Cyclistic Bike-Share Case Study Report

Mohammed Tayfour

28/12/2021

Table of Contents

1. Overview

This report is a part of one of the Google Data Analytics Professional Certificate capstone projects.

In this report, we will explore the case study that asks: **How does a Bike-Share Navigate Speedy Success?**

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."

In this case study, we will follow the six phases of the data analysis process:

1. Ask
2. Prepare
3. Process
4. Analyze
5. Share
6. Act

Ask

We need to understand the task at hand by identifying the business task as well as considering key stakeholders. We also need to ask ourselves two things:

● What is the problem we are trying to solve?

● How can our insights drive business decisions?

We want to understand and highlight the difference between casual riders, and annual members. The director of marketing believes that the company's future success depends on maximizing the number of annual memberships, hence why understanding the difference between the options offered could provide a clearer idea on what choices need to be made next.

2. Description of all data sources used

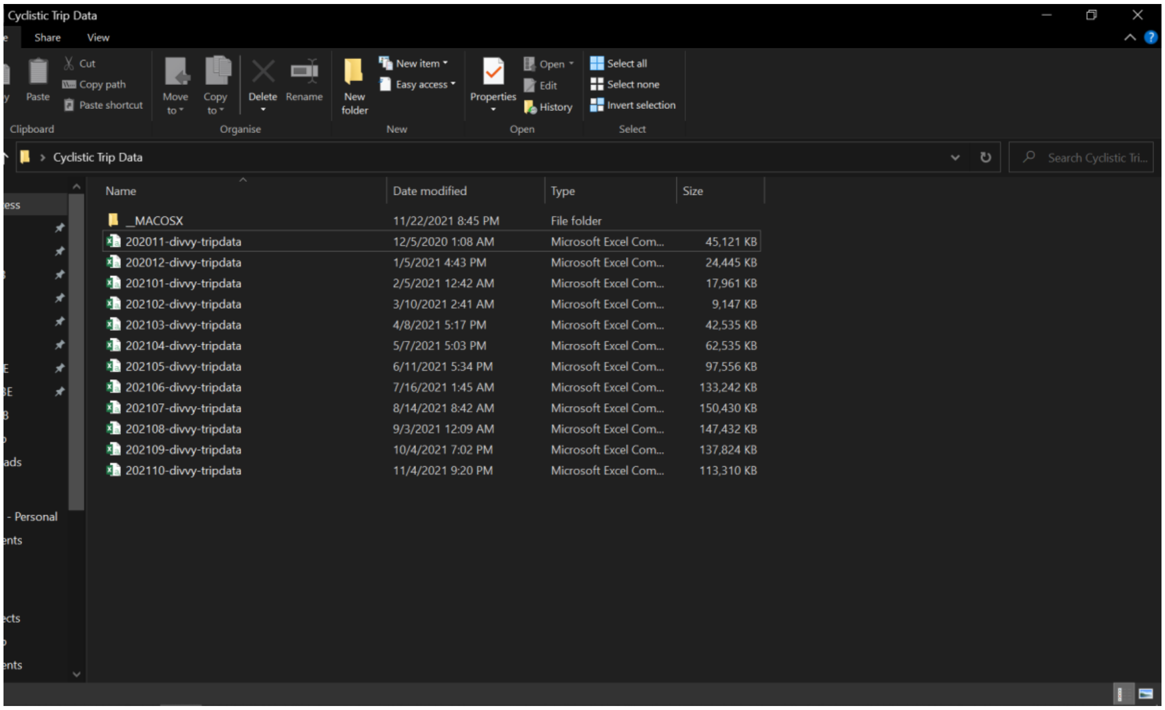
Prepare

The data needs to be prepared for analysis. I will be using [Cyclistic's historical trip data] to analyze and identify trends. This will be done by downloading the previous 12 months of Cyclistic trip data here. (Note: The datasets have a different name because Cyclistic is a fictional company. For the purposes of this case study, the datasets are appropriate and will enable me to answer the business questions. The data has been made available by Motivate International Inc. under this license: https://www.divvybikes.com/data-license-agreement .) This is public data that I will use to explore how different customer types are using Cyclistic bikes. But note that data-privacy issues prohibit you from using riders' personally identifiable information. This means that I won't be able to connect pass purchases to credit card numbers to determine if casual riders live in the Cyclistic service area or if they have purchased multiple single passes.

