Uber Data Analysis

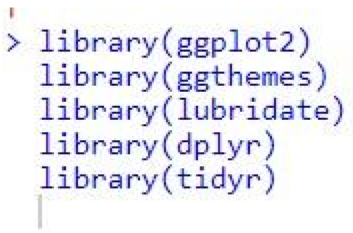
In this Small case Study, I'll be Working with raw trip data collected in 2014 regarding Uber rides. Data storytelling is an important component of Machine Learning through which companies can understand the background of various operations. With the help of visualization, companies can benefit from understanding complex data and gain insights that would help them craft better decisions. Therefore, I'll be showcasing different aspects of trip data that could impact product usage and providing Visualization using R to support and better understand how to achieve better business decisions.

I can only support three datasets, April, May, and June, at a time as I am using the free version of R otherwise the thing crashes simultaneously.

1. Importing the Essential Packages

In the first step, I will import the essential packages we will use in this project. Some of the important packages of R that we will use are

Screenshots:

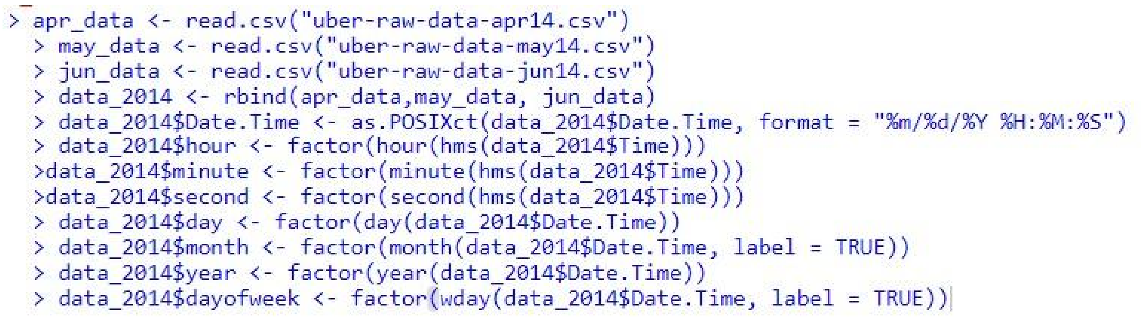


```
> library(ggplot2)
  library(ggthemes)
  library(lubridate)
  library(dplyr)
  library(tidyr)
```

Importing Datasets and creating Data Frames

Now, we will add three CSV files that contain the data from April 2014 to June 2014. We will store these in corresponding data frames like apr_data, may_data, etc. After we have added the files, we will combine all of this data into a single data frame called ‘data_2014’.

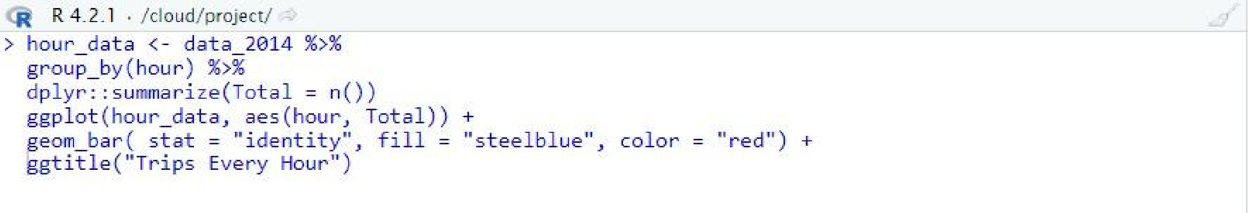
Then, in the next step, we will perform the appropriate formatting of the Date.Time column. Then, we will proceed to create time factors like day, month, and year.



