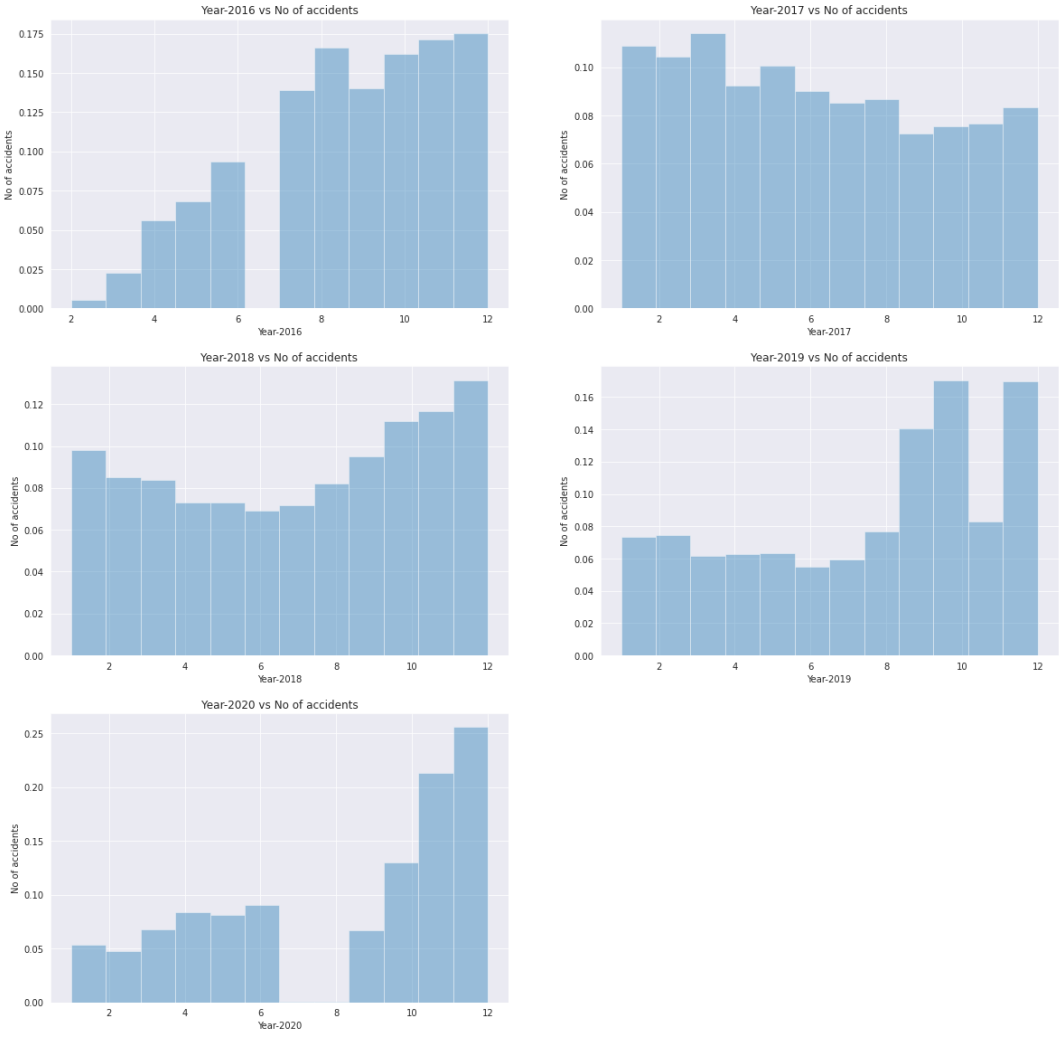
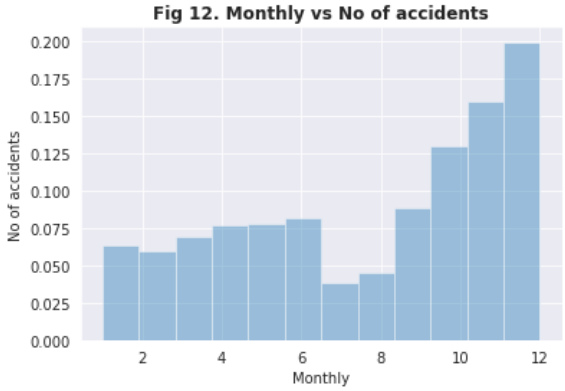


Fig 13. Year Vs No of accidents



4.1.7. Analyzing monthly trend of number of accidents

Accidents increase over a period of time in a year. Monthly trends have been seen to go up as we approach from Jan to Dec.



