Data Analysis R Project Uber Pickup Rides in New York City Between April and September of 2014

Madison Hively

5/7/2021

The business goal of this data analysis project is to use data visualizations to understand Uber pickup rides in New York City between April 2014 and September 2014. This project aims at using data storytelling through data visualizations from the ggplot2 library to gain insight about Uber pickup rides. The dataset includes variables that measure dates, times, latitude, longitude, and a base number to indicate where in New York the passengers were being taken.

This dataset was retrieved from the DataFlair website as part of their series of R projects that encompasses concepts in machine learning, AI, and data science. The link to access this R project can be found here: <https://data-flair.training/blogs/r-data-science-project-uber-data-analysis/>

#set working directory #load and read R packages

knitr::opts\_chunk$set(echo = TRUE)  
getwd()

## [1] "/Users/madisonhively/Documents/Data\_Analytics\_Program"

setwd("/Users/madisonhively/Documents/Data\_Analytics\_Program")  
  
library(ggplot2)  
library(ggthemes)  
library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyr)  
library(scales)

Import and read six .csv files into R, each file contains data from a separate month between May and September of 2014.

#load readr package #import and rename all .csv files

library(readr)

##   
## Attaching package: 'readr'

## The following object is masked from 'package:scales':  
##   
## col\_factor

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")  
jul\_data <- read.csv("uber-raw-data-jul14.csv")  
aug\_data <- read.csv("uber-raw-data-aug14.csv")  
sep\_data <- read.csv("uber-raw-data-sep14.csv")

#combine six datasets into one single dataframe named data\_2014

data\_2014 <- rbind(apr\_data, aug\_data, jul\_data, jun\_data, may\_data, sep\_data)

The data\_2014 dataframe contains 4 columns and 4534327 rows.

#create color vectors that will be used in plots for data visualization

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

#see first 10 rows of dataset

head(data\_2014, 10)

## Date.Time Lat Lon Base  
## 1 4/1/2014 0:11:00 40.7690 -73.9549 B02512  
## 2 4/1/2014 0:17:00 40.7267 -74.0345 B02512  
## 3 4/1/2014 0:21:00 40.7316 -73.9873 B02512  
## 4 4/1/2014 0:28:00 40.7588 -73.9776 B02512  
## 5 4/1/2014 0:33:00 40.7594 -73.9722 B02512  
## 6 4/1/2014 0:33:00 40.7383 -74.0403 B02512  
## 7 4/1/2014 0:39:00 40.7223 -73.9887 B02512  
## 8 4/1/2014 0:45:00 40.7620 -73.9790 B02512  
## 9 4/1/2014 0:55:00 40.7524 -73.9960 B02512  
## 10 4/1/2014 1:01:00 40.7575 -73.9846 B02512

The variables of interest in the data\_2014 dataset are Date.Time, latitude, longitude, and base.

#summary of variable types

str(data\_2014)

## 'data.frame': 4534327 obs. of 4 variables:  
## $ Date.Time: chr "4/1/2014 0:11:00" "4/1/2014 0:17:00" "4/1/2014 0:21:00" "4/1/2014 0:28:00" ...  
## $ Lat : num 40.8 40.7 40.7 40.8 40.8 ...  
## $ Lon : num -74 -74 -74 -74 -74 ...  
## $ Base : chr "B02512" "B02512" "B02512" "B02512" ...

#summary statistics for variables

summary(data\_2014)

## Date.Time Lat Lon Base   
## Length:4534327 Min. :39.66 Min. :-74.93 Length:4534327   
## Class :character 1st Qu.:40.72 1st Qu.:-74.00 Class :character   
## Mode :character Median :40.74 Median :-73.98 Mode :character   
## Mean :40.74 Mean :-73.97   
## 3rd Qu.:40.76 3rd Qu.:-73.97   
## Max. :42.12 Max. :-72.07

Both Date.Time and Base are classified as character variables. The Date.Time variable will be changed in the data preparation section to an appropriate format. Latitude and longitude are classified as numeric variables. Latitude has a minimum of 39.66, maximum of 42.12, and mean of 40.74. Longitude has a minimum of -74.93, maximum of -72.07, and mean of -73.97.

