
WELCOME
Project by :-
Prathamesh Anand Patole(12001)
Govinda Kishor N(12054)
Topic :- Uber Data Analysis Using Data Visualization

INDEX

INTRODUCTION

Data storytelling is an important component of Machine Learning through which companies are able to understand the background of various operations. With the help of visualization, companies can avail the benefit of understanding the complex data and gain insights that would help them to craft decisions. In this project we are going to analyze the data of Uber company. We will be using R commands to generate different graphs. *Hope you will enjoy the project!!!*

INTRODUCTION TO R LANGUAGE

R is a programming language for statistical computing and graphics supported by the R Core Team and the R Foundation for Statistical Computing. Created by statisticians Ross Ihaka and Robert Gentleman, R is used among data miners and statisticians for data analysis and developing statistical software. Users have created packages to augment the functions of the R language.

Polls, data mining surveys, and studies of scholarly literature databases show that R is highly popular; since January 2022, R ranks 12th in the TIOBE index, a measure of programming language popularity.

The official R software environment is an open-source free software environment within the GNU package, available under the GNU General Public License. It is written primarily in C, Fortran, and R itself (partially self-hosting). Precompiled executables are provided for various operating systems. R has a command line interface. Multiple third-party graphical user interfaces are also available, such as RStudio, an integrated development environment, and Jupyter, a notebook interface.

ABOUT DATA AND GRAPHS

In this project, we will analyze the ***Uber Pickups in New York City dataset.*** In which there is details of each and every pickups of Uber in New York city on specific months of 2014. Using R commands we will be generating graphs through which we can easily analyze the data. We will be using Bar Graph, Heat Map and Ride based map for our analysis.

Bar Graph:- Bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. A bar graph shows comparisons among discrete categories.

Heat Map:- A heat map or choropleth map is a data visualization that shows the relationship between two measures and provides rating information. The rating information is displayed using varying colors or saturation and can exhibit ratings such as high to low or bad to awesome, and needs improvement to working well. It can also be a thematic map in which the area inside recognized boundaries is shaded in proportion to the data being represented.

Ride based map:- This is overall map of the city based on Uber ride. It will show the pickup points in city based on latitude and the longitude of the location.

DATA PREPROCESSING

Data Cleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

Noisy Data:
Noisy data is a meaningless data that can't be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc.

Clustering:
This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters. We used clustering method to eliminate the noisy data or error data. And we used processed data for our analysis.

PACKAGES/LIBRARIES USED

In the first step of our project, we will import the essential packages that we will use in this uber data analysis project. Some of the important libraries of R that we will use are –

- **ggplot2**

  This is the backbone of this project. ggplot2 is the most popular data visualization library that is most widely used for creating aesthetic visualization plots.

- **ggthemes**

  This is more of an add-on to our main ggplot2 library. With this, we can create better create extra themes and scales with the mainstream ggplot2 package.

- **Lubridate**

  Our dataset involves various time-frames. In order to understand our data in separate time categories, we will make use of the lubridate package.

- **dplyr**

  This package is the lingua franca of data manipulation in R.

- **tidyr**

  This package will help you to tidy your data. The basic principle of tidyr is to tidy the columns where each variable is present in a column, each observation is represented by a row and each value depicts a cell.

- **DT**

  With the help of this package, we will be able to interface with the JavaScript Library called – Data tables.

- **scales**

  With the help of graphical scales, we can automatically map the data to the correct scales with well-placed axes and legends.

BASIC COMMANDS OF R

Commands to install libraries :-

```
library(ggplot2)
library(ggthemes)
library(lubridate)
library(dplyr)
library(tidyr)
library(DT)
library(scales)
```

- Create a vector of our colors

```
colors = c("#CC1011", "#665555", "#05a399", "#cfcaca", "#f5e840", "#0683c9", "#e075b0")
```

- Formatting of Date

```
apr_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-apr14.csv")
may_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-may14.csv")
jun_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-jun14.csv")
jul_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-jul14.csv")
aug_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-aug14.csv")
sep_data <- read.csv("D:/DEAP Project/Uber-dataset/uber-raw-data-sep14.csv")
data_2014 <- rbind(apr_data,may_data, jun_data, jul_data, aug_data, sep_data)
data_2014$Date.Time <- as.POSIXct(data_2014$Date.Time, format = "%m/%d/%Y %H:%M:%S")
data_2014$Time <- format(as.POSIXct(data_2014$Date.Time, format = "%m/%d/%Y %H:%M:%S"), format="%H:%M:%S")
data_2014$Date.Time <- ymd_hms(data_2014$Date.Time)
data_2014$day <- factor(day(data_2014$Date.Time))
data_2014$month <- factor(month(data_2014$Date.Time, label = TRUE))
data_2014$year <- factor(year(data_2014$Date.Time))
data_2014$dayofweek <- factor(wday(data_2014$Date.Time, label = TRUE))
data_2014$hour <- factor(hour(hms(data_2014$Time)))
data_2014$minute <- factor(minute(hms(data_2014$Time)))
data_2014$second <- factor(second(hms(data_2014$Time)))
```

These commands will arrange the data and time in standard format

HOUR WISE TRIPS CALCULATION

- **Command :-**

```
hour_data <- data_2014 %>%
 group_by(hour) %>%
 dplyr::summarize(Total = n())
 datatable(hour_data)
```

Output :-

Show 100 entries Search:

	hour	Total
1	0	103836
2	1	67227
3	2	45865
4	3	48287
5	4	55230
6	5	83939
7	6	143213
8	7	193094
9	8	190504
10	9	159967
11	10	159148
12	11	165703
13	12	170452
14	13	195877
15	14	230625
16	15	275466
17	16	313400
18	17	336190
19	18	324679
20	19	294513
21	20	284604
22	21	281460
23	22	241858
24	23	169190

