Caso Práctico del Certificado en Análisis de Datos de Google: Cyclistic

Sam Paolo Michael Pérez Pérez

2023-08-31

I. Introducción

El sector de las bicicletas compartidas ha experimentado un auge en los últimos años, debido a sus beneficios económicos, sociales y ambientales. Sin embargo, también se enfrenta retos, como la competencia y la fidelización de clientes. En este contexto, se plantea el siguiente caso práctico, correspondiente al último curso del certificado en análisis de datos de Google.

Escenario

El caso se basa en la empresa ficticia Cyclistic, que ofrece el servicio de bicicletas compartidas en la ciudad de Chicago, Estados Unidos. Su misión es proporcionar una alternativa de transporte accesible, saludable y sostenible a los habitantes y visitantes de la ciudad. Sus bicicletas eléctricas, clásicas y acopladas están georreferenciadas y distribuidas en más de 600 estaciones, lo que permite a los usuarios desbloquearlas en una estación y devolverlas en cualquier otra estación cercana.

Cyclistic cuenta con planes de precios flexibles, para satisfacer las necesidades de los usuarios. Los clientes que adquieren pases de un solo viaje o pases de un día completo son conocidos como **Ciclistas Casuales**, mientras que aquellos que optan por las membresías anuales son denominados **Miembros de Cyclistic**.

La directora de mercadotecnia considera que el éxito futuro de la empresa depende de maximizar la cantidad de miembros anuales, a partir de conseguir que los usuarios casuales adquieran las membresías. Con el fin de respaldar las decisiones empresariales, se requiere analizar datos históricos de los últimos 12 meses.

La pregunta de investigación que guía el análisis es la siguiente:

¿En qué se diferencian los miembros anuales y los ciclistas casuales en cuanto al uso de las bicicletas de Cyclistic?

Para responder a esta pregunta, se utilizaron las siguientes herramientas y técnicas: hojas de cálculo para un análisis exploratorio mensual, RStudio para limpiar y analizar los datos, y Tableau para visualizar los hallazgos obtenidos.

El análisis se centró en las variables relacionadas con las preferencias de los usuarios, incluida la frecuencia, duración, hora y el día de los viajes.

Durante el análisis de las preferencias en el uso de bicicletas para los usuarios casuales y miembros, se encontraron valores extremos en la duración de los viajes de bicicletas. Se utilizó el rango intercuartil para determinar qué valores se considerarían **atípicos**. El resultado fue, que los viajes en bicicleta con duración mayor a **36 minutos con 19 segundos** son considerados atípicos.

Sin perder el enfoque de la estrategia de Cyclistic de convertir a usuarios casuales en miembros y encontrar las diferencias entre estos dos tipos de usuarios, se realizó una comparación entre los siguientes conjuntos de datos:
1. **Todos los datos**: Es el marco original que incluye tanto los datos no atípicos como los atípicos (nombre en RStudio: "all_data12_v2").
2. **Datos no atípicos**: Es el marco que excluye los datos atípicos (nombre en RStudio: "datos_sin_atípicos").

Al final de este documento se encuentra disponible un enlace a una visualizacion creada en "Tableau Public". En esta visualización, se presentan los resultados de los dos marcos de datos mencionados anteriormente, así como un análisis que se enfoca exclusivamente en los valores identificados como:

3. **Datos atípicos**: Es el marco que solo incluye los valores atípicos (nombre en RStudio: "valores_atípicos").

Con este análisis comparativo, se espera proporcionar información valiosa para determinar la influencia de los valores atípicos. Así orientar las estrategias de marketing, para promover la fidelización de clientes, lo que contribuirá al crecimiento y éxito continuo de Cyclistic en el mercado de bicicletas compartidas.

Fuente de datos

Los datos históricos que se utilizaron para el análisis corresponden a un período de 12 meses, desde el 1 de mayo de 2022 hasta el 30 de abril de 2023. Estos datos fueron proporcionados por Motivate International Inc., una empresa que opera sistemas de bicicletas compartidas en Chicago. Contiene información sobre los viajes realizados por los usuarios (Miembro o Ciclista Ocasional). Incluyendo el mes, día y hora en que se realizaron y culminaron, así como el tipo de bicicleta utilizada.

Los datos están disponibles para su libre descarga y se pueden consultar en el siguiente "enlace a la fuente de datos", bajo el presente "enlace al acuerdo de licencia".

Credibilidad de los datos

La información se consideró original y confiable, ya que proviene directamente de una fuente primaria, la empresa encargada de prestar el servicio de bicicletas compartidas en colaboración con la ciudad de Chicago. Los datos son integrales, abarcando información relevante sobre los tipos de usuarios y los tiempos de inicio y fin de los viajes en bicicleta, entre otros aspectos.

Además, los datos son actuales, cubriendo de manera consecutiva los últimos 12 meses. Por último, es importante resaltar que estos datos son citables, ya que están

disponibles en el sitio web oficial y abarcan un periodo desde el año 2020 hasta 2023, lo que permite su consulta y referencia en futuros análisis o investigaciones. A continuación se precenta el proceso realizado en RStudio.

II. Preparar los datos para el análisis

2.1 Instalar y cargar los paquetes requeridos

```
install.packages(c("tidyverse", "lubridate")) #Para la manipulación de datos y fechas
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
## package 'lubridate' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\samp_\AppData\Local\Temp\RtmpyKzt9n\downloaded_packages
```

```
library(tidyverse) #Limpieza de datos
library(lubridate) #Manipular fechas y horas
```

2.2 Fuente de datos

Datos históricos del servicio de bicicletas compartidas, del periodo del 01 de mayo de 2022 al 30 de abril de 2023. Distribuidos en 12 archivos mensuales en formato de valores separados por comas (.csv).

```
# Crear una lista para almacenar los datos
lista_datos <- list()

# Nombres de los archivos originales
nombres_originales <- c(
  "Cyclistic_202205.csv",
  "Cyclistic_202206.csv",
  "Cyclistic_202207.csv",
  "Cyclistic_202208.csv",
  "Cyclistic_202209.csv",
  "Cyclistic_202210.csv",
  "Cyclistic_202211.csv",
  "Cyclistic_202212.csv",
  "Cyclistic_202301.csv",
  "Cyclistic_202302.csv",
  "Cyclistic_202303.csv",
  "Cyclistic_202304.csv"
)

# Iterar sobre los nombres de los archivos
for (i in seq_along(nombres_originales)) {
  # Cargar el archivo CSV y cambiar el nombre
  assign(paste0("data_", substr(nombres_originales[i], 11, 16)),
read.csv(nombres_originales[i]), envir = .GlobalEnv)
```

```
  # Almacenar el archivo en la lista de datos
  lista_datos[[i]] <- get(paste0("data_", substr(nombres_originales[i],
11, 16)))
}
```

Se creó una lista de datos con 12 archivos. Se renombraron, iniciando con la palabra "data", seguida de un guion bajo (_), año (yyyy) y mes al que corresponde (mm), (ejemplo: data_202205).

2.3 Procesar datos y combinar en un único marco de datos

Se identificaron las propiedades de los datos con la finalidad de detectar incongruencias que afecten la correcta integración en un solo marco de datos.