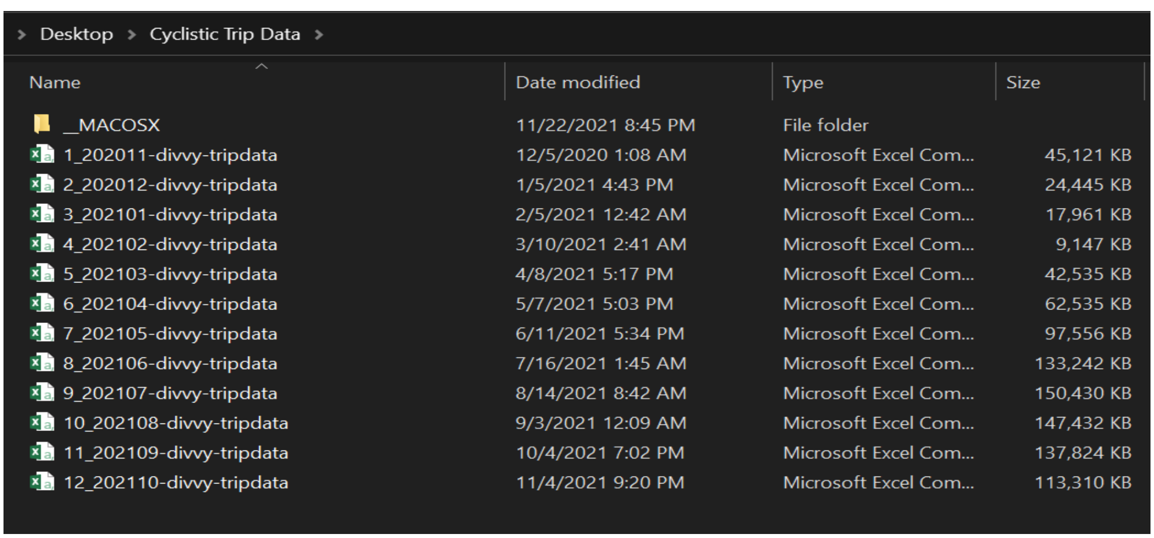
Here is a look of all the available data. I'll only be taking the past 12 months. The most recent data available was the 4th of November, 2021. Going back 12 months from that, we will go back to the 5th of December, 2021:

202011-divvy-tripdata.zip	Dec 5th 2020, 01:32:44 am	11.67 MB	ZIP file
202012-divvy-tripdata.zip	Jan 5th 2021, 04:56:54 pm	4.84 MB	ZIP file
202101-divvy-tripdata.zip	Feb 5th 2021, 12:52:59 am	3.66 MB	ZIP file
202102-divvy-tripdata.zip	Mar 10th 2021, 03:03:24 am	1.91 MB	ZIP file
202103-divvy-tripdata.zip	Apr 8th 2021, 05:28:53 pm	8.02 MB	ZIP file
202104-divvy-tripdata.zip	May 7th 2021, 05:52:05 pm	11.78 MB	ZIP file
202105-divvy-tripdata.zip	Jun 11th 2021, 08:10:18 pm	18.89 MB	ZIP file
202106-divvy-tripdata.zip	Jul 16th 2021, 02:22:05 am	26.52 MB	ZIP file
202107-divvy-tripdata.zip	Aug 14th 2021, 09:06:49 am	29.68 MB	ZIP file
202108-divvy-tripdata.zip	Sep 8th 2021, 09:10:46 pm	27.88 MB	ZIP file
202109-divvy-tripdata.zip	Oct 4th 2021, 08:21:39 pm	27.48 MB	ZIP file
202110-divvy-tripdata.zip	Nov 4th 2021, 10:58:36 pm	23.01 MB	ZIP file