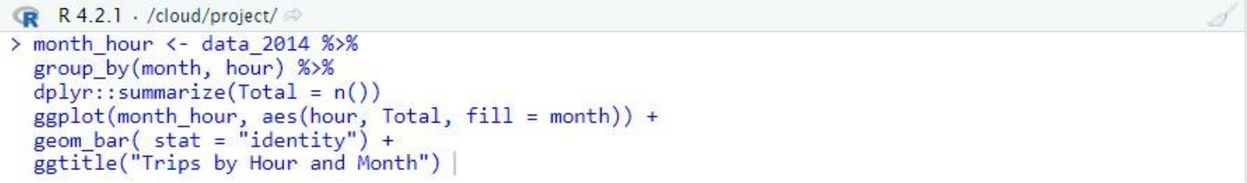
```
> apr_data <- read.csv("uber-raw-data-apr14.csv")
  > may_data <- read.csv("uber-raw-data-may14.csv")
  > jun_data <- read.csv("uber-raw-data-jun14.csv")
  > data_2014 <- rbind(apr_data,may_data, jun_data)
  > data_2014$Date.Time <- as.POSIXct(data_2014$Date.Time, format = "%m/%d/%Y %H:%M:%S")
  > 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)))
  > 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))
```

Plotting the trips by Every hour

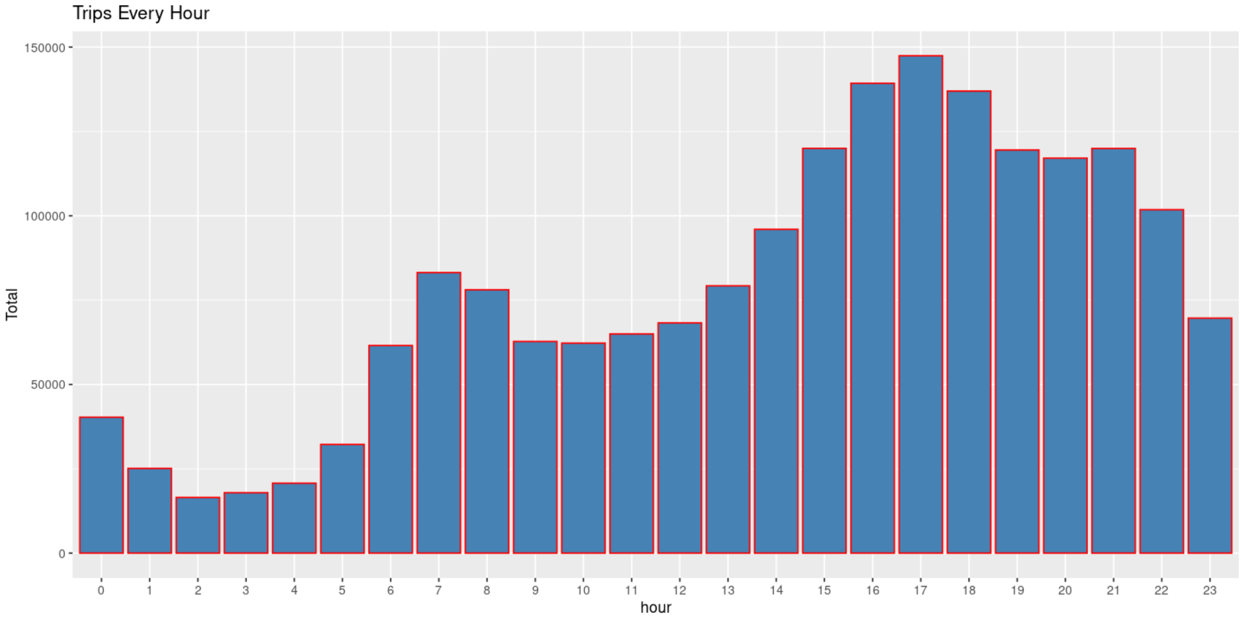
we will use ggplot function to plot the number of trips the passengers made every hour. We will also use dplyr to aggregate our data. In the resulting visualizations, we can understand how the number of passengers fares throughout the day.