Identificar la estructura de los datos

```
# Iterar sobre los archivos en la lista de datos
for (i in seq_along(lista_datos)) {
  glimpse(lista_datos[[i]])
}
```

```
## Rows: 634,858
## Columns: 13
## $ ride_id            <chr> "EC2DE40644C6B0F4", "1C31AD03897EE385",
"1542FBEC83…
## $ rideable_type      <chr> "classic_bike", "classic_bike",
"classic_bike", "cl…
## $ started_at         <chr> "2022-05-23 23:06:58", "2022-05-11
08:53:28", "2022…
## $ ended_at           <chr> "2022-05-23 23:40:19", "2022-05-11
09:31:22", "2022…
## $ start_station_name <chr> "Wabash Ave & Grand Ave", "DuSable Lake
Shore Dr & …
## $ start_station_id   <chr> "TA1307000117", "13300", "TA1305000032",
"TA1305000…
## $ end_station_name   <chr> "Halsted St & Roscoe St", "Field Blvd &
South Water…
## $ end_station_id     <chr> "TA1309000025", "15534", "13221",
"TA1305000030", "…
## $ start_lat          <dbl> 41.89147, 41.88096, 41.88224, 41.88224,
41.88224, 4…
## $ start_lng          <dbl> -87.62676, -87.61674, -87.64107, -87.64107,
-87.641…
## $ end_lat            <dbl> 41.94367, 41.88635, 41.90765, 41.88458,
41.88578, 4…
## $ end_lng            <dbl> -87.64895, -87.61752, -87.67255, -87.63189,
-87.651…
## $ member_casual      <chr> "member", "member", "member", "member",
"member", "…
## Rows: 769,204
```

```
## Columns: 13
## $ ride_id            <chr> "600CFD130D0FD2A4", "F5E6B5C1682C6464",
"B6EB6D27BA…
## $ rideable_type      <chr> "electric_bike", "electric_bike",
"electric_bike", …
## $ started_at         <chr> "2022-06-30 17:27:53", "2022-06-30
18:39:52", "2022…
## $ ended_at           <chr> "2022-06-30 17:35:15", "2022-06-30
18:47:28", "2022…
## $ start_station_name <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_lat          <dbl> 41.89, 41.91, 41.91, 41.80, 41.91, 42.03,
41.91, 41…
## $ start_lng          <dbl> -87.62, -87.62, -87.65, -87.66, -87.63, -
87.71, -87…
## $ end_lat            <dbl> 41.91, 41.93, 41.89, 41.80, 41.93, 42.06,
41.92, 41…
## $ end_lng            <dbl> -87.62, -87.63, -87.61, -87.65, -87.64, -
87.73, -87…
## $ member_casual      <chr> "casual", "casual", "casual", "casual",
"casual", "…
## Rows: 823,488
## Columns: 13
## $ ride_id            <chr> "954144C2F67B1932", "292E027607D218B6",
"5776585258…
## $ rideable_type      <chr> "classic_bike", "classic_bike",
"classic_bike", "cl…
## $ started_at         <chr> "2022-07-05 08:12:47", "2022-07-26
12:53:38", "2022…
## $ ended_at           <chr> "2022-07-05 08:24:32", "2022-07-26
12:55:31", "2022…
## $ start_station_name <chr> "Ashland Ave & Blackhawk St", "Buckingham
Fountain …
## $ start_station_id   <chr> "13224", "15541", "15541", "15541",
"TA1307000117",…
## $ end_station_name   <chr> "Kingsbury St & Kinzie St", "Michigan Ave &
8th St"…
## $ end_station_id     <chr> "KA1503000043", "623", "623",
"TA1307000164", "TA13…
## $ start_lat          <dbl> 41.90707, 41.86962, 41.86962, 41.86962,
41.89147, 4…
## $ start_lng          <dbl> -87.66725, -87.62398, -87.62398, -87.62398,
-87.626…
## $ end_lat            <dbl> 41.88918, 41.87277, 41.87277, 41.79526,
```

```
41.93625, 4…
## $ end_lng            <dbl> -87.63851, -87.62398, -87.62398, -87.59647,
-87.652…
## $ member_casual      <chr> "member", "casual", "casual", "casual",
"member", "…
## Rows: 785,932
## Columns: 13
## $ ride_id            <chr> "550CF7EFEAE0C618", "DAD198F405F9C5F5",
"E6F2BC47B6…
## $ rideable_type      <chr> "electric_bike", "electric_bike",
"electric_bike", …
## $ started_at         <chr> "2022-08-07 21:34:15", "2022-08-08
14:39:21", "2022…
## $ ended_at           <chr> "2022-08-07 21:41:46", "2022-08-08
14:53:23", "2022…
## $ start_station_name <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_lat          <dbl> 41.93, 41.89, 41.97, 41.94, 41.85, 41.79,
41.89, 41…
## $ start_lng          <dbl> -87.69, -87.64, -87.69, -87.65, -87.65, -
87.72, -87…
## $ end_lat            <dbl> 41.94, 41.92, 41.97, 41.97, 41.84, 41.82,
41.89, 41…
## $ end_lng            <dbl> -87.72, -87.64, -87.66, -87.69, -87.66, -
87.69, -87…
## $ member_casual      <chr> "casual", "casual", "casual", "casual",
"casual", "…
## Rows: 701,339
## Columns: 13
## $ ride_id            <chr> "5156990AC19CA285", "E12D4A16BF51C274",
"A02B53CD7D…
## $ rideable_type      <chr> "electric_bike", "electric_bike",
"electric_bike", …
## $ started_at         <chr> "2022-09-01 08:36:22", "2022-09-01
17:11:29", "2022…
## $ ended_at           <chr> "2022-09-01 08:39:05", "2022-09-01
17:14:45", "2022…
## $ start_station_name <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_name   <chr> "California Ave & Milwaukee Ave", "", "",
"", "", "…
## $ end_station_id     <chr> "13084", "", "", "", "", "", "", "", "",
```

```
"", "", ""…
## $ start_lat          <dbl> 41.93000, 41.87000, 41.87000, 41.93000,
41.92000, 4…
## $ start_lng          <dbl> -87.69000, -87.62000, -87.62000, -87.69000,
-87.730…
## $ end_lat            <dbl> 41.92269, 41.87000, 41.87000, 41.94000,
41.92000, 4…
## $ end_lng            <dbl> -87.69715, -87.62000, -87.62000, -87.67000,
-87.730…
## $ member_casual      <chr> "casual", "casual", "casual", "casual",
"casual", "…
## Rows: 558,685
## Columns: 13
## $ ride_id            <chr> "A50255C1E17942AB", "DB692A70BD2DD4E3",
"3C02727AAF…
## $ rideable_type      <chr> "classic_bike", "electric_bike",
"electric_bike", "…
## $ started_at         <chr> "2022-10-14 17:13:30", "2022-10-01
16:29:26", "2022…
## $ ended_at           <chr> "2022-10-14 17:19:39", "2022-10-01
16:49:06", "2022…
## $ start_station_name <chr> "Noble St & Milwaukee Ave", "Damen Ave &
Charleston…
## $ start_station_id   <chr> "13290", "13288", "655", "KA1504000133",
"13028", "…
## $ end_station_name   <chr> "Larrabee St & Division St", "Damen Ave &
Cullerton…
## $ end_station_id     <chr> "KA1504000079", "13089", "TA1307000140",
"620", "13…
## $ start_lat          <dbl> 41.90068, 41.92004, 41.97988, 41.90227,
41.87475, 4…
## $ start_lng          <dbl> -87.66260, -87.67794, -87.68190, -87.62769,
-87.649…
## $ end_lat            <dbl> 41.90349, 41.85497, 41.96640, 41.89820,
41.86610, 4…
## $ end_lng            <dbl> -87.64335, -87.67570, -87.68870, -87.63754,
-87.607…
## $ member_casual      <chr> "member", "casual", "member", "member",
"casual", "…
## Rows: 337,735
## Columns: 13
## $ ride_id            <chr> "BCC66FC6FAB27CC7", "772AB67E902C180F",
"585EAD07FD…
## $ rideable_type      <chr> "electric_bike", "classic_bike",
"classic_bike", "c…
## $ started_at         <chr> "2022-11-10 06:21:55", "2022-11-04
07:31:55", "2022…
## $ ended_at           <chr> "2022-11-10 06:31:27", "2022-11-04
07:46:25", "2022…
## $ start_station_name <chr> "Canal St & Adams St", "Canal St & Adams
```

```
St", "Indi…
## $ start_station_id   <chr> "13011", "13011", "SL-005", "SL-005", "SL-
005", "13…
## $ end_station_name   <chr> "St. Clair St & Erie St", "St. Clair St &
Erie St",…
## $ end_station_id     <chr> "13016", "13016", "13016", "13016",
"13016", "TA130…
## $ start_lat          <dbl> 41.87940, 41.87926, 41.86789, 41.86789,
41.86789, 4…
## $ start_lng          <dbl> -87.63985, -87.63990, -87.62304, -87.62304,
-87.623…
## $ end_lat            <dbl> 41.89435, 41.89435, 41.89435, 41.89435,
41.89435, 4…
## $ end_lng            <dbl> -87.62280, -87.62280, -87.62280, -87.62280,
-87.622…
## $ member_casual      <chr> "member", "member", "member", "member",
"member", "…
## Rows: 181,806
## Columns: 13
## $ ride_id            <chr> "65DBD2F447EC51C2", "0C201AA7EA0EA1AD",
"E0B148CCB3…
## $ rideable_type      <chr> "electric_bike", "classic_bike",
"electric_bike", "…
## $ started_at         <chr> "2022-12-05 10:47:18", "2022-12-18
06:42:33", "2022…
## $ ended_at           <chr> "2022-12-05 10:56:34", "2022-12-18
07:08:44", "2022…
## $ start_station_name <chr> "Clifton Ave & Armitage Ave", "Broadway &
Belmont A…
## $ start_station_id   <chr> "TA1307000163", "13277", "TA1306000015",
"KA1503000…
## $ end_station_name   <chr> "Sedgwick St & Webster Ave", "Sedgwick St &
Webster…
## $ end_station_id     <chr> "13191", "13191", "13016", "13134",
"13288", "KA150…
## $ start_lat          <dbl> 41.91824, 41.94011, 41.88592, 41.83846,
41.89595, 4…
## $ start_lng          <dbl> -87.65711, -87.64545, -87.65113, -87.63541,
-87.667…
## $ end_lat            <dbl> 41.92217, 41.92217, 41.89435, 41.88137,
41.92008, 4…
## $ end_lng            <dbl> -87.63889, -87.63889, -87.62280, -87.67493,
-87.677…
## $ member_casual      <chr> "member", "casual", "member", "member",
"casual", "…
## Rows: 190,301
## Columns: 13
## $ ride_id            <chr> "F96D5A74A3E41399", "13CB7EB698CEDB88",
"BD88A2E670…
## $ rideable_type      <chr> "electric_bike", "classic_bike",
```

```
"electric_bike", "…
## $ started_at         <chr> "2023-01-21 20:05:42", "2023-01-10
15:37:36", "2023…
## $ ended_at           <chr> "2023-01-21 20:16:33", "2023-01-10
15:46:05", "2023…
## $ start_station_name <chr> "Lincoln Ave & Fullerton Ave", "Kimbark Ave
& 53rd …
## $ start_station_id   <chr> "TA1309000058", "TA1309000037", "RP-005",
"TA130900…
## $ end_station_name   <chr> "Hampden Ct & Diversey Ave", "Greenwood Ave
& 47th …
## $ end_station_id     <chr> "202480.0", "TA1308000002", "599",
"TA1308000002", …
## $ start_lat          <dbl> 41.92407, 41.79957, 42.00857, 41.79957,
41.79957, 4…
## $ start_lng          <dbl> -87.64628, -87.59475, -87.69048, -87.59475,
-87.594…
## $ end_lat            <dbl> 41.93000, 41.80983, 42.03974, 41.80983,
41.80983, 4…
## $ end_lng            <dbl> -87.64000, -87.59938, -87.69941, -87.59938,
-87.599…
## $ member_casual      <chr> "member", "member", "casual", "member",
"member", "…
## Rows: 190,445
## Columns: 13
## $ ride_id            <chr> "CBCD0D7777F0E45F", "F3EC5FCE5FF39DE9",
"E54C1F27FA…
## $ rideable_type      <chr> "classic_bike", "electric_bike",
"classic_bike", "e…
## $ started_at         <chr> "2023-02-14 11:59:42", "2023-02-15
13:53:48", "2023…
## $ ended_at           <chr> "2023-02-14 12:13:38", "2023-02-15
13:59:08", "2023…
## $ start_station_name <chr> "Southport Ave & Clybourn Ave", "Clarendon
Ave & Go…
## $ start_station_id   <chr> "TA1309000030", "13379", "TA1309000030",
"TA1309000…
## $ end_station_name   <chr> "Clark St & Schiller St", "Sheridan Rd &
Lawrence A…
## $ end_station_id     <chr> "TA1309000024", "TA1309000041", "13156",
"TA1309000…
## $ start_lat          <dbl> 41.92077, 41.95788, 41.92077, 41.92087,
41.79483, 4…
## $ start_lng          <dbl> -87.66371, -87.64958, -87.66371, -87.66373,
-87.618…
## $ end_lat            <dbl> 41.90799, 41.96952, 41.88042, 41.87943,
41.78053, 4…
## $ end_lng            <dbl> -87.63150, -87.65469, -87.65552, -87.63550,
-87.605…
## $ member_casual      <chr> "casual", "casual", "member", "member",
```

```
"member", "…
## Rows: 258,678
## Columns: 13
## $ ride_id            <chr> "6842AA605EE9FBB3", "F984267A75B99A8C",
"FF7CF57CFE…
## $ rideable_type      <chr> "electric_bike", "electric_bike",
"classic_bike", "…
## $ started_at         <chr> "2023-03-16 08:20:34", "2023-03-04
14:07:06", "2023…
## $ ended_at           <chr> "2023-03-16 08:22:52", "2023-03-04
14:15:31", "2023…
## $ start_station_name <chr> "Clark St & Armitage Ave", "Public Rack -
Kedzie Av…
## $ start_station_id   <chr> "13146", "491", "620", "TA1306000003",
"18067", "62…
## $ end_station_name   <chr> "Larrabee St & Webster Ave", "", "Clark St
& Randol…
## $ end_station_id     <chr> "13193", "", "TA1305000030", "13154",
"TA1306000015…
## $ start_lat          <dbl> 41.91841, 41.97000, 41.89820, 41.88872,
41.91448, 4…
## $ start_lng          <dbl> -87.63645, -87.71000, -87.63754, -87.64445,
-87.668…
## $ end_lat            <dbl> 41.92182, 41.95000, 41.88458, 41.91052,
41.88578, 4…
## $ end_lng            <dbl> -87.64414, -87.71000, -87.63189, -87.65311,
-87.651…
## $ member_casual      <chr> "member", "member", "member", "member",
"member", "…
## Rows: 426,590
## Columns: 13
## $ ride_id            <chr> "8FE8F7D9C10E88C7", "34E4ED3ADF1D821B",
"5296BF07A2…
## $ rideable_type      <chr> "electric_bike", "electric_bike",
"electric_bike", …
## $ started_at         <chr> "2023-04-02 08:37:28", "2023-04-19
11:29:02", "2023…
## $ ended_at           <chr> "2023-04-02 08:41:37", "2023-04-19
11:52:12", "2023…
## $ start_station_name <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "", "",
"", "",…
## $ start_lat          <dbl> 41.80, 41.87, 41.93, 41.92, 41.91, 41.91,
41.93, 42…
## $ start_lng          <dbl> -87.60, -87.65, -87.66, -87.65, -87.65, -
```

```
87.63, -87…
## $ end_lat            <dbl> 41.79, 41.93, 41.93, 41.91, 41.91, 41.92,
41.91, 41…
## $ end_lng            <dbl> -87.60, -87.68, -87.66, -87.65, -87.63, -
87.65, -87…
## $ member_casual      <chr> "member", "member", "member", "member",
"member", "…
```

Número de columnas:

- El conjunto de datos consta de 13 columnas diferentes.

Información de los viajes:

- La columna ride_id almacena identificadores únicos para cada viaje.

- La columna rideable_type indica el tipo de bicicleta utilizada en cada viaje (por ejemplo, “electric_bike”).
- Las columnas started_at y ended_at registran la fecha y hora de inicio y finalización de cada viaje.

Información de las estaciones:

- Las columnas start_station_name y end_station_name contienen los nombres de las estaciones de inicio y fin, respectivamente. Sin embargo, se observa que algunas de estas celdas están vacías, lo que sugiere que no se dispone de información para todas las estaciones.

- Las columnas start_station_id y end_station_id, contienen identificadores de estaciones, y también se observa que algunas celdas están vacías.

