Análisis y Visualización

Uber Dataset

Álvaro Barrio Hernández (alvaro.barrio.hernandez@gmail.com)

Código: GitHub

la narración de datos es un componente importante del aprendizaje automático a través del cual las empresas pueden comprender los antecedentes de varias operaciones. Con la ayuda de la visualización, las empresas pueden aprovechar el beneficio de comprender los datos complejos y obtener información que les ayudaría a tomar decisiones. Aprenderemos a implementar ggplot2 en el conjunto de datos de Uber Pickups y, al final, dominaremos el arte de la visualización de datos en R.

---

1. Importación de los paquetes esenciales

En el primer paso de nuestro proyecto R, importaremos los paquetes esenciales que usaremos en este proyecto de análisis de datos uber. Algunas de las bibliotecas importantes de R que usaremos son:

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

- **ggplot2:** Esta es la columna vertebral de este proyecto. ggplot2 es la biblioteca de visualización de datos más popular que se usa más ampliamente para crear gráficos de visualización estética.
- **ggthemes:** Eseo es más un complemento de nuestra biblioteca principal ggplot2. Con esto, podemos crear mejores temas y escalas adicionales con el paquete ggplot2 convencional.
- **lubridate:** Nuestro conjunto de datos involucra varios marcos de tiempo. Para comprender nuestros datos en categorías de tiempo separadas, haremos uso del paquete lubridate.
- **dplyr**: Este paquete es la lengua franca de la manipulación de datos en R.
- **tidyr:** Este paquete le ayudará a ordenar sus datos. El principio básico de tidyr es ordenar las columnas donde cada variable está presente en una columna, cada observación está representada por una fila y cada valor representa una celda.
- **DT:** Con la ayuda de este paquete, podremos interactuar con la biblioteca JavaScript llamada - Datatables.
- **scales:** Con la ayuda de escalas gráficas, podemos mapear automáticamente los datos a las escalas correctas con ejes y leyendas bien colocados

---

2. Creación de vector de colores para implementar en nuestras parcelas

En este paso del proyecto de ciencia de datos, crearemos un vector de nuestros colores que se incluirá en nuestras funciones de trazado:

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

3. Leer los datos en sus variables designadas

Ahora, leeremos varios archivos csv que contienen los datos de abril de 2014 a septiembre de 2014. Los almacenaremos en los marcos de datos correspondientes como apr_data, may_data, etc. Después de leer los archivos, combinaremos todos estos datos en un marco de datos único llamado 'data_2014'.

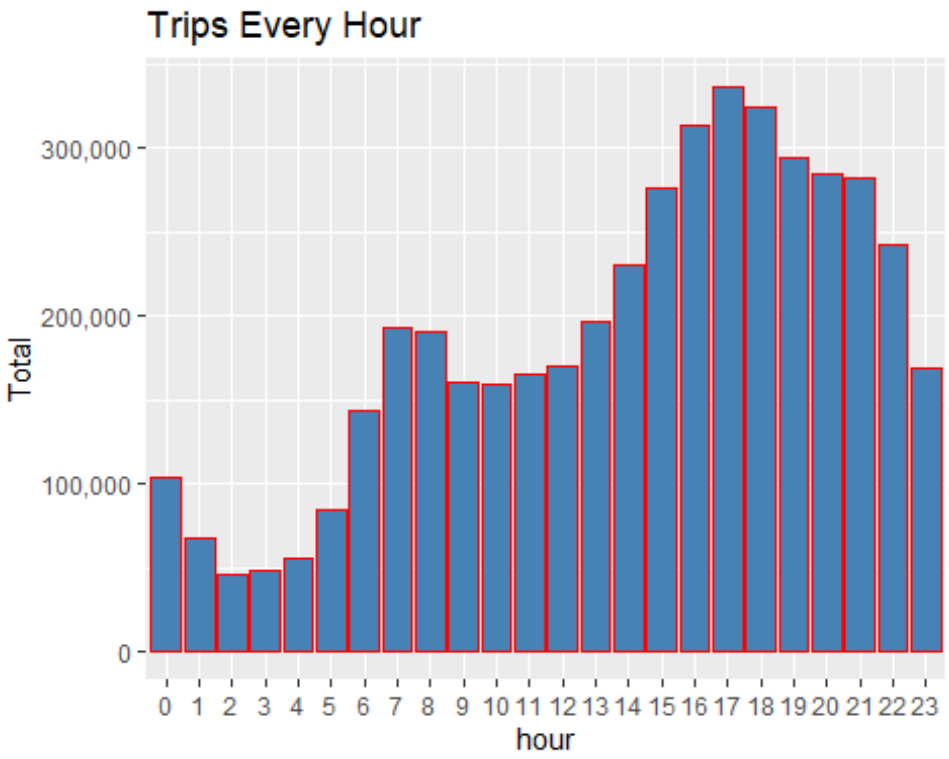
Luego, en el siguiente paso, realizaremos el formateo apropiado de la columna Date.Time. Por último, procederemos a crear factores de objetos de tiempo como día, mes, año, etc.

```
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")

data_2014 <- rbind(apr_data,may_data, jun_data, jul_data, aug_data, sep_d
ata)

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)))
```