Showing 1 to 24 of 24 entries Previous 1 Next

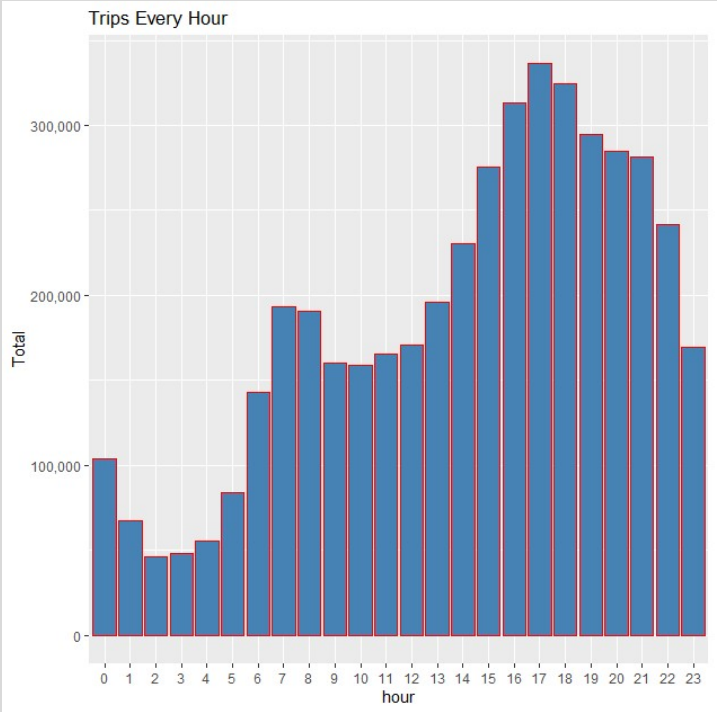
DATA VISUALIZATION

Bar Graph to show hour wise number of trips :-

```
ggplot(hour_data, aes(hour, Total)) +
geom_bar( stat = "identity", fill = "steelblue", color = "red") +
ggtitle("Trips Every Hour") +
theme(legend.position = "none") +
scale_y_continuous(labels = comma)
```

OUTPUT :-

So from above graph its clear that trips are maximum during 5PM-6PM. And minimum during 2 AM-3AM in New York city during month April, May, June, July, August, September.



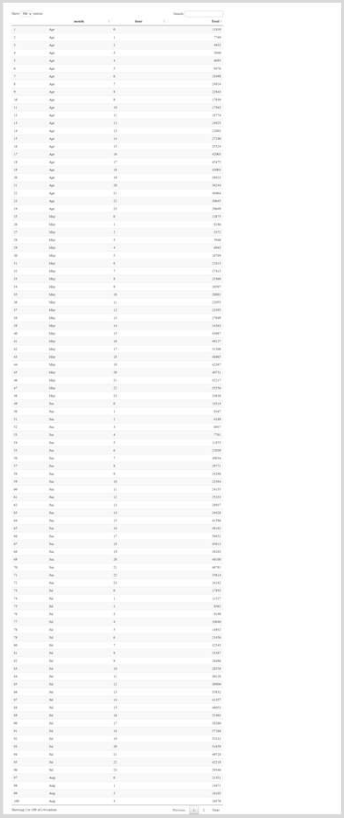
HOURWISE-MONTHWISE TRIPS CALCULATION

- **Command :-**

```
month_hour <- data_2014 %>%
group_by(month, hour) %>%
dplyr::summarize(Total = n())
datatable(month_hour)
```

OUTPUT :-

This table shows the hour wise- month wise number of trips.



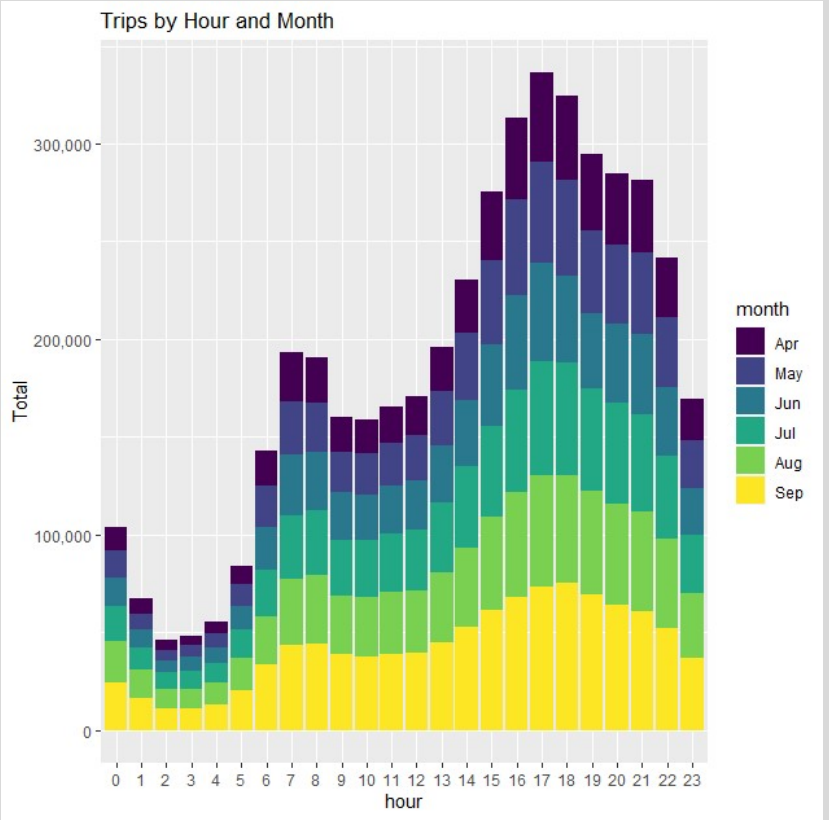
DATA VISUALIZATION

Command :-

```
ggplot(month_hour, aes(hour, Total, fill = month)) +
 geom_bar( stat = "identity") +
 ggtitle("Trips by Hour and Month") +
 scale_y_continuous(labels = comma)
```

OUTPUT :-

This graph shows the months wise share among trips in specific hours. Each and every color represent the each and every month as represented in the index.