Información geoespacial:

- Las columnas start_lat, start_lng, end_lat y end_lng contienen datos de latitud y longitud para las ubicaciones de inicio y fin de cada viaje.

Tipo de usuario:

- La columna member_casual, registra el tipo de usuario que realizó cada viaje, con valores como “Member” (Miembro) y “Casual” (Casual).

Identificar el nombre de las columnas

```r
# Iterar sobre los archivos en la lista de datos
for (i in seq_along(lista_datos)) {
  # Obtener los nombres de las columnas del archivo actual
  nombres_columnas <- colnames(lista_datos[[i]])

  # Imprimir los nombres de las columnas
  cat("Nombres de columnas del archivo", i, ":", paste(nombres_columnas,
```

```
collapse = ", "), "\n")
}

## Nombres de columnas del archivo 1 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 2 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 3 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 4 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 5 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 6 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 7 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 8 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 9 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 10 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 11 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
## Nombres de columnas del archivo 12 : ride_id, rideable_type,
started_at, ended_at, start_station_name, start_station_id,
```

```
end_station_name, end_station_id, start_lat, start_lng, end_lat, end_lng,
member_casual
```

Cada uno de los archivos está integrado por 13 columnas. Con el mismo nombre y orden, por lo que se puede integrar en un solo marco de datos.

Integrar los doce archivos en un único marco de datos.

```
library(dplyr) # Instalado en el paquete tidyverse, combina múltiples
conjuntos de datos en filas
all_data12 <- bind_rows(lista_datos)
```

El nuevo marco de datos "all_data12", contiene la información de los doce documentos.

III. Procesamiento de los datos

3.1 Inspeccionar el nuevo marco de datos "all_data12"

```
colnames(all_data12) #nombre de las columnas
```

```
##  [1] "ride_id"            "rideable_type"      "started_at"
##  [4] "ended_at"           "start_station_name" "start_station_id"
##  [7] "end_station_name"   "end_station_id"     "start_lat"
## [10] "start_lng"          "end_lat"            "end_lng"
## [13] "member_casual"
```

```
dim(all_data12) #dimension del marco de datos
```

```
## [1] 5859061      13
```

```
head(all_data12) #visualizar las primeras seis filas
```

```
##            ride_id rideable_type          started_at
ended_at
## 1 EC2DE40644C6B0F4  classic_bike 2022-05-23 23:06:58 2022-05-23
23:40:19
## 2 1C31AD03897EE385  classic_bike 2022-05-11 08:53:28 2022-05-11
09:31:22
## 3 1542FBEC830415CF  classic_bike 2022-05-26 18:36:28 2022-05-26
18:58:18
## 4 6FF59852924528F8  classic_bike 2022-05-10 07:30:07 2022-05-10
07:38:49
## 5 483C52CAAE12E3AC  classic_bike 2022-05-10 17:31:56 2022-05-10
17:36:57
## 6 C0A3AA5A614DCE01  classic_bike 2022-05-04 14:48:55 2022-05-04
14:56:04
##                   start_station_name start_station_id
## 1            Wabash Ave & Grand Ave     TA1307000117
## 2 DuSable Lake Shore Dr & Monroe St            13300
## 3           Clinton St & Madison St     TA1305000032
## 4           Clinton St & Madison St     TA1305000032
```

```
## 5           Clinton St & Madison St     TA1305000032
## 6           Carpenter St & Huron St            13196
##                end_station_name end_station_id start_lat start_lng
end_lat
## 1        Halsted St & Roscoe St   TA1309000025  41.89147 -87.62676
41.94367
## 2   Field Blvd & South Water St          15534  41.88096 -87.61674
41.88635
## 3       Wood St & Milwaukee Ave          13221  41.88224 -87.64107
41.90765
## 4        Clark St & Randolph St   TA1305000030  41.88224 -87.64107
41.88458
## 5           Morgan St & Lake St   TA1306000015  41.88224 -87.64107
41.88578
## 6 Sangamon St & Washington Blvd          13409  41.89456 -87.65345
41.88316
##     end_lng member_casual
## 1 -87.64895        member
## 2 -87.61752        member
## 3 -87.67255        member
## 4 -87.63189        member
## 5 -87.65102        member
## 6 -87.65110        member
```

```
str(all_data12) #tipos de datos
```

```
## 'data.frame':    5859061 obs. of  13 variables:
##  $ ride_id           : chr  "EC2DE40644C6B0F4" "1C31AD03897EE385"
"1542FBEC830415CF" "6FF59852924528F8" ...
##  $ rideable_type     : chr  "classic_bike" "classic_bike"
"classic_bike" "classic_bike" ...
##  $ started_at        : chr  "2022-05-23 23:06:58" "2022-05-11
08:53:28" "2022-05-26 18:36:28" "2022-05-10 07:30:07" ...
##  $ ended_at          : chr  "2022-05-23 23:40:19" "2022-05-11
09:31:22" "2022-05-26 18:58:18" "2022-05-10 07:38:49" ...
##  $ start_station_name: chr  "Wabash Ave & Grand Ave" "DuSable Lake
Shore Dr & Monroe St" "Clinton St & Madison St" "Clinton St & Madison St"
...
##  $ start_station_id  : chr  "TA1307000117" "13300" "TA1305000032"
"TA1305000032" ...
##  $ end_station_name  : chr  "Halsted St & Roscoe St" "Field Blvd &
South Water St" "Wood St & Milwaukee Ave" "Clark St & Randolph St" ...
##  $ end_station_id    : chr  "TA1309000025" "15534" "13221"
"TA1305000030" ...
##  $ start_lat         : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num  -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num  -87.6 -87.6 -87.7 -87.6 -87.7 ...
##  $ member_casual     : chr  "member" "member" "member" "member" ...
```

El marco de datos está estructurado por 13 columnas y un total de 5,859,061 registros.

En resumen, el marco de datos “all_data12” proporciona una cantidad significativa de información sobre los viajes en bicicleta, incluidos detalles sobre los viajes, ubicaciones geoespaciales y el tipo de usuario.

Identificar las variables en la columna “rideable_type”

La columna “rideable_type” hace referencia a los 3 tipos de bicicletas (classic_bike, docked_bike, electric_bike) disponibles. Es necesario validar que solo existan estos tipos de bicicletas.

```
#Obtener los valores únicos de la columna "rideable_type"
valores_unicos_bicicleta <- unique(all_data12$rideable_type)

# Verificar si hay valores adicionales
valores_adicionales <- setdiff(valores_unicos_bicicleta,
c("classic_bike", "docked_bike", "electric_bike"))

# Imprimir los valores adicionales, si los hay
if (length(valores_adicionales) > 0) {
  cat("Se encontraron valores adicionales en la columna
'rideable_type':\n")
  cat(valores_adicionales, sep = "\n")
} else {
  cat("La columna 'rideable_type' solo contiene las entradas
'classic_bike', 'docked_bike' y 'electric_bike'.\n")
}
```

```
## La columna 'rideable_type' solo contiene las entradas 'classic_bike',
'docked_bike' y 'electric_bike'.
```

Las conclusiones de la validación de las variables en la columna “rideable_type” son las siguientes:

- Validación Exitosa: Tras la inspección de la columna “rideable_type”, se pudo verificar que solo contiene los tres tipos de bicicletas deseados: “classic_bike”, “docked_bike” y “electric_bike”, por lo cual no presenta problemas de calidad de datos en este aspecto.

Identificar las variables en la columna “member_casual”

La columna “member_casual” hace referencia a los 2 tipos de usuarios, los que pagan una suscripción anual (member), y los que utilizan el servicio por hora o día (casual). Es necesario validar que solo existan estos dos tipos de usuario.

```
# Obtener los valores únicos de la columna "member_casual"
valores_unicos_usuario <- unique(all_data12$member_casual)

# Verificar si hay valores adicionales
```

```r
valores_adicionales <- setdiff(valores_unicos_usuario, c("member",
"casual"))

# Imprimir los valores adicionales, si los hay
if (length(valores_adicionales) > 0) {
  cat("Se encontraron valores adicionales en la columna
'member_casual':\n")
  cat(valores_adicionales, sep = "\n")
} else {
  cat("La columna 'member_casual' solo contiene las entradas 'member' y
'casual'.\n")
}
```

```
## La columna 'member_casual' solo contiene las entradas 'member' y
'casual'.
```

Las conclusiones de la validación de las variables en la columna "member_casual" son las siguientes:

- Cumplimiento de la Condición: Tras la inspección de la columna "member_casual", se pudo verificar que solo contiene los dos tipos de usuarios deseados: "member" (miembro) y "casual" (casual).

3.2 Agregar información específica del inicio de los viajes, a partir de la columna "started_at".

La columna "started_at" contiene información referente a la fecha y hora, en formato (yyyy-mm-dd hh:mm:ss) en la que inició cada uno de los viajes. Sin embargo, para facilitar el análisis temporal y segmentar los datos, se realizaron transformaciones para crear nuevas columnas específicas de fecha y hora.

```r
# A partir de la columna startet_at, crear la columna "date", fecha sin
hora
all_data12$date <- as.Date(all_data12$started_at) #En formato "yyyy-mm-
dd"
# A partir de la columna "date", extraer el día, mes y año
all_data12$month <- format(as.Date(all_data12$date), "%m") #Columna
"month" en formato numérico (1-12)
all_data12$day <- format(as.Date(all_data12$date), "%d") #Crear la
columna "day"
all_data12$year <- format(as.Date(all_data12$date), "%Y") #Columna "year"
en formato "yyyy"
all_data12$day_of_week <- format(as.Date(all_data12$date), "%A") #Crear
la columna "day_of_week", para el nombre del día de la semana
```

Las conclusiones obtenidas a partir de la información relacionada con la columna "started_at" y la creación de nuevas columnas específicas de fecha y hora son las siguientes:

Nuevas Columnas Agregadas: Se agregaron cinco nuevas columnas al marco de datos "all_data12" para desglosar la información temporal de manera más detallada:

- date: Contiene la fecha sin la parte de la hora, en formato "yyyy-mm-dd".
- month: Representa el mes en formato numérico (1-12).
- day: Contiene el día del mes en formato numérico.
- year: Almacena el año en formato "yyyy".
- day_of_week: Registra el nombre del día de la semana.

Facilitación del Análisis Temporal:

- Estas nuevas columnas proporcionan una mayor flexibilidad para realizar análisis temporales, como la identificación de patrones mensuales, estacionales o diarios en los viajes en bicicleta. Además, la columna "day_of_week" facilita la segmentación de datos según el día de la semana.

En resumen, la adición de estas nuevas columnas específicas de fecha y hora enriquece los datos y permite un análisis más detallado y efectivo de los patrones temporales relacionados con los viajes en bicicleta. Esto es esencial para comprender mejor las preferencias de los usuarios y tomar decisiones basadas en datos relacionadas con la temporalidad de los viajes.

3.3 Calcular la duración de los viajes en segundos

Las columnas "started_at" y "ended_at" contienen respectivamente información del inicio y término de los viajes en formato (yyyy-mm-dd hh:mm:ss). Para calcular la duración de los viajes en segundos, se realizó la operación de restar la columna "started_at" de "ended_at".

```
all_data12$ride_length <-
difftime(all_data12$ended_at,all_data12$started_at)
```

Las conclusiones obtenidas a partir del cálculo de la duración de los viajes en segundos son las siguientes:

- El cálculo de la duración de los viajes en segundos se realizó correctamente mediante la diferencia entre las columnas "started_at" (inicio del viaje) y "ended_at" (finalización del viaje) utilizando la función difftime.
- La duración de los viajes en segundos es una métrica crucial para comprender la eficiencia y la duración promedio de los viajes en bicicleta. Esto facilitará la identificación de tendencias, valores atípicos y patrones temporales en la duración de los viajes.