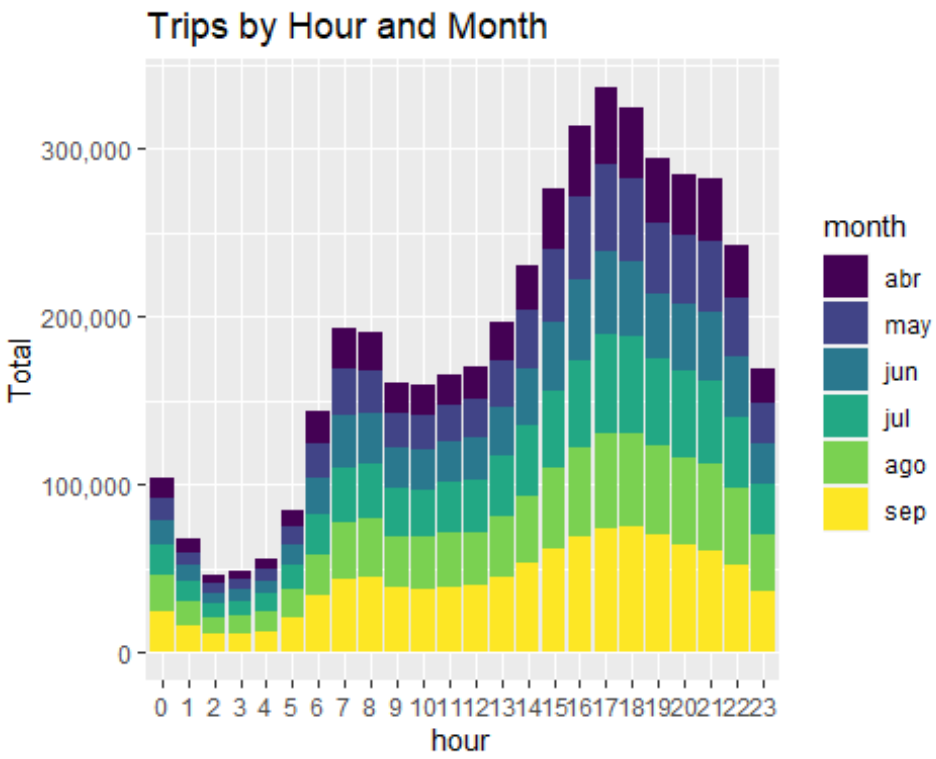
4. Trazar los viajes por horas en un día

En el siguiente paso o proyecto R, usaremos la función ggplot para trazar el número de viajes que los pasajeros realizaron en un día. También usaremos dplyr para agregar nuestros datos. En las visualizaciones resultantes, podremos entender cómo se van moviendo los pasajeros a lo largo del día. Observamos que el número de viajes es mayor en la tarde alrededor de las 5:00 y las 6:00 PM.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```



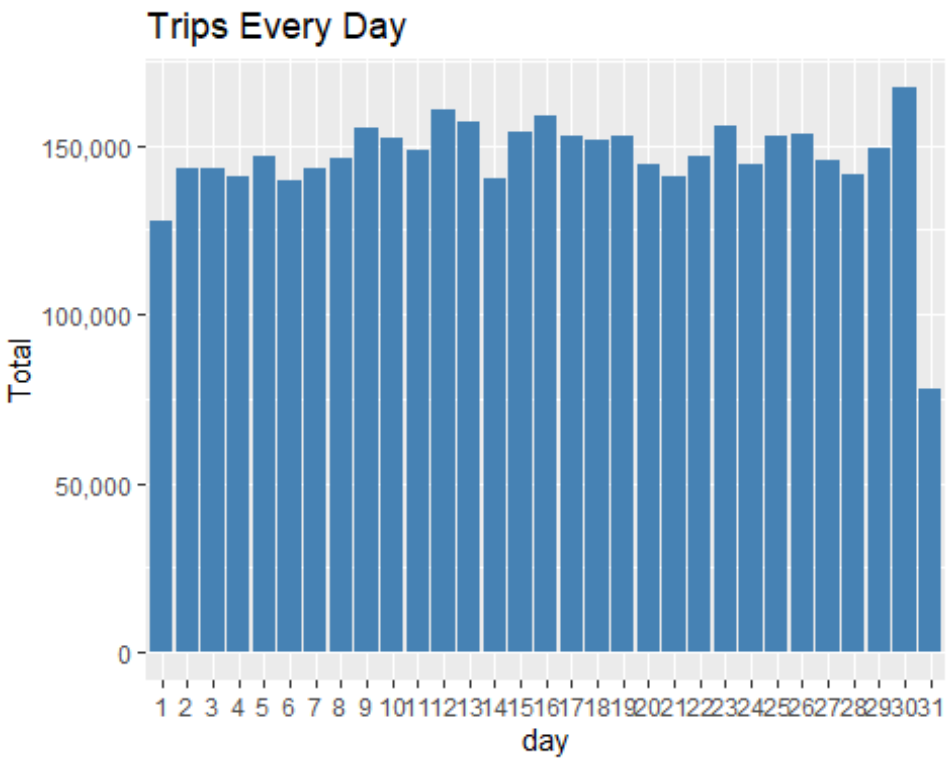

```
## `summarise()` regrouping output by 'month' (override with `.groups` ar
gument)
```