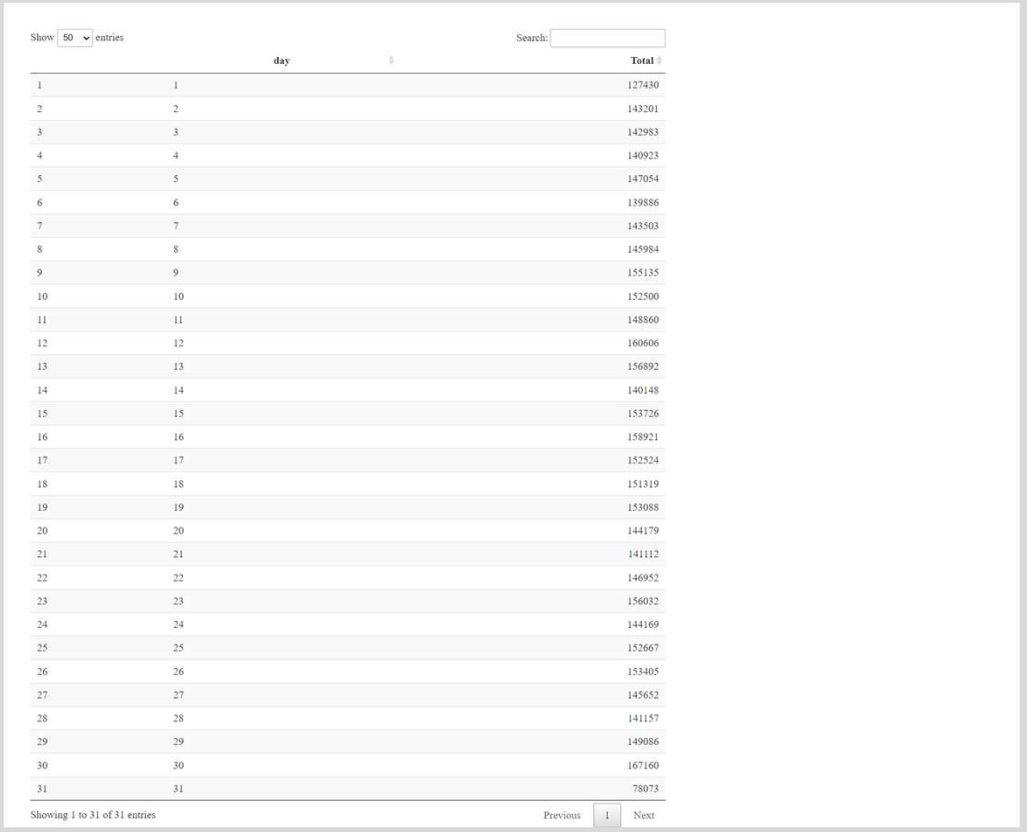
DAYWISE TRIP CALCULATION

Command :-

```
day_group <- data_2014 %>%
 group_by(day) %>%
 dplyr::summarize(Total = n())
 datatable(day_group)
```

Output :-

This table shows the day wise number of trips including all Six months.



Show 50 entries Search:

	day	Total
1	1	127430
2	2	143201
3	3	142983
4	4	140923
5	5	147054
6	6	139886
7	7	143503
8	8	145984
9	9	155135
10	10	152500
11	11	148860
12	12	160606
13	13	156892
14	14	140148
15	15	153726
16	16	158921
17	17	152524
18	18	151319
19	19	153088
20	20	144179
21	21	141112
22	22	146952
23	23	156032
24	24	144169
25	25	152667
26	26	153405
27	27	145652
28	28	141157
29	29	149086
30	30	167160
31	31	78073

Showing 1 to 31 of 31 entries Previous 1 Next

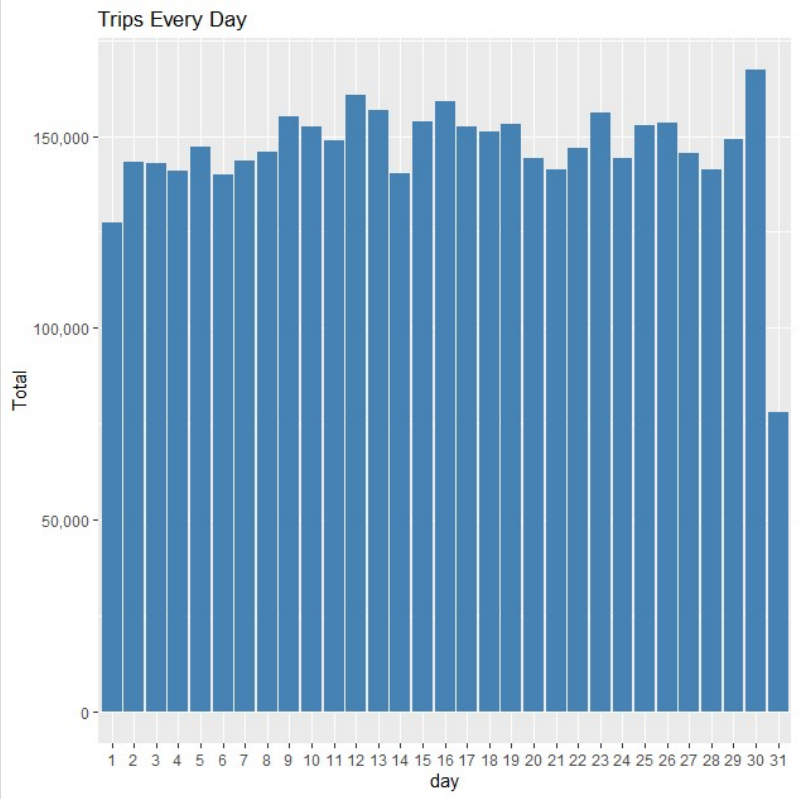
DATA VISUALIZATION

Command :-

```
ggplot(day_group, aes(day, Total)) +
geom_bar( stat = "identity", fill = "steelblue") +
ggtitle("Trips Every Day") +
theme(legend.position = "none") +
scale_y_continuous(labels = comma)
```

Output :-

We observe from the resulting visualization that 30th of the month had the highest trips in the year which is mostly contributed by the month of April. And lowest in 31st day but we can't say it as lowest because Every month does not contain 31 days.



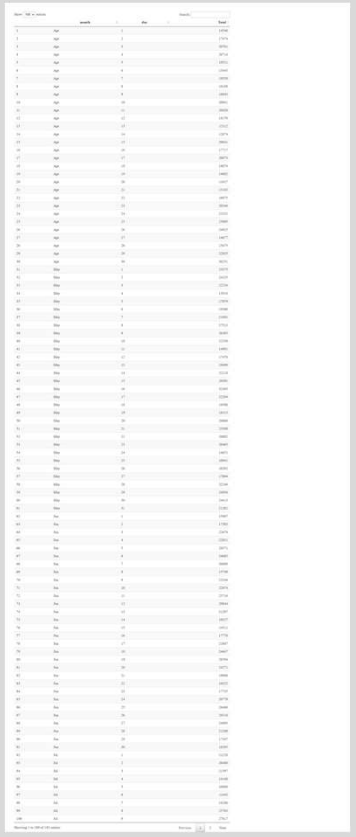
MONTHWISE-HOURWISE TRIPS

Command :-

```
day_month_group <- data_2014 %>%
group_by(month, day) %>%
dplyr::summarize(Total = n())
datatable(day_month_group)
```

Output :-

This table shows the month wise day wise number of trips.