4.1.8. Analyzing weekly trend of number of accidents

Most of the accidents are seen on weekdays rather than weekends. While Weekdays follow the general 24-hour pattern- two peaks coincide with work rush hours; the distribution for weekends sees a peak between 10 am and 2 pm. This could be due to people going out for leisurely activities on Sundays.

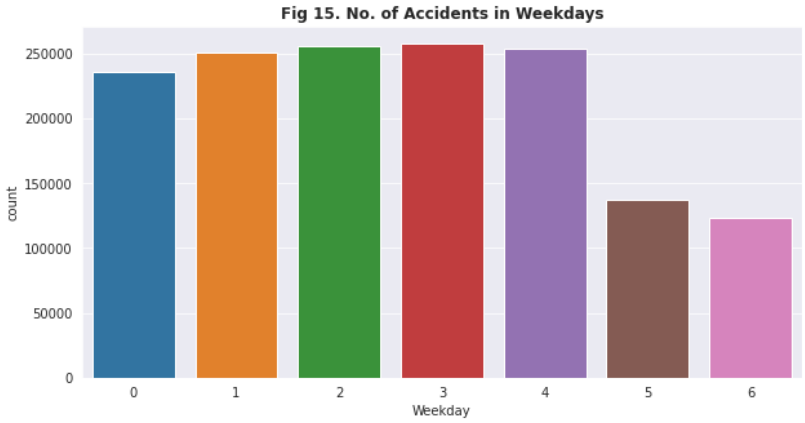
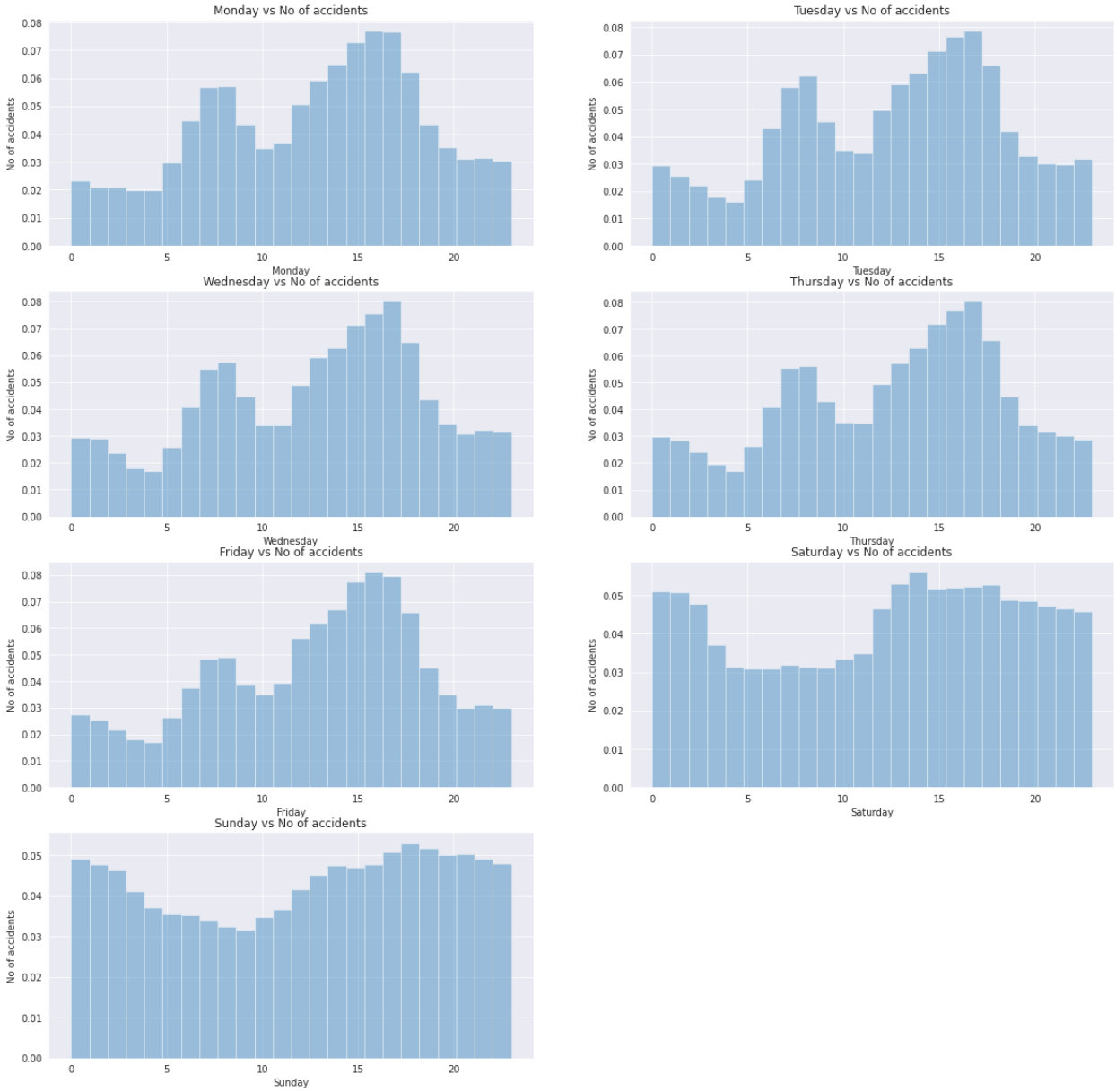
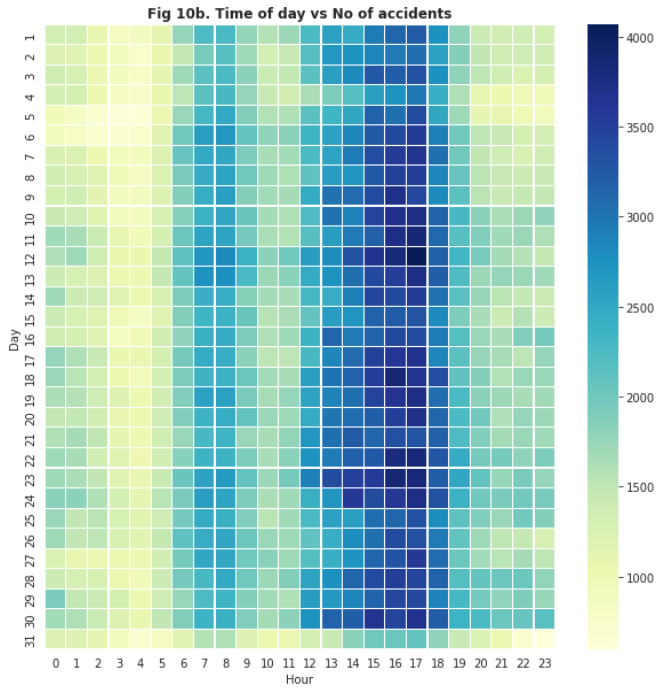