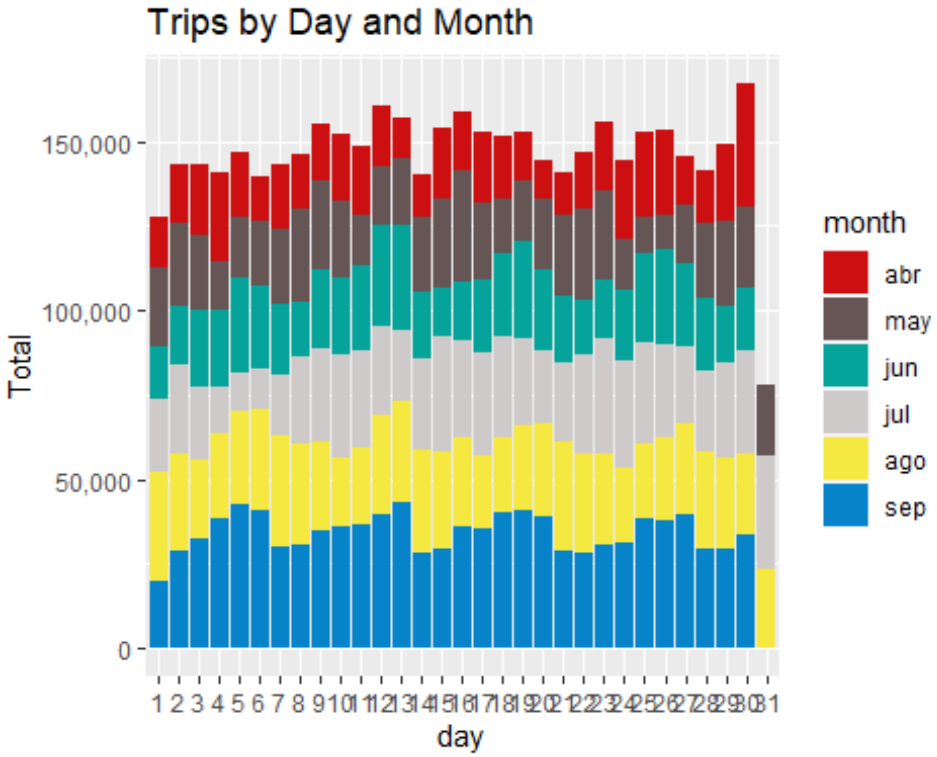
5. Trazar datos por viajes durante todos los días del mes

En esta sección del proyecto DataFlair R, aprenderemos cómo trazar nuestros datos en función de todos los días del mes. Observamos a partir de la visualización resultante que el 30 del mes tuvo los viajes más altos en el año, lo que se debe principalmente al mes de abril.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```



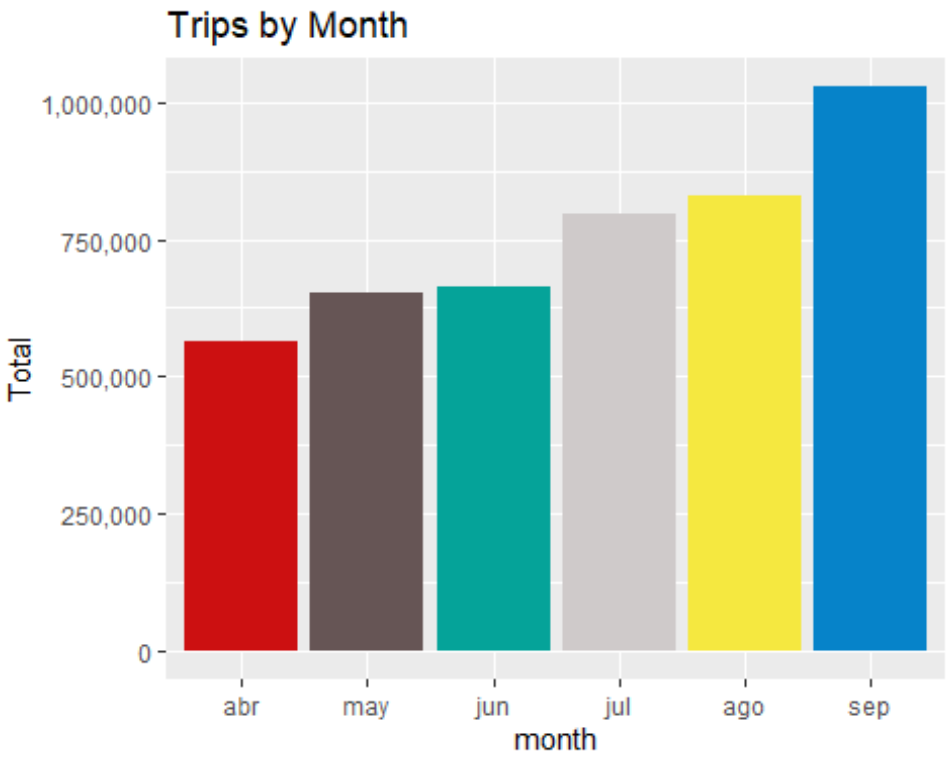

```
## `summarise()` regrouping output by 'month' (override with `.groups` ar
gument)
```



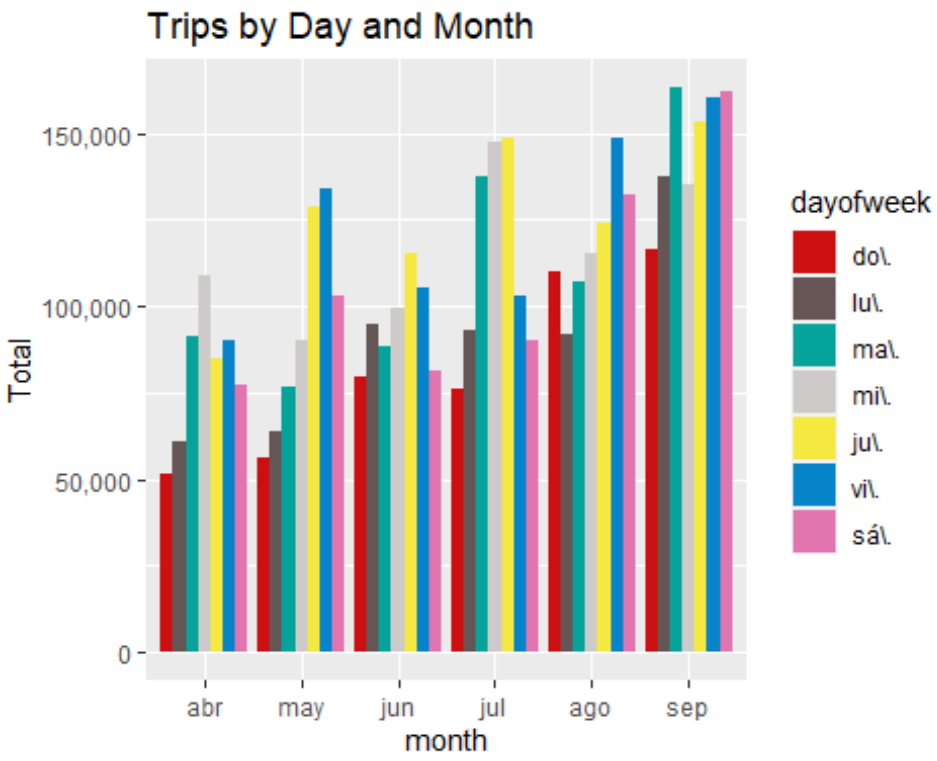
6. Número de viajes que se realizan durante meses en un año

