Google Data Analytics Course Capstone Project: Case Study 1, Cyclistic Bike Share

For this case study I followed the steps of the data analysis process: **ask**, **prepare**, **process**, **analyze**, **share**, and **act**.

Scenario

You are a junior data analyst working in the marketing analyst team at Cyclistic, a bike-share company in Chicago. The director of marketing believes the company's future success depends on maximizing the number of annual memberships. Therefore, your team wants to understand how casual riders and annual members use Cyclistic bikes differently. From these insights, your team will design a new marketing strategy to convert casual riders into annual members. But first, Cyclistic executives must approve your recommendations, so they must be backed up with compelling data insights and professional data visualizations.

Characters and teams

- **Cyclistic**: A bike-share program that features more than 5,800 bicycles and 600 docking stations. Cyclistic sets itself apart by also offering reclining bikes, hand tricycles, and cargo bikes, making bike-share more inclusive to people with disabilities and riders who can't use a standard two-wheeled bike. The majority of riders opt for traditional bikes; about 8% of riders use the assistive options. Cyclistic users are more likely to ride for leisure, but about 30% use them to commute to work each day.
- **Lily Moreno**: The director of marketing and your manager. Moreno is responsible for the development of campaigns and initiatives to promote the bike-share program. These may include email, social media, and other channels.
- **Cyclistic marketing analytics team**: A team of data analysts who are responsible for collecting, analyzing, and reporting data that helps guide Cyclistic marketing strategy. You joined this team six months ago and have been busy learning about Cyclistic's mission and business goals — as well as how you, as a junior data analyst, can help Cyclistic achieve them.
- **Cyclistic executive team:** The notoriously detail-oriented executive team will decide whether to approve the recommended marketing program.

About the company

In 2016, Cyclistic launched a successful bike-share offering. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system anytime.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes, and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

Cyclistic's finance analysts have concluded that annual members are much more profitable than casual riders. Although the pricing flexibility helps Cyclistic attract more customers, Moreno believes that maximizing the number of annual members will be key to future growth. Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a very good chance to convert casual riders into members. She notes that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.

Moreno has set a clear goal: Design marketing strategies aimed at converting casual riders into annual members. In order to do that, however, the marketing analyst team needs to better understand how annual members and casual riders differ, why casual riders would buy a membership, and how digital media could affect their marketing tactics. Moreno and her team are interested in analyzing the Cyclistic historical bike trip data to identify trends.

Key Insights

- **Three different pricing plans:** single-ride passes, full-day passes, and annual memberships
- Customers who purchase single-ride or full-day passes are referred to as **casual riders**
- Customers who purchase annual memberships are referred to as **members.**
- Cyclistic's finance analysts have concluded that **annual members are much more profitable than casual riders.** Although the pricing flexibility helps Cyclistic attract more customers.
- Moreno believes that **maximizing the number of annual members** will be **key to future growth**
- **Rather than creating a marketing campaign that targets all-new customers**, Moreno believes there is a very good chance to **convert casual riders into members**.

Key questions

1. How do annual members and casual riders use Cyclistic bikes differently?

2. Why would casual riders buy Cyclistic annual memberships?

3. How can Cyclistic use digital media to influence casual riders to become members?

Business task:

Identify differences between casual riders and annual members to find ways to convert casual riders into annual members.

Data information:

For this case study, I used 12 datasets representing Cyclistic's last 12 months of historical trip data. These datasets were made available by Motivate International Inc and contain real historical trip data from their customers. None of these datasets contain customers' personally identifiable information. All the data used in this case study is publicly available here under this license.

All the datasets contained the same 13 attributes (columns), and each of them contained hundreds of thousands of tuples (rows).

To improve processing time, I decided to remove the attributes start_station_id and end_station_id since their values represent the same entities as start_station_name and end_station_name.

Tools used:

For my case study I decided to use R. I decided to use R because I knew it was better suited to handle large datasets than spreadsheets. I could also have used SQL, but while I am familiar with it and other programming languages, it was my first time working with R and I wished to learn more about it.

Preparing RStudio:

```
> install.packages("tidyverse")

> library("tidyverse")
── Attaching packages ───────────────────
✓ ggplot2 3.3.6     ✓ purrr   0.3.4
✓ tibble  3.1.7     ✓ dplyr   1.0.9
✓ tidyr   1.2.0     ✓ stringr 1.4.0
✓ readr   2.1.2     ✓ forcats 0.5.1
── Conflicts ────────────────────────────
✕ dplyr::filter() masks stats::filter()
✕ dplyr::lag()    masks stats::lag()
>
```

Loading and combining datasets:

```
csv_files <- list.files(path = "..../Datasets", recursive = TRUE, full.names=TRUE)
merged_datasets <- do.call(rbind, lapply(csv_files, read.csv))
```

```
View(merged_datasets)
```

	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
1	99FEC93BA843FB20	electric_bike	2021-06-13 14:31:28	2021-06-13 14:34:11					41.80000	-87.59000	41.80000	-87.60000	member
2	06048DCFC8520CAF	electric_bike	2021-06-04 11:18:02	2021-06-04 11:24:19					41.79000	-87.59000	41.80000	-87.60000	member
3	9598066F68045DF2	electric_bike	2021-06-04 09:49:35	2021-06-04 09:55:34					41.80000	-87.60000	41.79000	-87.59000	member
4	B03C0FE48C412214	electric_bike	2021-06-03 19:56:05	2021-06-03 20:21:55					41.78000	-87.58000	41.80000	-87.60000	member
5	B9EEA89F8FEE73B7	electric_bike	2021-06-04 14:05:51	2021-06-04 14:09:59					41.80000	-87.59000	41.79000	-87.59000	member
6	62B943CEAAA420BA	electric_bike	2021-06-03 19:32:01	2021-06-03 19:38:46					41.78000	-87.58000	41.78000	-87.58000	member
7	7E2546FBA79C46EE	electric_bike	2021-06-10 16:30:10	2021-06-10 16:36:21					41.79000	-87.60000	41.79000	-87.59000	member
8	3DDF3BBF6C4C3C89	electric_bike	2021-06-10 17:00:30	2021-06-10 17:06:48					41.79000	-87.59000	41.80000	-87.59000	member
9	2608805637155AB6	electric_bike	2021-06-10 12:46:16	2021-06-10 12:55:02					41.93000	-87.67000	41.94000	-87.68000	member
10	AF529C946F28ED42	electric_bike	2021-06-23 17:57:29	2021-06-23 18:06:40			Michigan Ave & Oak St	13042	41.88000	-87.61000	41.90105	-87.62370	member
11	E6010941FB92E4A6	electric_bike	2021-06-22 19:28:02	2021-06-22 19:39:48					41.87000	-87.62000	41.87000	-87.64000	member
12	1149C0723F7AFFD5	electric_bike	2021-06-29 17:35:49	2021-06-29 17:55:11					41.90000	-87.63000	41.90000	-87.68000	member
13	8762DB62099E6011	electric_bike	2021-06-05 14:55:05	2021-06-05 15:13:29					41.89000	-87.62000	41.88000	-87.62000	member
14	BE3AC77CBFF17E6A	electric_bike	2021-06-05 14:05:00	2021-06-05 14:09:01					41.89000	-87.62000	41.89000	-87.62000	member
15	8E9F2CB0893B96A0	electric_bike	2021-06-05 13:39:04	2021-06-05 13:57:21					41.88000	-87.62000	41.89000	-87.62000	member
16	6344B71B7BB6E09E	electric_bike	2021-06-22 18:52:53	2021-06-22 18:59:13					41.79000	-87.59000	41.80000	-87.60000	member
17	59CE9444E2ED2530	electric_bike	2021-06-02 10:30:11	2021-06-02 10:37:03					41.79000	-87.60000	41.80000	-87.59000	member
18	2D6929277855EBE5	electric_bike	2021-06-08 13:49:03	2021-06-08 13:53:01					41.79000	-87.60000	41.78000	-87.60000	member