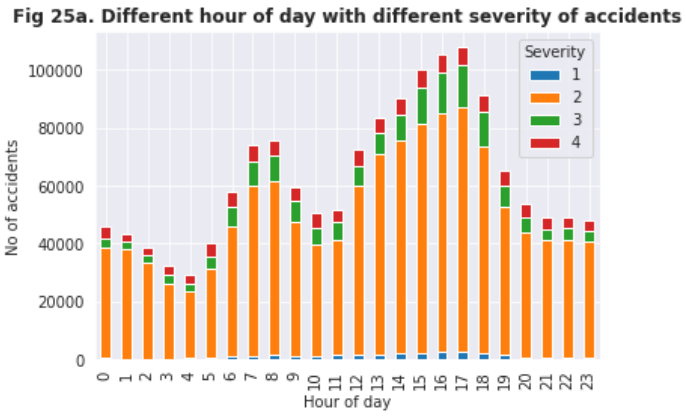
Fig 11. Day of Week vs No of accident



4.1.9. Analyzing hourly trend of number of accidents and severity

There are two peaks of time- one in the morning between 6 am to 9 am and another between 3 pm and 6 pm. This is consistent with the assumption that rush hours in the morning and evening could lead to more accidents. It is further seen that most of the accidents that have occured during night are of severity 2. Severity 1 and 2 has same percentage of accidents between day and night while 3 and 4 has more accidents percentage at nights





4.1.10. Analyzing Timezone and number of accidents and severity of accidents

Eastern followed by Pacific time zones have seen a higher number of accidents compared to others. Most severe 2 accidents were seen in all timezones. But Eastern time zone has seen high severe (severity -4) accidents more than any other timezone.