En esta sección, visualizaremos la cantidad de viajes que se están realizando cada mes del año. En la visualización de salida, observamos que la mayoría de los viajes se realizaron durante el mes de septiembre. Además, también obtenemos informes visuales del número de viajes que se realizaron todos los días de la semana.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

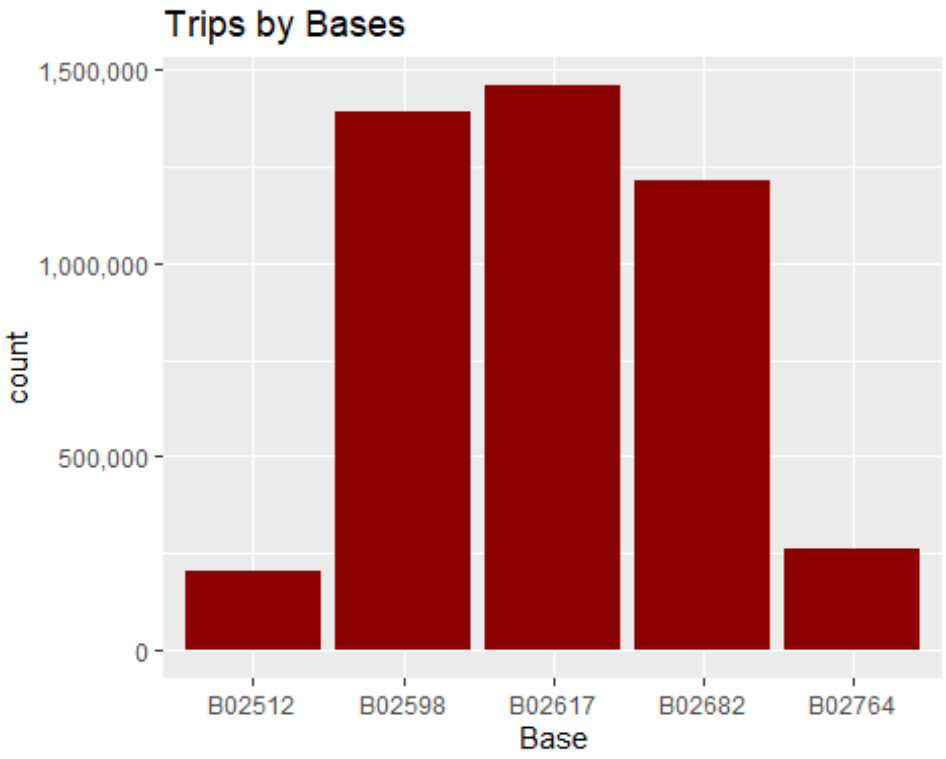


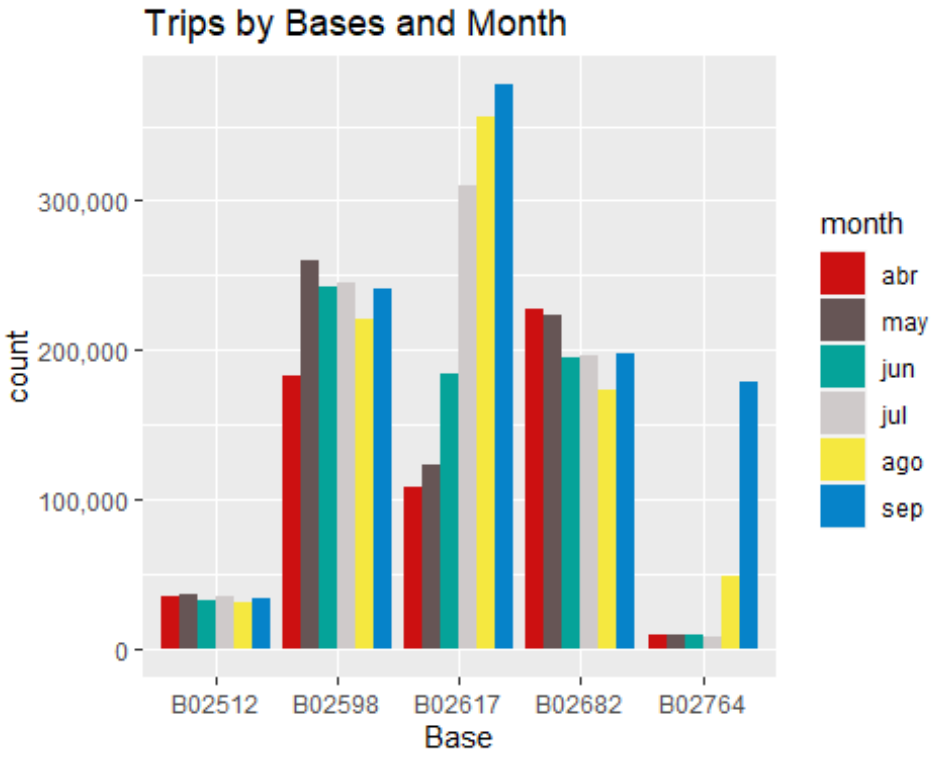

```
## `summarise()` regrouping output by 'month' (override with `.groups` ar
gument)
```