Showing 1 to 19 of 6,629,980 entries, 13 total columns

Creating a copy to work with

```
historical_data <- merged_datasets
```

Removing duplicate tuples

```
> historical_data <- historical_data[!duplicated(historical_data$ride_id),]
> tibble(historical_data)
# A tibble: 6,629,980 × 13
   ride_id          rideable_type started_at          ended_at            start_station_name start_station_id end_station_name        end_s…¹ start…² start…³ end_lat end_lng membe…⁴
   <chr>            <chr>         <chr>               <chr>               <chr>              <chr>            <chr>                   <chr>     <dbl>   <dbl>   <dbl>   <dbl> <chr>
 1 99FEC93BA843FB20 electric_bike 2021-06-13 14:31:28 2021-06-13 14:34:11 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 2 06048DCFC8520CAF electric_bike 2021-06-04 11:18:02 2021-06-04 11:24:19 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 3 9598066F68045DF2 electric_bike 2021-06-04 09:49:35 2021-06-04 09:55:34 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 4 B03C0FE48C412214 electric_bike 2021-06-03 19:56:05 2021-06-03 20:21:55 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 5 B9EEA89F8FEE73B7 electric_bike 2021-06-04 14:05:51 2021-06-04 14:09:59 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 6 62B943CEAAA420BA electric_bike 2021-06-03 19:32:01 2021-06-03 19:38:46 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 7 7E2546FBA79C46EE electric_bike 2021-06-10 16:30:10 2021-06-10 16:36:21 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 8 3DDF3BBF6C4C3C89 electric_bike 2021-06-10 17:00:30 2021-06-10 17:06:48 ""                 ""               ""                      ""         41.8   -87.6    41.8   -87.6 member
 9 2608805637155AB6 electric_bike 2021-06-10 12:46:16 2021-06-10 12:55:02 ""                 ""               ""                      ""         41.9   -87.7    41.9   -87.7 member
10 AF529C946F28ED42 electric_bike 2021-06-23 17:57:29 2021-06-23 18:06:40 ""                 ""               "Michigan Ave & Oak St" "13042"    41.9   -87.6    41.9   -87.6 member
```

R returned the same number of tuples meaning that the dataset did not contain duplicated tuples.

Dropping columns from dataset:

Before I dropped any columns, I verified that each value for start_station_name had a corresponding value in start_station_id and vice versa. I repeated the same process for end_station_id and end_station_name.

Only 3 tuples had unpaired values.

```
historical_data %>%
  select(ride_id, start_station_name,start_station_id) %>%
  filter(start_station_name == "" & !start_station_id == "") %>%
  view()
```

	ride_id	start_station_name	start_station_id
1	176105D1F8A1216B		13221
2	DE82A15026BA3056		20215
3	EE197EDA4CF8CFE5		WL-008

I filled each missing value using the primary key and the corresponding station ID

```
historical_data$start_station_name[historical_data$ride_id == "176105D1F8A1216B"] <- "Wood St & Milwaukee Ave"
historical_data$start_station_name[historical_data$ride_id == "DE82A15026BA3056"] <- "Hegewisch Metra Station"
historical_data$start_station_name[historical_data$ride_id == "EE197EDA4CF8CFE5"] <- "Clinton St & Roosevelt Rd"
```

After running the query for a second time it did not return any tuples, meaning that all the missing values were fixed correctly.

```
historical_data %>%
  select(ride_id, start_station_name,start_station_id) %>%
  filter(start_station_name == "" & !start_station_id == "") %>%
  view()
```

	ride_id	start_station_name	start_station_id
	No data available in table		

Dropping columns

```
> historical_data <- subset(historical_data, select = -c(start_station_id, end_station_id))
> tibble(historical_data)
# A tibble: 6,629,980 x 11
   ride_id          rideable_type started_at          ended_at            start_station_name end_station_name        start_lat start_lng end_lat end_lng member_casual
   <chr>            <chr>         <chr>               <chr>               <chr>              <chr>                       <dbl>     <dbl>   <dbl>   <dbl> <chr>
 1 99FEC93BA843FB20 electric_bike 2021-06-13 14:31:28 2021-06-13 14:34:11 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 2 06048DCFC8520CAF electric_bike 2021-06-04 11:18:02 2021-06-04 11:24:19 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 3 9598066F68045DF2 electric_bike 2021-06-04 09:49:35 2021-06-04 09:55:34 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 4 B03C0FE48C412214 electric_bike 2021-06-03 19:56:05 2021-06-03 20:21:55 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 5 B9EEA89F8FEE73B7 electric_bike 2021-06-04 14:05:51 2021-06-04 14:09:59 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 6 62B943CEAAA420BA electric_bike 2021-06-03 19:32:01 2021-06-03 19:38:46 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 7 7E2546FBA79C46EE electric_bike 2021-06-10 16:30:10 2021-06-10 16:36:21 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 8 3DDF3BBF6C4C3C89 electric_bike 2021-06-10 17:00:30 2021-06-10 17:06:48 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 9 2608805637155AB6 electric_bike 2021-06-10 12:46:16 2021-06-10 12:55:02 ""                 ""                           41.9     -87.7    41.9   -87.7 member
10 AF529C946F28ED42 electric_bike 2021-06-23 17:57:29 2021-06-23 18:06:40 ""                 "Michigan Ave & Oak St"      41.9     -87.6    41.9   -87.6 member
```

Parsing data

Previously I had observed that the attributes started_at and ended_at were not formatted correctly. Their values were strings when they should have been dates.

```
historical_data$started_at <- as.POSIXlt(historical_data$started_at,format = "%Y-%m-%d %H:%M:%S")
historical_data$ended_at <- as.POSIXlt(historical_data$ended_at,format = "%Y-%m-%d %H:%M:%S")
```

```
> tibble(historical_data)
# A tibble: 6,629,980 x 11
   ride_id          rideable_type started_at          ended_at            start_station_name end_station_name        start_lat start_lng end_lat end_lng member_casual
   <chr>            <chr>         <dttm>              <dttm>              <chr>              <chr>                       <dbl>     <dbl>   <dbl>   <dbl> <chr>
 1 99FEC93BA843FB20 electric_bike 2021-06-13 14:31:28 2021-06-13 14:34:11 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 2 06048DCFC8520CAF electric_bike 2021-06-04 11:18:02 2021-06-04 11:24:19 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 3 9598066F68045DF2 electric_bike 2021-06-04 09:49:35 2021-06-04 09:55:34 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 4 B03C0FE48C412214 electric_bike 2021-06-03 19:56:05 2021-06-03 20:21:55 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 5 B9EEA89F8FEE73B7 electric_bike 2021-06-04 14:05:51 2021-06-04 14:09:59 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 6 62B943CEAAA420BA electric_bike 2021-06-03 19:32:01 2021-06-03 19:38:46 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 7 7E2546FBA79C46EE electric_bike 2021-06-10 16:30:10 2021-06-10 16:36:21 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 8 3DDF3BBF6C4C3C89 electric_bike 2021-06-10 17:00:30 2021-06-10 17:06:48 ""                 ""                           41.8     -87.6    41.8   -87.6 member
 9 2608805637155AB6 electric_bike 2021-06-10 12:46:16 2021-06-10 12:55:02 ""                 ""                           41.9     -87.7    41.9   -87.7 member
10 AF529C946F28ED42 electric_bike 2021-06-23 17:57:29 2021-06-23 18:06:40 ""                 "Michigan Ave & Oak St"      41.9     -87.6    41.9   -87.6 member
```

Adding additional columns

To help my analysis, I added the columns ride_length_m (ride length in minutes), hour_of_day (hour at which the ride started), day_of_week (day in which the ride started), and month_of_year (the month in which the ride started).