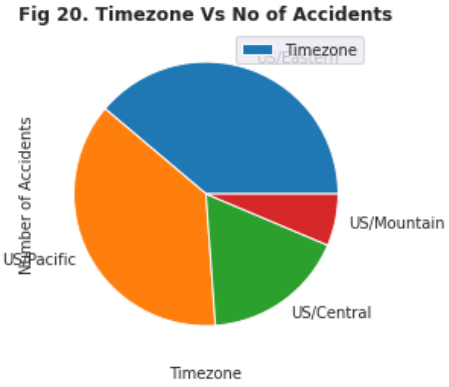


Fig 20. Timezone Vs No of Accidents

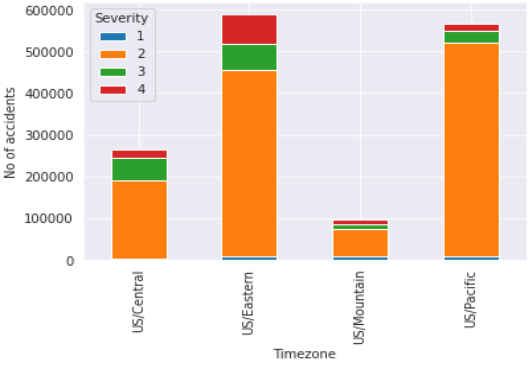
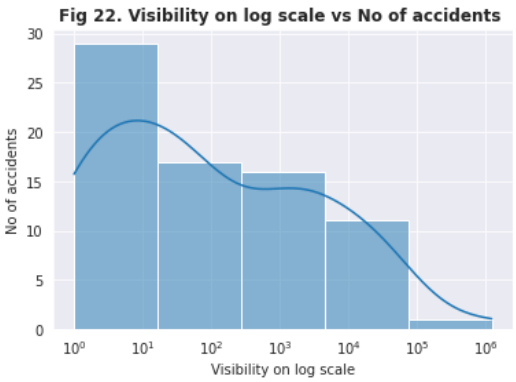
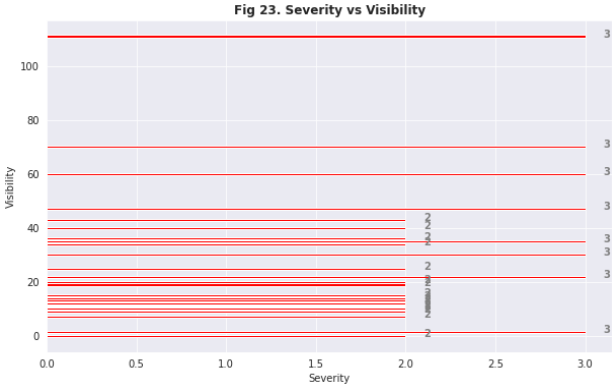


Fig 21b. Different timezones with different severity of accidents

4.1.11. Analyzing Visibility and number of accidents and severity of accidents

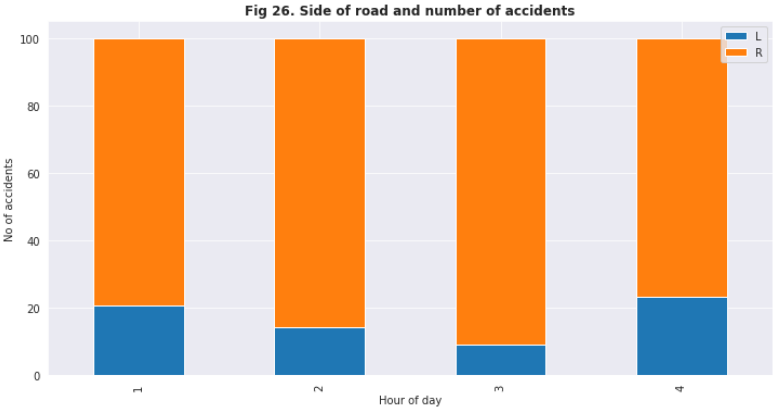
Average visibility range that has seen the most accidents came out to be 9.15 miles. Maximum number of accidents were seen with visibility between 1 to 10 miles. Ideally, 10 mile visibility is something which practically allows a person to see clearly in daytime. So as per data, visibility is not providing ample reasoning for the number of accidents as most of the accidents were seen in good visibility range. But data do suggest that when visibility range is more than 10 miles, majority accidents recorded were of higher severity(sev = 3). Which is logically correct as when visibility is above 10 miles that can cause high severe accidents.