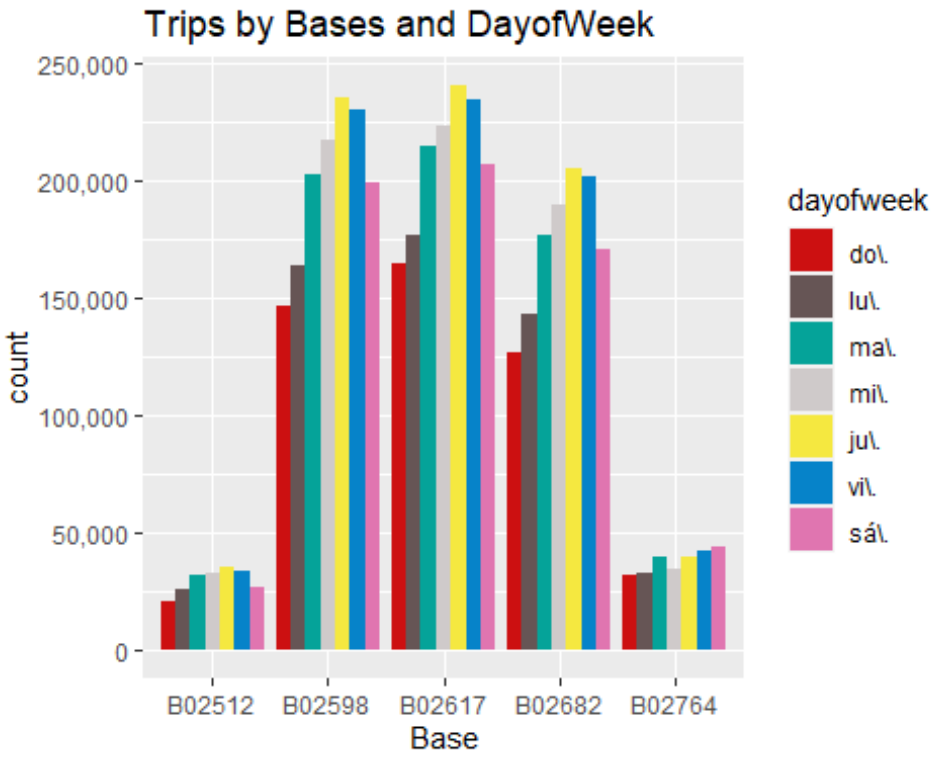


7. Averiguar el número de viajes por bases

En la siguiente visualización, graficamos el número de viajes que han realizado los pasajeros desde cada una de las bases. Hay cinco bases en total, de las cuales observamos que B02617 tuvo el mayor número de viajes. Además, esta base tuvo el mayor número de viajes en el mes B02617. El jueves se observaron los viajes más altos en las tres bases: B02598, B02617, B02682.


Trips by Bases
1,500,000
1,000,000
500,000
0
count
B02512
B02598
B02617
B02682
B02764
Base


Trips by Bases and Month
300,000
200,000
100,000
0
count
B02512
B02598
B02617
B02682
B02764
Base
month
abr
may
jun
jul
ago
sep