ride_length_m

```
historical_data <- historical_data %>%
  mutate(ride_length_m = round((as.numeric(historical_data$ended_at - historical_data$started_at)/60) , digits = 2))

historical_data %>%
  select(ride_id, ride_length_m) %>%
  head(10)
```

```
           ride_id ride_length_m
1  99FEC93BA843FB20          2.72
2  06048DCFC8520CAF          6.28
3  9598066F68045DF2          5.98
4  B03C0FE48C412214         25.83
5  B9EEA89F8FEE73B7          4.13
6  62B943CEAAA420BA          6.75
7  7E2546FBA79C46EE          6.18
8  3DDF3BBF6C4C3C89          6.30
9  2608805637155AB6          8.77
10 AF529C946F28ED42          9.18
```

hour_of_day

```
historical_data <- historical_data %>%
  mutate(hour_of_day = strftime(historical_data$started_at, "%H"))

historical_data %>%
  select(ride_id, ride_length_m, hour_of_day) %>%
  head(10)
```

```
           ride_id ride_length_m hour_of_day
1  99FEC93BA843FB20          2.72          14
2  06048DCFC8520CAF          6.28          11
3  9598066F68045DF2          5.98          09
4  B03C0FE48C412214         25.83          19
5  B9EEA89F8FEE73B7          4.13          14
6  62B943CEAAA420BA          6.75          19
7  7E2546FBA79C46EE          6.18          16
8  3DDF3BBF6C4C3C89          6.30          17
9  2608805637155AB6          8.77          12
10 AF529C946F28ED42          9.18          17
```

day_of_week

```
historical_data <- historical_data %>%
  mutate(day_of_week = strftime(historical_data$started_at, "%u"))

historical_data %>%
  select(ride_id, ride_length_m, hour_of_day, day_of_week) %>%
  head(10)
```

```
           ride_id ride_length_m hour_of_day day_of_week
1  99FEC93BA843FB20          2.72          14           7
2  06048DCFC8520CAF          6.28          11           5
3  9598066F68045DF2          5.98          09           5
4  B03C0FE48C412214         25.83          19           4
5  B9EEA89F8FEE73B7          4.13          14           5
6  62B943CEAAA420BA          6.75          19           4
7  7E2546FBA79C46EE          6.18          16           4
8  3DDF3BBF6C4C3C89          6.30          17           4
9  2608805637155AB6          8.77          12           4
10 AF529C946F28ED42          9.18          17           3
```

month_of_year

```
historical_data <- historical_data %>%
  mutate(month_of_year = strftime(historical_data$started_at, "%m"))
```

```
historical_data %>%
  select(ride_id, ride_length_m, hour_of_day, day_of_week, month_of_year) %>%
  head(10)

            ride_id ride_length_m hour_of_day day_of_week month_of_year
1  99FEC93BA843FB20          2.72          14           7            06
2  06048DCFC8520CAF          6.28          11           5            06
3  95980066F68045DF2         5.98          09           5            06
4  B03C0FE48C412214         25.83          19           4            06
5  B9EEA89F8FEE73B7          4.13          14           5            06
6  62B943CEAAA420BA          6.75          19           4            06
7  7E2546FBA79C46EE          6.18          16           4            06
8  3DDF3BBF6C4C3C89          6.30          17           4            06
9  2608805637155AB6          8.77          12           4            06
10 AF529C946F28ED42          9.18          17           3            06
```

Verifying data integrity

Dataset summary

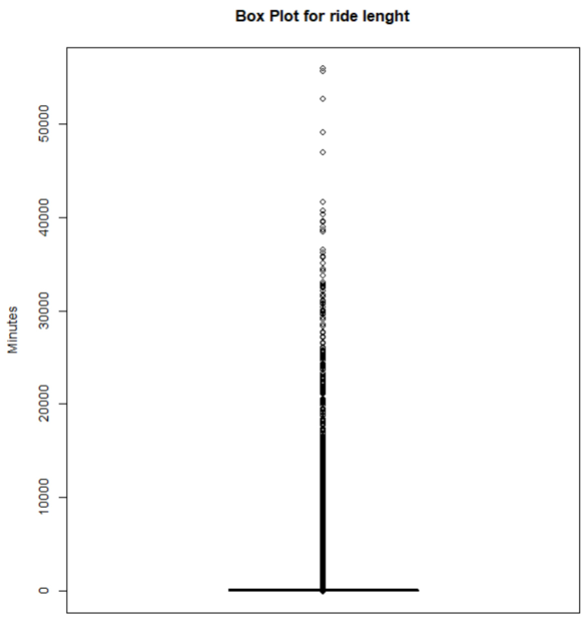
```
> summary(historical_data)
   ride_id          rideable_type        started_at                      ended_at                       start_station_name end_station_name    start_lat       start_lng
 Length:6629980     Length:6629980     Min.   :2021-06-01 00:00:38.00   Min.   :2021-06-01 00:06:22.00   Length:6629980     Length:6629980     Min.   :41.64   Min.   :-87.84
 Class :character   Class :character   1st Qu.:2021-08-05 07:46:52.75   1st Qu.:2021-08-05 08:02:29.00   Class :character   Class :character   1st Qu.:41.88   1st Qu.:-87.66
 Mode  :character   Mode  :character   Median :2021-10-09 20:38:10.00   Median :2021-10-09 20:59:35.50   Mode  :character   Mode  :character   Median :41.90   Median :-87.64
                                       Mean   :2021-11-22 04:59:00.73   Mean   :2021-11-22 05:19:55.99                                         Mean   :41.90   Mean   :-87.65
                                       3rd Qu.:2022-04-11 22:40:07.25   3rd Qu.:2022-04-11 23:03:12.75                                         3rd Qu.:41.93   3rd Qu.:-87.63
                                       Max.   :2022-06-30 23:59:58.00   Max.   :2022-07-13 04:21:06.00                                         Max.   :45.64   Max.   :-73.80

    end_lat         end_lng        member_casual      ride_length_m       hour_of_day        day_of_week        month_of_year
 Min.   :41.39   Min.   :-88.97   Length:6629980     Min.   : -137.42   Length:6629980     Length:6629980     Length:6629980
 1st Qu.:41.88   1st Qu.:-87.66   Class :character   1st Qu.:    6.40   Class :character   Class :character   Class :character
 Median :41.90   Median :-87.64   Mode  :character   Median :   11.40   Mode  :character   Mode  :character   Mode  :character
 Mean   :41.90   Mean   :-87.65                      Mean   :   20.92
 3rd Qu.:41.93   3rd Qu.:-87.63                      3rd Qu.:   20.65
 Max.   :42.17   Max.   :-87.49                      Max.   :55944.15
 NA's   :6091    NA's   :6091
```

While looking at the summary of the dataset, I noticed that the attributes end_lat and end_lng had 6091 null values. In addition, by looking at the differences between the min and the 1st quartile and max and the 3rd quartile in ride_length_m. I realized the dataset had outliers. A negative ride length and an extreme ride length time of 55944.15 minutes or 39.9 days.

Creating a boxplot

```
boxplot(historical_data$ride_length_m,
        main="Box Plot for ride lenght",
        ylab = "Minutes"
)
```



After looking at the boxplot, I saw that the previously observed outliers were not unique occurrences.