DATA VISUALIZATION

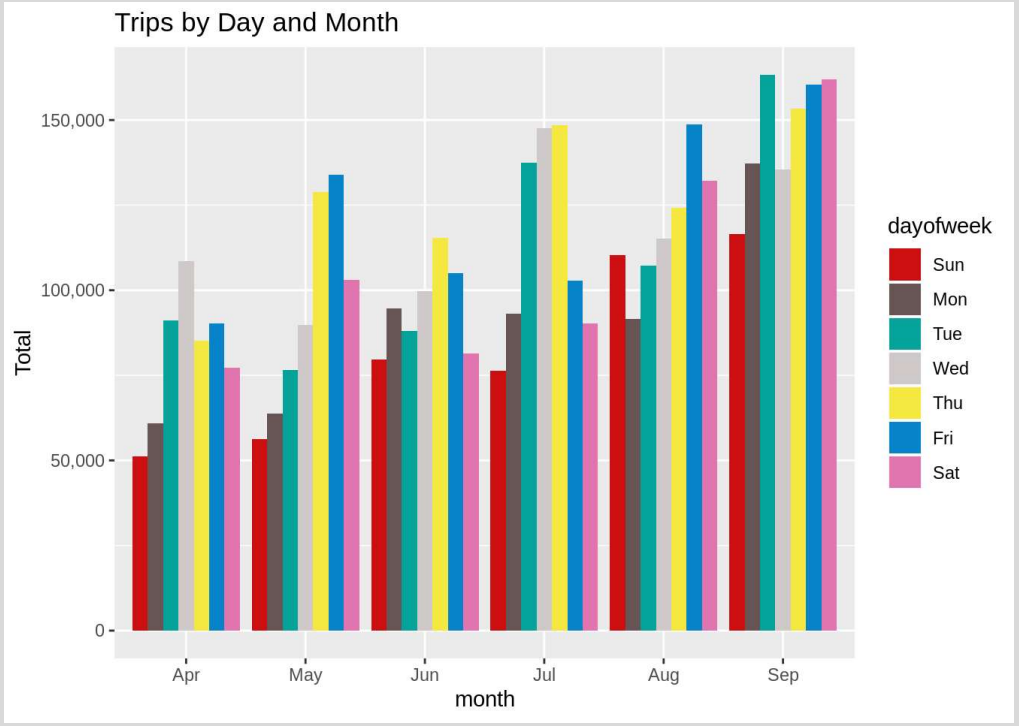
COMMAND :-

```
ggplot(month_weekday, aes(month, Total, fill = dayofweek)) +
geom_bar( stat = "identity", position = "dodge") +
ggtitle("Trips by Day and Month") +
scale_y_continuous(labels = comma) +
scale_fill_manual(values = colors)
```

OUTPUT :-

This graph basically shows the month wise-day wise number of trips. So

We can clearly conclude from above graph that in most of months Sunday has very lowest number of trips . But only in August Monday has lowest number of trips. Incase of every month highest number of trips changes as month changes. In April Wednesday got highest number of trips , where as in May and August Friday got highest number of trips. And in June-July on Thursday has highest demand. But in September we can see Tuesday got highest number of trips.



MONTHWISE TRIP CALCULATION

Command :-

```
month_group <- data_2014 %>%
 group_by(month) %>%
 dplyr::summarize(Total = n())
 datatable(month_group)
```

OUTPUT :-

This is simple table which shows Month wise number of trips.

Show 10 entries Search:

	month	Total
1	Apr	564516
2	May	652435
3	Jun	663844
4	Jul	796121
5	Aug	829275
6	Sep	1028136

Showing 1 to 6 of 6 entries Previous 1 Next

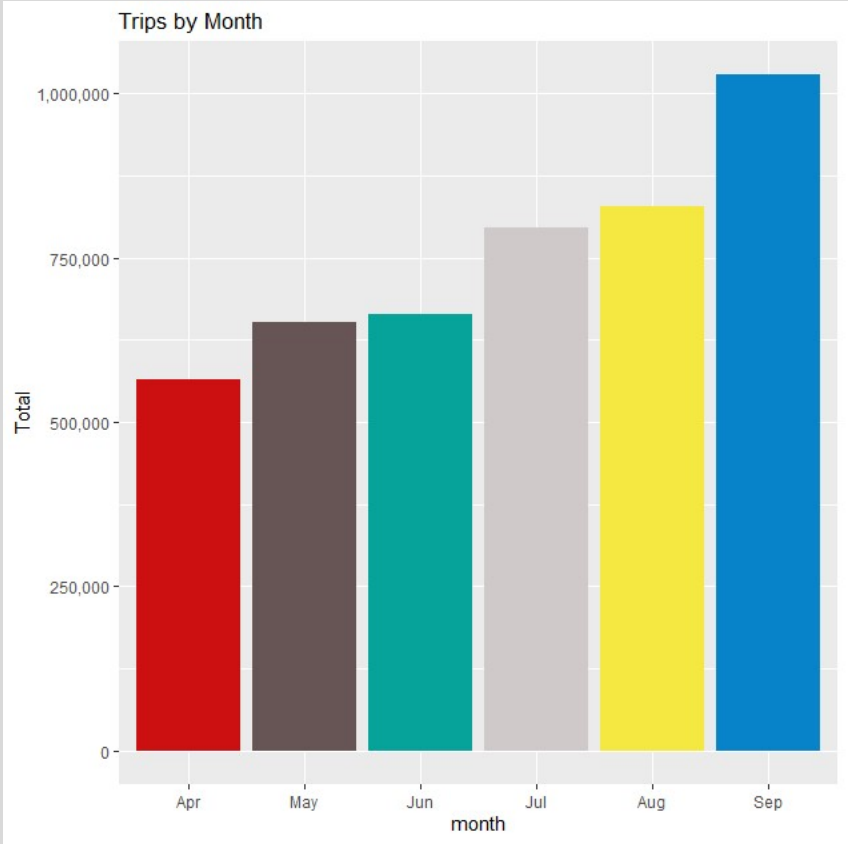
DATA VISUALIZATION

Command :-

```
ggplot(month_group , aes(month, Total, fill = month)) +
geom_bar( stat = "identity") +
ggtitle("Trips by Month") +
theme(legend.position = "none") +
scale_y_continuous(labels = comma) +
scale_fill_manual(values = colors)
```

OUTPUT :-

This is the basic bar graph which represents number of trips in each month. From The graph its clear that in September Uber got most number of trips where as in April Uber got least number of trips.