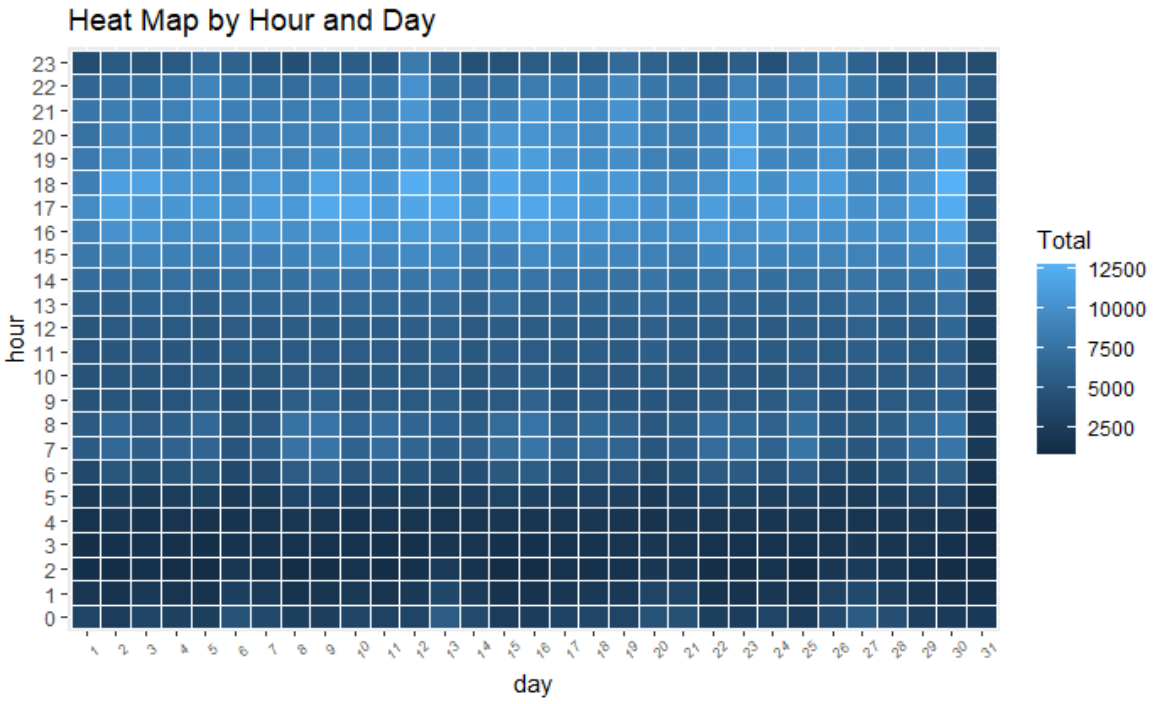
---

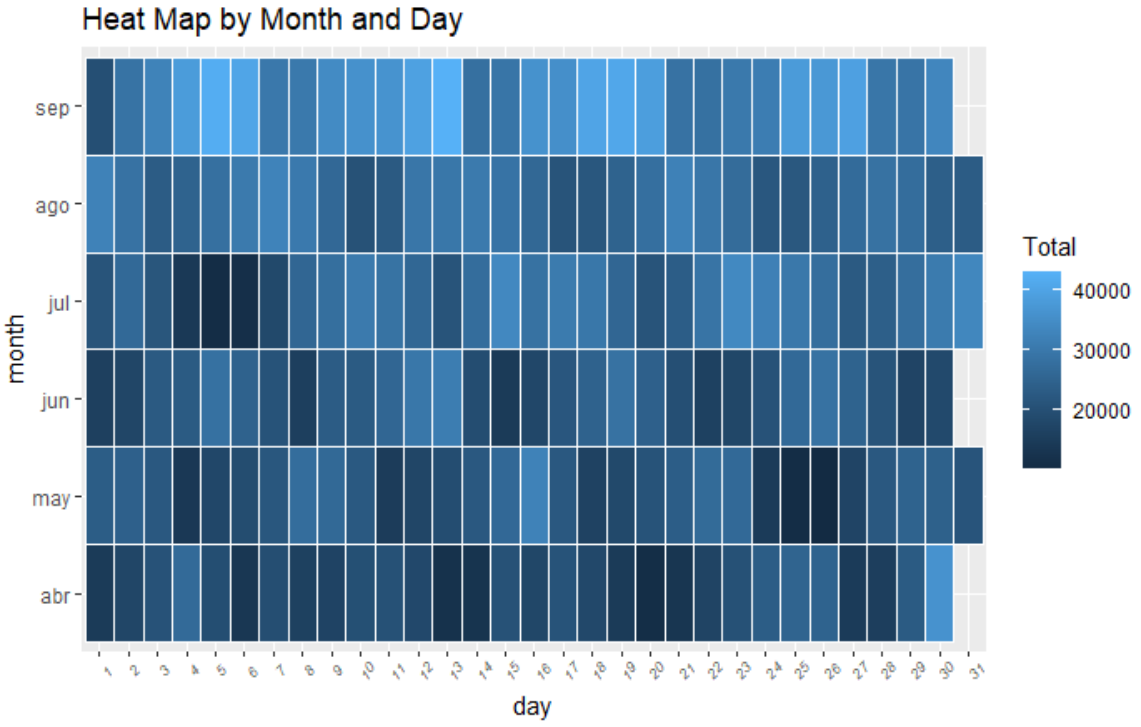
8. Creación de una visualización de mapa de calor de día, hora y mes

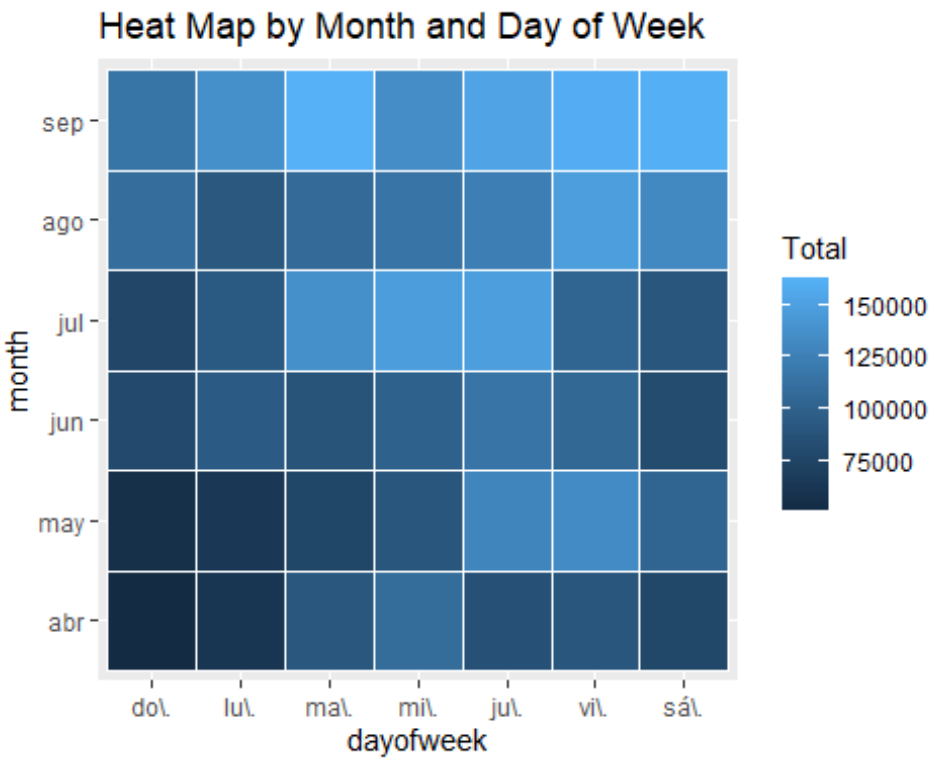
En esta sección, aprenderemos cómo trazar mapas de calor usando ggplot ().
Trazaremos cinco gráficos de mapas de calor:

- Primero, trazaremos el mapa de calor por hora y día.
- En segundo lugar, trazaremos el mapa de calor por mes y día.
- En tercer lugar, un mapa de calor por mes y día de la semana.
- Cuarto, un mapa de calor que delimita Mes y Bases.
- Finalmente, trazaremos el mapa de calor, por bases y día de la semana.