Inspeccionar el marco de datos con las nuevas columnas creadas

```
#Identificar la estructura del marco de datos
head(all_data12) #Muestra las seis primeras filas
```

```
##            ride_id rideable_type          started_at
ended_at
```

```
## 1 EC2DE40644C6B0F4  classic_bike 2022-05-23 23:06:58 2022-05-23
23:40:19
## 2 1C31AD03897EE385  classic_bike 2022-05-11 08:53:28 2022-05-11
09:31:22
## 3 1542FBEC830415CF  classic_bike 2022-05-26 18:36:28 2022-05-26
18:58:18
## 4 6FF59852924528F8  classic_bike 2022-05-10 07:30:07 2022-05-10
07:38:49
## 5 483C52CAAE12E3AC  classic_bike 2022-05-10 17:31:56 2022-05-10
17:36:57
## 6 C0A3AA5A614DCE01  classic_bike 2022-05-04 14:48:55 2022-05-04
14:56:04
##                   start_station_name start_station_id
## 1            Wabash Ave & Grand Ave     TA1307000117
## 2 DuSable Lake Shore Dr & Monroe St            13300
## 3           Clinton St & Madison St     TA1305000032
## 4           Clinton St & Madison St     TA1305000032
## 5           Clinton St & Madison St     TA1305000032
## 6           Carpenter St & Huron St            13196
##                end_station_name end_station_id start_lat start_lng
end_lat
## 1        Halsted St & Roscoe St   TA1309000025  41.89147 -87.62676
41.94367
## 2   Field Blvd & South Water St          15534  41.88096 -87.61674
41.88635
## 3       Wood St & Milwaukee Ave          13221  41.88224 -87.64107
41.90765
## 4        Clark St & Randolph St   TA1305000030  41.88224 -87.64107
41.88458
## 5           Morgan St & Lake St   TA1306000015  41.88224 -87.64107
41.88578
## 6 Sangamon St & Washington Blvd          13409  41.89456 -87.65345
41.88316
##     end_lng member_casual       date month day year day_of_week
ride_length
## 1 -87.64895        member 2022-05-23    05  23 2022       lunes   2001
secs
## 2 -87.61752        member 2022-05-11    05  11 2022   miércoles   2274
secs
## 3 -87.67255        member 2022-05-26    05  26 2022      jueves   1310
secs
## 4 -87.63189        member 2022-05-10    05  10 2022      martes    522
secs
## 5 -87.65102        member 2022-05-10    05  10 2022      martes    301
secs
## 6 -87.65110        member 2022-05-04    05  04 2022   miércoles    429
secs
```

```r
str(all_data12) #Tipo de dato y estructura
```

```
## 'data.frame':    5859061 obs. of  19 variables:
##  $ ride_id           : chr  "EC2DE40644C6B0F4" "1C31AD03897EE385"
"1542FBEC830415CF" "6FF59852924528F8" ...
##  $ rideable_type     : chr  "classic_bike" "classic_bike"
"classic_bike" "classic_bike" ...
##  $ started_at        : chr  "2022-05-23 23:06:58" "2022-05-11
08:53:28" "2022-05-26 18:36:28" "2022-05-10 07:30:07" ...
##  $ ended_at          : chr  "2022-05-23 23:40:19" "2022-05-11
09:31:22" "2022-05-26 18:58:18" "2022-05-10 07:38:49" ...
##  $ start_station_name: chr  "Wabash Ave & Grand Ave" "DuSable Lake
Shore Dr & Monroe St" "Clinton St & Madison St" "Clinton St & Madison St"
...
##  $ start_station_id  : chr  "TA1307000117" "13300" "TA1305000032"
"TA1305000032" ...
##  $ end_station_name  : chr  "Halsted St & Roscoe St" "Field Blvd &
South Water St" "Wood St & Milwaukee Ave" "Clark St & Randolph St" ...
##  $ end_station_id    : chr  "TA1309000025" "15534" "13221"
"TA1305000030" ...
##  $ start_lat         : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num  -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num  -87.6 -87.6 -87.7 -87.6 -87.7 ...
##  $ member_casual     : chr  "member" "member" "member" "member" ...
##  $ date              : Date, format: "2022-05-23" "2022-05-11" ...
##  $ month             : chr  "05" "05" "05" "05" ...
##  $ day               : chr  "23" "11" "26" "10" ...
##  $ year              : chr  "2022" "2022" "2022" "2022" ...
##  $ day_of_week       : chr  "lunes" "miércoles" "jueves" "martes" ...
##  $ ride_length       : 'difftime' num  2001 2274 1310 522 ...
##   ..- attr(*, "units")= chr "secs"
```

Las conclusiones obtenidas al inspeccionar el marco de datos con las nuevas columnas creadas son las siguientes:

- Se agregaron exitosamente cinco nuevas columnas al marco de datos "all_data12": "date" (fecha sin hora), "month" (mes en formato numérico), "day" (día del mes en formato numérico), "year" (año) y "day_of_week" (nombre del día de la semana).
- Se observa que la columna "ride_length" tiene el tipo de dato "difftime" con unidades en segundos. Para realizar cálculos numéricos, es necesario convertir esta columna a un tipo de dato numérico.

En resumen, las nuevas columnas agregadas proporcionan información valiosa y facilitan la realización de análisis más completos y detallados sobre los datos de los viajes en bicicleta. Además, se destaca la necesidad de convertir la columna "ride_length" al tipo de dato numérico para futuros cálculos.

Convertir la columna ride_length de tipo factor a numérico

```
is.factor(all_data12$ride_length) #Verificar si la columna es tipo factor
```

```
## [1] FALSE
```

```
all_data12$ride_length <-
as.numeric(as.character(all_data12$ride_length)) #Convertir a una cadena
de caracteres y luego a numérico
is.numeric(all_data12$ride_length) #Verificar si la columna es tipo
numérico
```

```
## [1] TRUE
```

Las conclusiones obtenidas al convertir la columna "ride_length" de tipo factor a numérico son las siguientes:

Verificación del tipo de dato Inicial:

- Se verificó inicialmente que la columna "ride_length" no es de tipo factor, ya que su resultado fue FALSE.

Conversión exitosa:

- Se procedió a convertir la columna "ride_length" a tipo numérico de manera exitosa, primero convirtiéndola a una cadena de caracteres y luego a numérica. El resultado de la verificación posterior fue TRUE, confirmando que ahora la columna es de tipo numérico.

Preparación para cálculos:

- La conversión de la columna "ride_length" a tipo numérico es fundamental para realizar cálculos estadísticos y análisis cuantitativos relacionados con la duración de los viajes en bicicleta.

3.4 Identificar datos inconsistentes

Identificar valores negativos y cero en la duración de los viajes.

```
#Identificar el número de viajes con duración negativa
viajes_duracion_negativa <- sum(all_data12$ride_length < 0)
print(viajes_duracion_negativa)
```

```
## [1] 103
```

Viajes con duración negativa:

- Se identificaron 103 registros con duración negativa en la columna "ride_length" de un total de 5,859,061. **Consideración No. 1** No fue posible corroborar la naturaleza exacta de estos resultados, pero se decidió no contemplarlos para los siguientes pasos del análisis. La duración (tiempo) de los viajes no puede ser negativa, lo que sugiere posibles errores en los datos o problemas en la forma en que se registraron los tiempos de inicio y finalización de los viajes.

```r
#Identificar el número de viajes con duración cero
viajes_duracion_cero <- sum(all_data12$ride_length == 0)
print(viajes_duracion_cero)

## [1] 441
```

Viajes con duración cero:

- Se identificaron 441 registros con una duración de cero segundos en la columna "ride_length" de un total de 5,859,061. **Consideración No. 2** A pesar de que no se pudo corroborar la naturaleza exacta de estos resultados, se decidió contemplarlos para los siguientes pasos del análisis.
- La inclusión o exclusión de los registros con duración cero puede tener un efecto en los resultados del análisis posterior. Esto sugiere la importancia de analizar cómo estos viajes de duración cero pueden afectar las conclusiones del análisis.

Comparar el efecto de la duración cero en los viajes de bicicleta

Se crean dos marcos de datos ambos sin los valores negativos de la duración de los viajes, el primero considerando las entradas cero en la columna "ride_length" y el segundo sin considerar estos valores.

```r
##Crear el marco de datos "all_data12_v2" sin los registros negativos de la columna ride_length
all_data12_v2 <- all_data12[!(all_data12$ride_length < 0), ]

# Utilizar "all_data_v2" para crear el marco de datos "all_data_v3" en el que no se incluya el valor cero
all_data12_v3 <- all_data12_v2[all_data12_v2$ride_length != 0, ]

##Comparar las estadísticas
summary(all_data12_v2$ride_length)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     339     599    1136    1075 2486835

summary(all_data12_v3$ride_length)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1     339     599    1136    1075 2486835
```

Las conclusiones obtenidas al comparar el efecto de la duración cero en los viajes de bicicleta son las siguientes:

Creación de marcos de datos sin duraciones negativas:

- Se crearon dos marcos de datos, "all_data12_v2" y "all_data12_v3", ambos excluyendo los registros con duraciones negativas en la columna "ride_length". Esto se hizo para garantizar que no se incluyan valores de duración que no tengan sentido en el análisis.

Comparación de estadísticas: Se compararon las estadísticas de la duración de los viajes en ambos marcos de datos.

- **Para "all_data12_v2" (que incluye duraciones de cero):**
  - Valor mínimo: 0 segundos
  - Primer cuartil: 339 segundos
  - Mediana: 599 segundos
  - Media: 1,136 segundos
  - Tercer cuartil: 1,075 segundos
  - Valor Máximo: 2,486,835 segundos

- **Para "all_data12_v3" (que excluye duraciones de cero):**
  - Valor mínimo: 1 segundo
  - Primer Cuartil: 339 segundos
  - Mediana: 599 segundos
  - Media: 1,136 segundos
  - Tercer cuartil: 1,075 segundos
  - Valor máximo: 2,486,835 segundos

Comparación de resultados:

- Se observa que, al excluir los registros con duración cero, la media, mediana y el valor máximo de la duración de los viajes no cambian. Sin embargo, el valor mínimo cambia de 0 segundos (considerando el valor cero) a 1 segundo (excluyendo el valor cero).

Decisión de inclusión:

- Se concluyó que, para este análisis, es apropiado considerar los valores de duración de cero, ya que su exclusión no altera significativamente las estadísticas resumidas. Esto significa que se pueden incluir los viajes de duración cero en el análisis sin distorsionar de manera significativa los resultados finales.

Valores atípicos en la duración de los viajes

En el contexto del análisis del marco de datos "all_data12_v2", es importante destacar que la medida de tendencia central, representada por la media (1,136 segundos), muestra un sesgo hacia la derecha en comparación con el valor de la mediana (599 segundos). Por lo que se debe considerar la posible presencia de valores atípicos, que se pueden atribuir a duración de los viajes en bicicleta extremadamente altos.

IV. Análisis del marco de datos “all_data12_v2” (Todos los datos)

4.1. Resumen estadístico

Ya que se decidió considerar la duración de valores cero y excluir los valores negativos. A continuación, un resumen de las estadísticas de duración de los viajes, considerando a ambos tipos de usuarios.

```
summary(all_data12_v2$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     339     599    1136    1075 2486835
```

A partir de estos datos, podemos observar que la duración promedio de los viajes es de aproximadamente 18 minutos y 56 segundos. Sin embargo, y como se mencionó anteriormente, también es evidente que existen valores extremadamente altos, como el máximo de 2,486,835 segundos, que podrían considerarse atípicos. El cual se retomará al concluir con el análisis del marco de datos “all_data12_v2”.

Comparativa de las estadísticas de duración de viajes entre usuarios “Miembros” y “Casuales”

```
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual, FUN =
mean)
```

```
##   all_data12_v2$member_casual all_data12_v2$ride_length
## 1                      casual                 1709.8826
## 2                      member                  750.0753
```

```
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual, FUN =
median)
```

```
##   all_data12_v2$member_casual all_data12_v2$ride_length
## 1                      casual                       751
## 2                      member                       520
```

```
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual, FUN =
max)
```

```
##   all_data12_v2$member_casual all_data12_v2$ride_length
## 1                      casual                   2486835
## 2                      member                     93597
```

```
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual, FUN =
min)
```

```
##   all_data12_v2$member_casual all_data12_v2$ride_length
## 1                      casual                         0
## 2                      member                         0
```

Para una mejor comprensión, los resultados se expresan en minutos.