Data Preparation Prepare the data before analysis.

#change Date.Time column to POSIXct format #assign associated time zone to Date.Time column #create additional Time column based on Date.Time column formatting.

data\_2014$Date.Time <- as.POSIXct(data\_2014$Date.Time, format = "%m/%d/%Y %H:%M:%S")  
data\_2014$Date.Time <- ymd\_hms(data\_2014$Date.Time)  
  
data\_2014$Time <- format(as.POSIXct(data\_2014$Date.Time, format = "%m/%d/%Y %H:%M:%S"), format="%H:%M:%S")  
head(data\_2014, 5)

## Date.Time Lat Lon Base Time  
## 1 2014-04-01 00:11:00 40.7690 -73.9549 B02512 00:11:00  
## 2 2014-04-01 00:17:00 40.7267 -74.0345 B02512 00:17:00  
## 3 2014-04-01 00:21:00 40.7316 -73.9873 B02512 00:21:00  
## 4 2014-04-01 00:28:00 40.7588 -73.9776 B02512 00:28:00  
## 5 2014-04-01 00:33:00 40.7594 -73.9722 B02512 00:33:00

#create factors of data for hour, day, month, abd day of the week

data\_2014$Day <- factor(day(data\_2014$Date.Time))  
data\_2014$Month <- factor(month(data\_2014$Date.Time, label = TRUE))  
data\_2014$Day.of.Week <- factor(wday(data\_2014$Date.Time, label = TRUE))  
data\_2014$Hour <- factor(hour(hms(data\_2014$Time)))  
head(data\_2014, 5)

## Date.Time Lat Lon Base Time Day Month Day.of.Week  
## 1 2014-04-01 00:11:00 40.7690 -73.9549 B02512 00:11:00 1 Apr Tue  
## 2 2014-04-01 00:17:00 40.7267 -74.0345 B02512 00:17:00 1 Apr Tue  
## 3 2014-04-01 00:21:00 40.7316 -73.9873 B02512 00:21:00 1 Apr Tue  
## 4 2014-04-01 00:28:00 40.7588 -73.9776 B02512 00:28:00 1 Apr Tue  
## 5 2014-04-01 00:33:00 40.7594 -73.9722 B02512 00:33:00 1 Apr Tue  
## Hour  
## 1 0  
## 2 0  
## 3 0  
## 4 0  
## 5 0

There were no errors within this dataset, therefore minimal preparation was needed before creating any visualizations.

Modeling The modeling used for this data analysis project is data storytelling by using data visualization plots from the ggplot2 library in R. A variety of different plots will be created including bar graphs, heatmap visualization, and a map visualization to gain insight into Uber pickup rides by customers in New York City.

Visualizations - Bar Graphs

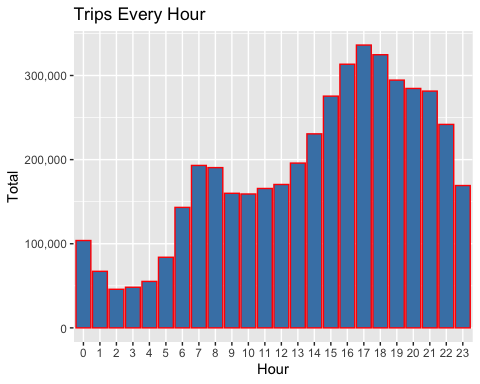
#group data by hour of the day

hour\_data <- data\_2014 %>%  
 group\_by(Hour) %>%  
 summarize(Total = n())  
print(hour\_data)

## # A tibble: 24 x 2  
## Hour Total  
## <fct> <int>  
## 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  
## # … with 14 more rows

#create bar graph of trips by hour in the day

ggplot(hour\_data, aes(Hour, Total)) +  
 geom\_bar(stat = "identity", fill = "steelblue", color = "red") +  
 ggtitle("Trips Every Hour") +  
 scale\_y\_continuous(labels = comma)



The three hours that account for the greatest number of Uber rides are 4 pm, 5 pm, and 6 pm. All three hours account for more than 300,000 rides each. No other hour within the day exceeds 300,000 rides. The three hours that account for the fewest number of Uber rides are 2 am, 3 am, and 4 am, which would be expected since most people are asleep at those times.