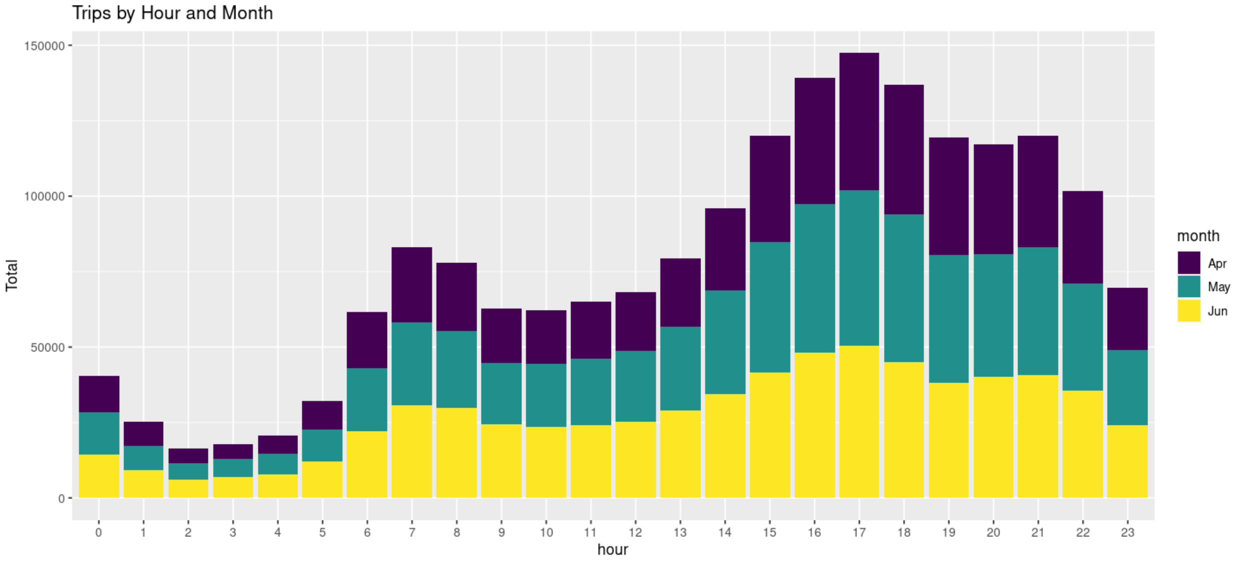


```
> hour_data <- data_2014 %>%
  group_by(hour) %>%
  dplyr::summarize(Total = n())
  ggplot(hour_data, aes(hour, Total)) +
  geom_bar( stat = "identity", fill = "steelblue", color = "red") +
  ggtitle("Trips Every Hour")
```



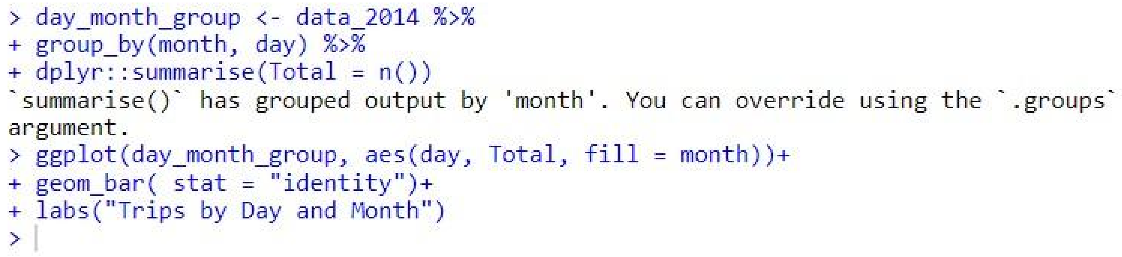
```
> month_hour <- data_2014 %>%
  group_by(month, hour) %>%
  dplyr::summarize(Total = n())
  ggplot(month_hour, aes(hour, Total, fill = month)) +
  geom_bar( stat = "identity") +
  ggtitle("Trips by Hour and Month")
```