Removing outliers

For my data cleaning process, I considered any ride length values below the 1th percentile (0.43 minutes) and above the 99th percentile (122.25 minutes) outliers.

```
> print(quantile(historical_data$ride_length_m, 0.010))
  1%
0.43
> print(quantile(historical_data$ride_length_m, 0.990))
   99%
122.25
```

Removing values outside acceptable range

```
historical_data <- historical_data[!(historical_data$ride_length_m < 0.43),]
historical_data <- historical_data[!(historical_data$ride_length_m > 122.25),]
```

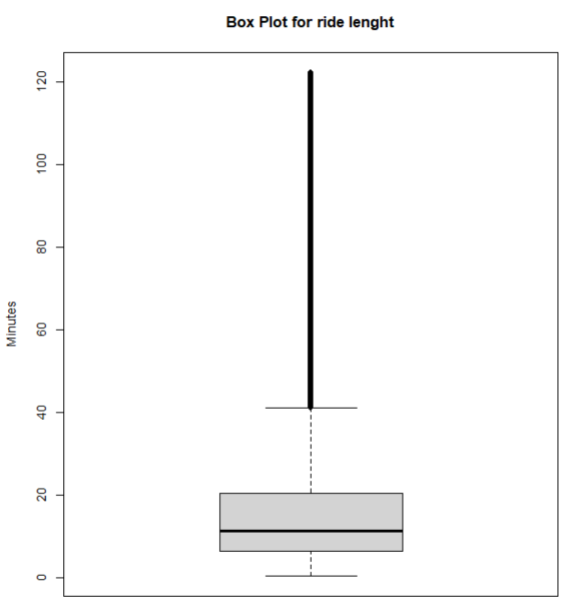
New dataset summary

```
> summary(historical_data)
   ride_id          rideable_type        started_at                        ended_at                         start_station_name end_station_name     start_lat        start_lng
 Length:6498540     Length:6498540     Min.   :2021-06-01 00:00:38.00   Min.   :2021-06-01 00:06:22.00   Length:6498540     Length:6498540     Min.   :41.64   Min.   :-87.84
 Class :character   Class :character   1st Qu.:2021-08-05 13:29:57.75   1st Qu.:2021-08-05 13:48:59.00   Class :character   Class :character   1st Qu.:41.88   1st Qu.:-87.66
 Mode  :character   Mode  :character   Median :2021-10-10 01:37:25.00   Median :2021-10-10 01:52:27.50   Mode  :character   Mode  :character   Median :41.90   Median :-87.64
                                       Mean   :2021-11-22 07:07:08.74   Mean   :2021-11-22 07:23:35.38                                         Mean   :41.90   Mean   :-87.65
                                       3rd Qu.:2022-04-11 19:52:03.00   3rd Qu.:2022-04-11 20:06:18.75                                         3rd Qu.:41.93   3rd Qu.:-87.63
                                       Max.   :2022-06-30 23:59:53.00   Max.   :2022-07-01 01:36:57.00                                         Max.   :45.64   Max.   :-73.80

    end_lat         end_lng        member_casual      ride_length_m     hour_of_day        day_of_week        month_of_year
 Min.   :41.39   Min.   :-88.97   Length:6498540     Min.   :  0.43   Length:6498540     Length:6498540     Length:6498540
 1st Qu.:41.88   1st Qu.:-87.66   Class :character   1st Qu.:  6.48   Class :character   Class :character   Class :character
 Median :41.90   Median :-87.64   Mode  :character   Median : 11.40   Mode  :character   Mode  :character   Mode  :character
 Mean   :41.90   Mean   :-87.65                      Mean   : 16.44
 3rd Qu.:41.93   3rd Qu.:-87.63                      3rd Qu.: 20.35
 Max.   :42.13   Max.   :-87.49                      Max.   :122.25
 NA's   :816     NA's   :816
```

After removing 2% of the dataset, I got more reasonable min/max ride length values. I based my decision of what was reasonable on the statement made in the case study scenario that mentioned that Cyclistic users were more likely to ride for leisure and me to commute to work each day.

New boxplot



By looking at the summary and the new boxplot it could had seem like the dataset still had outliers. I did not remove any more values since I did not consider them to be erroneous values, it may just be that some people enjoyed longer rides. In addition, the number of tuples with ride lengths longer than 40 minutes just accounted for 7% of the whole dataset.

Lastly, while I still had 816 tuples with null values in the attributes end_lat and end_lng, I decided to not remove those tuples since I was not planning on using those attributes in my analysis.

Looking for empty strings

While I was learning how to detect null values in the dataset, I noticed that some empty cells were not considered null even though they were empty. Then I realized that those cells were not null they just had empty strings. I did not look for empty strings in start_station_id and end_station_id since empty strings were intended in those attributes.

Query to find empty strings

```
historical_data %>%
  select(ride_id, rideable_type, started_at, ended_at, start_lat, start_lng, end_lat, end_lng, member_casual, ride_length_m, hour_of_day, day_of_week
         , month_of_year) %>%
  filter(ride_id == "" | rideable_type == "" | start_lat == "" | start_lng == "" | end_lat == "" |
           end_lng == "" | member_casual == "" | ride_length_m == "" | hour_of_day == "" | day_of_week == "" | month_of_year == "") %>%
  view()
```

ride_id	rideable_type	started_at	ended_at	start_lat	start_lng	end_lat	end_lng	member_casual	ride_length_m	hour_of_day	day_of_week	month_of_year
No data available in table												

The query did not return any tuples. Meaning that there were no empty strings in the tested attributes.

Looking for unique values

hour_of_day

```
> unique(historical_data$hour_of_day)
 [1] "14" "11" "09" "19" "16" "17" "12" "13" "18" "10" "22" "21" "15" "02" "23" "00" "07" "08" "05" "20" "01" "06" "03" "04"
```

The values of this attribute were accurate and unique.

day_of_week

```
> unique(historical_data$day_of_week)
[1] "7" "5" "4" "3" "2" "6" "1"
```

The values of this attribute were accurate and unique.

month_of_year

```
> unique(historical_data$month_of_year)
 [1] "06" "07" "08" "09" "10" "11" "12" "01" "02" "03" "04" "05"
```

The values of this attribute were accurate and unique.

rideable_type

```
> unique(historical_data$rideable_type)
[1] "electric_bike" "classic_bike"  "docked_bike"
```

The values of this attribute were accurate and unique.

member_casual

```
> unique(historical_data$member_casual)
[1] "member" "casual"
```

The values of this attribute were accurate and unique.

Analysis

Rides distribution difference between casual and member riders

```
historical_data %>%
  group_by(member_casual) %>%
  summarise(total = length(ride_id), "percentage" = round(length(ride_id) / nrow(historical_data) * 100, digits = 2)) %>%
  view()
```

```
chart = ggplot(historical_data, aes(member_casual, fill=member_casual)) +
  geom_bar() +
  labs(y="Total rides",x="Rider type", title="Rides distrubutsion Casual vs Members")+
  geom_text(aes(label = ..count..), stat = "count", vjust = 1.5, colour = "white")


chart +                                        # Modify formatting of axis
  scale_y_continuous(labels = comma)
```

	member_casual	total	percentage
1	casual	2843549	43.76
2	member	3654991	56.24

Table 1 Ride distribution - Members vs Casuals

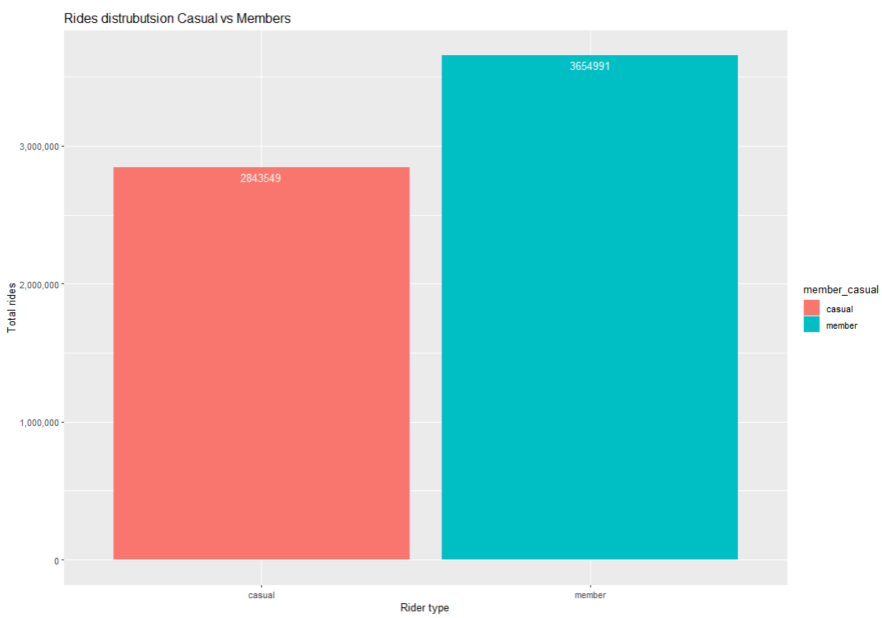


Figure 1 Ride distribution - Members vs Casuals

Most rides were done by member riders (56.24% of total rides). That is a difference of 12.48% of the total rides or 811,442 more rides than casual riders.

