Olufemi Babalola

Practicum II

Topic: Data Visualization and Analysis of US Accidents

Overview:

In the USA, over 37,000 people die in road crashes each year, and 2.35 million are injured or disabled. Road crashes cost the U.S. $230.6 billion per year or an average of $820 per person. Road crashes are the single greatest annual cause of death of healthy U.S. citizens travelling abroad.

There are many factors responsible for road accidents in the United States. This study attempts to visualize and analyze features of US Accidents records with the aim of revealing valuable insight.

The insights reveal:

- States and Cities reporting highest record of accidents
- The severity of these occurrences.
- The duration of each accidents
- The Impact of weather conditions and other traffic objects

The dataset used was download from Kaggle (https://www.kaggle.com/sobhanmoosavi/us-accidents).

The US Accidents data is a countrywide traffic accident dataset, which covers 49 states of the United States. The data is continuously being collected from February 2016 to March 2019 and consists of 2.974 million rows with 49 variables.

Feature description of the dataset

S.No.	Attribute	Description
1	ID	This is a unique identifier of the accident record.
2	Source	Indicates source of the accident report (i.e. the API which reported the accident.).
3	TMC	A traffic accident may have a Traffic Message Channel (TMC) code which provides more detailed description of the event.
4	Severity	Shows the severity of the accident, a number between 1 and 4, where 1 indicates the least impact on traffic (i.e., short delay as a result of the accident) and 4 indicates a significant impact on traffic (i.e., long delay).
5	Start_Time	Shows start time of the accident in local time zone.
6	End_Time	Shows end time of the accident in local time zone.
7	Start_Lat	Shows latitude in GPS coordinate of the start point.
8	Start_Lng	Shows longitude in GPS coordinate of the start point.
9	End_Lat	Shows latitude in GPS coordinate of the end point.
10	End_Lng	Shows longitude in GPS coordinate of the end point.
11	Distance(mi)	The length of the road extent affected by the accident.
12	Description	Shows natural language description of the accident.
13	Number	Shows the street number in address field.
14	Street	Shows the street name in address field.
15	Side	Shows the relative side of the street (Right/Left) in address field.
16	City	Shows the city in address field.
17	County	Shows the county in address field.
18	State	Shows the state in address field.
19	Zipcode	Shows the zipcode in address field.
20	Country	Shows the country in address field.
21	Timezone	Shows timezone based on the location of the accident (eastern, central, etc.).
22	Airport_Code	Denotes an airport-based weather station which is the closest one to location of the accident.
23	Weather_Timestamp	Shows the time-stamp of weather observation record (in local time).
24	Temperature(F)	Shows the temperature (in Fahrenheit).
25	Wind_Chill(F)	Shows the wind chill (in Fahrenheit).
26	Humidity(%)	Shows the humidity (in percentage).
27	Pressure(in)	Shows the air pressure (in inches).
28	Visibility(mi)	Shows visibility (in miles).
29	Wind_Direction	Shows wind direction.
30	Wind_Speed(mph)	Shows wind speed (in miles per hour).
31	Precipitation(in)	Shows precipitation amount in inches, if there is any.
32	Weather_Condition	Shows the weather condition (rain, snow, thunderstorm, fog, etc.)
33	Amenity	A POI annotation which indicates presence of amenity in a nearby location.
34	Bump	A POI annotation which indicates presence of speed bump or hump in a nearby location.
35	Crossing	A POI annotation which indicates presence of crossing in a nearby location.
36	Give_Way	A POI annotation which indicates presence of give_way in a nearby location.
37	Junction	A POI annotation which indicates presence of junction in a nearby location.
38	No_Exit	A POI annotation which indicates presence of no_exit in a nearby location.
39	Railway	A POI annotation which indicates presence of railway in a nearby location.
40	Roundabout	A POI annotation which indicates presence of roundabout in a nearby location.
41	Station	A POI annotation which indicates presence of station in a nearby location.
42	Stop	A POI annotation which indicates presence of stop in a nearby location.
43	Traffic_Calming	A POI annotation which indicates presence of traffic_calming in a nearby location.
44	Traffic_Signal	A POI annotation which indicates presence of traffic_signal in a nearby location.
45	Turning_Loop	A POI annotation which indicates presence of turning_loop in a nearby location.
46	Sunrise_Sunset	Shows the period of day (i.e. day or night) based on sunrise/sunset.
47	Civil_Twilight	Shows the period of day (i.e. day or night) based on civil twilight.
48	Nautical_Twilight	Shows the period of day (i.e. day or night) based on nautical twilight.
49	Astronomical_Twili	Shows the period of day (i.e. day or night) based on astronomical twilight.

Data Exploration, Visualization and Analysis

Installing r packages and loading libraries

```
> # installing required packages
>
> install.packages("lubridate")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/lubridate_1.7.8.zi
p'
Content type 'application/zip' length 1591310 bytes (1.5 MB)
downloaded 1.5 MB

package 'lubridate' successfully unpacked and MD5 sums checked
Error in install.packages : ERROR: failed to lock directory 'C:\Users\lenovo\Doc
uments\R\win-library\3.6' for modifying
Try removing 'C:\Users\lenovo\Documents\R\win-library\3.6/00LOCK'
> install.packages("corrplot")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/corrplot_0.84.zip'
Content type 'application/zip' length 5450693 bytes (5.2 MB)
downloaded 5.2 MB

package 'corrplot' successfully unpacked and MD5 sums checked
Error in install.packages : ERROR: failed to lock directory 'C:\Users\lenovo\Doc
uments\R\win-library\3.6' for modifying
Try removing 'C:\Users\lenovo\Documents\R\win-library\3.6/00LOCK'
> install.packages("data.table")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:



> install.packages("tidyverse")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/tidyverse_1.3.0.zi
p'
Content type 'application/zip' length 440141 bytes (429 KB)
downloaded 429 KB

package 'tidyverse' successfully unpacked and MD5 sums checked
Error in install.packages : ERROR: failed to lock directory 'C:\Users\lenovo\Doc
uments\R\win-library\3.6' for modifying
Try removing 'C:\Users\lenovo\Documents\R\win-library\3.6/00LOCK'
> install.packages("ggmap")
WARNING: Rtools is required to build R packages but is not currently installed.
 Please download and install the appropriate version of Rtools before proceedin
g:
```

```
> # load tidyverse package into library
> library(tidyverse)
-- Attaching packages --------------------------------------- tidyverse 1.3.0 -
-
v ggplot2 3.3.0     v purrr   0.3.3
v tibble  3.0.0     v dplyr   0.8.5
v tidyr   1.0.2     v stringr 1.4.0
v readr   1.3.1     v forcats 0.5.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() -
-
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
>
```

Downloading the dataset

```
> # import the US_Accidents dataset into rstudio
> US_Accidents <- read.csv("~/US_Accidents.csv")
> View(US_Accidents)
```