Trips Every Hour

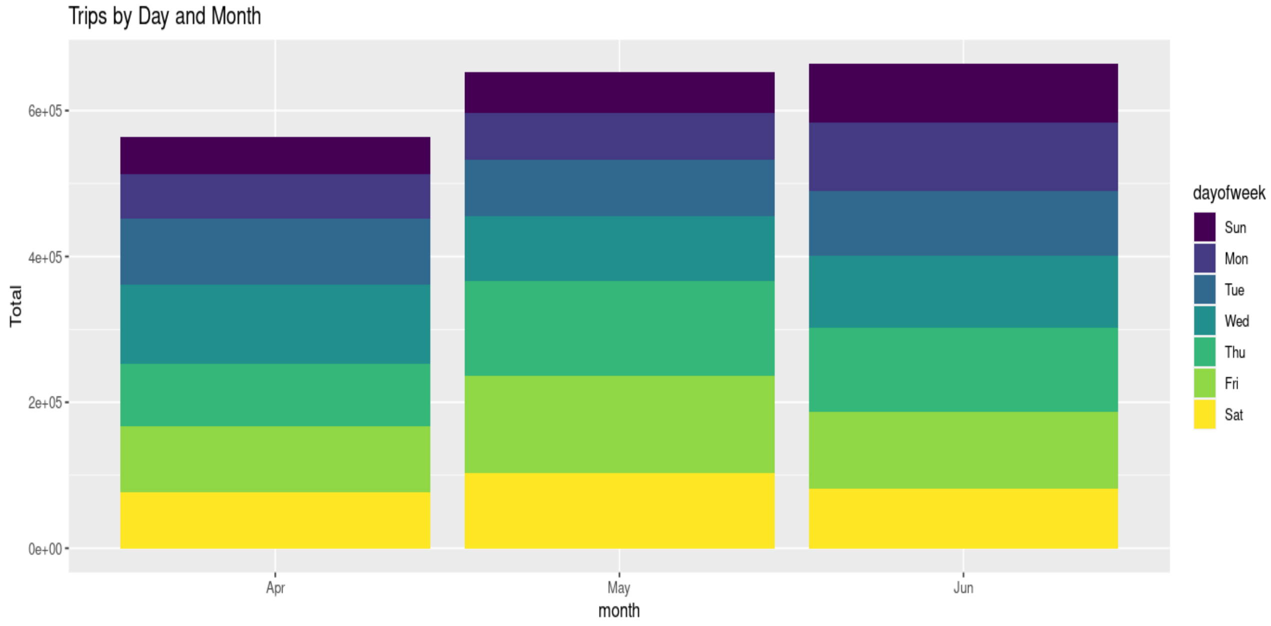

Trips by Hour and Month

We can clearly see that the peak usage is between 4:00 and 10:00 pm in all three months.

Plotting the trips by Day and Month

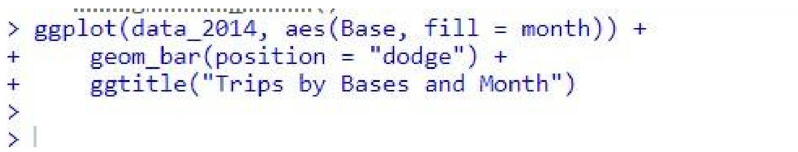


```
> day_month_group <- data_2014 %>%
+ group_by(month, day) %>%
+ dplyr::summarise(Total = n())
`summarise()` has grouped output by 'month'. You can override using the `.groups`
argument.
> ggplot(day_month_group, aes(day, Total, fill = month))+
+ geom_bar( stat = "identity")+
+ labs("Trips by Day and Month")
> 
```

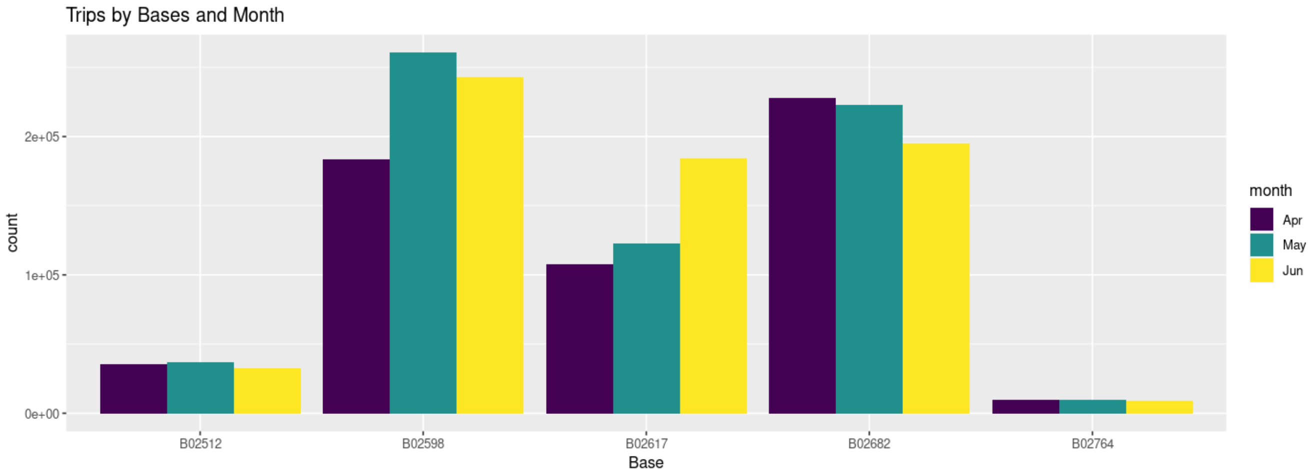


In the above chart, it is clear that in spring people like to travel more on the weekends as compared to the working days.