#group data by month and hour

month\_hour <- data\_2014 %>%  
 group\_by(Month, Hour) %>%  
 summarize(Total = n())

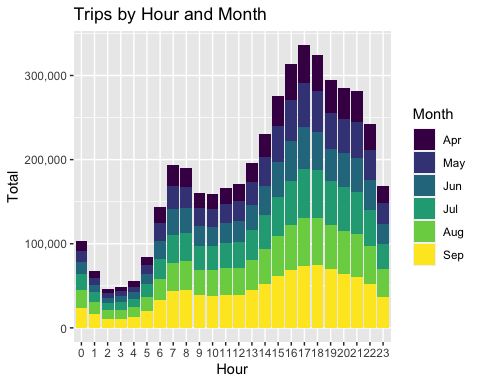
## `summarise()` has grouped output by 'Month'. You can override using the `.groups` argument.

print(month\_hour)

## # A tibble: 144 x 3  
## # Groups: Month [6]  
## Month Hour Total  
## <ord> <fct> <int>  
## 1 Apr 0 11910  
## 2 Apr 1 7769  
## 3 Apr 2 4935  
## 4 Apr 3 5040  
## 5 Apr 4 6095  
## 6 Apr 5 9476  
## 7 Apr 6 18498  
## 8 Apr 7 24924  
## 9 Apr 8 22843  
## 10 Apr 9 17939  
## # … with 134 more rows

#create bar graph of trips by month and hour

ggplot(month\_hour, aes(Hour, Total, fill = Month)) +   
 geom\_bar( stat = "identity") +  
 ggtitle("Trips by Hour and Month") +  
 scale\_y\_continuous(labels = comma)



September appears to have the highest number of rides and April appears to have the lowest number of rides for most hours in the day. Of the three hours that had the highest number of rides, September seems to account for the most while April accounts for the least. This is also true for the three hours that account for the fewest number of rides.

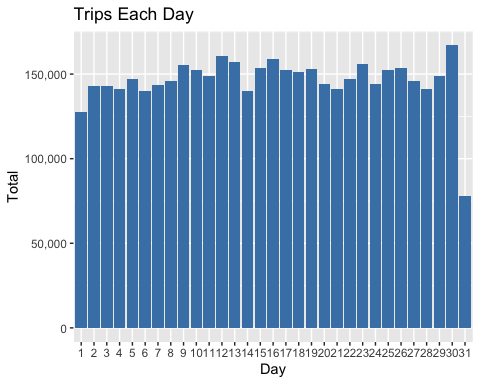
#group data by day

day\_data <- data\_2014 %>%  
 group\_by(Day) %>%  
 summarize(Total =n())  
print(day\_data)

## # A tibble: 31 x 2  
## Day Total  
## <fct> <int>  
## 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  
## # … with 21 more rows

#create bar graph of trips by each day of the month

ggplot(day\_data, aes(Day, Total)) +  
 geom\_bar(stat = "identity", fill = "steelblue") +  
 ggtitle("Trips Each Day") +  
 scale\_y\_continuous(labels = comma)



The number of Uber rides is fairly steady throughout all days in a month. Day 31 has the lowest number of trips which is expected because April, June, and September do not have 31 days in the month. The 30th day has the highest number of Uber rides compared to any other day in the month.

#group data by day of the week and month

day0fweek\_month\_group <- data\_2014 %>%  
 group\_by(Month, Day.of.Week) %>%  
 summarize(Total = n())

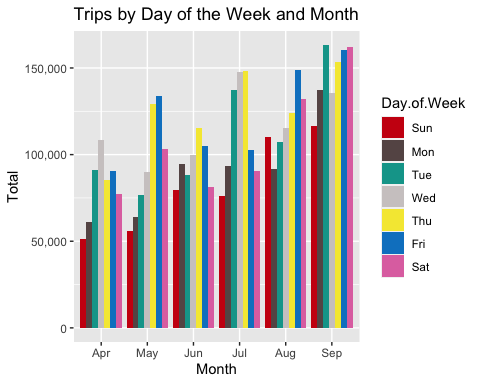
## `summarise()` has grouped output by 'Month'. You can override using the `.groups` argument.