	ID	Source	TMC	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	End_Lat	End_Lng	Distance.mi.
1	A-1	MapQuest	201	3	2016-02-08 05:46:00	2016-02-08 11:00:00	39.86515	-84.05872	NA	NA	0.01
2	A-2	MapQuest	201	2	2016-02-08 06:07:59	2016-02-08 06:37:59	39.92806	-82.83118	NA	NA	0.01
3	A-3	MapQuest	201	2	2016-02-08 06:49:27	2016-02-08 07:19:27	39.06315	-84.03261	NA	NA	0.01
4	A-4	MapQuest	201	3	2016-02-08 07:23:34	2016-02-08 07:53:34	39.74775	-84.20558	NA	NA	0.01
5	A-5	MapQuest	201	2	2016-02-08 07:39:07	2016-02-08 08:09:07	39.62778	-84.18835	NA	NA	0.01
6	A-6	MapQuest	201	3	2016-02-08 07:44:26	2016-02-08 08:14:26	40.10059	-82.92519	NA	NA	0.01
7	A-7	MapQuest	201	2	2016-02-08 07:59:35	2016-02-08 08:29:35	39.75827	-84.23051	NA	NA	0.00
8	A-8	MapQuest	201	3	2016-02-08 07:59:58	2016-02-08 08:29:58	39.77038	-84.19490	NA	NA	0.01
9	A-9	MapQuest	201	2	2016-02-08 08:00:40	2016-02-08 08:30:40	39.77806	-84.17200	NA	NA	0.00
10	A-10	MapQuest	201	3	2016-02-08 08:10:04	2016-02-08 08:40:04	40.10059	-82.92519	NA	NA	0.01

Showing 1 to 12 of 2,974,335 entries, 49 total columns

Checking for missing values and Cleaning the dataset

```
> ### exploring and cleaning the dataset ####
>
> # checking the class of the dataset
> class(US_Accidents)
[1] "data.frame"
```

```
> # check the Structure of this dataset
> glimpse(US_Accidents)
Rows: 2,974,335
Columns: 49
$ ID                    <fct> A-1, A-2, A-3, A-4, A-5, A-6, A-7, A-8, A-...
$ Source                <fct> MapQuest, MapQuest, MapQuest, MapQuest, Ma...
$ TMC                   <dbl> 201, 201, 201, 201, 201, 201, 201, 201, 20...
$ Severity              <int> 3, 2, 2, 3, 2, 3, 2, 3, 2, 3, 3, 3, 2, 2, ...
$ Start_Time            <fct> 2016-02-08 05:46:00, 2016-02-08 06:07:59, ...
$ End_Time              <fct> 2016-02-08 11:00:00, 2016-02-08 06:37:59, ...
$ Start_Lat             <dbl> 39.86515, 39.92806, 39.06315, 39.74775, 39...
$ Start_Lng             <dbl> -84.05872, -82.83118, -84.03261, -84.20558...
$ End_Lat               <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ End_Lng               <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
$ Distance.mi.          <dbl> 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.00, ...
$ Description           <fct> Right lane blocked due to accident on I-70...
$ Number                <dbl> NA, 2584, NA, NA, NA, NA, 376, NA, 99, NA,...
$ Street                <fct> I-70 E, Brice Rd, State Route 32, I-75 S, ...
$ Side                  <fct> R, L, R, R, R, R, R, R, L, R, R, R, R, L, ...
$ City                  <fct> Dayton, Reynoldsburg, Williamsburg, Dayton...
$ County                <fct> Montgomery, Franklin, Clermont, Montgomery...
$ State                 <fct> OH, OH, OH, OH, OH, OH, OH, OH, OH, OH, OH...
$ Zipcode               <fct> 45424, 43068-3402, 45176, 45417, 45459, 43...
$ Country               <fct> US, US, US, US, US, US, US, US, US, US, US...
$ Timezone              <fct> US/Eastern, US/Eastern, US/Eastern, US/Eas...
$ Airport_Code          <fct> KFFO, KCMH, KI69, KDAY, KMGY, KCMH, KDAY, ...
$ Weather_Timestamp     <fct> 2016-02-08 05:58:00, 2016-02-08 05:51:00, ...

$ Temperature.F.        <dbl> 36.9, 37.9, 36.0, 35.1, 36.0, 37.9, 34.0, ...
$ Wind_Chill.F.         <dbl> NA, NA, 33.3, 31.0, 33.3, 35.5, 31.0, 31.0...
$ Humidity...           <dbl> 91, 100, 100, 96, 89, 97, 100, 100, 99, 10...
$ Pressure.in.          <dbl> 29.68, 29.65, 29.67, 29.64, 29.65, 29.63, ...
$ Visibility.mi.        <dbl> 10, 10, 10, 9, 6, 7, 7, 7, 5, 3, 5, 3, 3, ...
$ Wind_Direction        <fct> Calm, Calm, SW, SW, SW, SSW, WSW, WSW, SW,...
$ Wind_Speed.mph.       <dbl> NA, NA, 3.5, 4.6, 3.5, 3.5, 3.5, 3.5, 1.2,...
$ Precipitation.in.     <dbl> 0.02, 0.00, NA, NA, NA, 0.03, NA, NA, NA, ...
$ Weather_Condition     <fct> Light Rain, Light Rain, Overcast, Mostly C...
$ Amenity               <fct> False, False, False, False, False, False, ...
$ Bump                  <fct> False, False, False, False, False, False, ...
$ Crossing              <fct> False, False, False, False, False, False, ...
$ Give_Way              <fct> False, False, False, False, False, False, ...
$ Junction              <fct> False, False, False, False, False, False, ...
$ No_Exit               <fct> False, False, False, False, False, False, ...
$ Railway               <fct> False, False, False, False, False, False, ...
$ Roundabout            <fct> False, False, False, False, False, False, ...
$ Station               <fct> False, False, False, False, False, False, ...
$ Stop                  <fct> False, False, False, False, False, False, ...
$ Traffic_Calming       <fct> False, False, False, False, False, False, ...
$ Traffic_Signal        <fct> False, False, True, False, True, False, Fa...
$ Turning_Loop          <fct> False, False, False, False, False, False, ...
$ Sunrise_Sunset        <fct> Night, Night, Night, Night, Day, Day, Day,...
$ Civil_Twilight        <fct> Night, Night, Night, Day, Day, Day, Day, D...
$ Nautical_Twilight     <fct> Night, Night, Day, Day, Day, Day, Day, Day...
$ Astronomical_Twilight <fct> Night, Day, Day, Day, Day, Day, Day, Day, ...
```

```
> # check column names in the dataset
> colnames(US_Accidents)
 [1] "ID"                    "Source"                "TMC"
 [4] "Severity"              "Start_Time"            "End_Time"
 [7] "Start_Lat"             "Start_Lng"             "End_Lat"
[10] "End_Lng"               "Distance.mi."          "Description"
[13] "Number"                "Street"                "Side"
[16] "City"                  "County"                "State"
[19] "Zipcode"               "Country"               "Timezone"
[22] "Airport_Code"          "Weather_Timestamp"     "Temperature.F."
[25] "Wind_Chill.F."         "Humidity..."           "Pressure.in."
[28] "Visibility.mi."        "Wind_Direction"        "Wind_Speed.mph."
[31] "Precipitation.in."     "Weather_Condition"     "Amenity"
[34] "Bump"                  "Crossing"              "Give_Way"
[37] "Junction"              "No_Exit"               "Railway"
[40] "Roundabout"            "Station"               "Stop"
[43] "Traffic_Calming"       "Traffic_Signal"        "Turning_Loop"
[46] "Sunrise_Sunset"        "Civil_Twilight"        "Nautical_Twilight"
[49] "Astronomical_Twilight"
>
```

```
> # check for missing values in dataframe
> any(is.na(US_Accidents))
[1] TRUE
>
>
> # listing all columns with missing values in our dataframe
> colnames(US_Accidents)[!complete.cases(t(US_Accidents))]
 [1] "TMC"               "End_Lat"           "End_Lng"
 [4] "Number"            "Temperature.F."    "Wind_Chill.F."
 [7] "Humidity..."       "Pressure.in."      "Visibility.mi."
[10] "Wind_Speed.mph."   "Precipitation.in."
>
>
> # Next, remove all columns containing at least one na values from the dataset
> # and assigned as Accidentsdata
> Accidentsdata <- US_Accidents[, colSums(is.na(US_Accidents)) == 0]
> View(Accidentsdata)
```