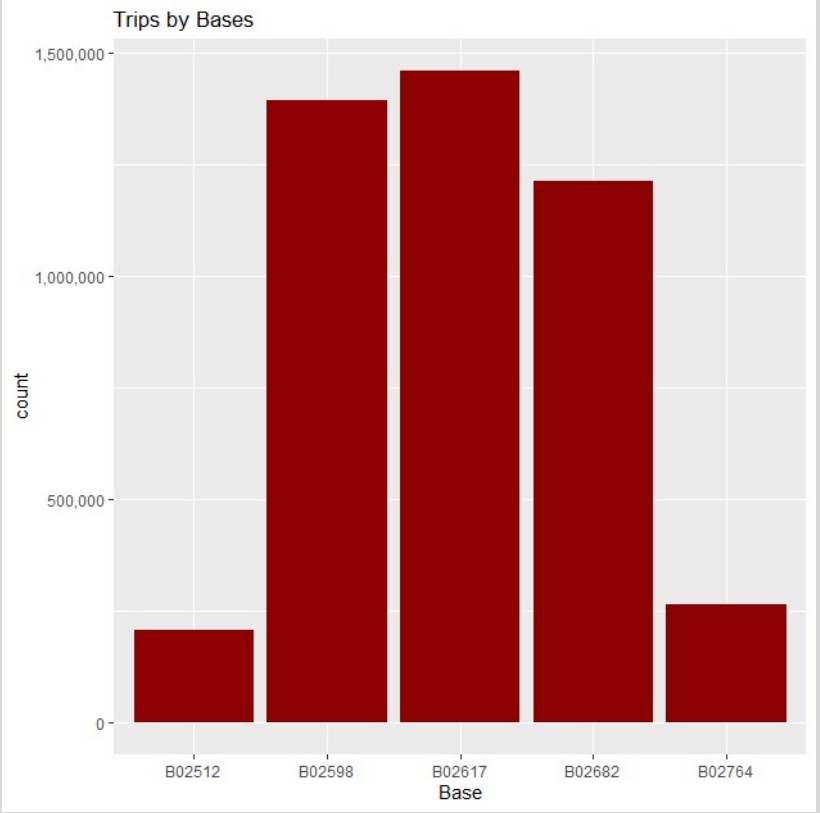
BASEWISE NUMBER OF TRIPS

Command :-

```
ggplot(data_2014, aes(Base)) +
geom_bar(fill = "darkred") +
scale_y_continuous(labels = comma) +
ggtitle("Trips by Bases")
```

Output :-

There are five bases in all out of which, we observe that B02617 had the highest number of trips. And B02512 had lowest number of trips. Its around 1,450,000 and 200,000 trips respectively.



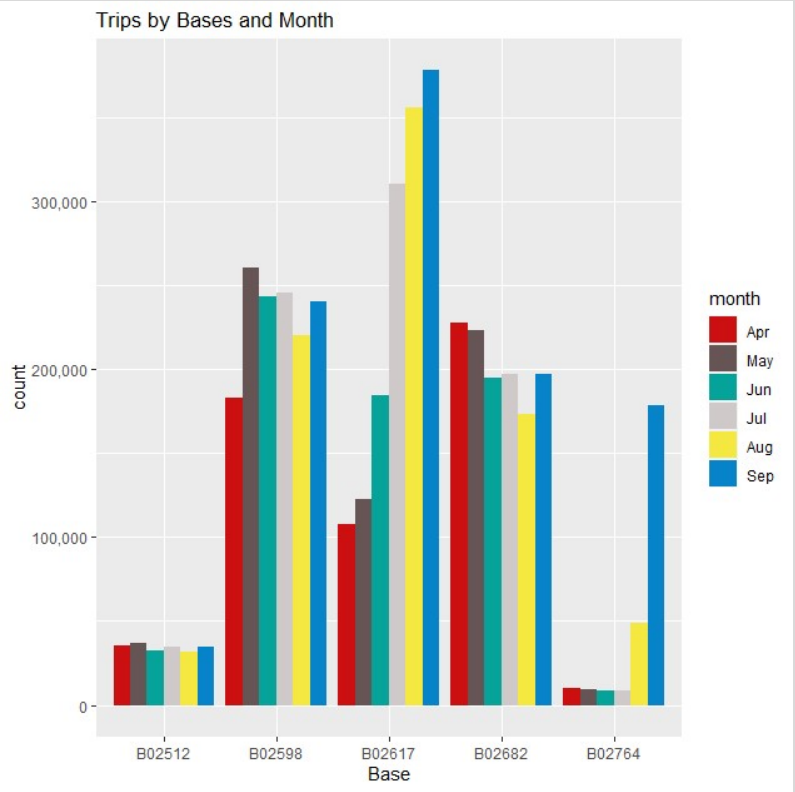
TRIPS BY BASES AND MONTHS

Commands :-

```
ggplot(data_2014, aes(Base, fill = month)) +
geom_bar(position = "dodge") +
scale_y_continuous(labels = comma) +
ggtitle("Trips by Bases and Month") +
scale_fill_manual(values = colors)
```

Output :-

So above graph clearly shows number of trips in different month from different bases. So there was maximum trips from base B02617 in the month of September. Where as minimum number of trips from base B02764 in the month of July. In most of the base we can see most number of trips in month of September where as least number of trips changes randomly from month to month.



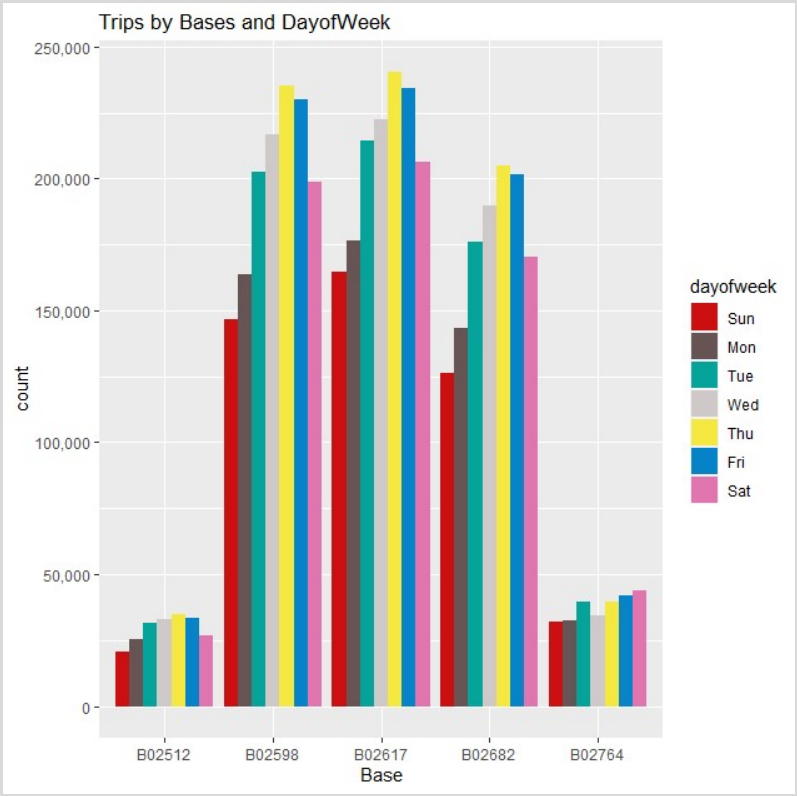
TRIPS BY BASES WEEKDAY WISE

Command :-

```
ggplot(data_2014, aes(Base, fill = dayofweek)) +
  geom_bar(position = "dodge") +
  scale_y_continuous(labels = comma) +
  ggtitle("Trips by Bases and DayofWeek") +
  scale_fill_manual(values = colors)
```

Output :-

Above graph represent number of trips from each base with respect to week days. In general there is most number of trips from each base on Thursday and lowest number of trips mostly in Sunday as it comes Holliday.