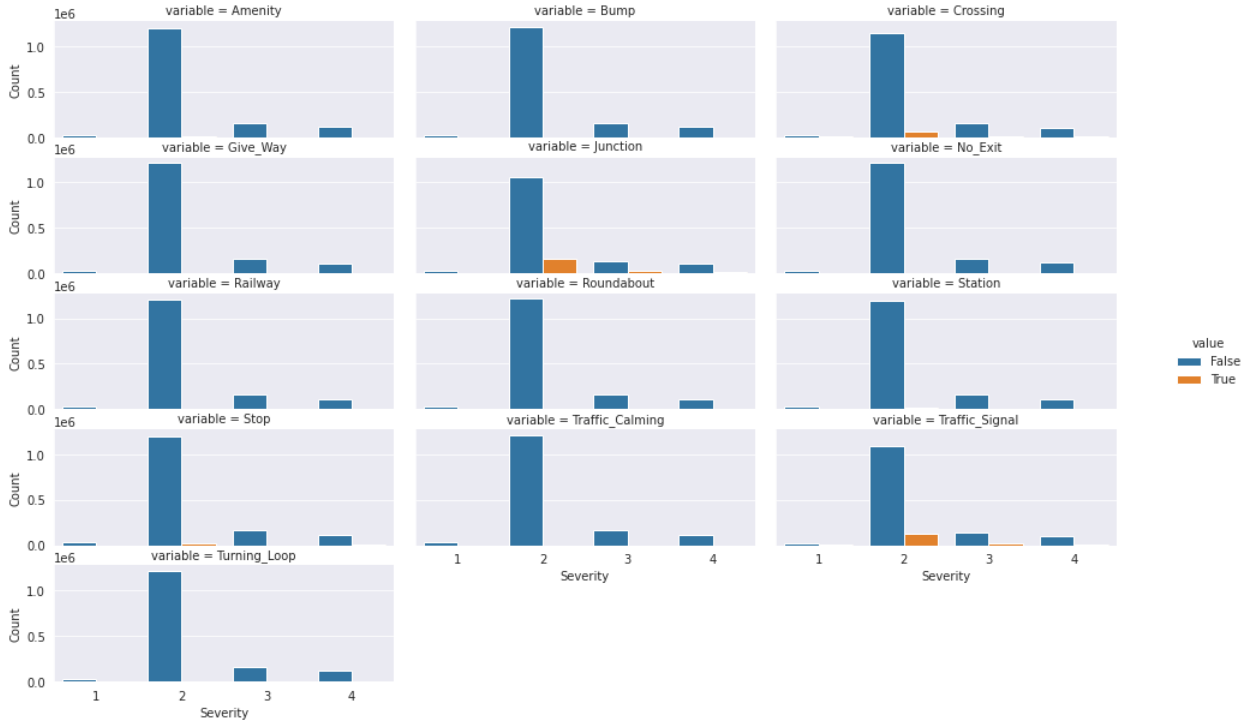




4.1.12. Analyzing Road Conditions and number of accidents

Right side of the road has seen most accidents. Crossing, Junction, Traffic Signal have some impact on the number of accidents.





4.1.13. Analyzing Severity with Distance and duration of accident

It is seen that more severe accidents will affect longer distances and last longer time.