The data for each month was downloaded in .ZIP files. I extracted and placed them in a folder titled: 'Cyclistic Trip Data'. The data was in .CSV format as shown below:



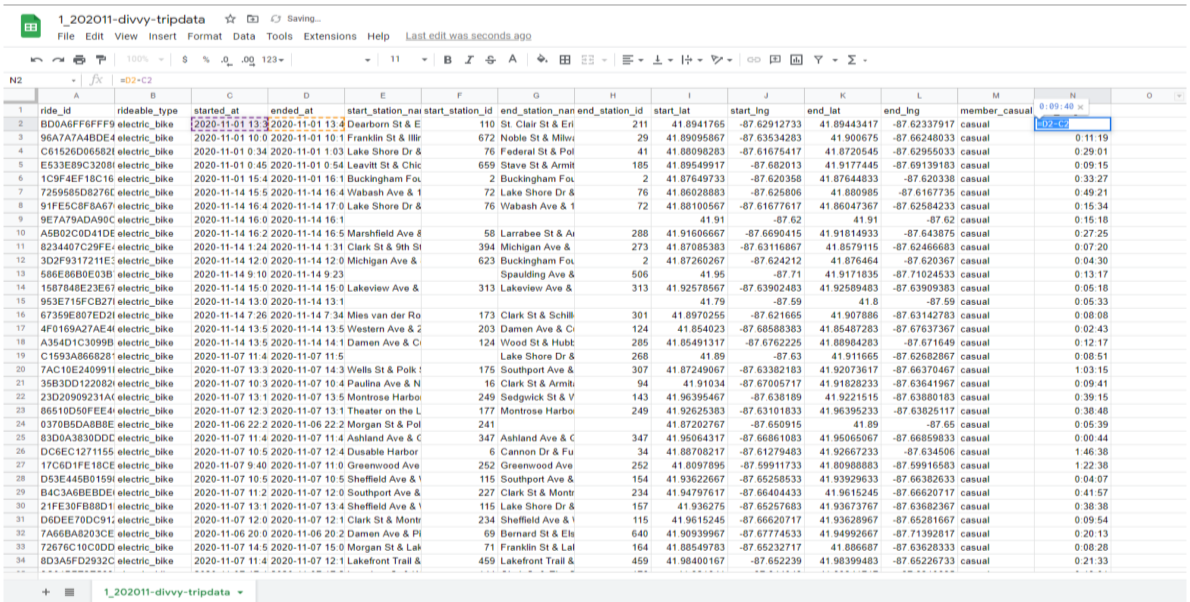
The files were renamed to follow proper naming conventions to help us know what data is in there. They were renamed with prefixes starting from "1_" to "12_" for each month to help make viewing them in chronological order easier:



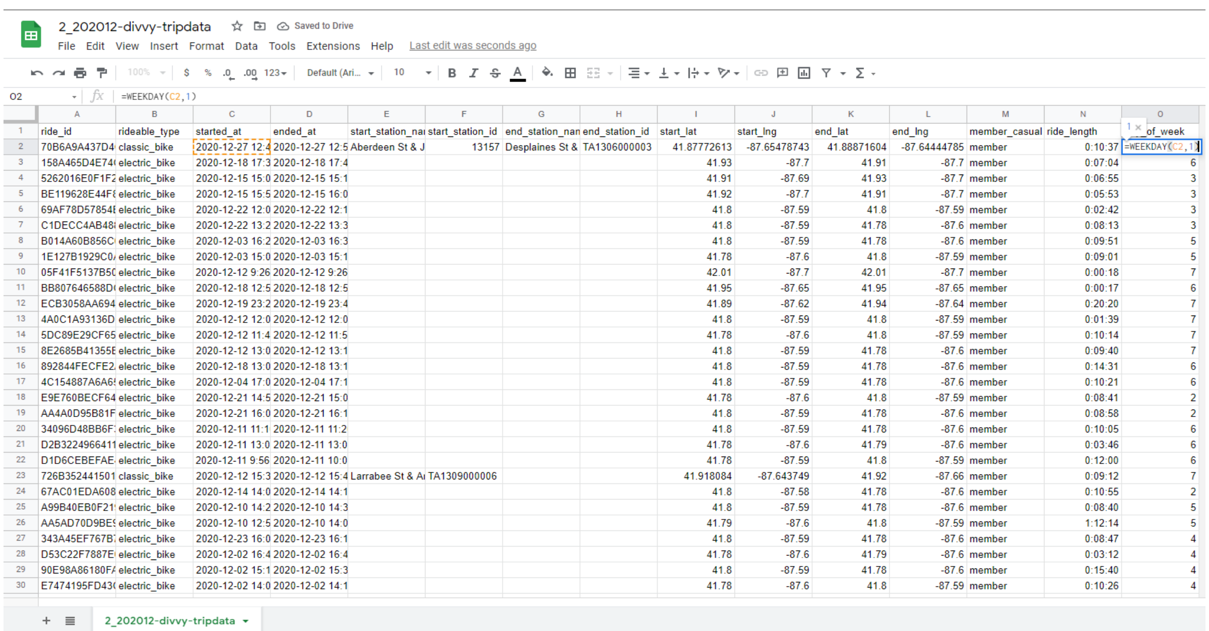
3. Documentation of any cleaning or manipulation of data