	ID	Source	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	Distance.mi.
1	A-1	MapQuest	3	2016-02-08 05:46:00	2016-02-08 11:00:00	39.86515	-84.05872	0.01
2	A-2	MapQuest	2	2016-02-08 06:07:59	2016-02-08 06:37:59	39.92806	-82.83118	0.01
3	A-3	MapQuest	2	2016-02-08 06:49:27	2016-02-08 07:19:27	39.06315	-84.03261	0.01
4	A-4	MapQuest	3	2016-02-08 07:23:34	2016-02-08 07:53:34	39.74775	-84.20558	0.01
5	A-5	MapQuest	2	2016-02-08 07:39:07	2016-02-08 08:09:07	39.62778	-84.18835	0.01
6	A-6	MapQuest	3	2016-02-08 07:44:26	2016-02-08 08:14:26	40.10059	-82.92519	0.01
7	A-7	MapQuest	2	2016-02-08 07:59:35	2016-02-08 08:29:35	39.75827	-84.23051	0.00
8	A-8	MapQuest	3	2016-02-08 07:59:58	2016-02-08 08:29:58	39.77038	-84.19490	0.01
9	A-9	MapQuest	2	2016-02-08 08:00:40	2016-02-08 08:30:40	39.77806	-84.17200	0.00
10	A-10	MapQuest	3	2016-02-08 08:10:04	2016-02-08 08:40:04	40.10059	-82.92519	0.01

Showing 1 to 12 of 2,974,335 entries, 38 total columns

```
> # check the structure of Accidentsdata with glimpse functions
> glimpse(Accidentsdata)
Rows: 2,974,335
Columns: 38
$ ID                    <fct> A-1, A-2, A-3, A-4, A-5, A-6, A-7, A-8, A-...
$ Source                <fct> MapQuest, MapQuest, MapQuest, MapQuest, Ma...
$ Severity              <int> 3, 2, 2, 3, 2, 3, 2, 3, 2, 3, 3, 3, 2, 2, ...
$ Start_Time            <fct> 2016-02-08 05:46:00, 2016-02-08 06:07:59, ...
$ End_Time              <fct> 2016-02-08 11:00:00, 2016-02-08 06:37:59, ...
$ Start_Lat             <dbl> 39.86515, 39.92806, 39.06315, 39.74775, 39...
$ Start_Lng             <dbl> -84.05872, -82.83118, -84.03261, -84.20558...
$ Distance.mi.          <dbl> 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.00, ...
$ Description           <fct> Right lane blocked due to accident on I-70...
$ Street                <fct> I-70 E, Brice Rd, State Route 32, I-75 S, ...
$ Side                  <fct> R, L, R, R, R, R, R, R, L, R, R, R, R, L, ...
$ City                  <fct> Dayton, Reynoldsburg, Williamsburg, Dayton...
$ County                <fct> Montgomery, Franklin, Clermont, Montgomery...
$ State                 <fct> OH, OH, OH, OH, OH, OH, OH, OH, OH, OH, OH...
$ Zipcode               <fct> 45424, 43068-3402, 45176, 45417, 45459, 43...
$ Country               <fct> US, US, US, US, US, US, US, US, US, US, US...
$ Timezone              <fct> US/Eastern, US/Eastern, US/Eastern, US/Eas...
$ Airport_Code          <fct> KFFO, KCMH, KI69, KDAY, KMGY, KCMH, KDAY, ...
$ Weather_Timestamp     <fct> 2016-02-08 05:58:00, 2016-02-08 05:51:00, ...
$ Wind_Direction        <fct> Calm, Calm, SW, SW, SW, SSW, WSW, WSW, SW,...
$ Weather_Condition     <fct> Light Rain, Light Rain, Overcast, Mostly C...
$ Amenity               <fct> False, False, False, False, False, False, ...
$ Bump                  <fct> False, False, False, False, False, False, ...
$ Crossing              <fct> False, False, False, False, False, False, ...
$ Give_Way              <fct> False, False, False, False, False, False, ...
$ Junction              <fct> False, False, False, False, False, False, ...
$ No_Exit               <fct> False, False, False, False, False, False, ...
$ Railway               <fct> False, False, False, False, False, False, ...
$ Roundabout            <fct> False, False, False, False, False, False, ...
$ Station               <fct> False, False, False, False, False, False, ...
$ Stop                  <fct> False, False, False, False, False, False, ...
$ Traffic_Calming       <fct> False, False, False, False, False, False, ...
$ Traffic_Signal        <fct> False, False, True, False, True, False, Fa...
$ Turning_Loop          <fct> False, False, False, False, False, False, ...
$ Sunrise_Sunset        <fct> Night, Night, Night, Night, Day, Day, Day,...
$ Civil_Twilight        <fct> Night, Night, Night, Day, Day, Day, Day, D...
$ Nautical_Twilight     <fct> Night, Night, Day, Day, Day, Day, Day, Day...
$ Astronomical_Twilight <fct> Night, Day, Day, Day, Day, Day, Day, Day, ...
>
>
> # Convert factors in data.frame columns to characters
> i <- sapply(Accidentsdata, is.factor)
> Accidentsdata[i] <- lapply(Accidentsdata[i], as.character)
```

```
> # check the structure of Accidentsdata with glimpse functions
> glimpse(Accidentsdata)
Rows: 2,974,335
Columns: 38
$ ID                    <chr> "A-1", "A-2", "A-3", "A-4", "A-5", "A-6", ...
$ Source                <chr> "MapQuest", "MapQuest", "MapQuest", "MapQu...
$ Severity              <int> 3, 2, 2, 3, 2, 3, 2, 3, 2, 3, 3, 3, 2, 2, ...
$ Start_Time            <chr> "2016-02-08 05:46:00", "2016-02-08 06:07:5...
$ End_Time              <chr> "2016-02-08 11:00:00", "2016-02-08 06:37:5...
$ Start_Lat             <dbl> 39.86515, 39.92806, 39.06315, 39.74775, 39...
$ Start_Lng             <dbl> -84.05872, -82.83118, -84.03261, -84.20558...
$ Distance.mi.          <dbl> 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.00, ...
$ Description           <chr> "Right lane blocked due to accident on I-7...
$ Street                <chr> "I-70 E", "Brice Rd", "State Route 32", "I...
$ Side                  <chr> "R", "L", "R", "R", "R", "R", "R", "R", "L...
$ City                  <chr> "Dayton", "Reynoldsburg", "Williamsburg", ...
$ County                <chr> "Montgomery", "Franklin", "Clermont", "Mon...
$ State                 <chr> "OH", "OH", "OH", "OH", "OH", "OH", "OH", ...
$ Zipcode               <chr> "45424", "43068-3402", "45176", "45417", "...
$ Country               <chr> "US", "US", "US", "US", "US", "US", "US", ...
$ Timezone              <chr> "US/Eastern", "US/Eastern", "US/Eastern", ...
$ Airport_Code          <chr> "KFFO", "KCMH", "KI69", "KDAY", "KMGY", "K...
$ Weather_Timestamp     <chr> "2016-02-08 05:58:00", "2016-02-08 05:51:0...
$ Wind_Direction        <chr> "Calm", "Calm", "SW", "SW", "SW", "SSW", "...
$ Weather_Condition     <chr> "Light Rain", "Light Rain", "Overcast", "M...
$ Amenity               <chr> "False", "False", "False", "False", "False...
$ Bump                  <chr> "False", "False", "False", "False", "False...

$ Crossing              <chr> "False", "False", "False", "False", "False...
$ Give_Way              <chr> "False", "False", "False", "False", "False...
$ Junction              <chr> "False", "False", "False", "False", "False...
$ No_Exit               <chr> "False", "False", "False", "False", "False...
$ Railway               <chr> "False", "False", "False", "False", "False...
$ Roundabout            <chr> "False", "False", "False", "False", "False...
$ Station               <chr> "False", "False", "False", "False", "False...
$ Stop                  <chr> "False", "False", "False", "False", "False...
$ Traffic_Calming       <chr> "False", "False", "False", "False", "False...
$ Traffic_Signal        <chr> "False", "False", "True", "False", "True",...
$ Turning_Loop          <chr> "False", "False", "False", "False", "False...
$ Sunrise_Sunset        <chr> "Night", "Night", "Night", "Night", "Day",...
$ Civil_Twilight        <chr> "Night", "Night", "Night", "Day", "Day", "...
$ Nautical_Twilight     <chr> "Night", "Night", "Day", "Day", "Day", "Da...
$ Astronomical_Twilight <chr> "Night", "Day", "Day", "Day", "Day", "Day"...
```