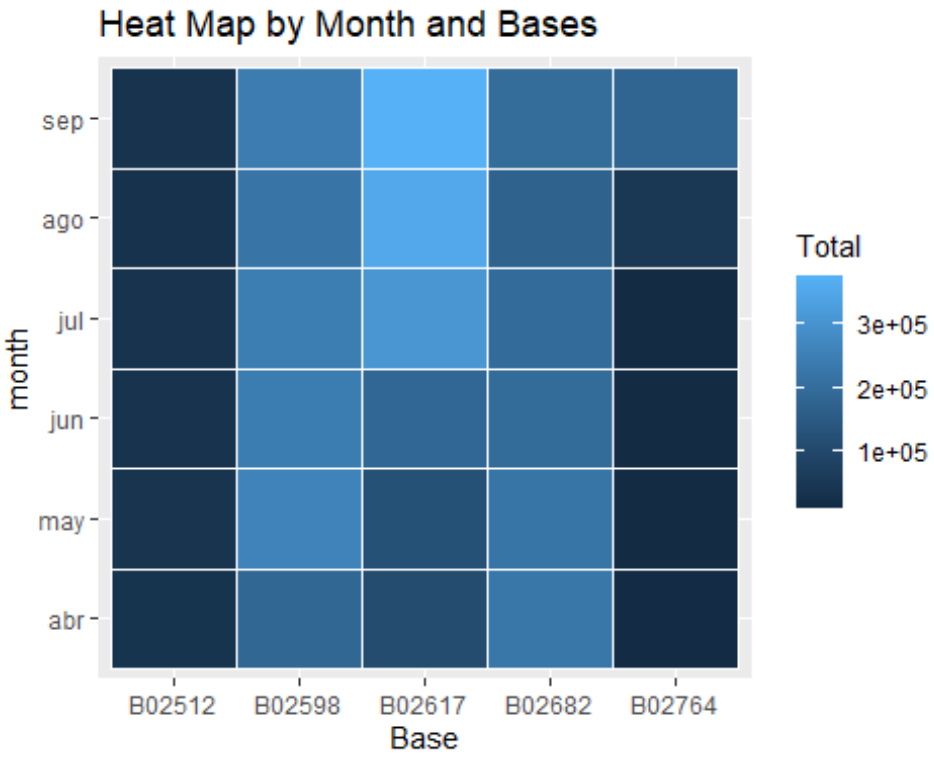
```
## `summarise()` regrouping output by 'day' (override with `.groups` argu
ment)
```

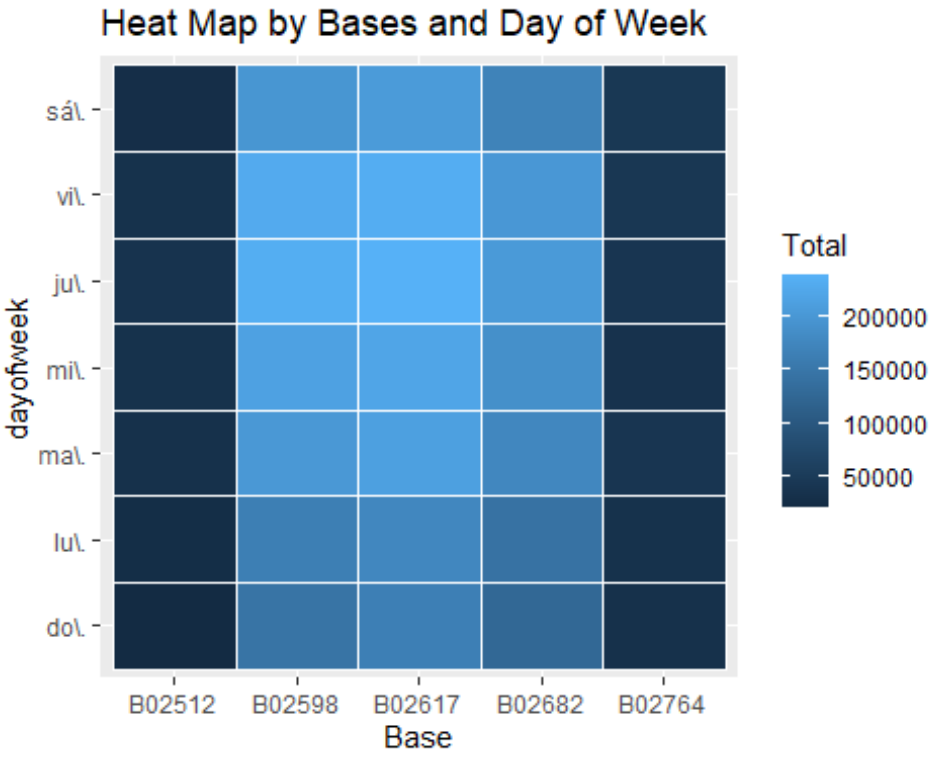

Heat Map by Hour and Day
hour
day
Total
12500
10000
7500
5000
2500


Heat Map by Month and Day
month
day
Total
40000
30000
20000
sep
ago
jul
jun
may
abr