print(day0fweek\_month\_group)

## # A tibble: 42 x 3  
## # Groups: Month [6]  
## Month Day.of.Week Total  
## <ord> <ord> <int>  
## 1 Apr Sun 51251  
## 2 Apr Mon 60861  
## 3 Apr Tue 91185  
## 4 Apr Wed 108631  
## 5 Apr Thu 85067  
## 6 Apr Fri 90303  
## 7 Apr Sat 77218  
## 8 May Sun 56168  
## 9 May Mon 63846  
## 10 May Tue 76662  
## # … with 32 more rows

#create bar graph of trips by day of the week and month

ggplot(day0fweek\_month\_group, aes(Month, Total, fill = Day.of.Week)) +  
 geom\_bar(stat = "identity", position = "dodge") +  
 ggtitle("Trips by Day of the Week and Month") +  
 scale\_y\_continuous(labels = comma) +  
 scale\_fill\_manual(values = colors)



There does not seem to be a specific day of the week that has the greatest number of rides each month. Wednesday was highest in April, Friday was highest in May, Thursday was highest in June, Thursday was highest in July, Friday was highest in August, and Tuesday was highest in September. It is surprising that Tuesday had the greatest number of rides for the month of September. I would have thought that Friday or Saturday would have been the day of the week that had the greatest number of rides for all months, but this was not the case. Sunday and Monday were the two days each week that had the fewest number of Uber rides for all months.

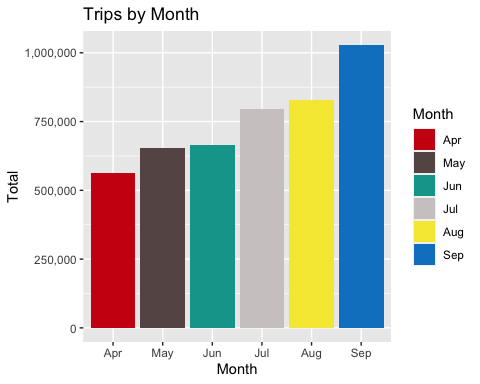
#group data by month

month\_data <- data\_2014 %>%  
 group\_by(Month) %>%  
 summarize(Total = n())  
print(month\_data)

## # A tibble: 6 x 2  
## Month Total  
## <ord> <int>  
## 1 Apr 564516  
## 2 May 652435  
## 3 Jun 663844  
## 4 Jul 796121  
## 5 Aug 829275  
## 6 Sep 1028136

#create bar graph of trips per month

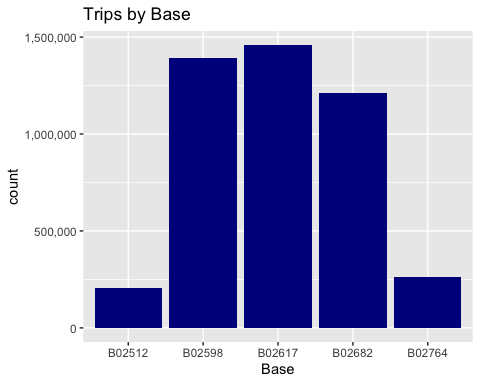
ggplot(month\_data, aes(Month, Total, fill = Month)) +  
 geom\_bar(stat = "identity") +  
 ggtitle("Trips by Month") +  
 scale\_y\_continuous(labels = comma) +  
 scale\_fill\_manual(values = colors)



September has the greatest number of trips, almost double that of April which had the lowest number of trips. The number of rides steadily increases each month and then spikes in the month of September. It is surprising that September has the greatest number of trips, I would have thought that one of the summer months would have had the greatest number of trips.