Ride length difference between casual and member riders

```
historical_data %>%
  group_by(member_casual) %>%
  summarise(ride_length_m_mean = mean(ride_length_m)) %>%
  view()
```

	member_casual	ride_length_m_mean
1	casual	21.51312
2	member	12.50027

Table 2 Ride length - Members vs Casuals

On average, casual riders with a ride length average of 21.51 minutes took longer rides than member riders that have a ride length average of 12.5 minutes.

Rides per type of bike

```
historical_data %>%
  group_by(rideable_type) %>%
  summarise(total = length(ride_id), "percentage" = round(length(ride_id) / nrow(historical_data) * 100, digits = 2),
            "casual" = sum(member_casual == "casual"),"% casual" = round((sum(member_casual == "casual")
                                                            /length(ride_id))*100, digits = 2),
            "member" = sum(member_casual == "member"),"% member"= round((sum(member_casual == "member")
                                                            /length(ride_id))*100, digits = 2)) %>%
  view()

chart = ggplot(historical_data, aes(rideable_type, fill=member_casual)) +
  geom_bar() +
  labs(y="Total rides",x="Bike type", title="Rides distrubutsion - Bike type")+
  facet_wrap(vars(member_casual))+
  geom_text(aes(label = ..count..), stat = "count", vjust = 1.5, colour = "white")

chart +                                   # Modify formatting of axis
  scale_y_continuous(labels = comma)
```

	rideable_type	total	percentage	casual	% casual	member	% member
1	classic_bike	3563663	54.84	1370146	38.45	2193517	61.55
2	docked_bike	280197	4.31	280197	100.00	0	0.00
3	electric_bike	2654680	40.85	1193206	44.95	1461474	55.05

Table 3 Ride distribution - Bike type

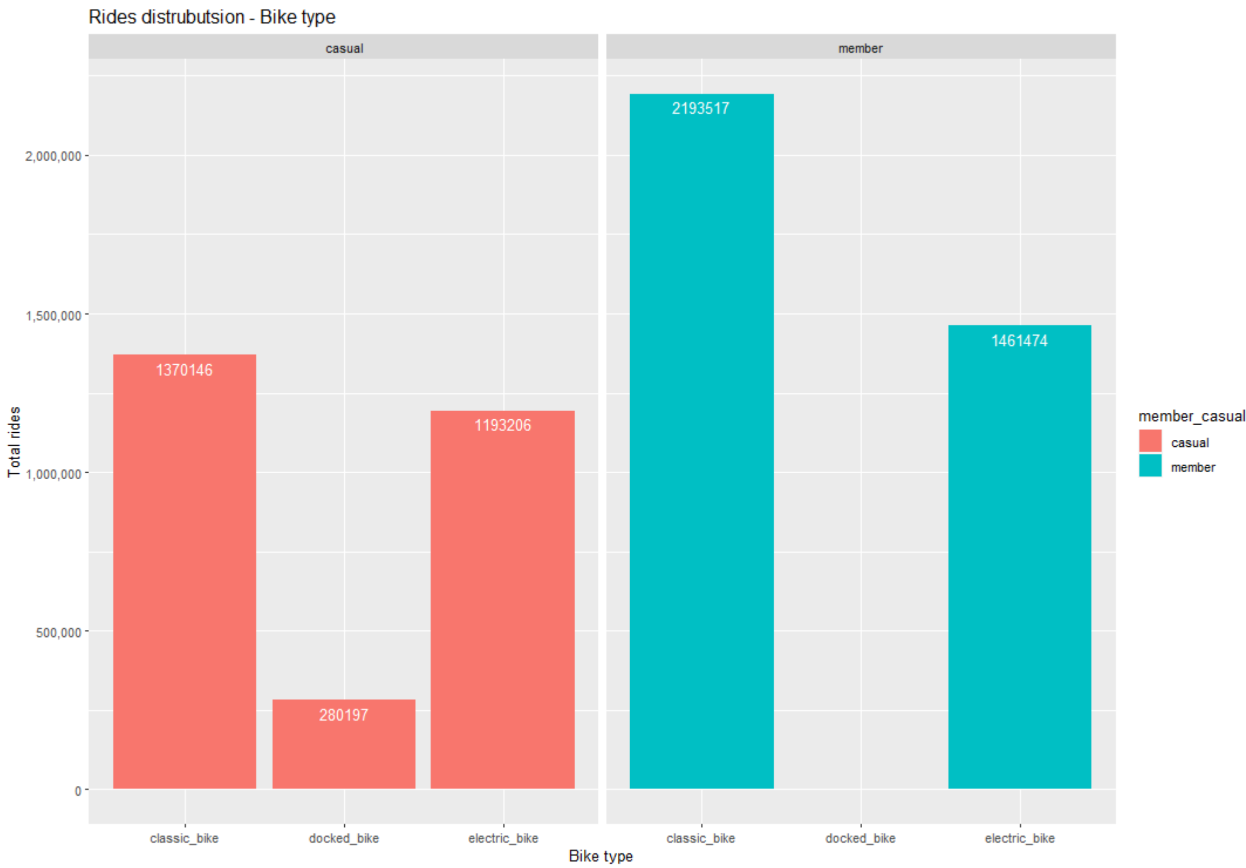


Figure 2 Ride distribution - Bike type

The most popular bike type was the classic bike (54.84 % of total rides), followed by the electric bike (40.85 % of total rides) and lastly the docked bike (4.31% of total rides). Member riders performed more rides than casual riders using classic bikes and electric bikes. 61.55% of the classic bike rides were done by member riders while 38.45% were done by casual riders. Similarly, 55.05% of the electric bike rides were done by member riders while 40.85% were done by casual riders. All the docked bike rides were done by casual riders. I suspect that all the docked bike rides were removed with the outliers maybe due to members not docking their bikes back at the stations.

Distributions by hour of the day

```
historical_data %>%
  group_by(hour_of_day) %>%
  summarise(total = length(ride_id), "percentage" = round(length(ride_id) / nrow(historical_data) * 100, digits = 2),
            "casual" = sum(member_casual == "casual"),"% casual" = round((sum(member_casual == "casual")
                                                                       /length(ride_id))*100, digits = 2),
            "member" = sum(member_casual == "member"),"% member"= round((sum(member_casual == "member")
                                                                      /length(ride_id))*100, digits = 2),
            "% Difference" = abs(round((sum(member_casual == "casual")/length(ride_id))*100, digits = 2)
                                 -round((sum(member_casual == "member")/length(ride_id))*100, digits = 2))) %>%
  view()
```

```
chart = ggplot(historical_data, aes(hour_of_day, fill=member_casual)) +
  geom_bar() +
  labs(y="Total rides",x="Hour of day", title="Rides distribution - Hour")+
  facet_wrap(vars(member_casual))+
  coord_flip()

chart +                                    # Modify formatting of axis
  scale_y_continuous(labels = comma)
```