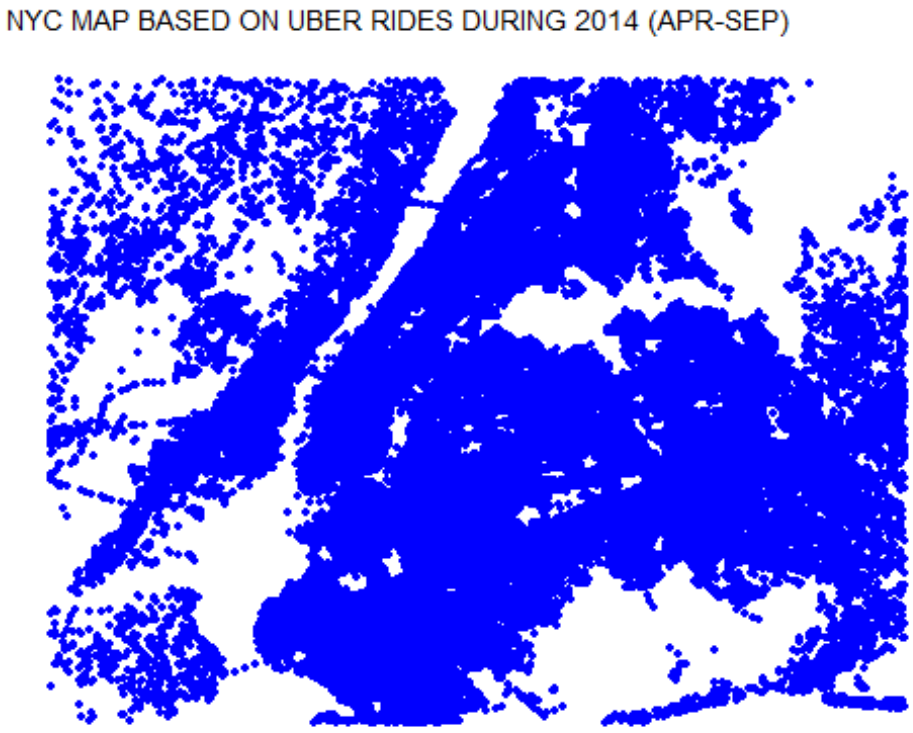

```
## `summarise()` regrouping output by 'Base' (override with `.groups` arg
ument)

## `summarise()` regrouping output by 'Base' (override with `.groups` arg
ument)
```

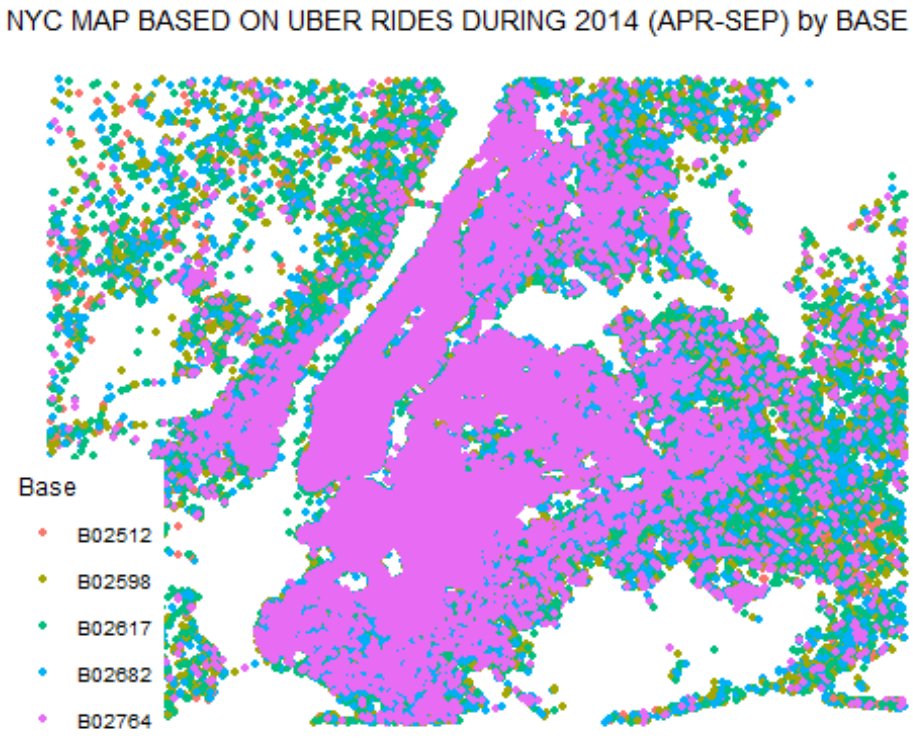

Heat Map by Month and Bases
sep
ago
jul
jun
may
abr
month
B02512
B02598
B02617
B02682
B02764
Base
Total
3e+05
2e+05
1e+05


Heat Map by Bases and Day of Week
sá\.
vi\.
ju\.
mi\.
ma\.
lu\.
do\.
dayofweek
B02512
B02598
B02617
B02682
B02764
Base
Total
200000
150000
100000
50000

9. Creación de una visualización de mapas de atracciones en Nueva York

En la sección final, visualizaremos las atracciones en la ciudad de Nueva York mediante la creación de una trama geográfica que nos ayudará a visualizar las atracciones durante 2014 (Abril - Septiembre) y por las bases en el mismo período.





Con este proyecto en R de análisis de datos de Uber, observamos cómo se crean visualizaciones de datos. Hicimos uso de paquetes como ggplot2 que nos permitieron trazar varios tipos de visualizaciones que pertenecían a varios períodos de tiempo del año. Finalmente, realizamos una representación del mapa de Nueva York que nos proporcionó los detalles de cómo varios usuarios realizaban viajes desde diferentes bases.

Pueden encontrar todo el código disponible en mi GitHub

¡Saludos!