Process

The data has now been prepared for processing. Here, I will perform all of the basic aggregations and cleaning required to eliminate errors and inaccuracies that can get in the way of any results we obtain.



To better understand the data we've obtained, I calculated the ride length for each bike ride a user took. Using the WEEKDAY() to calculate the day of the week that each ride started.



4. Summary of analysis

Analyze

Now that the data is stored appropriately, it's time to move on to the analysis phase of this case study. For my analysis, I will be using RStudio. R was chosen because it can manage large data sets much quicker than spreadsheets. It can also visualize data and create well-formatted reports, complete with code chunks and pictures.

After setting the working directory, and loading all the necessary libraries for the analysis. I uploaded the datasets into dataframes which were named after the month the data covers. For example, the data collected from November 2020 will be stored in a data frame named "NOV20."

```
NOV20 <- read_csv("1_202011-divvy-tripdata.csv")
DEC20 <- read_csv("2_202012-divvy-tripdata.csv")
JAN21 <- read_csv("3_202101-divvy-tripdata.csv")
FEB21 <- read_csv("4_202102-divvy-tripdata.csv")
MAR21 <- read_csv("5_202103-divvy-tripdata.csv")
APR21 <- read_csv("6_202104-divvy-tripdata.csv")
MAY21 <- read_csv("7_202105-divvy-tripdata.csv")
JUN21 <- read_csv("8_202106-divvy-tripdata.csv")
JUL21 <- read_csv("9_202107-divvy-tripdata.csv")
AUG21 <- read_csv("10_202108-divvy-tripdata.csv")
SEP21 <- read_csv("11_202109-divvy-tripdata.csv")
OCT21 <- read_csv("12_202110-divvy-tripdata.csv")
```

After the data is imported and loaded into dataframes, the next step is to make sure that the column names are consistent and then merging them all into a single dataframe.

```
> colnames(NOV20)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
> colnames(DEC20)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
> colnames(JAN21)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
> colnames(FEB21)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
> colnames(MAR21)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
> colnames(APR21)
 [1] "ride_id"            "rideable_type"      "started_at"         "ended_at"           "start_station_name"
 [6] "start_station_id"   "end_station_name"   "end_station_id"     "start_lat"          "start_lng"
[11] "end_lat"            "end_lng"            "member_casual"
```

All of the column names were found to be consistent. Using the str() function, I tried looking for any incongruencies with the data types. It was revealed that the start_station_id and end_station_id were both assigned as double types, instead of the character type in NOV20. These errors were rectified by re-assigning the correct data types to those columns in both data frames.

```
NOV20 <-  transform(NOV20, start_station_id = as.character(start_station_id)
                   ,end_station_id = as.character(end_station_id))
```