	hour_of_day	total	percentage	casual	% casual	member	% member	% Difference
1	00	98968	1.52	59534	60.15	39434	39.85	20.30
2	01	67277	1.04	41844	62.20	25433	37.80	24.40
3	02	42855	0.66	27985	65.30	14870	34.70	30.60
4	03	24360	0.37	15562	63.88	8798	36.12	27.76
5	04	21037	0.32	11010	52.34	10027	47.66	4.68
6	05	51728	0.80	14839	28.69	36889	71.31	42.62
7	06	129778	2.00	31033	23.91	98745	76.09	52.18
8	07	243526	3.75	57278	23.52	186248	76.48	52.96
9	08	294341	4.53	76566	26.01	217775	73.99	47.98
10	09	243805	3.75	86432	35.45	157373	64.55	29.10
11	10	265734	4.09	115122	43.32	150612	56.68	13.36
12	11	330352	5.08	149513	45.26	180839	54.74	9.48
13	12	385013	5.92	176247	45.78	208766	54.22	8.44
14	13	391279	6.02	186918	47.77	204361	52.23	4.46
15	14	394710	6.07	194015	49.15	200695	50.85	1.70
16	15	445417	6.85	210110	47.17	235307	52.83	5.66
17	16	541759	8.34	231906	42.81	309853	57.19	14.38
18	17	649129	9.99	267415	41.20	381714	58.80	17.60
19	18	562072	8.65	243586	43.34	318486	56.66	13.32
20	19	421186	6.48	190700	45.28	230486	54.72	9.44
21	20	301651	4.64	141486	46.90	160165	53.10	6.20
22	21	241389	3.71	119619	49.55	121770	50.45	0.90
23	22	204240	3.14	110891	54.29	93349	45.71	8.58
24	23	146934	2.26	83938	57.13	62996	42.87	14.26

Table 4 – Ride distribution – Hour of the day

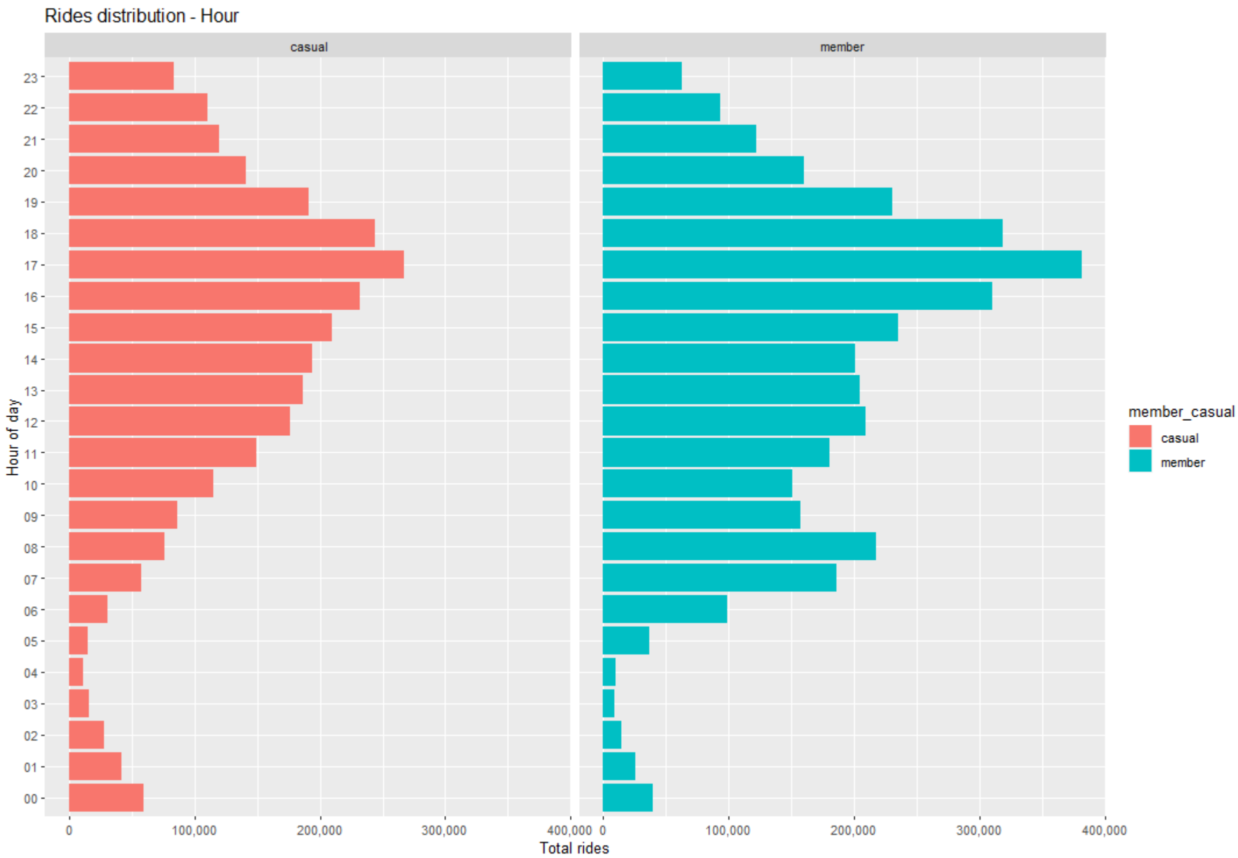


Figure 3 Ride distribution – Hour of the day

Both rider types had similar hourly ride distributions, but from 5 am to 9 am member riders were 29.10% to 53% more active than casual riders. I believe this could be due to member riders using the bikes to commute to work more than casual riders do. Similarly, from 12 am to 3 am casual riders were 20.30% to 30.60% more active that member riders. Both riders showed an increasing ride trend that stopped at 5 pm (17) the most active hour for both rider types.

Distribution by day of the week

```
historical_data %>%
  group_by(day_of_week) %>%
  summarise(total = length(ride_id), "percentage" = round(length(ride_id) / nrow(historical_data) * 100, digits = 2),
            "casual" = sum(member_casual == "casual"),"% casual" = round((sum(member_casual == "casual")
                                                              /length(ride_id))*100, digits = 2),
            "member" = sum(member_casual == "member"),"% member"= round((sum(member_casual == "member")
                                                             /length(ride_id))*100, digits = 2),
            "% Difference" = abs(round((sum(member_casual == "casual")/length(ride_id))*100, digits = 2)
                                 -round((sum(member_casual == "member")/length(ride_id))*100, digits = 2))) %>%
  view()
```

```
chart = ggplot(historical_data, aes(day_of_week, fill=member_casual)) +
  geom_bar() +
  labs(y="Total rides",x="Hour of day", title="Rides distribution - Hour")+
  facet_wrap(vars(member_casual))+
  geom_text(aes(label = ..count..), stat = "count", vjust = 1.5, colour = "white")


chart +                                    # Modify formatting of axis
  scale_y_continuous(labels = comma)
```

	day_of_week	total	percentage	casual	% casual	member	% member	% Difference
1	1	835682	12.86	328952	39.36	506730	60.64	21.28
2	2	890549	13.70	317488	35.65	573061	64.35	28.70
3	3	900567	13.86	326055	36.21	574512	63.79	27.58
4	4	925798	14.25	357226	38.59	568572	61.41	22.82
5	5	915631	14.09	404985	44.23	510646	55.77	11.54
6	6	1075580	16.55	591619	55.00	483961	45.00	10.00
7	7	954733	14.69	517224	54.17	437509	45.83	8.34