Plotting the trips by Bases

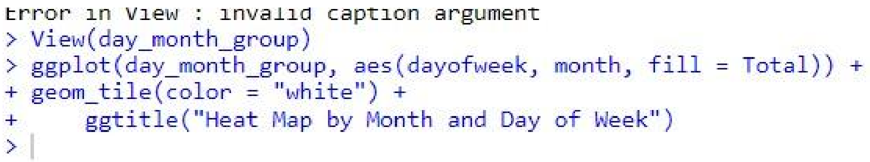


```
> ggplot(data_2014, aes(Base, fill = month)) +
+     geom_bar(position = "dodge") +
+     ggtitle("Trips by Bases and Month")
>
> |
```

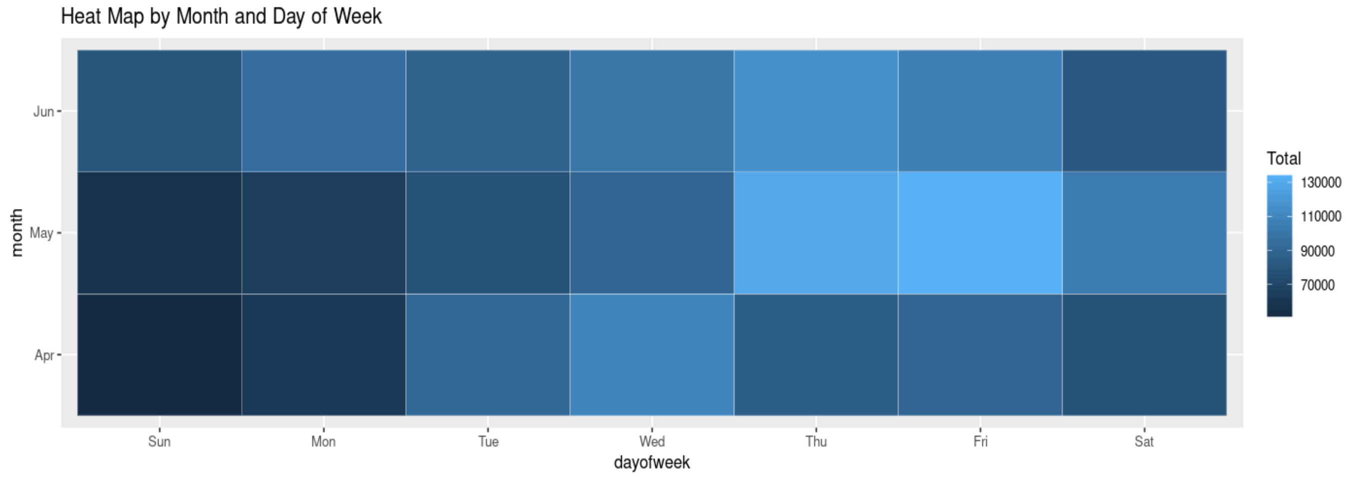


In the following visualization, we plot the number of trips taken by the passengers from each of the bases. There are five bases in all, out of which we observe that B02598 had the highest number of trips. Furthermore, B02682 is clearly busier in spring and Base B02764 is the least Busy.

I would also like to see traffic on a heat map by the day of the week.



```
Error in View : invalid caption argument
> View(day_month_group)
> ggplot(day_month_group, aes(dayofweek, month, fill = Total)) +
+ geom_tile(color = "white") +
+     ggtitle("Heat Map by Month and Day of Week")
> |
```



Summary

With the data available, we can conclude how time and day of the week affected customer trips. Furthermore, we can incentivize drivers to be more active on weekends to maximize readership and consumer engagement.

Resources

Uber Datasets and Github

Prepared By: Gursimrat Singh