The started_at and ended_at values were also incorrectly assigned as characters, instead of date-time for all the months, so I rectified them using the as.POSIXct() function

```
NOV20[['started_at']] <- as.POSIXct(NOV20[['started_at']], format = "%Y-%m-%d %H:%M:%S")
NOV20[['ended_at']] <- as.POSIXct(NOV20[['ended_at']], format = "%Y-%m-%d %H:%M:%S")
```

To make sure everything stacks correctly, I also converted ride_id and rideable_type to character types.

```
NOV20 <-  mutate(NOV20, ride_id = as.character(ride_id)
                 ,rideable_type = as.character(rideable_type))
```

With that done, I proceeded to merge everything into a single dataframe using the bind() function.

```
# Stacking the individual month data frames into one big data frame
all_trips <- bind_rows(NOV20,DEC20,JAN21,FEB21,MAR21,APR21,MAY21,JUN21,JUL21,AUG21,SEP21,OCT21)
```

With that out of the way, it's time to clean the data to prepare for analysis. I began by inspecting the new all_trips table using the following functions

```
# Inspect the new table that has been created
colnames(all_trips)  #List of column names
nrow(all_trips)  #How many rows are in data frame?
dim(all_trips)  #Dimensions of the data frame?
head(all_trips)  #See the first 6 rows of data frame.  Also tail(all_trips)
str(all_trips)  #See list of columns and data types (numeric, character, etc)
summary(all_trips)  #Statistical summary of data. Mainly for numerics
```

There are a few problems that will need to be fixed:

1. In the "member_casual" column, there are two names for members ("member" and "Subscriber") and two names for casual riders ("Customer" and "casual"). We will need to consolidate that from four to two labels.

2. The data can only be aggregated at the ride-level, which is too granular. We will want to add some additional columns of data -- such as day, month, year -- that provide additional opportunities to aggregate the data.

3. I want to add a calculated field for length of ride since we dropped the field earlier for inconsistencies. I will add "ride_length" to the entire dataframe for consistency.

4. Before dropping the column, there were some rides where ride_length showed up as negative, including several hundred rides where the bikes were taken out of circulation for Quality Control reasons. We will want to delete these rides.

To differentiate between subscribers, and casual users, in the "member_casual" column, I proceeded to replace "Subscriber" with "member" and "Customer" with "casual". I began by seeing how many observations fall under each usertype.

```
> table(all_trips$member_casual)

 casual  member 
2470517 2908317 
> 
```

There are 2,470,517 casual users, and 2,908,317 members. I reassigned the values and then checked again to make sure they were still intact after the reassignment.

```
> all_trips <-  all_trips %>%
+   mutate(member_casual = recode(member_casual
+                                ,"Subscriber" = "member"
+                                ,"Customer" = "casual"))
> table(all_trips$member_casual)

 casual  member 
2470517 2908317 
> 
```

I proceeded to add columns that list the date, month, day, and year of each ride. This will allow us to aggregate ride data for each month, day, or year. Before completing these operations we could only aggregate at the ride level

```
all_trips$date <- as.Date(all_trips$started_at) #The default format is yyyy-mm-dd
all_trips$month <- format(as.Date(all_trips$date), "%m")
all_trips$day <- format(as.Date(all_trips$date), "%d")
all_trips$year <- format(as.Date(all_trips$date), "%Y")
all_trips$day_of_week <- format(as.Date(all_trips$date), "%A")
```

We can now proceed to add a ride_length calculation to all_trips (in seconds)

```
> str(all_trips)
'data.frame':	5378834 obs. of  15 variables:
 $ ride_id           : chr  "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E533E89C32080B9E" ...
 $ rideable_type     : chr  "electric_bike" "electric_bike" "electric_bike" "electric_bike" ...
 $ started_at        : POSIXct, format: "2020-11-01 13:36:00" ...
 $ ended_at          : POSIXct, format: "2020-11-01 13:45:40" ...
 $ start_station_name: chr  "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake Shore Dr & Monroe St" "Le
avitt St & Chicago Ave" ...
 $ start_station_id  : chr  "110" "672" "76" "659" ...
 $ end_station_name  : chr  "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federal St & Polk St" "Stave S
t & Armitage Ave" ...
 $ end_station_id    : chr  "211" "29" "41" "185" ...
 $ member_casual     : chr  "casual" "casual" "casual" "casual" ...
 $ date              : Date, format: "2020-11-01" ...
 $ month             : chr  "11" "11" "11" "11" ...
 $ day               : chr  "01" "01" "01" "01" ...
 $ year              : chr  "2020" "2020" "2020" "2020" ...
 $ day_of_week       : chr  "Sunday" "Sunday" "Sunday" "Sunday" ...
 $ ride_length       : num  580 679 1741 555 2007 ...
> 
```