Exploring the dataset

```
> # Drop the some columns of the dataframe
> RoadAccidents <- select (Accidentsdata,-c(Description,Street,Side,Zipcode,Country,Ai
rport_Code,Weather_Timestamp,Turning_Loop,Civil_Twilight,Nautical_Twilight,Astronomica
l_Twilight))
>
> View(RoadAccidents)
```

	ID	Source	Severity	Start_Time	End_Time	Start_Lat	Start_Lng	Distance.mi.
1	A-1	MapQuest	3	2016-02-08 05:46:00	2016-02-08 11:00:00	39.86515	-84.05872	0.01
2	A-2	MapQuest	2	2016-02-08 06:07:59	2016-02-08 06:37:59	39.92806	-82.83118	0.01
3	A-3	MapQuest	2	2016-02-08 06:49:27	2016-02-08 07:19:27	39.06315	-84.03261	0.01
4	A-4	MapQuest	3	2016-02-08 07:23:34	2016-02-08 07:53:34	39.74775	-84.20558	0.01
5	A-5	MapQuest	2	2016-02-08 07:39:07	2016-02-08 08:09:07	39.62778	-84.18835	0.01
6	A-6	MapQuest	3	2016-02-08 07:44:26	2016-02-08 08:14:26	40.10059	-82.92519	0.01
7	A-7	MapQuest	2	2016-02-08 07:59:35	2016-02-08 08:29:35	39.75827	-84.23051	0.00
8	A-8	MapQuest	3	2016-02-08 07:59:58	2016-02-08 08:29:58	39.77038	-84.19490	0.01
9	A-9	MapQuest	2	2016-02-08 08:00:40	2016-02-08 08:30:40	39.77806	-84.17200	0.00
10	A-10	MapQuest	3	2016-02-08 08:10:04	2016-02-08 08:40:04	40.10059	-82.92519	0.01

Showing 1 to 12 of 2,974,335 entries, 27 total columns

Duration for each accident

```
> # calculate duration for each accident
> RoadAccidents$duration_Hours <- with(RoadAccidents, difftime(End_Time,Start_Ti
me, units = "hours"))
> RoadAccidents$duration_Mins <- with(RoadAccidents, difftime(End_Time,Start_Tim
e, units = "mins"))
```

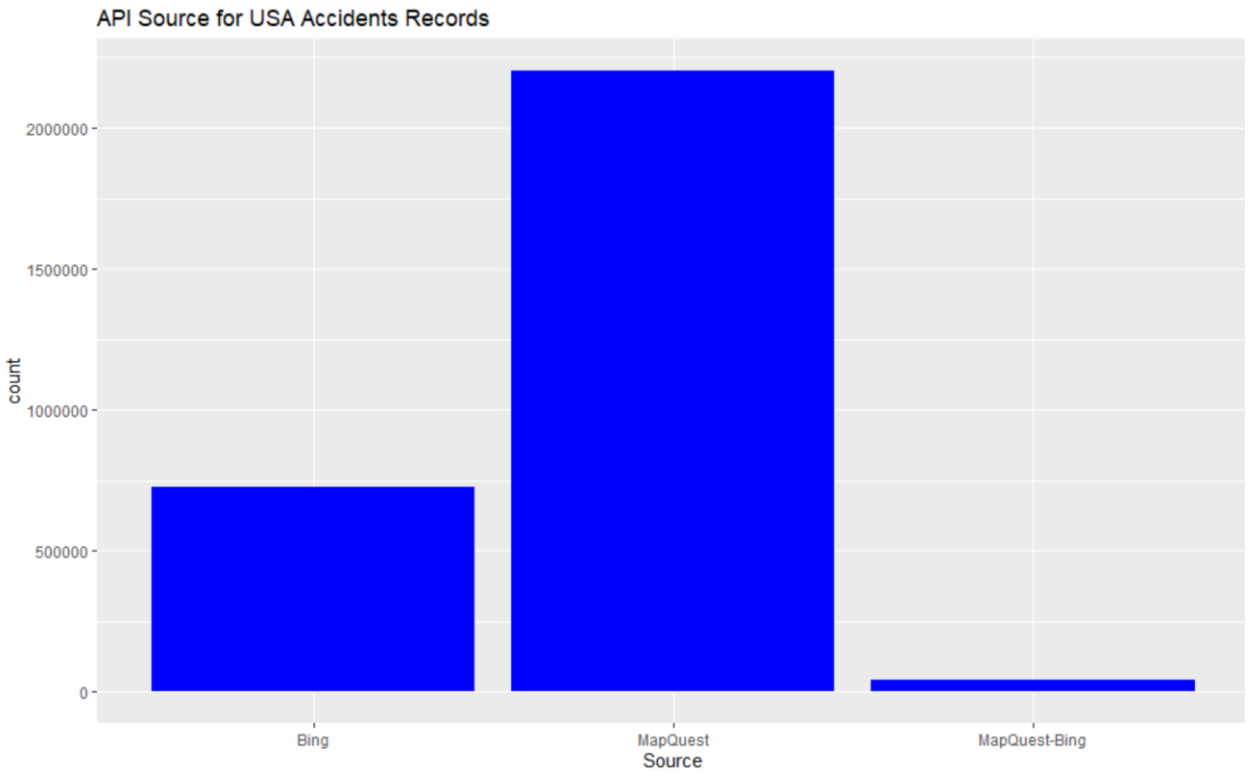
RoadAccidents | US_Accidents

Sunrise_Sunset	duration_Hours	duration_Mins
Night	5.23333333333333	314
Night	0.5	30
Night	0.5	30
Night	0.5	30
Day	0.5	30
Day	0.5	30
Day	0.5	30

Data visualization and analysis

There are only three API sources that reported the accidents. It can be observed that most of the accidents (over 2, 000, 000) were reported by MapQuest, followed by Bing.