Duración promedio:

- Usuarios casuales: 28 minutos y 30 segundos.
- Usuarios miembros: 12 minutos y 30 segundos.

Esto indica que, en promedio, los usuarios ocasionales tienen viajes significativamente más largos en comparación con los usuarios miembros. La duración promedio de los viajes para los usuarios ocasionales es aproximadamente 2.28 veces que la de los usuarios miembros.

Duración (valor de la mediana):

- Usuarios casuales: 12 minutos y 31 segundos.
- Usuarios miembros: 8 minutos y 40 segundos.

La mediana, que representa el valor en el centro de la distribución, también muestra que los usuarios casuales tienen viajes más largos en comparación con los usuarios miembros. El valor de la mediana de los viajes para los usuarios ocasionales es aproximadamente 1.44 veces que la de los usuarios miembros.

4.2 Duración promedio de los viajes por dia de semana

Duración promedio de viajes por día de la semana para usuarios "Casuales" y "Miembros".

```
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual +
all_data12_v2$day_of_week, FUN = mean)
```

```
##    all_data12_v2$member_casual all_data12_v2$day_of_week
## 1                       casual                   domingo
## 2                       member                   domingo
## 3                       casual                    jueves
## 4                       member                    jueves
## 5                       casual                     lunes
## 6                       member                     lunes
## 7                       casual                    martes
## 8                       member                    martes
## 9                       casual                 miércoles
## 10                      member                 miércoles
## 11                      casual                    sábado
## 12                      member                    sábado
## 13                      casual                   viernes
## 14                      member                   viernes
##    all_data12_v2$ride_length
## 1                  2007.2362
## 2                   829.8643
## 3                  1482.0060
## 4                   726.0527
## 5                  1702.1177
## 6                   718.6881
```

```
## 7                  1520.1645
## 8                   717.0647
## 9                  1450.3947
## 10                  713.4792
## 11                 1934.7530
## 12                  835.9565
## 13                 1651.1706
## 14                  741.8887
```

```
#Para que el resultado se muestre de domingo a sábado, se ordena la
columna day_of_week
all_data12_v2$day_of_week <- ordered(all_data12_v2$day_of_week,
levels=c("domingo", "lunes", "martes", "miércoles", "jueves", "viernes",
"sábado"))
#Comparar el promedio de la duración de viajes por día de semana para
usuarios y miembros.
aggregate(all_data12_v2$ride_length ~ all_data12_v2$member_casual +
all_data12_v2$day_of_week, FUN = mean)
```

```
##    all_data12_v2$member_casual all_data12_v2$day_of_week
## 1                       casual                   domingo
## 2                       member                   domingo
## 3                       casual                     lunes
## 4                       member                     lunes
## 5                       casual                    martes
## 6                       member                    martes
## 7                       casual                  miércoles
## 8                       member                  miércoles
## 9                       casual                     jueves
## 10                      member                     jueves
## 11                      casual                   viernes
## 12                      member                   viernes
## 13                      casual                    sábado
## 14                      member                    sábado
##    all_data12_v2$ride_length
## 1                  2007.2362
## 2                   829.8643
## 3                  1702.1177
## 4                   718.6881
## 5                  1520.1645
## 6                   717.0647
## 7                  1450.3947
## 8                   713.4792
## 9                  1482.0060
## 10                  726.0527
## 11                 1651.1706
## 12                  741.8887
## 13                 1934.7530
## 14                  835.9565
```

Los resultados se expresan en términos del total de viajes (anual) y el promedio de duración de los viajes.

Usuarios casuales:

- Los días con la mayor duración promedio de viajes son el domingo (33 minutos y 27 segundos) y el sábado (32 minutos y 15 segundos).
- El día con la menor duración promedio es miércoles (24 minutos y 10 segundos).
- La diferencia entre el día de mayor y menor duración es de 9 minutos y 16 segundos.

Usuarios miembros:

- Los días con la mayor duración promedio de viajes son sábado (13 minutos y 49 segundos) y domingo (13 minutos y 56 segundos).
- El día con la menor duración promedio es miércoles (11 minutos y 53 segundos).
- La diferencia entre el día de mayor y menor duración es de 2 minutos y 3 segundos.

En resumen, para ambos usuarios los fines de semana son los días con la mayor duración promedio de los viajes. Así como el miércoles es el día con menor duración promedio.
4.3 Número de viajes por día de semana

```
all_data12_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)
```

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

```
## # A tibble: 14 × 4
## # Groups:   member_casual [2]
##    member_casual weekday  number_of_rides average_duration
##    <chr>         <ord>              <int>            <dbl>
##  1 casual        "dom\\."          388811            2007.
##  2 casual        "lun\\."          275748            1702.
##  3 casual        "mar\\."          272648            1520.
##  4 casual        "mié\\."          284575            1450.
##  5 casual        "jue\\."          318467            1482.
##  6 casual        "vie\\."          350081            1651.
##  7 casual        "sáb\\."          467923            1935.
##  8 member        "dom\\."          402066             830.
##  9 member        "lun\\."          484560             719.
```

```
## 10 member        "mar\\."          544393            717.
## 11 member        "mié\\."          556913            713.
## 12 member        "jue\\."          560877            726.
## 13 member        "vie\\."          497473            742.
## 14 member        "sáb\\."          454423            836.
```

Los resultados se expresan con respecto al total de cada tipo de usuario.

Usuarios casuales:

- El día con mayor número de viajes es el sábado, con un total de 467,923 registros, lo que equivale al 19.84% del total. En contraste, el día con la menor cantidad de viajes es el martes, con 272,648 registros, representando el 11.73% del total.

Usuarios miembros:

- El día con mayor número de viajes es el jueves, con un total de 560,877 registros, lo que equivale al 16.02% del total. En contraste, el día con la menor cantidad de viajes es el domingo, con 402,066 registros, representando el 11.49% del total.

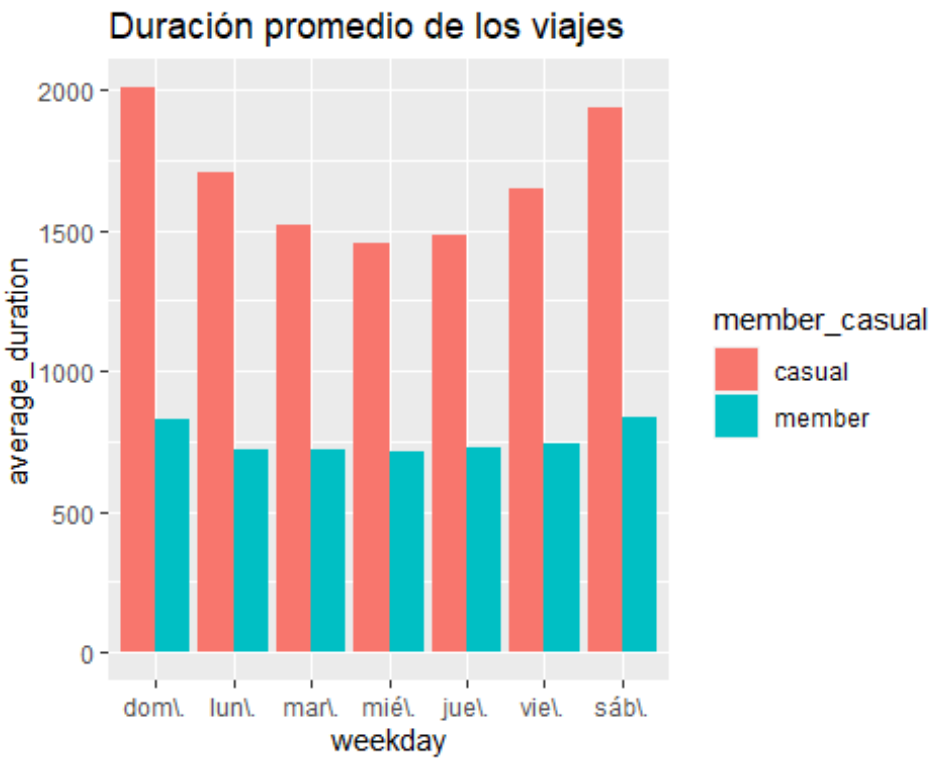
Estos hallazgos sugieren diferencias en los patrones de uso de bicicletas entre usuarios casuales y miembros. Los usuarios casuales tienden a utilizar más el servicio los fines de semana, con un pico notable los sábados. Por otro lado, los miembros tienden a utilizar el servicio en días laborables, con el jueves como el día de mayor actividad.

4.4. Visualización de resultados

Visualización de la duración promedio

Después de analizar el promedio de duración de viajes por día de la semana para usuarios "Casuales" y "Miembros," se procede a representar visualmente esta información a través de un gráfico de barras.

```
# Visualización para la duración promedio de los viajes por tipo de
usuario y día de semana
all_data12_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title="Duración promedio de los viajes")

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



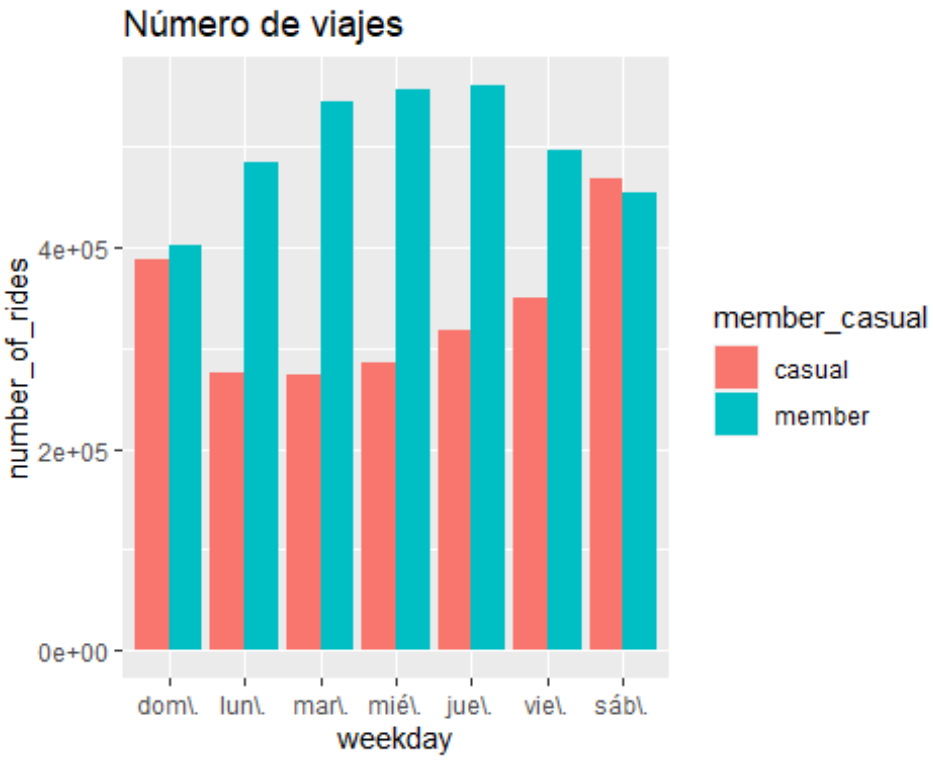
Estos hallazgos indican que, en general, tanto para usuarios casuales como para miembros, los fines de semana (sábado y domingo) tienden a tener viajes más largos en promedio. Además, la variación en la duración promedio es más pronunciada para los usuarios casuales en comparación con los miembros.

Visualización del número de viajes

Después de analizar el número de viajes por día de la semana para usuarios "Casuales" y "Miembros," se procede a representar visualmente esta información a través de un gráfico de barras.

