# My First case Study

Nwani Stanley

3/9/2022

## 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.

The data used was from February,2021 to January,2022. i.e.12 months of data. The data was downloaded here

### Loading the required libraries

**tidyverse** for data import and wrangling, **lubridate** for date functions, **ggplot** for visualization

Set working directory to simplify calls for data

```
setwd("/Users/ugonn/Videos/DS/Google Data Analytics Certificate/Datasets/Case Study 1/2")
```

### Collect Data

```
m02_2021 <- read_csv("202102-divvy-tripdata.csv")
```

```
## Rows: 49622 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m03_2021 <- read_csv("202103-divvy-tripdata.csv")
```

```
## Rows: 228496 Columns: 13
## -- Column specification --------------------------------------------------------
```

```
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m04_2021 <- read_csv("202104-divvy-tripdata.csv")
```

```
## Rows: 337230 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m05_2021 <- read_csv("202105-divvy-tripdata.csv")
```

```
## Rows: 531633 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m06_2021 <- read_csv("202106-divvy-tripdata.csv")
```

```
## Rows: 729595 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m07_2021 <- read_csv("202107-divvy-tripdata.csv")
```

```
## Rows: 822410 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
```

```
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m08_2021 <- read_csv("202108-divvy-tripdata.csv")
```

```
## Rows: 804352 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m09_2021 <- read_csv("202109-divvy-tripdata.csv")
```

```
## Rows: 756147 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m10_2021 <- read_csv("202110-divvy-tripdata.csv")
```

```
## Rows: 631226 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m11_2021 <- read_csv("202111-divvy-tripdata.csv")
```

```
## Rows: 359978 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m12_2021 <- read_csv("202112-divvy-tripdata.csv")
```

```
## Rows: 247540 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
m01_2022 <- read_csv("202201-divvy-tripdata.csv")
```

```
## Rows: 103770 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

### Wrangle data and combine into a single variable

Inspect column names for the different months and ensure they are the same for successful joining

```
colnames(m02_2021)
```

```
##  [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"
```

```
colnames(m03_2021)
```

```
##  [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"
```

```
colnames(m04_2021)
```

```
##  [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"
```

```
colnames(m05_2021)
```

```
##  [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"
```

```
colnames(m06_2021)
```

```
##  [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"
```

```
colnames(m07_2021)
```

```
##  [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"
```

```
colnames(m08_2021)
```

```
##  [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"
```

```
colnames(m09_2021)
```

```
##  [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"
```

```
colnames(m10_2021)
```

```
##  [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"
```

```
colnames(m11_2021)
```

```
##  [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"
```

```
colnames(m12_2021)
```

```
##  [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"
```

```
colnames(m01_2022)
```

```
##  [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"
```

### Combine Data and Inspect

Stack individual data frames into one big dataframe

```
all_trips <- bind_rows(m02_2021, m03_2021, m04_2021, m05_2021, m06_2021, m07_2021,
                       m08_2021, m09_2021, m10_2021, m11_2021, m12_2021, m01_2022)
```

Inspect new data frame that has been created

```
# Column names
colnames(all_trips)
```

```
##  [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"
```

```
# Dimension of the data frame
dim(all_trips)
```

```
## [1] 5601999      13
```

```r
# Sample of the data frame
head(all_trips)
```

```
## # A tibble: 6 x 13
##   ride_id rideable_type started_at          ended_at            start_station_n~
##   <chr>   <chr>         <dttm>              <dttm>              <chr>
## 1 89E7AA~ classic_bike  2021-02-12 16:14:56 2021-02-12 16:21:43 Glenwood Ave & ~
## 2 0FEFDE~ classic_bike  2021-02-14 17:52:38 2021-02-14 18:12:09 Glenwood Ave & ~
## 3 E6159D~ electric_bike 2021-02-09 19:10:18 2021-02-09 19:19:10 Clark St & Lake~
## 4 B32D31~ classic_bike  2021-02-02 17:49:41 2021-02-02 17:54:06 Wood St & Chica~
## 5 83E463~ electric_bike 2021-02-23 15:07:23 2021-02-23 15:22:37 State St & 33rd~
## 6 BDAA7E~ electric_bike 2021-02-24 15:43:33 2021-02-24 15:49:05 Fairbanks St & ~
## # ... with 8 more variables: start_station_id <chr>, end_station_name <chr>,
## #   end_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, member_casual <chr>
```