HOUR WISE NUMBER OF TRIPS IN EACH DAY

Command :-

```
day_and_hour <- data_2014 %>%
group_by(day, hour) %>%
dplyr::summarize(Total = n())
datatable(day_and_hour)
```

Output :-

This table basically represent hour wise number of trips in each and every day.

Let do some analysis on this table.

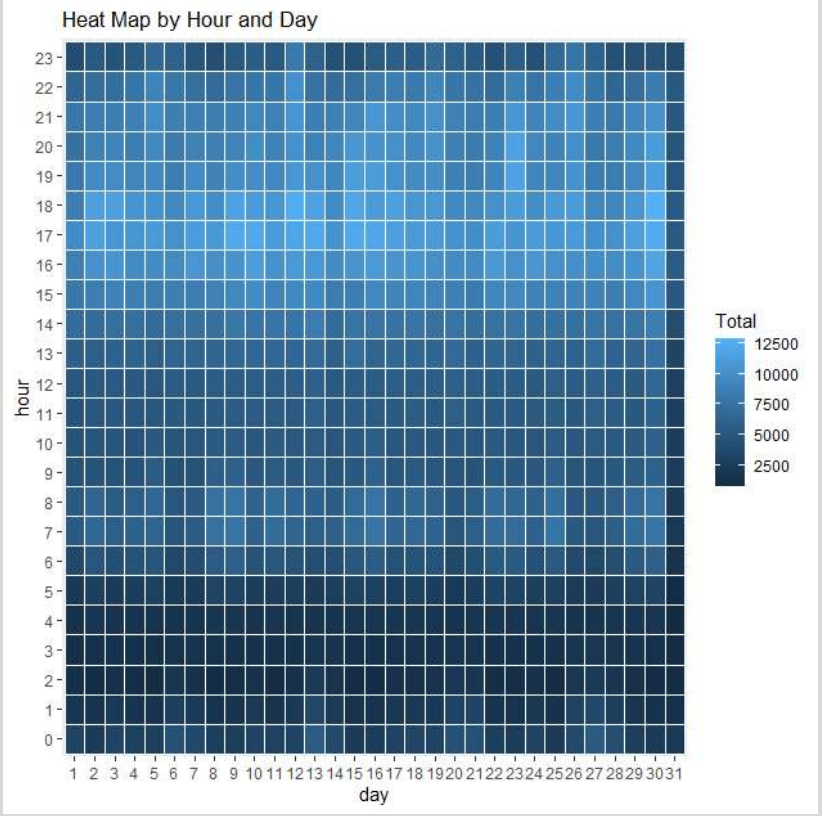
DATA VISUALIZATION

Command :-

```
ggplot(day_and_hour, aes(day, hour, fill = Total)) +
 geom_tile(color = "white") +
 ggtitle("Heat Map by Hour and Day")
```

Output :-

This heat map just shows relationship between hours and days in month. We can clearly see from index that number of trips is very low during early morning like 2AM-3AM.And we can also see number of trips are intense during evening like 4PM-around 6PM.If we see day wise trips are some what intense on 30th day of month and lowest in 31st day of month. But we cant say lowest because every month doesn't contain 31 days.



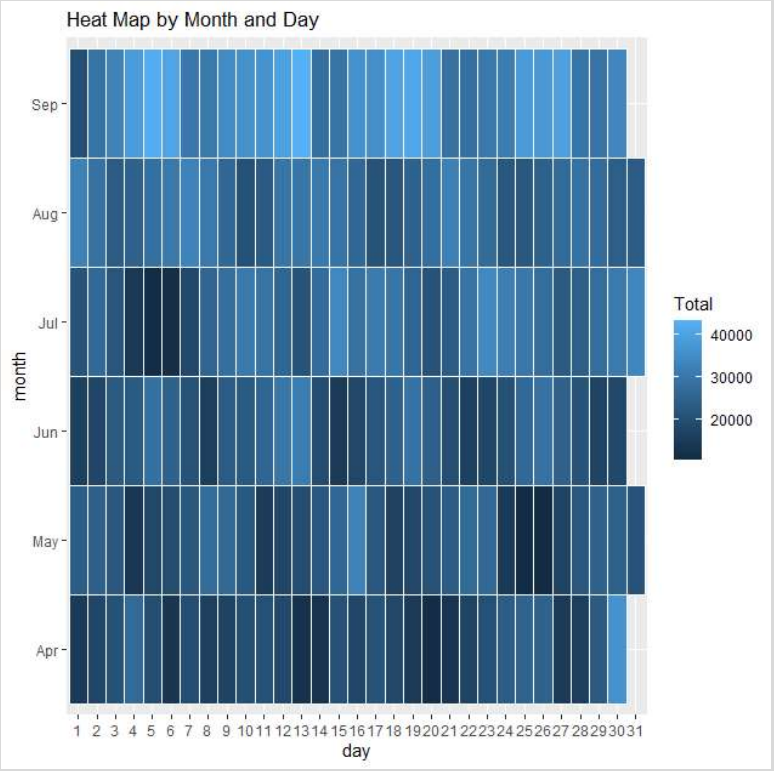
HEAT MAP BY HOURS AND DAYS

Command :-

```
ggplot(day_month_group, aes(day, month, fill = Total)) +
geom_tile(color = "white") +
ggtitle("Heat Map by Month and Day")
```

Output :-

From heat map its clear that September is most demanded month for the Uber where as April is least demanded month. On the middle of each and every month there is no such great demand for the trips. So from this data Uber can keep exciting offers for customer attraction in middle of every month and make more and more profit.