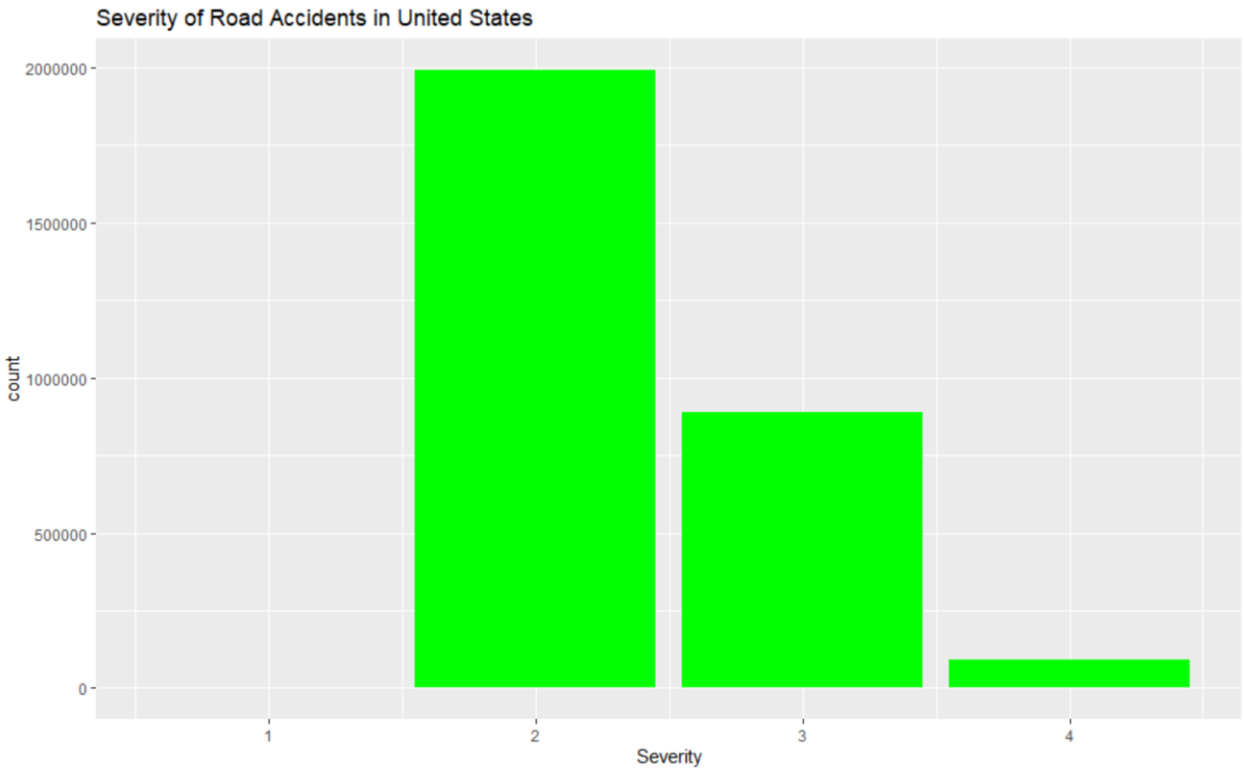
```
> # bar plot of Source
> ggplot(RoadAccidents) +
+   geom_bar(aes(x = Source), fill = 'blue') +
+   ggtitle("API Source for USA Accidents Records")
>
```



Severity of road accidents

Here, I am showcasing the severity of the road accidents.

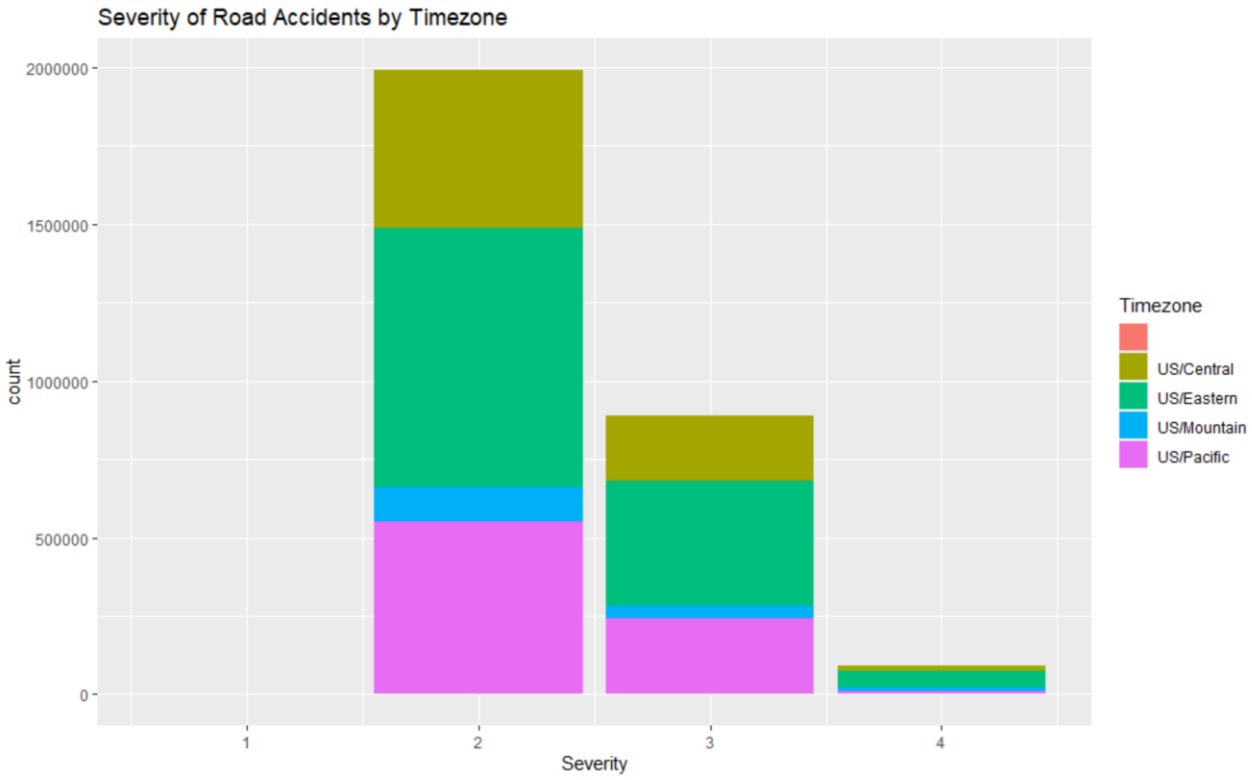
```
> # barplot of severity
> ggplot(RoadAccidents) +
+   geom_bar(aes(x = Severity), fill = 'green') +
+   ggtitle("Severity of Road Accidents in United States")
>
```