#create a bar graph of trips by the number of bases

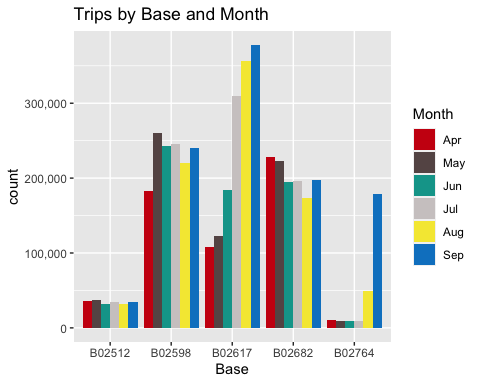
ggplot(data\_2014, aes(Base)) +  
 geom\_bar(fill = "darkblue") +  
 scale\_y\_continuous(labels = comma) +  
 ggtitle("Trips by Base")



Base B02617 had the greatest number of Uber rides of all five bases. Bases B02598, B02617, and B02682 have almost four to five times the number of Uber rides than bases B02512 and B02764, which appear to be much less popular.

#create a bar graph of trips by base and month

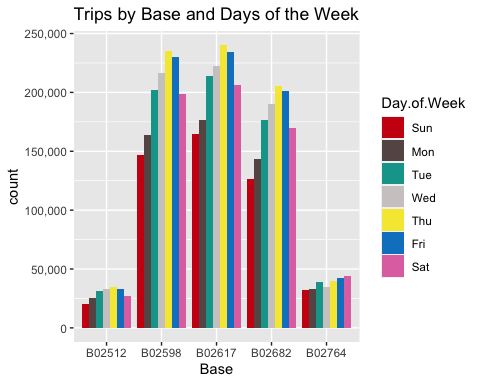
ggplot(data\_2014, aes(Base, fill = Month)) +  
 geom\_bar(position = "dodge") +  
 scale\_y\_continuous(labels = comma) +  
 ggtitle("Trips by Base and Month") +  
 scale\_fill\_manual(values = colors)



The the amount of Uber rides for each month does not have a clear pattern across all base locations. Bases B02512, B02598, and B02682 have relatively similar number of Uber rides each month. Bases B02617 and B02764 have obvious months that had greater number of Uber rides, which were July, August and September for B02617 and August and September for B02764.

#create bar graph of trips by bases and days of the week

ggplot(data\_2014, aes(Base, fill = Day.of.Week)) +  
 geom\_bar(position = "dodge") +  
 scale\_y\_continuous(labels = comma) +  
 ggtitle("Trips by Base and Days of the Week") +  
 scale\_fill\_manual(values = colors)



The bases B02598, B02617, and B02682 all have a similar pattern with the greatest number of rides on Thursday and the fewest number of rides on Sunday. Base B02512 the number of Uber rides increases toward the beginning of the week and decreases toward the end of the week. Base B02764 has a steady increase of Uber rides throughout the week and has the greatest number of rides on Saturday.

Visualizations - Heatmaps

#factor data by day and hour

day\_and\_hour <- data\_2014 %>%  
 group\_by(Day, Hour) %>%  
 summarize(Total = n())

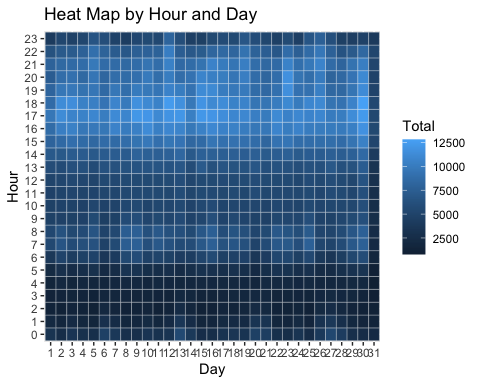
## `summarise()` has grouped output by 'Day'. You can override using the `.groups` argument.

print(day\_and\_hour)

## # A tibble: 744 x 3  
## # Groups: Day [31]  
## Day Hour Total  
## <fct> <fct> <int>  
## 1 1 0 3247  
## 2 1 1 1982  
## 3 1 2 1284  
## 4 1 3 1331  
## 5 1 4 1458  
## 6 1 5 2171  
## 7 1 6 3717  
## 8 1 7 5470  
## 9 1 8 5376  
## 10 1 9 4688  
## # … with 734 more rows

#create heatmap by hour and day

ggplot(day\_and\_hour, aes(Day, Hour, fill = Total)) +  
 geom\_tile(color = "white") +  
 ggtitle("Heat Map by Hour and Day")



The most popular hours for Uber rides are in the early evening hours, specifically 4pm, 5pm, and 6pm. The early morning hours are the least popular times for Uber rides, specifically 1 am, 2 am, 3 am, and 4 am. The amount of pickups increases at 7 am and 8 am, as I would assume people are going into work at those times and need transportation.