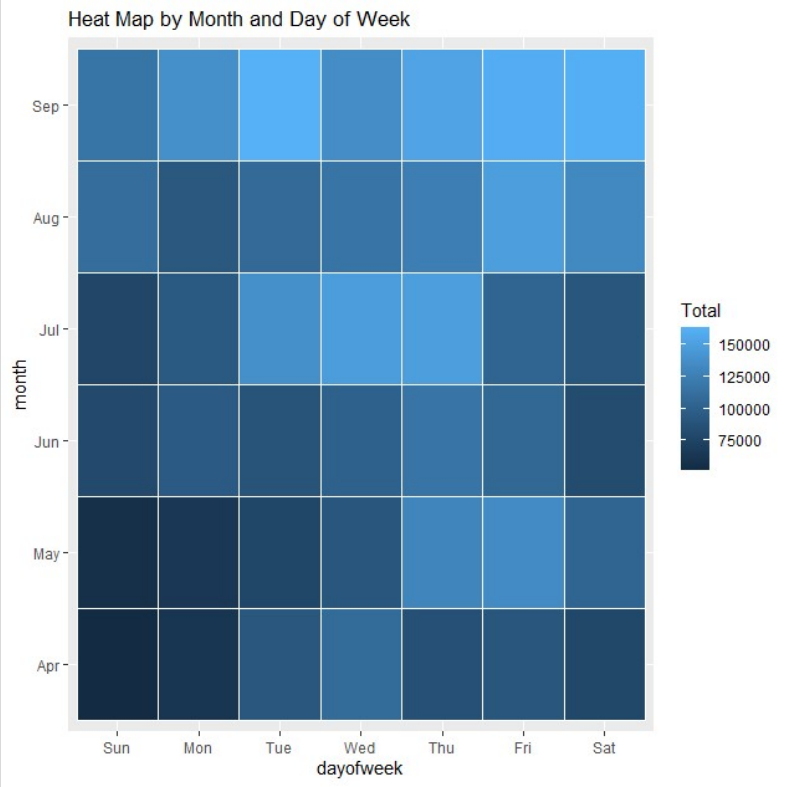
HEAT MAP BY MONTH AND DAY OF WEEK

Command :-

```
ggplot(month_weekday, aes(dayofweek, month, fill = Total)) +
geom_tile(color = "white") +
ggtitle("Heat Map by Month and Day of Week")
```

Output :-

This heat map clearly shows that Sundays of April month see very less number of trips which is less than 75000 trips overall. In the same way Saturdays of September is maximum number of trips its more than 150000 trips. And middle week days of July also saw some peak in demand but thing we should note is that there is less demand for Uber in starting of every month.



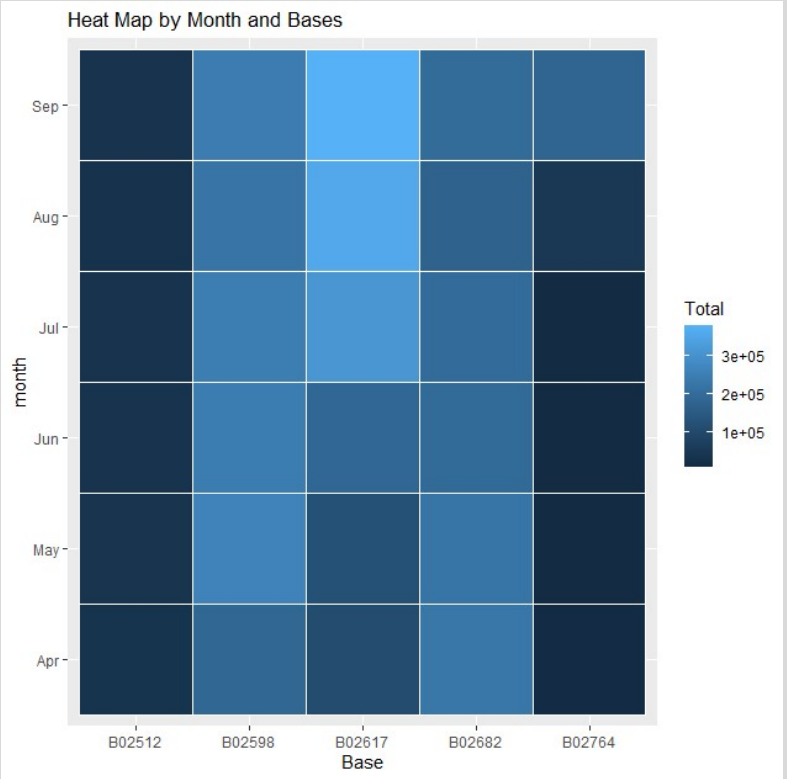
HEAT MAP BY MONTH WITH RESPECT TO BASE

Command :-

```
month_base <-  data_2014 %>%
            group_by(Base, month) %>%
              dplyr::summarize(Total = n())
day0fweek_bases <-  data_2014 %>%
            group_by(Base, dayofweek) %>%
              dplyr::summarize(Total = n())
ggplot(month_base, aes(Base, month, fill = Total)) +
        geom_tile(color = "white") +
         ggtitle("Heat Map by Month and Bases")
```

Output :-

From heat map its clear that base B02512 and B02764 is bases with least number of trips and B02598 and B02617 are the bases with most number of trips. And again its clear that September is peak season and April is month with least number of trips for each base.



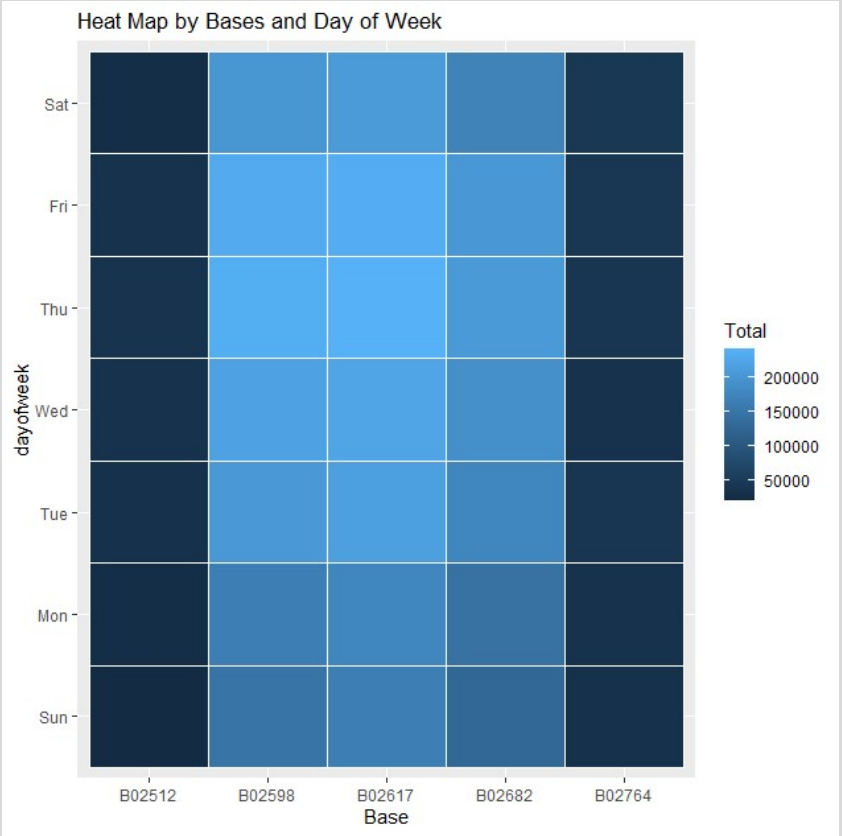
HEAT MAP BY BASE AND DAYS OF WEEK

Command :-

```
ggplot(day0fweek_bases, aes(Base, dayofweek, fill = Total)) +
geom_tile(color = "white") +
ggtitle("Heat Map by Bases and Day of Week")
```

Output :-

From this heat map its crystal clear that Base B02512 and B02764 is the base with least number of trips where as B02617 is base with most number of trips. And its also clear that Sunday has least demand among week days and Thursday and well as Friday have most demand for Uber trips.