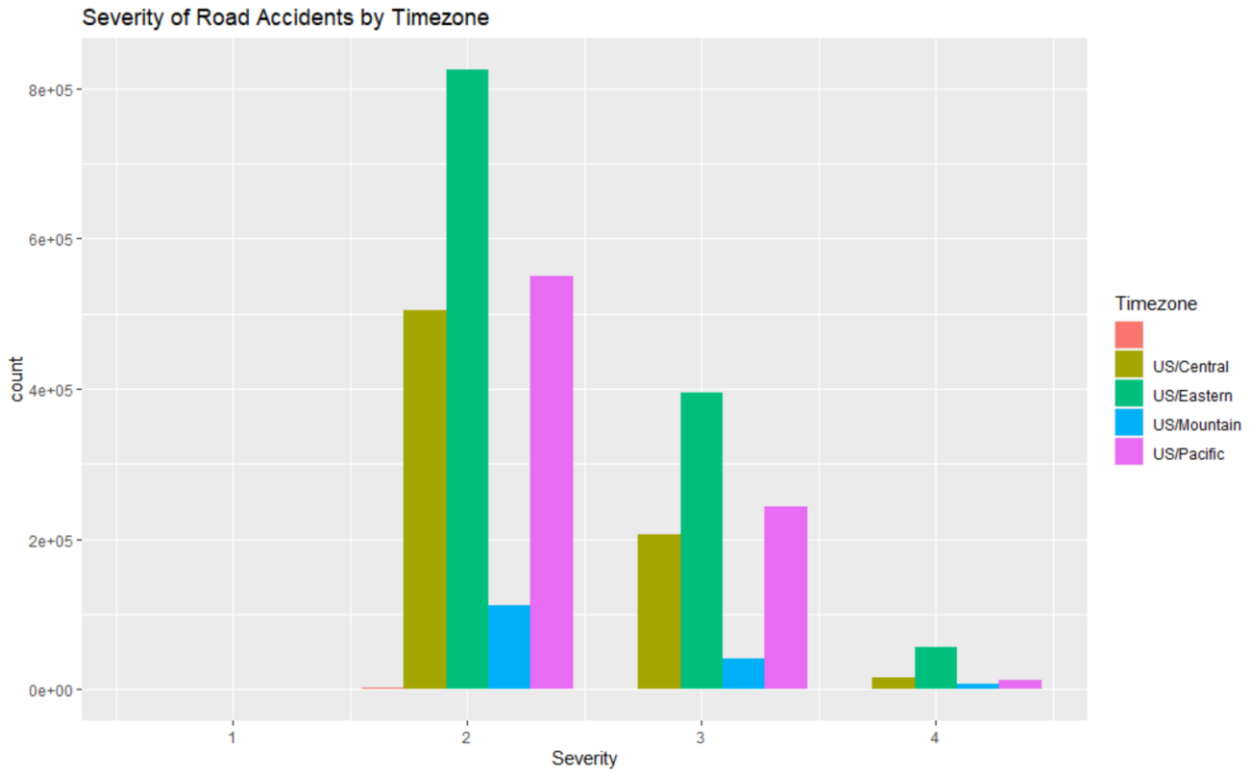
The plot portrays that commonly the accidents had severity equal to 2 (average) followed by 3 (above average), which is unfortunate. There were hardly any accidents with very low severity (0 and 1).

Next, I displayed the severity of the road accidents according to the time zones.

```
> # barplot of severity by Time zone
> ggplot(RoadAccidents, aes(Severity)) +
+   geom_bar(aes(fill = Timezone)) +
+   ggtitle("Severity of Road Accidents by Timezone")
>
```



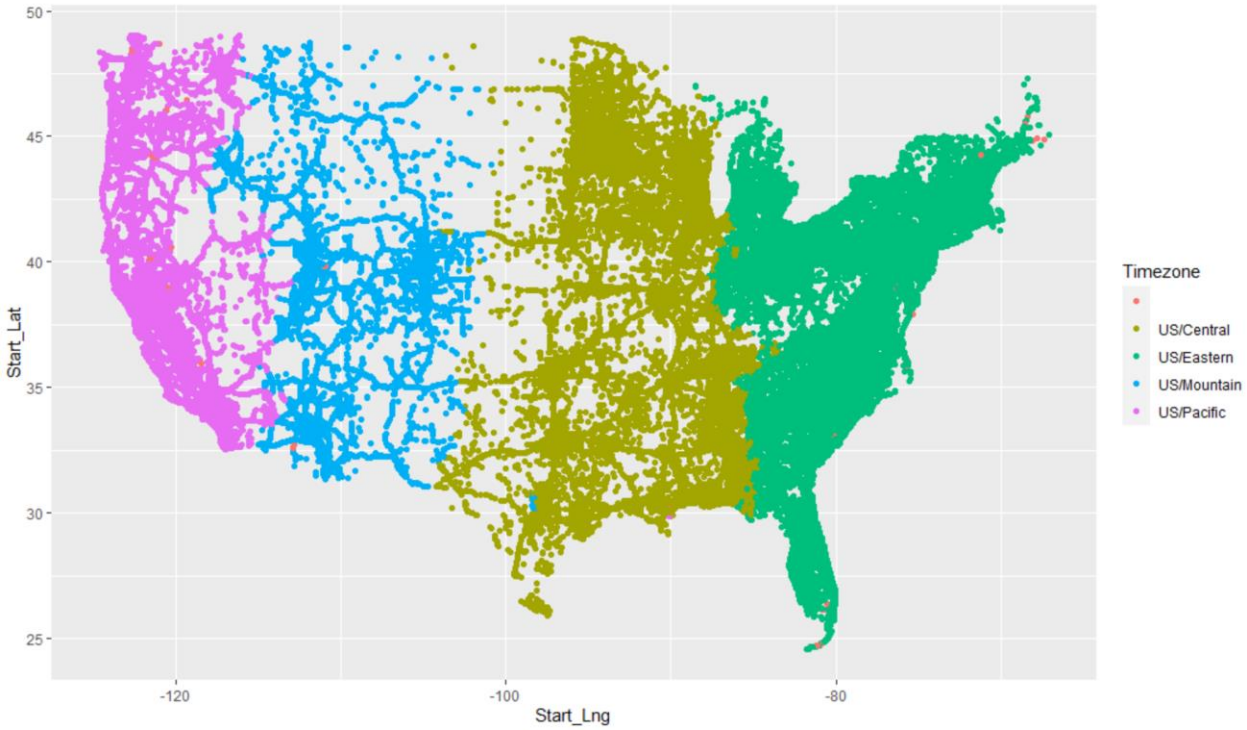

```
> ggplot(data = RoadAccidents, mapping = aes(x = Severity, fill = Timezone))
 +
+   geom_bar(position = "dodge") +
+   ggtitle("Severity of Road Accidents by Timezone")
>
```



Accidents spread by Time zones

The Start_Lat and Start_Lng features can be plotted on a map to get the exact location of the accident. A scatterplot visualization will showcase this.

```
> # scatterplot() of RoadAccidents by Time zones
> ggplot(data = RoadAccidents) +
+        geom_point(mapping = aes(x = Start_Lng, y = Start_Lat, fill=Timezone))
>
```