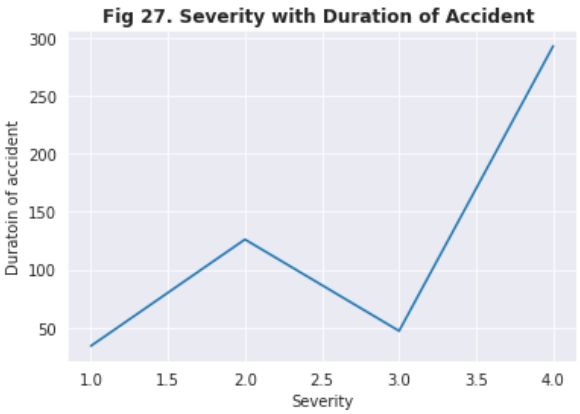


Fig 27. Severity with Duration of Accident

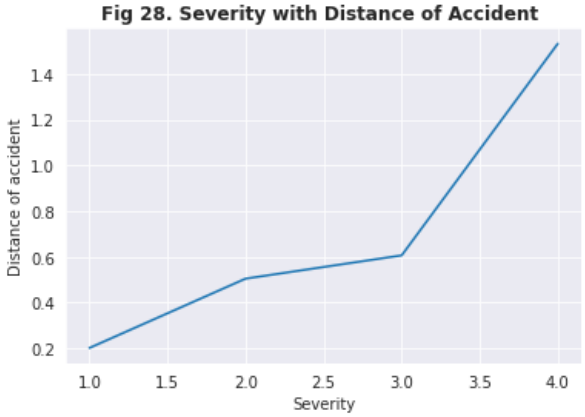


Fig 28. Severity with Distance of Accident

4.1.14. Analyzing in each state , which county has recorded the highest number of accidents

The following table highlights the county which has record highest number of accidents in each state.

	State	County
0	AL	Jefferson
1	AR	Pulaski
2	AZ	Maricopa
3	CA	Los Angeles
4	CO	Denver
5	CT	Hartford
6	DC	District of Columbia
7	DE	New Castle
8	FL	Miami-Dade
9	GA	Fulton
10	IA	Polk
11	ID	Ada
12	IL	Cook
13	IN	Marion
14	KS	Johnson
15	KY	Jefferson
16	LA	East Baton Rouge
17	MA	Middlesex
18	MD	Prince George's
19	ME	Cumberland
20	MI	Wayne
21	MN	Hennepin
22	MO	Jackson

4.2. Methods used to visualize the data

- Method 1: We have used the .value_counts() to first count the values of each unique value in the column.
- Method 2: We can group the data by a particular column, count total rows corresponding to each group and sort them in descending order using .groupby() , .count() , .sort_values(). This is useful when we must find values in a column with high occurring or low occurring accident.
- Method 3: We have used methods to convert the timestamp field in date-time format using pd.to_datetime() and then fetched the hourly value of each row of the data by .dt.hour.
- Method 4: To understand and visualize the data on a daily, monthly, and yearly manner we have adopted the in-built functions mentioned below :

- To pull the day of the week from the timestamp format, we have used function [.dt.dayofweek == n], where n =0,1,2,3,4,5,6 for each day of the week
- To pull the month of the year from the timestamp format we can use function like .dt.month
- To pull the year from timestamp format we can use function [.dt.year == n], with n = 2016, 2018, etc.
- Method 5: We have grouped the data by 'column1' and 'column2' using groupby(). We have further calculated the number of accidents in each 'column' grouped in 'column1' by using .size(). Then again grouped by 'column1' using groupby(level=0) to deal with multilevel index and then used .idxmax() to get 'column2' with maximum number of accidents in each 'column1' .
    - We have visualized the data with respect to 2 different columns with the help of the groupby() method and have used additional functions such as
    - .count()/.size()- to get the count of a particular column after grouping the data on another column.
    - .idxmax()- to get the highest value of one column while grouping the data on another column.
    - .apply() - to pass columns as series to the function
- Method 6: We have also used the .mean(), .median() functions to get the data for certain columns after appropriate grouping of the data.

4.3. Types of Plot used to visualize the data

We have used both the matplotlib and the seaborn libraries to get appropriate graphs based on the visualizations required for our dataset.

Graphs used from the Matplotlib library:

- Bar graph: We have used both the horizontal bar graph(.barh()) and the traditional bar graph(.plot(kind='bar')) wherever required.
- Pie chart: We have used the pie chart (.plot(kind='pie')) for columns that have Categorical values to understand what percentage of the data was recorded under those unique values in that column.
- Scatter plot: We have used the scatterplot (.scatter()) to understand the trend of the data from left to right related to two columns to depict the nature of their relationship – Positive or Negative.
- Line Plot: We have used line graphs for depicting x and y variables.

Graphs used from the Seaborn library:

- Histplot: We have used the histogram plot (.histplot()) to interpret the nature of the graph – Symmetric, Skewed Right or Skewed Left.
- Distplot: We have used the distribution plot(.distplot()) to depict the variation of the overall data distribution.
- Countplot
- Heatmap: We have used heatmap to depict variation in the number of accidents throughout the day.
- Boxplot

- Catplot: We have used this plot to see all boolean columns in one figure and depict which one is impacting the number of accidents.