```
# Visualización del numero de viaje por tipo de usuario
all_data12_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title="Número de viajes")
```

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



Los usuarios casuales tienden a utilizar más el servicio los fines de semana, con un pico notable los sábados. Por otro lado, los miembros tienden a utilizar el servicio en días laborables, con el jueves como el día de mayor actividad.

Distribución geográfica del inicio y final de los viajes

En esta sección, exploramos la distribución geográfica de los puntos de inicio de los viajes para los usuarios “Casuales” y “Miembros”.

```
install.packages("maps")
```

```
## Installing package into 'C:/Users/samp_/AppData/Local/R/win-
library/4.3'
## (as 'lib' is unspecified)
```

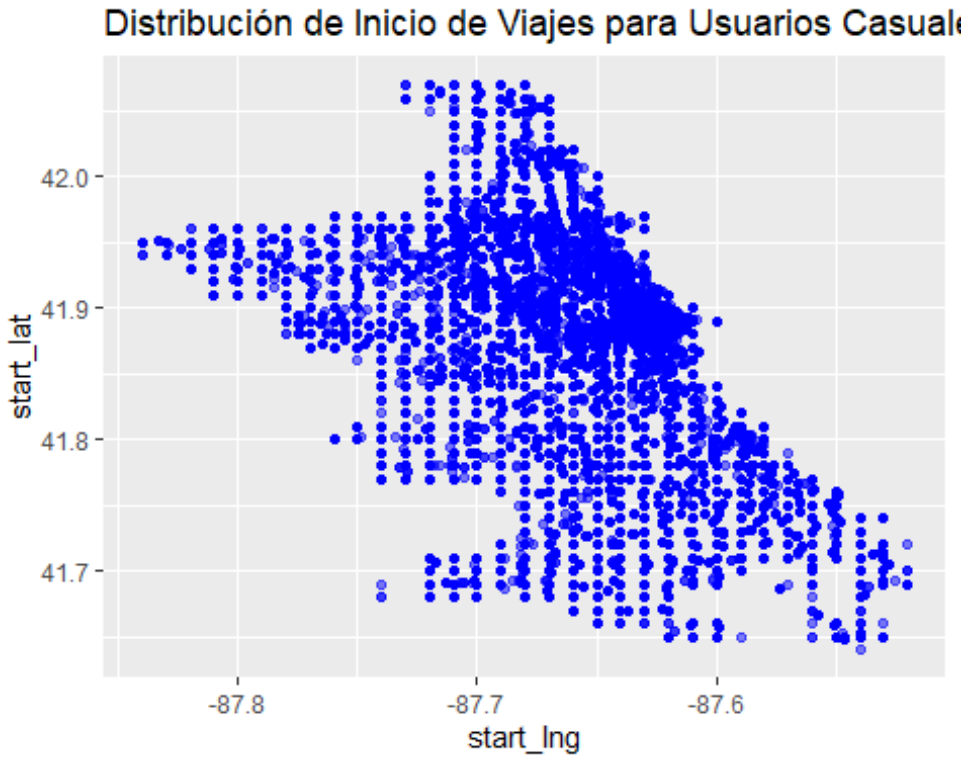
```
## package 'maps' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\samp_\AppData\Local\Temp\RtmpyKzt9n\downloaded_packages
```

```
library("maps")
```

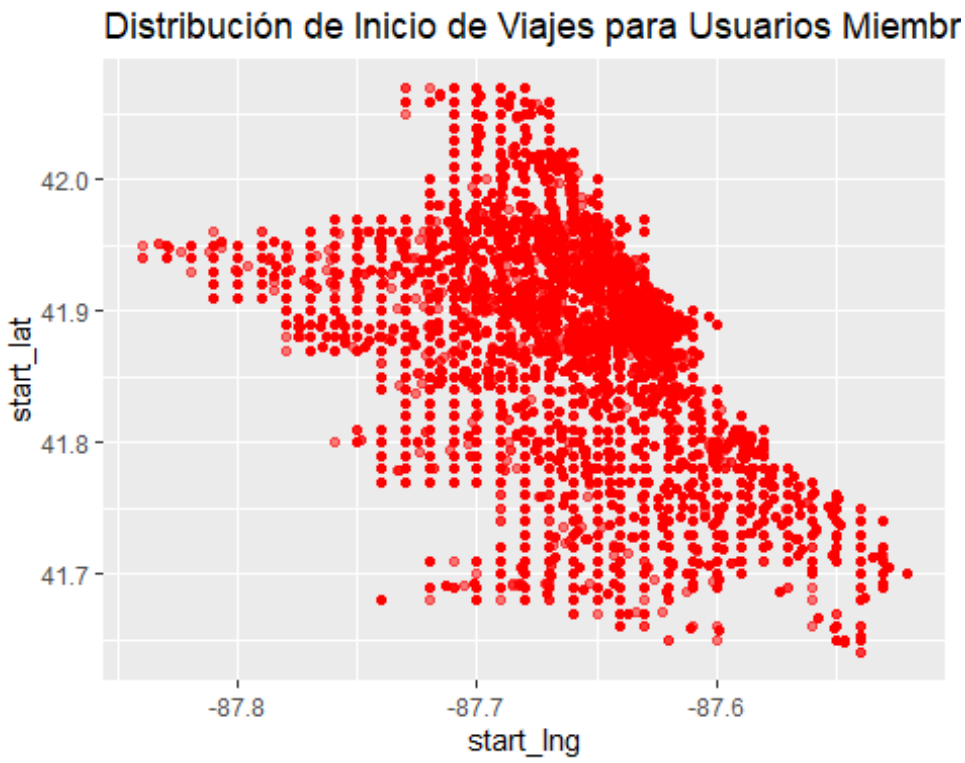
```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```r
# Filtrar el dataframe por usuarios "casuales" y "member"
casual_data_v2 <- all_data12_v2[all_data12_v2$member_casual == "casual",
]
member_data_v2 <- all_data12_v2[all_data12_v2$member_casual == "member",
]
# Crear el mapa para usuarios "casuales"
ggplot(casual_data_v2, aes(x = start_lng, y = start_lat)) +
  geom_point(alpha = 0.5, color = "blue") +
  labs(title = "Distribución de Inicio de Viajes para Usuarios Casuales
con valores atípicos")
```




```r
# Crear el mapa para usuarios "member"
ggplot(member_data_v2, aes(x = start_lng, y = start_lat)) +
  geom_point(alpha = 0.5, color = "red") +
  labs(title = "Distribución de Inicio de Viajes para Usuarios Miembros
con valores atípicos")
```



* La distribución geográfica de inicio de los viajes es similar tanto para los usuarios casuales como para los miembros. Esta delimitación puede estar asociada a la "zona de servicio" de Cyclistic. * Se observa una región de mayor densidad de puntos de inicio de viajes, que corresponde al centro de la ciudad de Chicago. Esto indica que esta área es especialmente popular para iniciar viajes en bicicleta, ya sea para usuarios casuales o miembros.

V. Datos atípicos

En el proceso de exploración y análisis de los datos (all_data12_v2), se identificó un sesgo hacia la derecha de la duración promedio de los viajes. Este sesgo sugiere la posible presencia de valores atípicos, es decir, datos que se apartan significativamente de la mayoría de las observaciones. Un ejemplo de tal valor atípico es el registro con una duración máxima de 2,486,835 segundos.

Se abordó la detección de valores atípicos, un aspecto crítico para comprender la influencia de estos datos en nuestras conclusiones sobre el uso de bicicletas por parte de "miembros" y "casuales."

Se utilizo el rango intercuartíl (IQR), para identificar los valores atípicos.

5.1 Determinación de valores atípicos

Determinar el límite inferior y superior para hallar valores atípicos.

```r
#Calcular el primer y tercer cuartil (Q1 y Q3) y el rango intercuartílico
(IQR)
Q1 <- quantile(all_data12_v2$ride_length, 0.25)
Q3 <- quantile(all_data12_v2$ride_length, 0.75)
IQR <- Q3 - Q1

# Definir los límites superior e inferior para detectar valores atípicos
limite_inferior <- Q1 - 1.5 * IQR
limite_superior <- Q3 + 1.5 * IQR

# Crear un nuevo marco de datos excluyendo los valores atípicos
datos_sin_atipicos <- all_data12_v2 %>%
  filter(ride_length >= limite_inferior & ride_length <= limite_superior)
```

A continuación, se presentan los hallazgos clave:

Cuartiles y duración de Viajes:

- El primer cuartil (Q1) reveló que el 25% de los viajes, tanto de "casuales" como de "miembros," tuvieron una duración igual o inferior a 339 segundos (5 minutos y 39 segundos).

- El tercer cuartil (Q3) indicó que el 75% de los viajes tuvieron una duración igual o inferior a 1075 segundos (17 minutos y 55 segundos).

Límites para la detección de valores atípicos:

- El límite inferior se estableció en -775 segundos. Es importante destacar que el marco de datos no incluye duraciones negativas.

- El límite superior se definió en 2179 segundos (alrededor de 36 minutos y 19 segundos).

Valores Atípicos:

- Se consideran valores atípicos todos aquellos que se ubican por debajo del límite inferior y por encima del límite superior establecido.

Consideración No. 3: Gestión de Valores Atípicos

La naturaleza precisa de los valores atípicos no pudo ser confirmada en esta etapa del análisis. Por lo tanto, se ha tomado la decisión de considerar estos valores en el proceso de análisis para evaluar su efecto en las conclusiones finales. Para lograr esto, se llevará a cabo una comparación entre los siguientes marcos de datos:

- Todos los datos (all_data12_v2): Este marco de datos representa la versión original que incluye tanto los datos no atípicos como los valores atípicos.

- Datos no atípicos(datos_sin_atípicos): Se trata del marco de datos que excluye los valores atípicos, brindando una visión de los datos sin la influencia de estos valores extremos.
- Datos atípicos(valores_atípicos): Este marco de datos se limita exclusivamente a los valores atípicos, lo que permitirá un análisis enfocado en entender la naturaleza y el impacto de estos datos fuera de lo común.

La comparación entre estos tres conjuntos de datos nos proporcionará una comprensión más completa de cómo los valores atípicos pueden afectar nuestras conclusiones sobre el uso de bicicletas por parte de usuarios "miembros" y "casuales."

5.2. Marco de datos "datos_sin_atipicos"

En el apartado 5.1 (Determinar Datos Atípicos), se creó el marco de datos "datos_sin_atípicos". En el que se excluyen los valores atípicos.