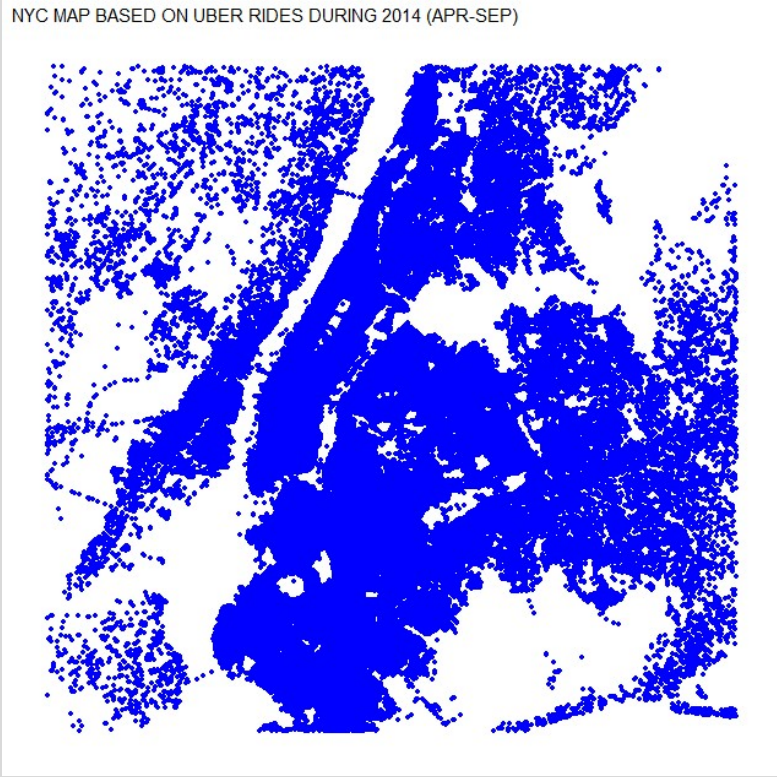
CITY MAP BASED UBER RIDES

Command :-

```
min_lat <- 40.5774
max_lat <- 40.9176
min_long <- -74.15
max_long <- -73.7004
ggplot(data_2014, aes(x=Lon, y=Lat)) +
geom_point(size=1, color = "blue") +
scale_x_continuous(limits=c(min_long, max_long)) +
scale_y_continuous(limits=c(min_lat, max_lat)) +
theme_map() +
ggtitle("NYC MAP BASED ON UBER RIDES DURING 2014 (APR-SEP)")
```

Output :-

This is trip based New York City map. Each dot represents each trip and we can clearly see that middle of city has most number of trips. And out skirt of the city has less number of trips.



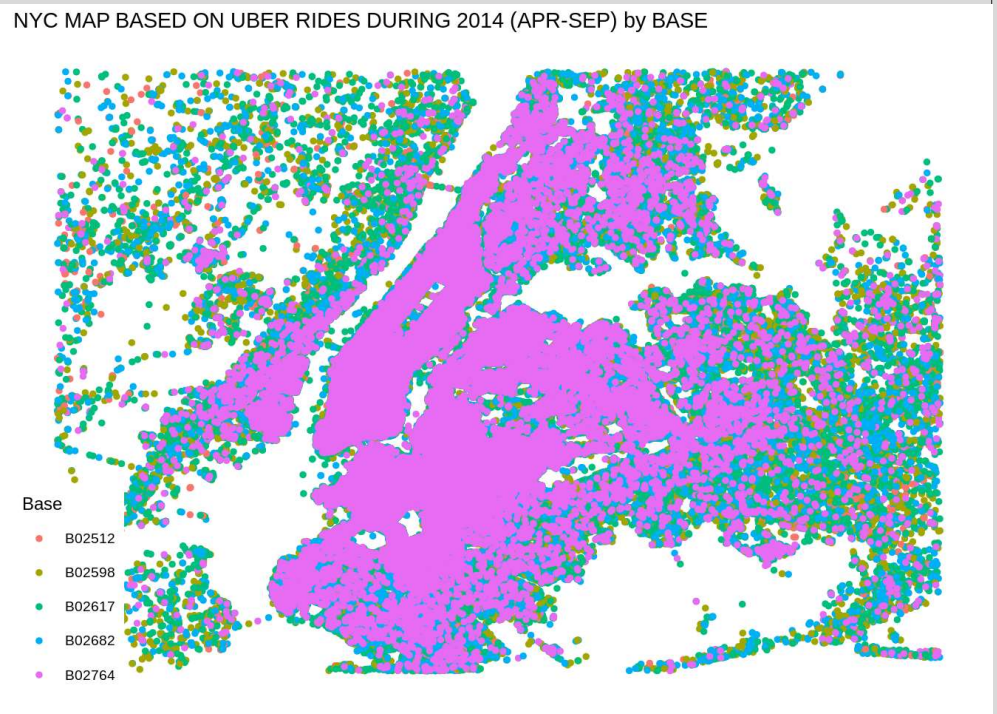
CITY MAP BASED ON UBER RIDES WITH RESPECT TO BASE

Command :-

```
ggplot(data_2014, aes(x=Lon, y=Lat, color = Base)) +
geom_point(size=1) +
scale_x_continuous(limits=c(min_long, max_long)) +
scale_y_continuous(limits=c(min_lat, max_lat)) +
theme_map() +
ggtitle("NYC MAP BASED ON UBER RIDES DURING 2014 (APR-SEP) by BASE")
```

Output :-

This is the City map of trips with respect to base. Different color represent different bases as given in index. So basically trips are dense in middle part of city . And as we interpreted less in outer part of city.



SUMMARY

At the end of the Uber data analysis this project, we observed how to create data visualizations and interpreted so many conclusions through heat map, bar graph and city map . We made use of packages like ggplot2 that allowed us to plot various types of visualizations that pertained to several time-frames of the year. With this, we could conclude how time affected customer trips. Finally, we made a geo plot of New York that provided us with the details of how various users made trips from different bases.

REFERENCES

- https://www.kaggle.com/
- https://drive.google.com/file/d/1emopjfEkTt59jJoBH9L9bSdmlDC4AR87/view
- REFERENCE NOTES

THANK YOU