The dataframe included a few hundred entries when bikes were taken out of docks and checked for quality/ or the ride_length value was negative. I proceeded to remove those and load the new values in another dataframe.

```r
# Remove "bad" data
all_trips_v2 <- all_trips[!(all_trips$start_station_name == "HQ QR" | all_trips$ride_length<0),]
```

With that, the data is clean, and we can now move on to conducting descriptive analysis.

Using the summary() function, I proceeded to perform descriptive analysis on ride_length. Here, we can see the minimum, maximum, mean, median, and mode values for that specific attribute.

```
> summary(all_trips_v2$ride_length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0     426     752    1396    1363 3356649
```

Using the aggregate() function, I also compared casual users and members in terms of ride_length. There is a higher average/mean number of casual users compared to full time members. The maximum number of casual users appears to be significantly higher than members as well.

```
> aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = mean)
  all_trips_v2$member_casual all_trips_v2$ride_length
1                     casual                2057.4935
2                     member                 839.6265
> aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = median)
  all_trips_v2$member_casual all_trips_v2$ride_length
1                     casual                     1009
2                     member                      598
> aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = max)
  all_trips_v2$member_casual all_trips_v2$ride_length
1                     casual                  3356649
2                     member                    93596
> aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = min)
  all_trips_v2$member_casual all_trips_v2$ride_length
1                     casual                        0
2                     member                        0
>
```

I also compared the average ride time by each day of the week for casual users and members. Casual riders use their bikes for a longer time, on average, compared to members on every day of the week.

```
> aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual + all_trips_v2$day_of_week, FUN = mean)
   all_trips_v2$member_casual all_trips_v2$day_of_week all_trips_v2$ride_length
1                      casual                   Sunday                2396.2825
2                      member                   Sunday                 961.6521
3                      casual                   Monday                2044.7720
4                      member                   Monday                 812.4559
5                      casual                  Tuesday                1810.3490
6                      member                  Tuesday                 787.3170
7                      casual                Wednesday                1780.7362
8                      member                Wednesday                 789.7363
9                      casual                 Thursday                1784.0406
10                     member                 Thursday                 787.0021
11                     casual                   Friday                1961.9236
12                     member                   Friday                 820.0691
13                     casual                 Saturday                2206.4562
14                     member                 Saturday                 938.4813
>
```

We can also analyze the ridership data by type AND weekday. Casual users took more rides on their bikes than members did on weekends, whereas members used them on weekdays longer. Casual members did however use them for a significantly longer duration on average as well.

```
# Groups:   member_casual [3]
   member_casual weekday number_of_rides average_duration
   <chr>         <ord>             <int>            <dbl>
 1 casual        Sun              428796            2396.
 2 casual        Mon              243995            2045.
 3 casual        Tue              229012            1810.
 4 casual        Wed              231926            1781.
 5 casual        Thu              240503            1784.
 6 casual        Fri              310258            1962.
 7 casual        Sat              497014            2206.
 8 member        Sun              327800             962.
 9 member        Mon              350646             812.
10 member        Tue              386661             787.
11 member        Wed              398513             790.
12 member        Thu              380131             787.
13 member        Fri              377294             820.
14 member        Sat              374471             938.
```

The next step of data analysis could finally be commenced.