#factor data by day and month

day\_month\_group <- data\_2014 %>%  
 group\_by(Month, Day) %>%  
 summarize(Total = n())

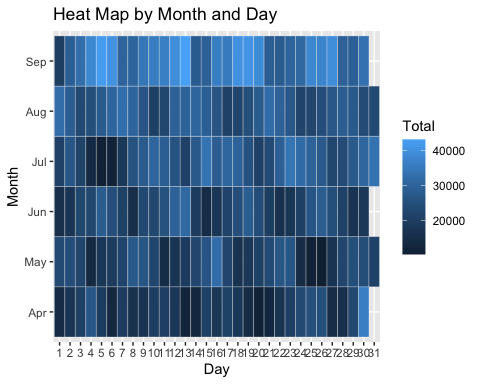
## `summarise()` has grouped output by 'Month'. You can override using the `.groups` argument.

print(day\_month\_group)

## # A tibble: 183 x 3  
## # Groups: Month [6]  
## Month Day Total  
## <ord> <fct> <int>  
## 1 Apr 1 14546  
## 2 Apr 2 17474  
## 3 Apr 3 20701  
## 4 Apr 4 26714  
## 5 Apr 5 19521  
## 6 Apr 6 13445  
## 7 Apr 7 19550  
## 8 Apr 8 16188  
## 9 Apr 9 16843  
## 10 Apr 10 20041  
## # … with 173 more rows

#create heatmap by month and day

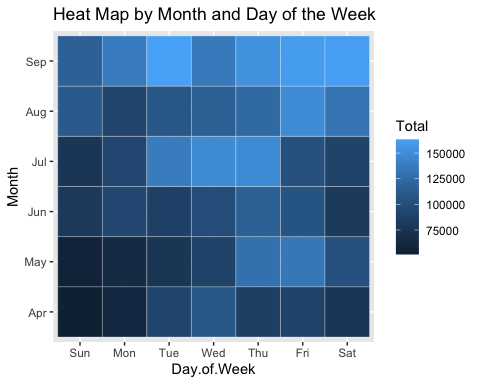
ggplot(day\_month\_group, aes(Day, Month, fill = Total)) +  
 geom\_tile(color = "white") +  
 ggtitle("Heat Map by Month and Day")



September appears to have the greatest number of rides and April appears to have the fewest number of rides for days in a month.

#create heatmap by Month and Day of Week

ggplot(day0fweek\_month\_group, aes(Day.of.Week, Month, fill = Total)) +  
 geom\_tile(color = "white") +  
 ggtitle("Heat Map by Month and Day of the Week")



Both Thursday and Friday appear to be the two days of the week that are most popular for all months. Sunday and Monday appear to be the two days of the week that are the least popular for all months. April overall has the fewest number of rides, while September has the greatest number of rides during a week.

#factor data by base and month

month\_base <- data\_2014 %>%  
 group\_by(Base, Month) %>%  
 summarize(Total = n())

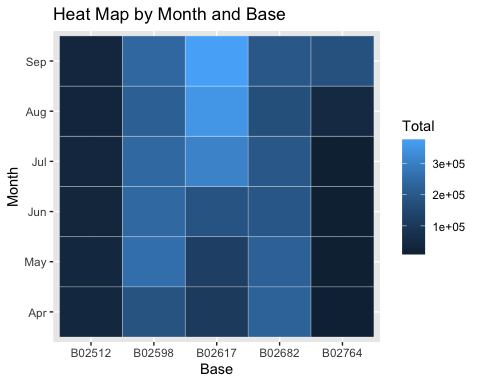
## `summarise()` has grouped output by 'Base'. You can override using the `.groups` argument.

print(month\_base)

## # A tibble: 30 x 3  
## # Groups: Base [5]  
## Base Month Total  
## <chr> <ord> <int>  
## 1 B02512 Apr 35536  
## 2 B02512 May 36765  
## 3 B02512 Jun 32509  
## 4 B02512 Jul 35021  
## 5 B02512 Aug 31472  
## 6 B02512 Sep 34370  
## 7 B02598 Apr 183263  
## 8 B02598 May 260549  
## 9 B02598 Jun 242975  
## 10 B02598 Jul 245597  
## # … with 20 more rows

#create heatmap by base and month

ggplot(month\_base, aes(Base, Month, fill = Total)) +  
 geom\_tile(color = "white") +  
 ggtitle("Heat Map by Month and Base")



Base B02617 gains more rides as the months progress and is the most popular base for all months. Base B02512 is the leas popular base for all months.