Table 5 Ride distribution - Day of the week

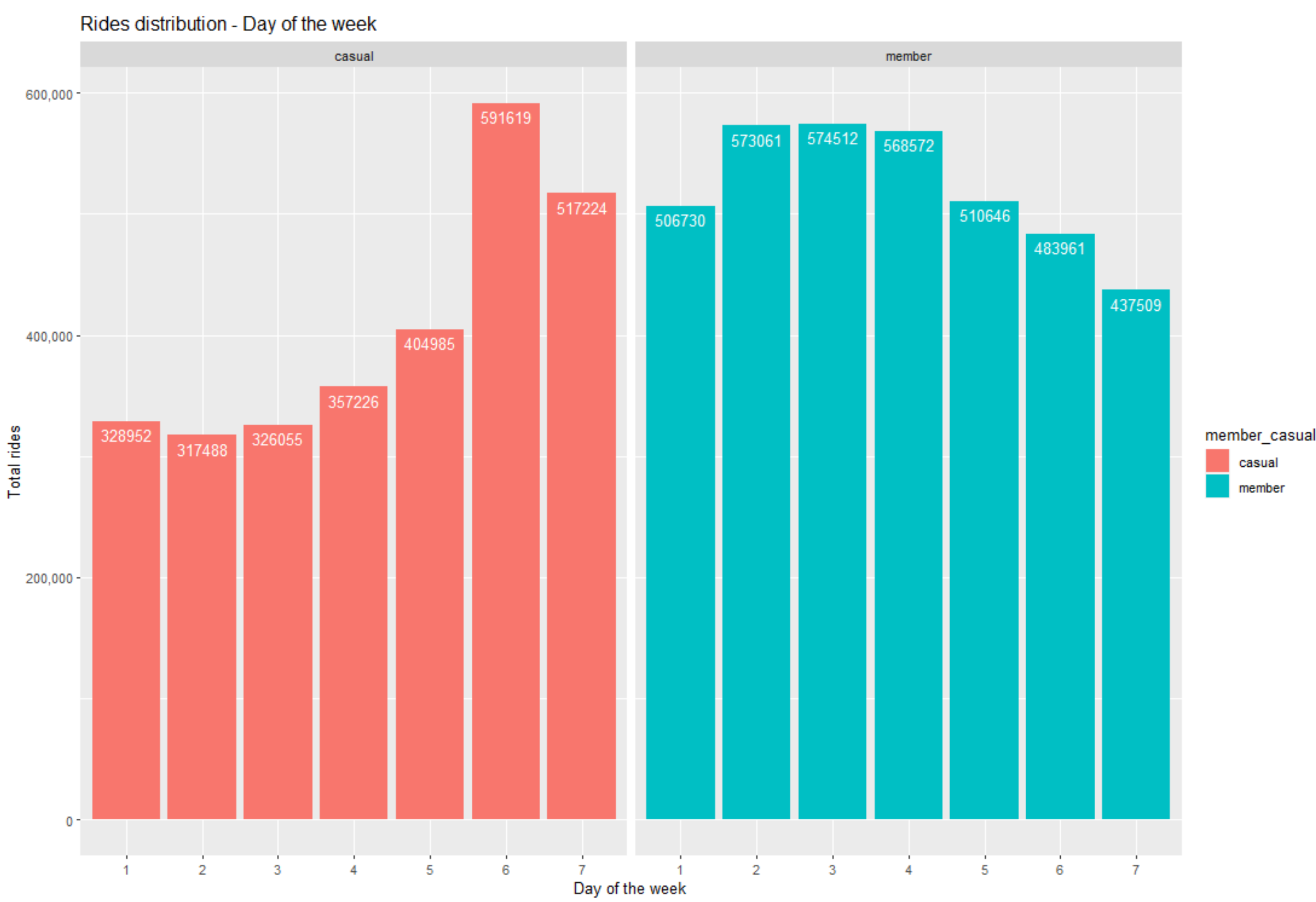


Figure 4 Ride distribution - Day of the week

The ride distribution per day between casual riders and member riders was quite different. It looked as if their distributions were inverses of each other. Casual riders were least active during weekdays, and they were most active during weekends, on the other hand, member riders were most active during weekdays and least active during weekends. I believe this could support the idea that member riders used bikes to commute to work more than casual riders did, and casual riders used the bikes for leisure more than members did.

Distribution by month of the year

```
historical_data %>%
  group_by(month_of_year) %>%
  summarise(total = length(ride_id), "percentage" = round(length(ride_id) / nrow(historical_data) * 100, digits = 2),
            "casual" = sum(member_casual == "casual"),"% casual" = round((sum(member_casual == "casual")
                                                               /length(ride_id))*100, digits = 2),
            "member" = sum(member_casual == "member"),"% member"= round((sum(member_casual == "member")
                                                              /length(ride_id))*100, digits = 2),
            "% Difference" = abs(round((sum(member_casual == "casual")/length(ride_id))*100, digits = 2)
                          -round((sum(member_casual == "member")/length(ride_id))*100, digits = 2))) %>%
  view()
```

```
chart = ggplot(historical_data, aes(month_of_year, fill=member_casual)) +
  geom_bar() +
  labs(y="Total rides",x="Month of the year", title="Rides distribution - Month of the year")+
  facet_wrap(vars(member_casual))+
  geom_text(aes(label = ..count..), stat = "count", vjust = 1.5, colour = "black")+
  coord_flip()


chart +                                  # Modify formatting of axis
  scale_y_continuous(labels = comma)
```

	month_of_year	total	percentage	casual	% casual	member	% member	% Difference
1	01	102047	1.57	17978	17.62	84069	82.38	64.76
2	02	113312	1.74	20660	18.23	92652	81.77	63.54
3	03	279061	4.29	87338	31.30	191723	68.70	37.40
4	04	364185	5.60	122764	33.71	241421	66.29	32.58
5	05	621390	9.56	271567	43.70	349823	56.30	12.60
6	06	1464654	22.54	715699	48.86	748955	51.14	2.28
7	07	803789	12.37	428036	53.25	375753	46.75	6.50
8	08	788075	12.13	400944	50.88	387131	49.12	1.76
9	09	742107	11.42	354363	47.75	387744	52.25	4.50
10	10	620797	9.55	251139	40.45	369658	59.55	19.10
11	11	355010	5.46	104775	29.51	250235	70.49	40.98
12	12	244113	3.76	68286	27.97	175827	72.03	44.06

Table 6 Ride distribution - Month

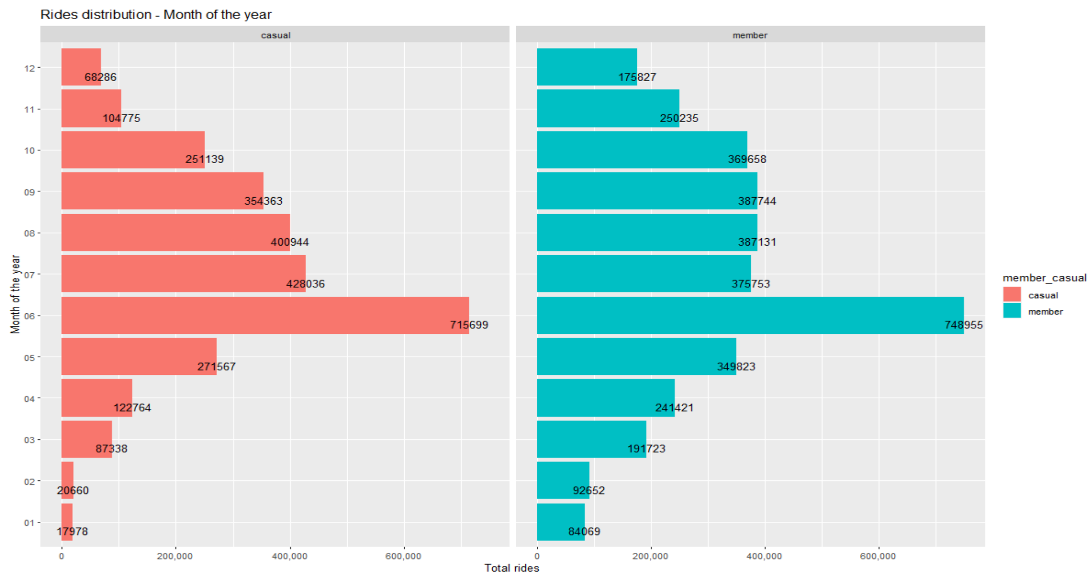
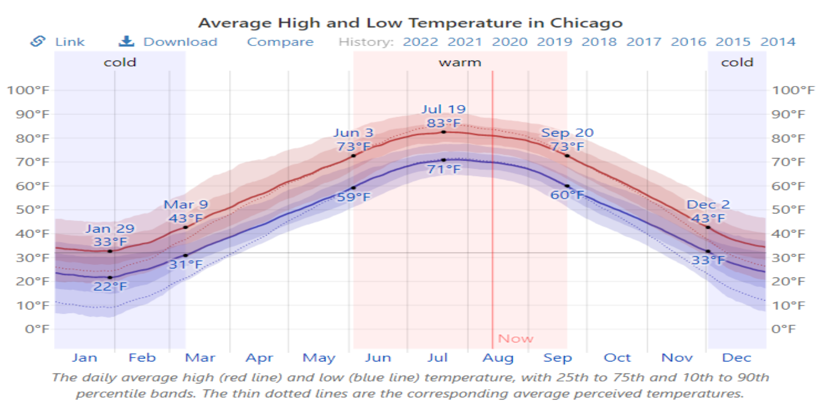


Figure 5 Ride distribution - Month

Both rider types had similar monthly ride distributions. Casual and member riders were much more active during June in comparison to other months of the year. In June, casual riders had 287,663 more rides than their second most active month (July), which was a 67% difference in total rides. Similarly, members had 361,211 (93.16%) more rides than their second most active month (September), which represented a 93.16% difference in total rides.