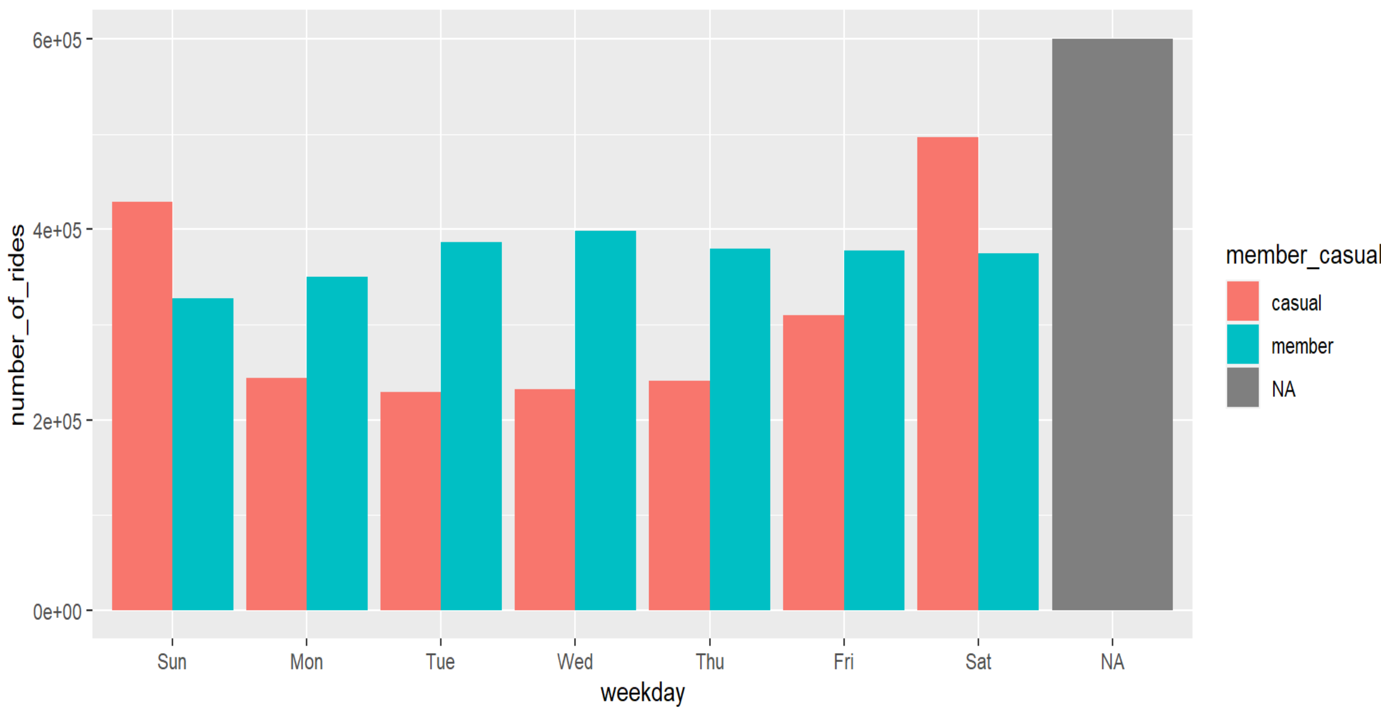
5. Supporting visualizations and key findings