#factor data by base and days of the week

day0fweek\_base <- data\_2014 %>%  
 group\_by(Base, Day.of.Week) %>%  
 summarize(Total = n())

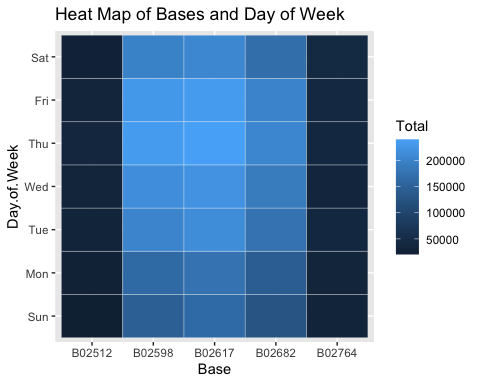
## `summarise()` has grouped output by 'Base'. You can override using the `.groups` argument.

print(day0fweek\_base)

## # A tibble: 35 x 3  
## # Groups: Base [5]  
## Base Day.of.Week Total  
## <chr> <ord> <int>  
## 1 B02512 Sun 20490  
## 2 B02512 Mon 25460  
## 3 B02512 Tue 31670  
## 4 B02512 Wed 32929  
## 5 B02512 Thu 35032  
## 6 B02512 Fri 33319  
## 7 B02512 Sat 26773  
## 8 B02598 Sun 146652  
## 9 B02598 Mon 163542  
## 10 B02598 Tue 202378  
## # … with 25 more rows

#create heatmap by days of week and month

ggplot(day0fweek\_base, aes(Base, Day.of.Week, fill = Total)) +  
 geom\_tile(color = "white") +  
 ggtitle("Heat Map of Bases and Day of Week")



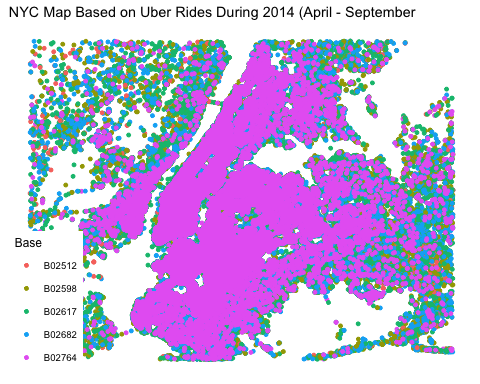
For bases B02598, B02617, and B02682 the number of rides is greatest during Tuesday, Wednesday, Thursday, and Friday. Bases B02512 and B02764 are not popular bases among Uber customers.

Visualization - Map

#create map visualization

min\_lat <- 40.5774  
max\_lat <- 40.9176  
min\_long <- -74.15  
max\_long <- -73.7004  
  
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 (April - September")

## Warning: Removed 71701 rows containing missing values (geom\_point).



Rides from base B02764 occur the greatest number of times on the New York City map. This base dominates the area and seems to be most popular with Uber customers in New York City. This base location is less popular as you move away toward the outskirts of the city.

Summary

This data analysis project went into depth with creating different types of data visualizations which included bar graphs, heatmaps, and a normal map. By creating these visualization I was able to gain insight into Uber pickup rides in New York City between April 2014 and September 2014. It was determined that April was the least popular month and September was the most popular month for Uber pickup rides. There was a consistent pattern in time of day when Uber pickup rides were at a highest and lowest. The most popular times were around 4 pm, 5 pm, and 6pm, while the least popular times were 2 am, 3 am, and 4 am for all days in the month. There was no consistent pattern with popularity in days of the week for Uber pickup rides for each month. Base B02764 was the most popular of all five base locations by month and days of the week.