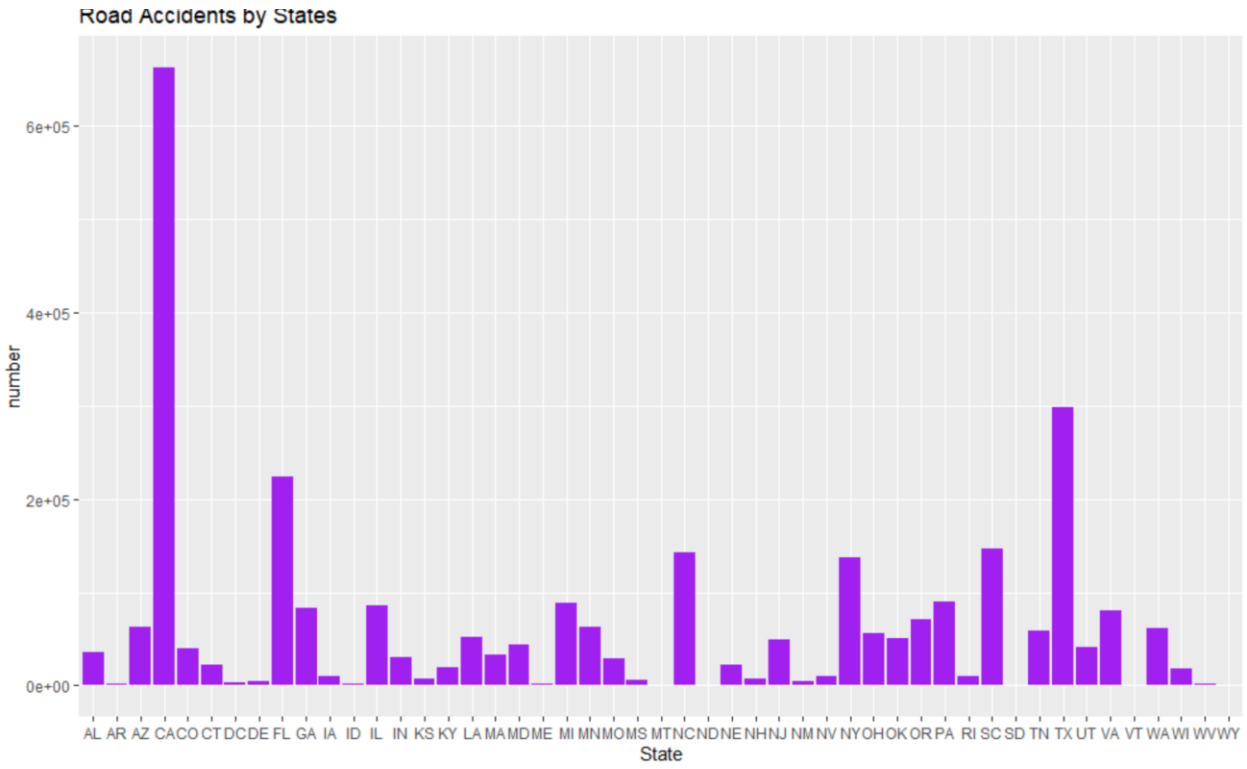


Here, the plot shows four regions where accidents took place in the US. Also, we can see that most of the US accidents occur in the Eastern Time zone.

Accidents ranking by States and Cities

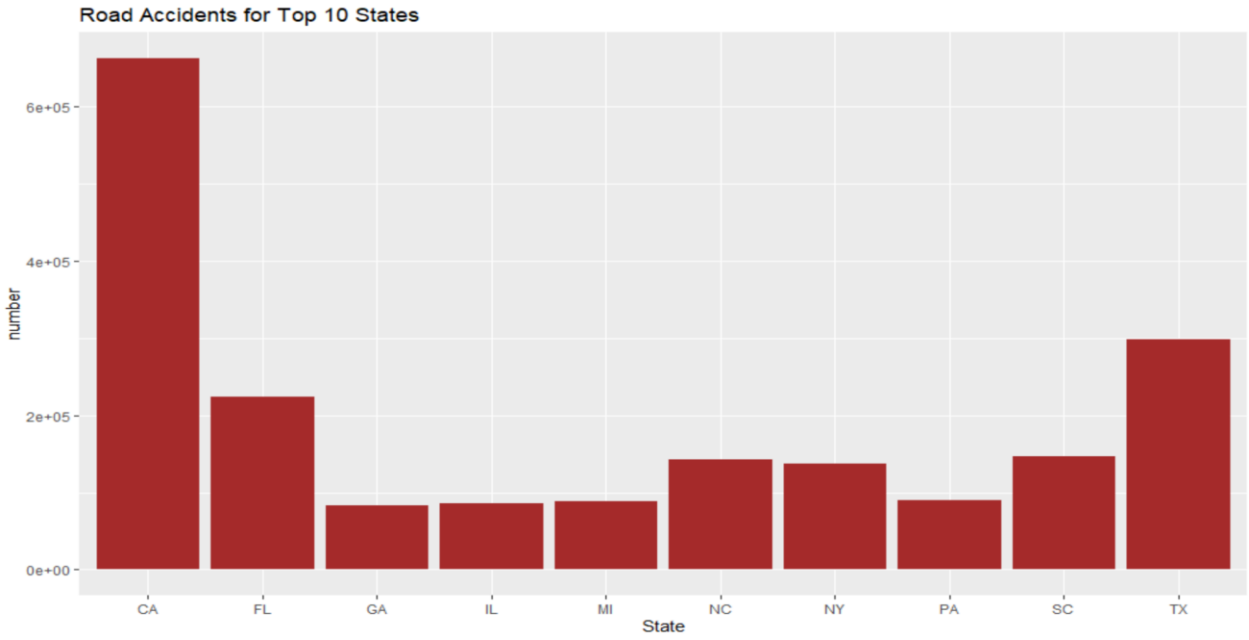
I decided to visualize the most accident-prone states and cities in the USA.

```
> # the total number of Road accidents
> df_State <- RoadAccidents %>%
+   group_by(State) %>%
+   summarize(number=n())
> df_State
# A tibble: 49 x 2
   State number
   <chr>  <int>
 1 AL     36369
 2 AR      1749
 3 AZ     62330
 4 CA    663204
 5 CO     40124
 6 CT     22803
 7 DC      3653
 8 DE      4434
 9 FL    223746
10 GA     83620
# ... with 39 more rows
> ggplot(df_State, aes(x=State, y=number)) +
+   geom_col(fill="purple") +
+   ggtitle("Road Accidents by States")
>
```




```
> # sort df_State data
> data <- df_State[order(-df_State$number),]
>
> # check the number of road accidents for top 10 State
> # and assigned to new data frame "States"
> States <- head(data, 10)
>

> # plot the number of road accidents by each State
> ggplot(States, aes(x=State, y=number)) +
+   geom_col(fill="brown") +
+   ggtitle("Road Accidents for Top 10 States")
>
```



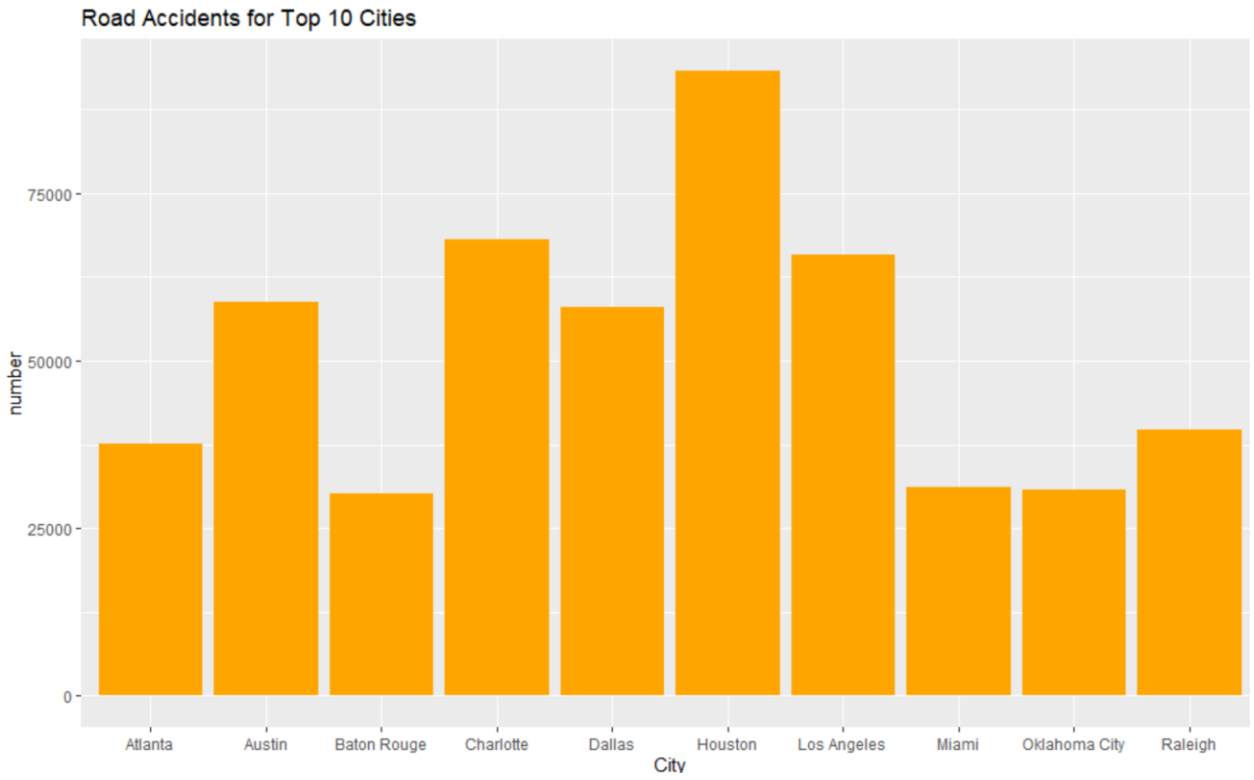
The plot illustrates that California (CA) has the greatest number of accidents followed by Texas (TX) and Florida (FL). The number of accidents in California is almost twice the number of accidents in Texas.

```
> # the total number of Road accidents by Cities
> df_City <- RoadAccidents %>%
+   group_by(City) %>%
+   summarize(number=n())
> df_City
# A tibble: 11,686 x 2
   City                        number
   <chr>                        <int>
 1 ""                              83
 2 "Aaronsburg"                     4
 3 "Abbeville"                    224
 4 "Abbotsford"                    14
 5 "Abbott"                        30
 6 "Abbottstown"                   37
 7 "Aberdeen"                     521
 8 "Aberdeen Proving Ground"        1
 9 "Abernathy"                      3
10 "Abilene"                       41
# ... with 11,676 more rows
>
>
> # sort df_City data
> data_City <- df_City[order(-df_City$number),]
>
> # check the number of road accidents for the top 10 City
> # and assigned to new data frame "Cities"
> Cities <- head(data_City, 10)
>
```