```
dim(datos_sin_atipicos)

## [1] 5425219      19
```

Este marco de datos está compuesto por un total de 5,425,219 registros y consta de 19 columnas.

5.3 Marco de datos "valores_atípicos"

```
valores_atipicos <- all_data12_v2 %>%
  filter(ride_length < limite_inferior | ride_length > limite_superior)
# Identificar las dimensiones del marco de datos "valores_atípicos"
dim(valores_atipicos)

## [1] 433739     19
```

A partir de la información contenida en "all_data12_v2," se generó un nuevo marco de datos denominado "valores_atípicos." Este conjunto de datos reúne exclusivamente los registros previamente identificados como valores atípicos en términos de duración de viajes. Está compuesto por 433,739 registros, lo que representa aproximadamente el 7.40% del total de los datos originales contenidos en "all_data12_v2."

Determinar la cantidad de valores atípicos para "casual" y "member"

```
# Contar la cantidad de valores atípicos por tipo de usuario
cantidad_atipicos_por_usuario <- table(valores_atipicos$member_casual)

# Mostrar los resultados
print(cantidad_atipicos_por_usuario)

##
## casual member
## 312542 121197
```

Los resultados revelan que, de los valores considerados atípicos, el 72% (312,542 registros) corresponde a usuarios clasificados como “casuales,” mientras que el 28% restante (121,197 registros) corresponde a usuarios “miembros”.

VI. Análisis del marco de datos “datos_sin_atípicos” (Datos No Atípicos)

6.1 Resumen estadístico

A continuación, un resumen de las estadísticas de duración de los viajes, considerando a ambos tipos de usuarios

```
summary(datos_sin_atipicos$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   322.0   554.0   680.1   928.0  2179.0
```

Comparativa de las estadísticas de duración de viajes entre usuarios “Miembros” y “Casuales”

```
aggregate(datos_sin_atipicos$ride_length ~
datos_sin_atipicos$member_casual, FUN = mean)
```

```
##   datos_sin_atipicos$member_casual datos_sin_atipicos$ride_length
## 1                           casual                        769.0971
## 2                           member                        626.1686
```

```
aggregate(datos_sin_atipicos$ride_length ~
datos_sin_atipicos$member_casual, FUN = median)
```

```
##   datos_sin_atipicos$member_casual datos_sin_atipicos$ride_length
## 1                           casual                             650
## 2                           member                             502
```

```
aggregate(datos_sin_atipicos$ride_length ~
datos_sin_atipicos$member_casual, FUN = max)
```

```
##   datos_sin_atipicos$member_casual datos_sin_atipicos$ride_length
## 1                           casual                            2179
## 2                           member                            2179
```

```
aggregate(datos_sin_atipicos$ride_length ~
datos_sin_atipicos$member_casual, FUN = min)
```

```
##   datos_sin_atipicos$member_casual datos_sin_atipicos$ride_length
## 1                           casual                               0
## 2                           member                               0
```

Una comparativa de las estadisticas, entre los marcos de datos “all_data12_v2” y “datos_sin_atípicos” se presenta a continuación:

Duración promedio de Viajes:

- Para los usuarios “Casuales” la duración promedio disminuyó 55%. Al pasar de 28 minutos con 30 segundos (en “all_data12_v2”) a 12 minutos con 49 segundos (en “datos_sin_atípicos”).
- En el caso de los usuarios “Miembros” la duración promedio de los viajes disminuyó 16 %. Al pasar de 12 minutos con 30 segundos (en “all_data12_v2”) a 10 minutos con 20 segundos (en “datos_sin_atípicos”)

Comportamiento de usuarios miembros:

- Es interesante destacar que, a pesar de la exclusión de valores atípicos, el comportamiento en el uso de bicicletas por parte de los usuarios miembros se mantiene relativamente consistente en términos de duración de viajes.

En resumen, al eliminar los valores atípicos, se observa una reducción en la duración promedio de los viajes para ambos grupos de usuarios, siendo más notable en el caso de los usuarios casuales. Esto puede indicar una mayor variabilidad en los viajes de los usuarios casuales en comparación con los usuarios miembros, que mantienen una duración de viaje más constante.

6.2 Duración promedio de los viajes por día de semana

Duración promedio de los viajes por día de semana para usuarios “Casuales” y “Miembros”.

```
# Duración promedio de los viajes
aggregate(datos_sin_atipicos$ride_length ~
datos_sin_atipicos$member_casual + datos_sin_atipicos$day_of_week, FUN =
mean)
```

```
##    datos_sin_atipicos$member_casual datos_sin_atipicos$day_of_week
## 1                            casual                         domingo
## 2                            member                         domingo
## 3                            casual                           lunes
## 4                            member                           lunes
## 5                            casual                          martes
## 6                            member                          martes
## 7                            casual                       miércoles
## 8                            member                       miércoles
## 9                            casual                          jueves
## 10                           member                          jueves
## 11                           casual                         viernes
## 12                           member                         viernes
## 13                           casual                          sábado
## 14                           member                          sábado
##    datos_sin_atipicos$ride_length
## 1                        828.1910
## 2                        661.5589
## 3                        751.8217
```

```
## 4                          604.6716
## 5                          718.3392
## 6                          608.8689
## 7                          717.3431
## 8                          613.1713
## 9                          729.6426
## 10                         618.4139
## 11                         761.1863
## 12                         619.8562
## 13                         830.9603
## 14                         672.3972
```

A continuación, se presenta una comparativa de las preferencias de los usuarios casuales y miembros al pasar del marco de datos "all_data12_v2" al "datos_sin_atípicos."

Usuarios casuales:

- Mayor duración promedio: La preferencia cambió de domingo a sábado, y se redujo significativamente en un 58.6%, pasando de 33 minutos y 27 segundos a 13 minutos y 51 segundos.
- Menor duración promedio: El miércoles mantuvo la menor duración promedio en ambos marcos de datos. Sin embargo, esta duración disminuyó en un 50.54%, pasando de 24 minutos y 10 segundos a 11 minutos y 57 segundos.

Usuarios miembros:

- Mayor duración promedio: El sábado mantuvo la mayor duración promedio en ambos marcos de datos.Sin embargo, esta duración disminuyó en un 19.57%, pasando de 13 minutos y 56 segundos a 11 minutos y 12 segundos.
- Menor duración promedio: La preferencia cambió de miércoles a lunes y disminuyó en un 15.25%, pasando de 11 minutos y 53 segundos a 10 minutos y 05 segundos.

Estos cambios sugieren una adaptación en las preferencias de los usuarios en función de la eliminación de valores atípicos, lo que podría indicar una mayor estabilidad en la duración promedio de los viajes para ambos grupos de usuarios.

6.3 Número de viajes por día de semana

Identificar la relación entre el número de viajes por usuario y el día de semana

```
datos_sin_atipicos %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)
```

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

## # A tibble: 14 × 4
## # Groups:   member_casual [2]
##    member_casual weekday  number_of_rides average_duration
##    <chr>         <ord>              <int>            <dbl>
##  1 casual        "dom\\."          323374             828.
##  2 casual        "lun\\."          237186             752.
##  3 casual        "mar\\."          243161             718.
##  4 casual        "mié\\."          256619             717.
##  5 casual        "jue\\."          284862             730.
##  6 casual        "vie\\."          307924             761.
##  7 casual        "sáb\\."          392585             831.
##  8 member        "dom\\."          382122             662.
##  9 member        "lun\\."          469410             605.
## 10 member        "mar\\."          528953             609.
## 11 member        "mié\\."          540944             613.
## 12 member        "jue\\."          544165             618.
## 13 member        "vie\\."          481148             620.
## 14 member        "sáb\\."          432766             672.
```

A continuación, se presenta una comparativa de las preferencias de los usuarios casuales y miembros al pasar del marco de datos “all_data12_v2” al “datos_sin_atípicos.”. Los porcentajes representan la relación entre la cantidad de viajes de un día específico, con respecto al total de viajes de los 7 días para cada usuario.

Usuarios casuales:

- Mayor cantidad de viajes: El sábado mantuvo el mayor número de viajes, en ambos marcos de datos. Sin embargo, esta cantidad disminuyó, pasando de 467,923 (19.84%) a 392,585 (19.19%) viajes.
- Menor cantidad de Viajes: La preferencia cambió de martes a lunes, y disminuyó al pasar de 276,648 (11.73%) a 237,184 (11.59%).

Usuarios miembros:

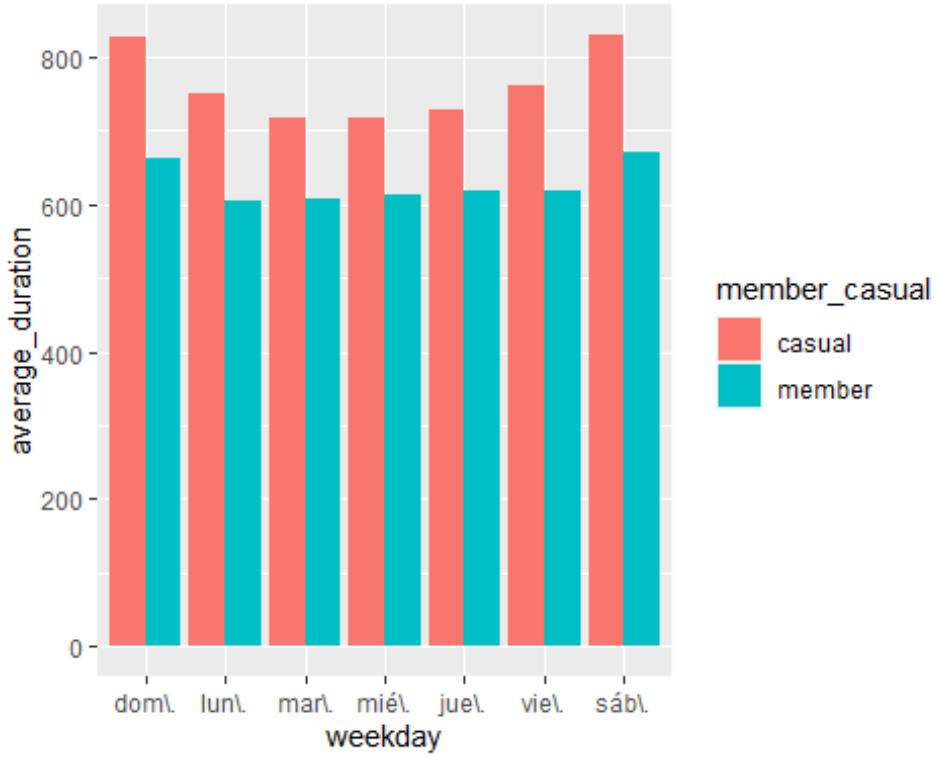
- Mayor cantidad de viajes: El jueves mantuvo el mayor número de viajes. Sin embargo, esta cantidad disminuyó, pasando de 560,877 (16.02%) a 544,165 (16.10%).
- Menor cantidad de viajes: El domingo mantuvo el menor número de viajes, en ambos marcos de datos. Sin embargo, esta cantidad disminuyó, pasando de 402,066 (11.49%) a 382,122 (11.31%) viajes.

6.4 Visualización de resultados

Visualización de la duración promedio

```
## Visualización de la duración promedio
datos_sin_atipicos %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```

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



A continuación, se presentan los hallazgos considerando únicamente el marco de datos "datos_sin_atípicos".

Usuarios casuales

- Los fines de semana son los días con la mayor duración promedio de los viajes.
- La duración promedio se encuentra entre un máximo de 13 minutos y 51 segundos (sábado) y un mínimo de 11 minutos y 57 segundos (miércoles).

Usuarios miembros

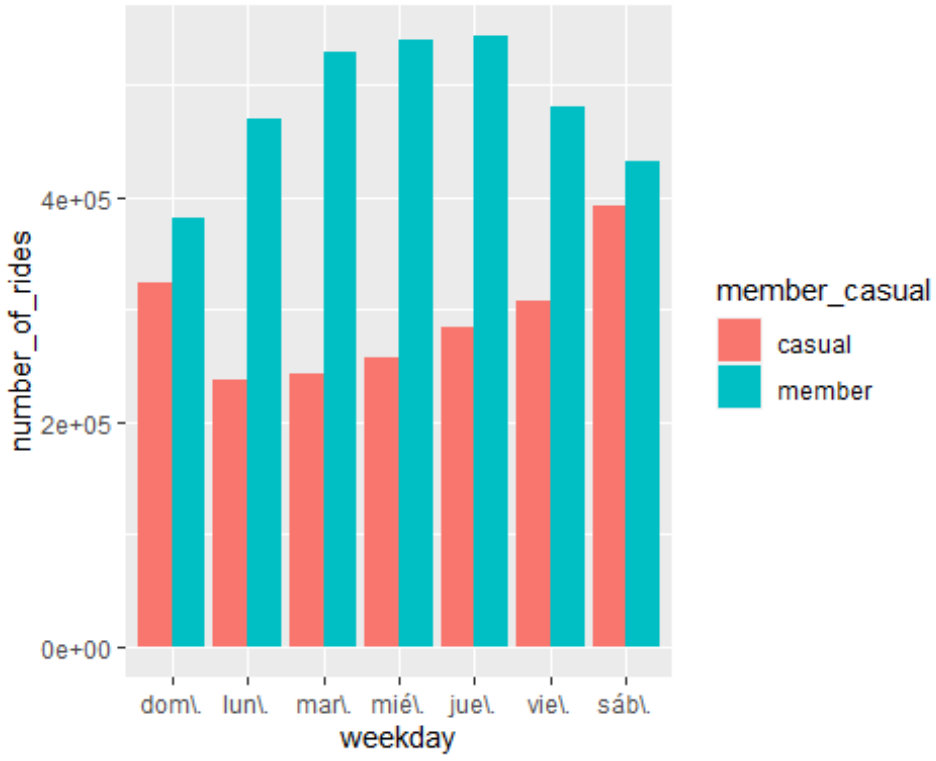
- Los fines de semana son los días con la mayor duración promedio de los viajes.
- La duración promedio se encuentra entre un máximo de 11 minutos y 12 segundos (sábado) y un mínimo de 10 minutos y 05 segundos (lunes).

Los usuarios miembros mantienen una duración promedio bastante consistente a lo largo de la semana laboral, es decir, de lunes a viernes. En cambio, los usuarios casuales muestran variaciones significativas en sus duraciones promedio en esos días.

Visualización del número de viajes