5. DATA MODELLING

5.1. Preparing Data for Modelling

- We have assumed that the Severity of the accident to be the target variable as the parameters in the dataset affect the severity directly.
- In the whole dataset, we take only those columns that directly affect the level of severity of a given accident.
- The columns that have been taken into consideration are: 'County', 'City', 'State' ,'Timezone', 'Temperature(F)', 'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Direction', 'Wind_Speed(mph)', 'Weather_Condition', 'Severity'.
- Now, we will perform label encoding on the columns that contain categorical values to convert them into integers which will help us in finding the best fit algorithm to predict the severity of each accident

5.2. Splitting our data into Train and test

We will split our data into two sets -

- Training set : The data that will be used to train our model and understand the parameters.
- Testing set: The data on which we will use our model to test the performance of our model.

- We will be using the train_test_split function from the sklearn.model_selection library to help us in splitting in data.
- We will split 70% of the data as our training data and 30% of the data as the testing data

5.3 Standardizing

- We have observed that the data contains columns with values under 10 and also columns that have values greater than 4500. So, we will be standardizing all the values with the help of the function StandardScaler from the sklearn.preprocessing library
- After standardizing the values, we will now fit the Linear regression model from the sklearn library on our dataset
- After fitting the model, we will check the accuracy of our model by comparing the values that have been generated by our model and the values in our dataset with the help of the function .score()

5.4 Model Fitting

- We have decided to use the Linear Regression Model from the sklearn library. As severity is the target variable in this dataset, we will be predicting the severity of a particular accident.
- After fitting the Linear Regression Model, we have come to the conclusion that the values of the coefficients are below 0 and the accuracy is just 4.13%. So we can say that the Linear Regression Model is not the best fit algorithm for this dataset.

	COEFFICIENTS
Intercept	1.6624911
County	-5.737125
City	-3.623324
State	8.642032
Timezone	-7.746209
Temperature (F)	-1.132101
Humidity (%)	2.221175
Pressure (in)	2.279186
Visibility	2.7136642
Wind Direction	-4.653437
Wind Speed (mph)	2.629752
Weather Condition	1.031194

- To our curiosity, we have also tried to implement the Random Forest Regression Model from the sklearn library to understand the scope of finding the better algorithm for this dataset.
- After calculating the accuracy score for this model, we have seen that 32.43% of the test data were predicted accurately. So we can say that out of the Linear Regression Model and the Random Forest Regression Model, the latter has yielded a better accuracy score and we have learnt that each dataset is different and we have to find the best fit algorithm after clearly understanding the dataset and the functionality of the model.

6. CONCLUSION:

- By understanding the data in the project, by applying different methods and functions in python to evaluate various aspects, and by deriving insights from meaningful graphs, we have been able to conclude that predicting the severity of any given accident requires a lot of parameters to be taken into consideration which have been inferred from the data cleaning and visualization processes . Some of them being:
- Visibility(mi) - whether the visibility falls under critical range.
- Weather_condition – whether the condition is fair or cloudy or snowy.
- Timezone & County – what is the trend of accidents in a particular time zone in the given county and many more.
- We will predict the severity of a particular accident once we have the required data using the best-fit algorithm according to our analysis which is the Random Forest Regressor.

7. FUTURE SCOPE

- We were able to successfully implement our findings and models to the data set for the number of accidents in the United States of America. Like we have concluded, we were able to find the related factors and draw insightful inferences.
- As a part of future scope of our project, we can consider other models and methods such as feature scaling, correlation matrix, one hot encoding to generate more meaningful insights.
- In future, we can also implement similar analyses on different countries of the world. We can use similar data for different countries and draw conclusions based on the

demographics of that region. These conclusions would be of help to the concerned authorities, who would be able to take the right measures and save many lives across the globe.

8. CODE

Below is the link of the code:

https://colab.research.google.com/drive/1tznB5Bg-HayX1iXCjrLgDW3h3fVSiJMl?usp=sharing

9. REFERENCES:

- https://www.kaggle.com/sobhanmoosavi/us-accidents
- https://melbournedrivinglesson.blogspot.com/2020/04/complete-your-goal-of-driving-by.html
- http://wilmingtonjournal.com/heroic-police-officer-risks-life-to-save-injured-young-driver-who-rolled-off-road/