	City	number
1	Houston	93289
2	Charlotte	68054
3	Los Angeles	65851
4	Austin	58703
5	Dallas	58036

Showing 1 to 6 of 10 entries, 2 total columns

```
> # plot the number of road accidents by each State
> ggplot(Cities, aes(x=City, y=number)) +
+   geom_col(fill="Orange") +
+   ggtitle("Road Accidents for Top 10 Cities")
>
```

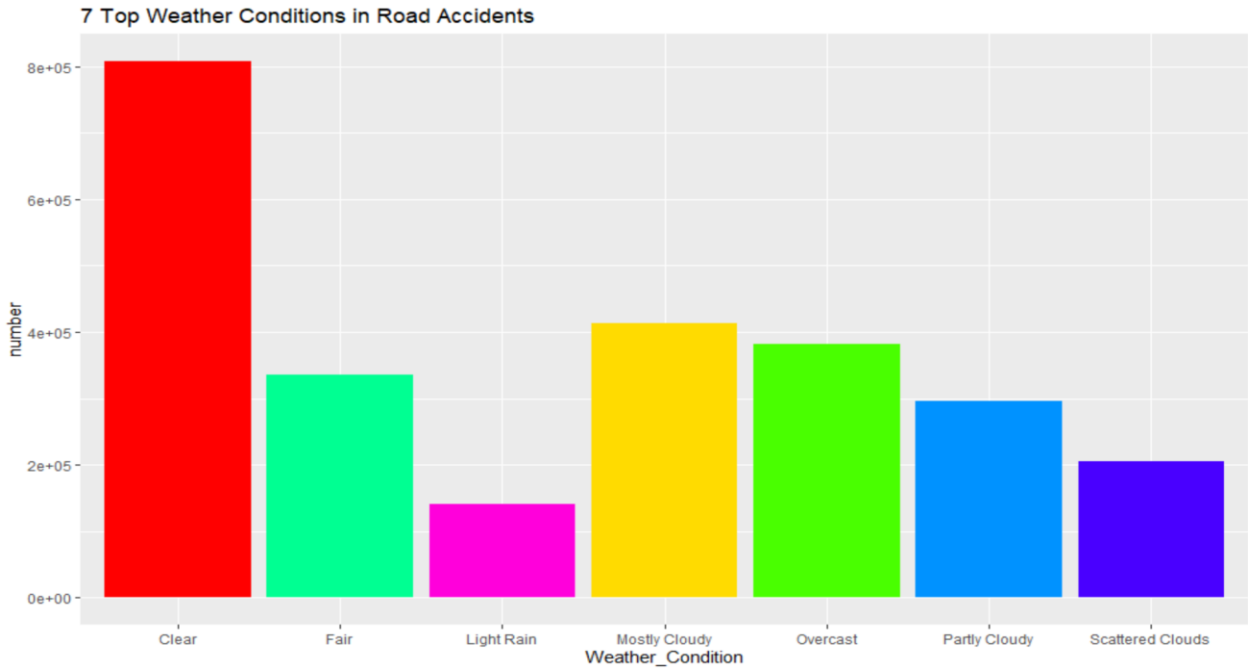


From this plot, we can see that most of the accidents occured in Houston, followed by Charlotte and Los Angeles.

Impact of Weather Conditions

```
> # plot of weather conditions
> ggplot(RoadAccidents) +
+   geom_bar(aes(x = Weather_Condition), fill = 'red')
>
> # the Weather Conditions in Road accidents
> Wc <- RoadAccidents %>%
+   group_by(Weather_Condition) %>%
+   summarize(number=n())
> Wc
# A tibble: 121 x 2
   Weather_Condition      number
   <chr>                  <int>
 1 ""                     65932
 2 "Blowing Dust"            44
 3 "Blowing Dust / Windy"    64
 4 "Blowing Sand"             1
 5 "Blowing Snow"           268
 6 "Blowing Snow / Windy"    10
 7 "Clear"                808171
 8 "Cloudy"               115496
 9 "Cloudy / Windy"         2097
10 "Drizzle"                2044
# ... with 111 more rows
>
```

```
> # sort Wc data
> data_Wc <- Wc[order(-Wc$number),]
>
> # check the top 7 weather conditions for road accidents
> # and assigned to new data frame "weather"
> data_Wc2 <- head(data_Wc, 7)
>
> # plot of 7 Top Weather Conditions in road accidents
> ggplot(data_Wc2, aes(x=Weather_Condition, y=number)) +
+   geom_col(fill=rainbow(7)) +
+   ggtitle("7 Top Weather Conditions in Road Accidents")
>
```



The plot shows that the weather condition for most of the accidents were clear, followed by mostly cloudy and overcast. Overcast and mostly cloudy are reasonable factors for accidents unlike clear, which means that weather conditions do not play an important role.

Conclusion

The US Accidents dataset downloaded from Kaggle was visualized and analyzed after thorough cleaning of missing values and eliminating other less important features of the dataset. From this study, it is interesting to see many exciting results – one of which is that California had the highest number of accidents among the states in North America. While Houston, a city in Texas, recorded the highest number of accidents among the cities in North America. I also discovered that most of the accidents occurred under clear weather conditions. Also, majority of the accidents took place in the Eastern Time Zone. The most important feature of this data is the severity of the accidents reported. It is unfortunate to know that no record of accidents between 0 and 1 severity were recorded and most of the severity where in severity 2. Another factor that was deduced from the data was the duration of accidents.

Reference

Rawat, S. (2020, February 21). USA Accidents Data Analysis. Retrieved May 04, 2020, from https://towardsdatascience.com/usa-accidents-data-analysis-d130843cde02

Moosavi, S. (2020, January 17). US Accidents (3.0 million records). Retrieved May 04, 2020, from https://www.kaggle.com/sobhanmoosavi/us-accidents

Road Safety Facts. (n.d.). Retrieved May 04, 2020, from https://www.asirt.org/safe-travel/road-safety-facts/