```
datos_sin_atipicos %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge")
```

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



A continuación se presentan los hallazgos considerando unicamente el marco de datos “datos_sin_atípicos”.

Usuarios casuales

- Los fines de semana son los días con la mayor cantidad de viajes.

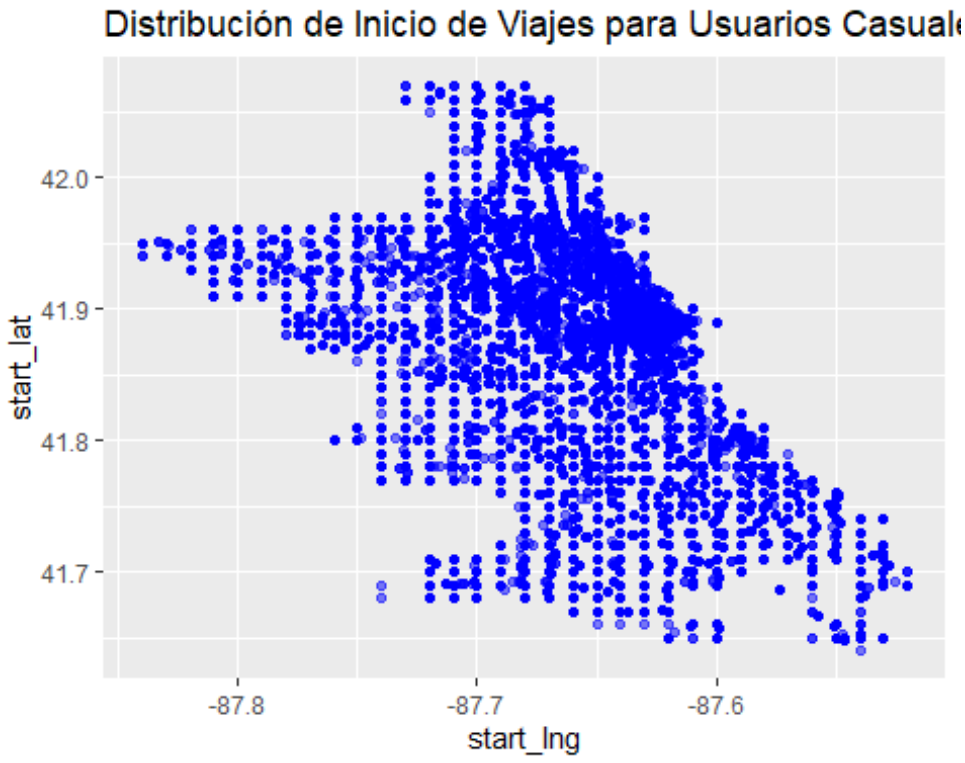
- La cantidad de viajes se encuentra entre un máximo de 392,585 (sábado) y un mínimo de 237,186 (lunes).

Usuarios miembros.

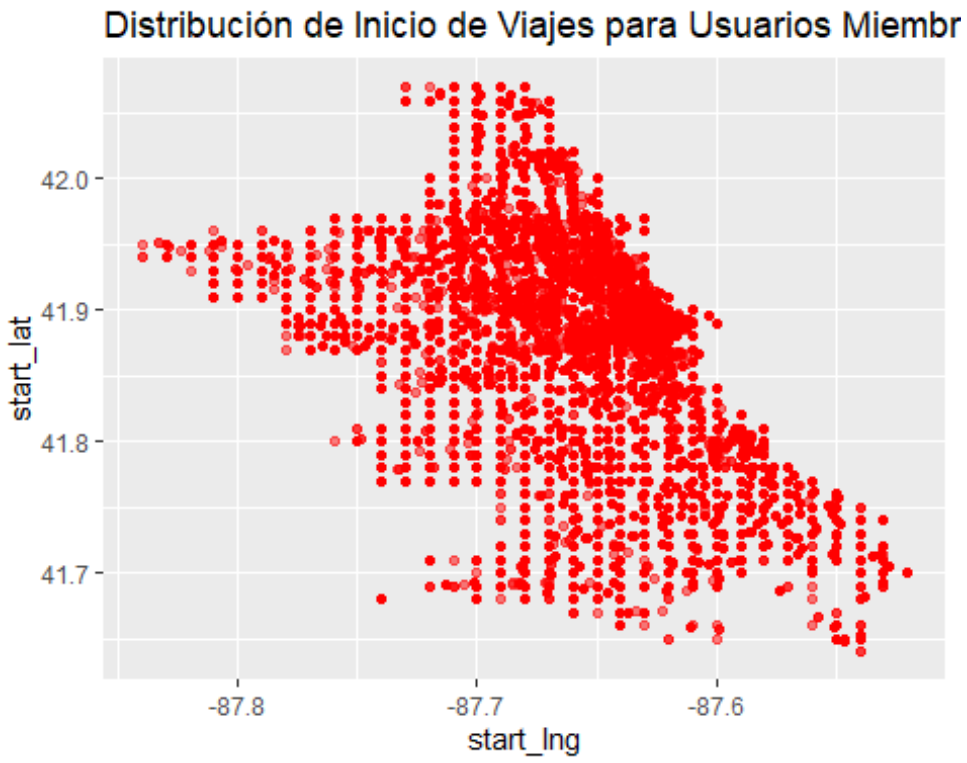
- Los días con la mayor cantidad de viajes son miércoles y jueves.
- La cantidad de viajes se encuentra entre un máximo de 544,165 (jueves) y un mínimo de 382,122 (domingo).

Distribución geográfica del inicio y final de los viajes

```
install.packages("maps")

## Warning: package 'maps' is in use and will not be installed

library("maps")

# Filtrar el dataframe por usuarios "casuales" y "member"
casual_datos_sin_atipicos <-
datos_sin_atipicos[datos_sin_atipicos$member_casual == "casual", ]
member_datos_sin_atipicos <-
datos_sin_atipicos[datos_sin_atipicos$member_casual == "member", ]

# Crear el mapa para usuarios "casuales"
ggplot(casual_datos_sin_atipicos, aes(x = start_lng, y = start_lat)) +
  geom_point(alpha = 0.5, color = "blue") +
  labs(title = "Distribución de Inicio de Viajes para Usuarios Casuales
sin valores atípicos")
```



```r
# Crear el mapa para usuarios "member"
ggplot(member_datos_sin_atipicos, aes(x = start_lng, y = start_lat)) +
  geom_point(alpha = 0.5, color = "red") +
  labs(title = "Distribución de Inicio de Viajes para Usuarios Miembros
sin valores atípicos")
```



A continuación, se presentan los hallazgos considerando únicamente el marco de datos "datos_sin_atípicos".

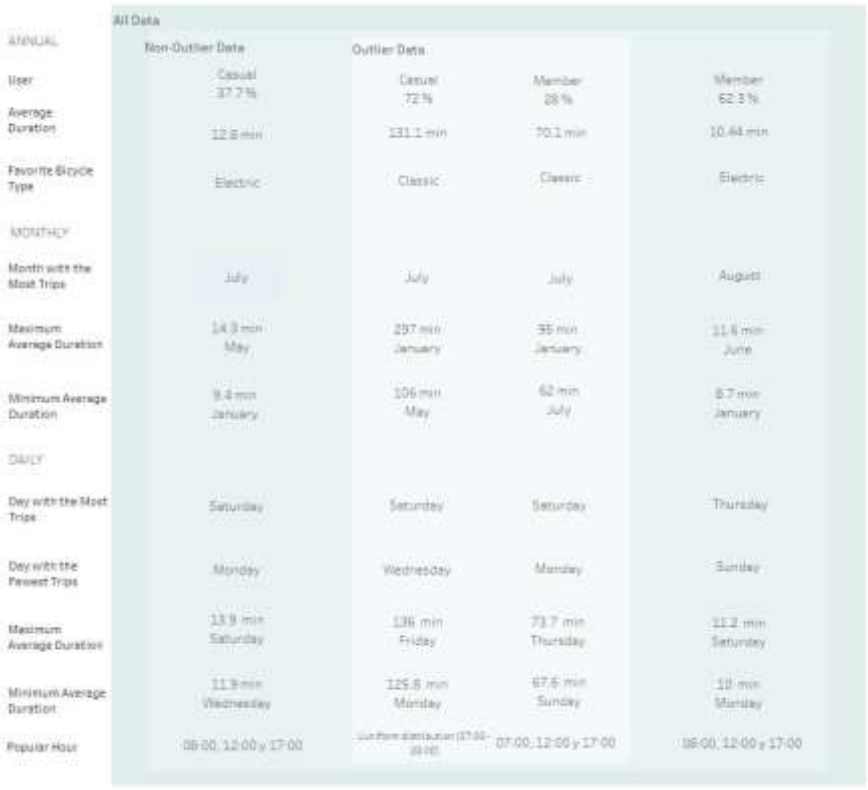
La distribución de inicio de viajes para ambos tipos de usuarios es notablemente similar, lo cual puede atribuirse a la zona geográfica limitada en la que operan las estaciones de "Cyclistic". Además, se observa claramente una zona de alta densidad de inicio de viajes, situada en el centro de Chicago.

VII. Los resultados finales se presentan en una visualización disponible en "Tableau Public"

Con la finalidad de integrar los resultados e identificar los cambios que se presentaron al considerar o no los datos atípicos, así como un análisis que se enfoca exclusivamente en los valores identificados como "atípicos". Está disponible una "historia" en Tableau Public, la cual se puede consultar en el siguiente enlace.

Google Capstone Project: Cyclistic

Un resumen de estos resultados se presenta a continuación.



VIII. Conclusión

Los valores atípicos tienen gran relavancia sobre las preferencias en el uso de bicicletas tanto para los usuarios “miembros” y “casuales”.

Con la información y hallazgos actuales, no fue posible determinar si los valores atípicos representan datos válidos o errores en los registros. Se requiere información adicional, como la distancia recorrida, la velocidad promedio, el número de usuarios únicos, entre otros, para llevar a cabo un análisis más profundo y utilizar métodos estadísticos robustos.

Por lo tanto, se plantean las siguientes hipótesis:

1. Los valores atípicos son errores: En este caso, se eliminarían los datos atípicos, y las conclusiones clave se basarían en el conjunto de datos “Datos sin Atípicos”. En este conjunto se destacan las diferencias, pero también se observan algunas similitudes en el uso de bicicletas entre los usuarios Casuales y Miembros.

2. Los valores atípicos son válidos: Se consideraría la segmentación de usuarios. Es decir, los usuarios casuales se dividirían en dos grupos: aquellos que pertenecen al conjunto de datos “Datos Sin Atípicos”, que utilizan el servicio de manera recurrente, que se pueden atribuir a usuarios locales; y aquellos que forman parte del conjunto de datos “Datos Atípicos”, que usan el servicio de forma ocasional, recreativa o turística, que se puden a tribuir a usuarios foráneos.

Si el objetivo es convertir a usuarios casuales en miembros de Cyclistic, las estrategias deberían centrarse en aquellos usuarios casuales que hacen uso recurrente del servicio, es decir, los usuarios locales. Es poco probable que los usuarios que utilizan el servicio de forma esporádica o que están de visita en la ciudad opten por adquirir una membresía anual.

Una forma de validar esta hipótesis es mediante la implementación de paquetes especiales dirigidos a los visitantes de la ciudad (foráneos) y posteriormente medir la demanda que generan.