While their distributions were similar after looking at the % percentage difference in the table, I saw that Member riders were 32.58% to 64.76% more active than casual riders during January, February, March, April, November, and December. I believe that these drastic differences were due to the temperature differences throughout the year.



The daily average high (red line) and low (blue line) temperature, with 25th to 75th and 10th to 90th percentile bands. The thin dotted lines are the corresponding average perceived temperatures.

Average	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
High	33°F	36°F	46°F	57°F	67°F	77°F	82°F	80°F	74°F	62°F	49°F	37°F
Temp.	27°F	30°F	39°F	49°F	59°F	70°F	76°F	75°F	67°F	55°F	43°F	32°F
Low	22°F	25°F	34°F	44°F	54°F	64°F	70°F	69°F	61°F	50°F	38°F	27°F

Figure 6 Chicago yearly average temperature

After looking at the Average High and low Temperature in Chicago chart from weatherspark.com I saw that the previously mentioned months are the coldest months of the year in Chicago. I believe that since casual riders are more likely to ride for leisure, they are less likely to ride in months with uncomfortable weather. On the other hand. while a ride activity decrease for members riders can be observed during those months, the decrease was not as drastic as it was for casual riders.

Rides that started at a station

```
historical_data %>%
  summarise("start_at_station_#" = sum(!start_station_name == ""),
            "start_at_station_%" = round((sum(!start_station_name == "")/length(ride_id))*100, digits = 2),
            "member_start_at_station_%" = round(((sum(!start_station_name == ""  & member_casual == "member"))/sum(!start_station_name == ""))*100, digits = 2),
            "casual_start_at_station_%" = round(((sum(!start_station_name == ""  & member_casual == "casual"))/sum(!start_station_name == ""))*100, digits = 2),
            "start_no_at_station" = sum(start_station_name == ""),
            "start_no_at_station_%" = round((sum(start_station_name == "")/length(ride_id))*100, digits = 2),
            "member_start_no_at_station_%" = round(((sum(start_station_name == ""  & member_casual == "member"))/sum(start_station_name == ""))*100, digits = 2),
            "casual_start_no_at_station_%" = round(((sum(start_station_name == ""  & member_casual == "casual"))/sum(start_station_name == ""))*100, digits = 2)
  ) %>%
  view()
```

```
historical_data %>%
  summarise("start_at_station_#" = sum(!start_station_name == ""),
            "member_start_at_station" = (sum(!start_station_name == ""  & member_casual == "member")),
            "casual_start_at_station" = (sum(!start_station_name == ""  & member_casual == "casual")),
            "start_no_at_station" = sum(start_station_name == ""),
            "member_start_no_at_station" =(sum(start_station_name == ""  & member_casual == "member")),
            "casual_start_no_at_station" = (sum(start_station_name == ""  & member_casual == "casual"))
  ) %>%
  view()
```

	start_at_station_#	start_at_station_%	member_start_at_station_%	casual_start_at_station_%	start_no_at_station	start_no_at_station_%	member_start_no_at_station_%	casual_start_no_at_station_%
1	5604040	86.24	56.23	43.77	894500	13.76	56.3	43.7

start_at_station_#	member_start_at_station	casual_start_at_station	start_no_at_station	member_start_no_at_station	casual_start_no_at_station
5604040	3151387	2452653	894500	503604	390896

Figure 7 Ride distribution - Start at station

Most rides started at a station (86.24% of them). Of the rides that started at a station, 56.23% of them were done by member riders, while 43.77% were not. There was a 12.46% (698,734 rides) difference between members and casual riders that started their rides at stations.

Of the rides that did not start at a station (13.76% of the rides), 56.3% of them were done by member riders and 43.7% were not. There was a 12.6% (112,708 rides) difference between member and casual riders that started their rides at stations.

Share

Main findings

- Most rides were done by member riders (56.24% of total rides). That is a difference of 12.48% of the total rides or 811,442 more rides than casual riders.
- On average, casual riders with a ride length average of 21.51 minutes took longer rides than member riders who had a ride length average of 12.5 minutes.
- The most popular bike type was the classic bike (54.84% of total rides), followed by the electric bike (40.85% of total rides), and lastly the docked bike (4.31% of total rides).

- Both rider types had similar hourly ride distributions, but from 5 am to 9 am member riders were 29.10% to 53% more active than casual riders. I believe this could be due to member riders using the bikes to commute to work more than casual riders do. Similarly, from 12 am to 3 am casual riders were 20.30% to 30.60% more active that member riders. Both riders showed an increasing ride trend that stopped at 5 pm (17) the most active hour for both rider types.
- Casual riders were least active during weekdays, and they were most active during weekends, on the other hand, member riders were most active during weekdays and least active during weekends. I believe this could support the idea that member riders used bikes to commute to work more than casual riders did, and casual riders used the bikes for leisure more than members did.
- Casual and member riders were much more active during June in comparison to other months of the year. In June, casual riders had 287,663 more rides than their second most active month (July), which was a 67% difference in total rides. Similarly, members had 361,211 (93.16%) more rides than their second most active month (September), which represented a 93.16% difference in total rides.
- Member riders were 32.58% to 64.76% more active than casual riders during January, February, March, April, November, and December. I believe that these drastic differences were due to the temperature differences throughout the year. After looking at the Average High and low Temperature in Chicago chart from weatherspark.com I saw that the previously mentioned months are the coldest months of the year in Chicago. I believe that since casual riders are more likely to ride for leisure, they are less likely to ride in months with uncomfortable weather. On the other hand. while a ride activity decrease for members riders can be observed during those months, the decrease was not as drastic as it was for casual riders
- Most rides started at a station (86.24% of them). Of the rides that started at a station, 56.23% of them were done by member riders, while 43.77% were not. There was a 12.46% (698,734 rides) difference between members and casual riders that started their rides at stations.
- Of the rides that did not start at a station (13.76% of the rides), 56.3% of them were done by member riders and 43.7% were not. There was a 12.6% (112,708 rides) difference between member and casual riders that started their rides at stations.

Act

My recommendations to encourage casual riders to become members:

- At stations implemented priority bike access to members during peak hours (5 am to 9 am and 4 pm to 6 pm). Considering that 43.7% of the rides that started at a station were done by casual riders there must be some competition for bikes between casual and member rides during peak hours.
- Implement other pricing plans, such as a monthly or seasonal plan. My analysis showed that ride activity for both ride types (especially for casual riders) decreases considerably during the coldest months of the year. Some people may not want to pay for a full-year membership when they know they will not want to ride for half the year.

- Price increasing for daily single-ride passes and full-day passes during peak months (June, August, and September). While both groups show an activity increase during those months, the increase is more drastic for casual riders.

Conclusion:

I enjoyed working on this case study. Is amazing how I started with millions of rows that on their own were meaningless but by combining them and analyzing them I was able to find trends and extract meaningful information.

I still have a lot to learn, but I am excited to see how much more I still have to learn from data analytics.