Share

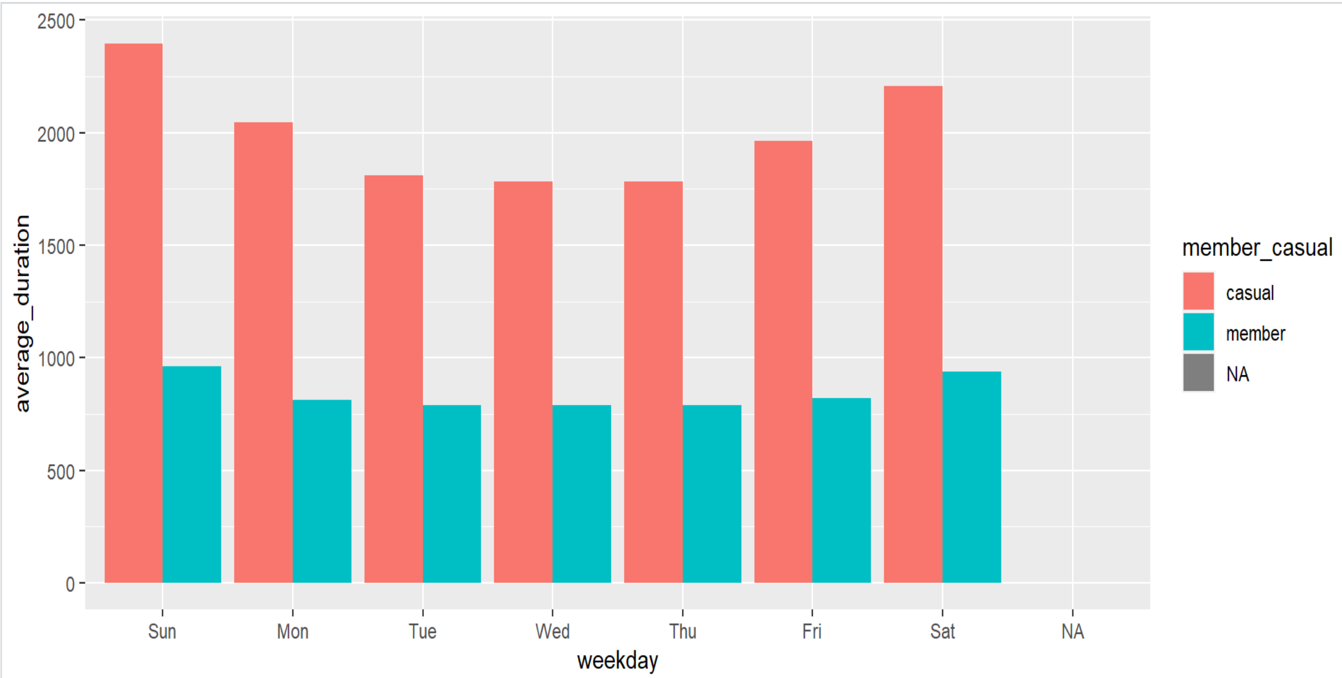
In this part of the report, insights extracted from the data with supporting visualizations will be discussed.

The purpose of this case study was to tackle the question: **How do annual members and casual riders use Cyclistic bikes differently?**

Based on the results of the descriptive analysis and backed by our visual below, casual users tend to use the bikes more often on the weekends. The assumption here is that they could be using their bikes for leisure activities on the weekends. Members however use their bikes more often on the weekdays compared to casual users. This could be the case because members may need to use their bikes to commute to work. It could also be their main source of transportation.



Based on the visual below, and backed by the descriptive analysis conducted earlier, I concluded that casual users did use their bikes for a significantly longer duration on average compared to members. This could further be backed by the assumption that casual riders use their bikes for leisure activities.



6. Top recommendations based on analysis

Act

Based on the analysis performed, it’s clear that the majority of casual and member customer groups use Cyclistic’s bikes for different purposes. Members use the bikes more often on the weekdays. The assumption is that they use the bikes to commute to and from their workplaces or their other regular destinations, while casual users on the other hand might be using the bikes for leisure activities. This conclusion was drawn from the fact that casual customers peak during weekends, while the opposite was observed for the member group.

Some recommendations that Cyclistic can make use of based on the analysis performed are:

1. Cyclistic can try to convert casual customers to member customers by offering more flexible membership prices. For example, there should be a membership price for only weekend users which should be much cheaper than the regular users’ price.

2. One other angle, through the use of social media, Cyclistic’s social media team could influence potential or existing casual customers to bend on using Cyclistic’s bike for their recreational activities, and then provide them with special offers for becoming a member. Cyclistic can also use the knowledge gained from the insights about peak days for casual customers to spread the news about its services. The company can initiate a social media campaign such that casual customers would be able to receive reduced membership prices if they share about their “weekend experience with Cyclistic” on social media platforms.

3. Cyclistic can also consider expanding and opening more branches, to lessen the chances of visitors and tourists having to use their bikes for casual reasons, and provide them with more incentive to use their bikes as members instead.