```r
# Summary of the data
summary(all_trips)
```

```
##    ride_id          rideable_type        started_at
##  Length:5601999     Length:5601999     Min.   :2021-02-01 00:55:44
##  Class :character   Class :character   1st Qu.:2021-06-11 12:40:12
##  Mode  :character   Mode  :character   Median :2021-08-04 22:01:30
##                                        Mean   :2021-08-04 20:30:49
##                                        3rd Qu.:2021-09-28 16:39:49
##                                        Max.   :2022-01-31 23:58:37
##
##     ended_at                   start_station_name start_station_id
##  Min.   :2021-02-01 01:22:48   Length:5601999     Length:5601999
##  1st Qu.:2021-06-11 13:03:36   Class :character   Class :character
##  Median :2021-08-04 22:23:12   Mode  :character   Mode  :character
##  Mean   :2021-08-04 20:52:45
##  3rd Qu.:2021-09-28 16:55:21
##  Max.   :2022-02-01 01:46:16
##
##  end_station_name   end_station_id       start_lat       start_lng
##  Length:5601999     Length:5601999     Min.   :41.64   Min.   :-87.84
##  Class :character   Class :character   1st Qu.:41.88   1st Qu.:-87.66
##  Mode  :character   Mode  :character   Median :41.90   Median :-87.64
##                                        Mean   :41.90   Mean   :-87.65
##                                        3rd Qu.:41.93   3rd Qu.:-87.63
##                                        Max.   :45.64   Max.   :-73.80
##
##     end_lat         end_lng       member_casual
##  Min.   :41.39   Min.   :-88.97   Length:5601999
##  1st Qu.:41.88   1st Qu.:-87.66   Class :character
##  Median :41.90   Median :-87.64   Mode  :character
##  Mean   :41.90   Mean   :-87.65
##  3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.17   Max.   :-87.49
##  NA's   :4754    NA's   :4754
```

Distribution of riders in the dataset

```r
# How many observations fall under each usertype
table(all_trips$member_casual)
```

```
##
##  casual  member
## 2529408 3072591
```

### Clean up and Add data to prepare for analysis

Add columns that list the date, month, day, and year of each ride

```r
all_trips$date <- as.Date(all_trips$started_at) # 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")
```

Calculate trip length and store as a new column

```r
all_trips$ride_length <- difftime(all_trips$ended_at, all_trips$started_at)

# Convert "ride_length" from Factor to numeric
all_trips$ride_length <- as.numeric(as.character(all_trips$ride_length))

# Check
is.numeric(all_trips$ride_length)
```

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

### Remove bad data

The data frame includes a few hundred entries when bikes were taken out of docks and checked for quality by Divvy or ride_length was negative.

```r
all_trips_v2 <- subset(all_trips, start_station_name != "HQ QR" & ride_length > 0)

summary(all_trips_v2)
```

```
##    ride_id          rideable_type        started_at
##  Length:4902928     Length:4902928     Min.   :2021-02-01 00:55:44
##  Class :character   Class :character   1st Qu.:2021-06-09 09:38:21
##  Mode  :character   Mode  :character   Median :2021-08-01 17:12:45
##                                        Mean   :2021-08-01 22:29:14
##                                        3rd Qu.:2021-09-24 18:36:04
##                                        Max.   :2022-01-31 23:58:37
##
##     ended_at                   start_station_name start_station_id
##  Min.   :2021-02-01 01:22:48   Length:4902928     Length:4902928
##  1st Qu.:2021-06-09 10:03:40   Class :character   Class :character
##  Median :2021-08-01 17:41:45   Mode  :character   Mode  :character
```

```
##  Mean   :2021-08-01 22:52:03
##  3rd Qu.:2021-09-24 18:52:27
##  Max.   :2022-02-01 01:46:16
##
##  end_station_name   end_station_id       start_lat       start_lng
##  Length:4902928     Length:4902928     Min.   :41.65   Min.   :-87.83
##  Class :character   Class :character   1st Qu.:41.88   1st Qu.:-87.66
##  Mode  :character   Mode  :character   Median :41.90   Median :-87.64
##                                        Mean   :41.90   Mean   :-87.64
##                                        3rd Qu.:41.93   3rd Qu.:-87.63
##                                        Max.   :45.64   Max.   :-73.80
##
##     end_lat         end_lng       member_casual           date
##  Min.   :41.39   Min.   :-88.97   Length:4902928     Min.   :2021-02-01
##  1st Qu.:41.88   1st Qu.:-87.66   Class :character   1st Qu.:2021-06-09
##  Median :41.90   Median :-87.64   Mode  :character   Median :2021-08-01
##  Mean   :41.90   Mean   :-87.64                      Mean   :2021-08-01
##  3rd Qu.:41.93   3rd Qu.:-87.63                      3rd Qu.:2021-09-24
##  Max.   :42.17   Max.   :-87.50                      Max.   :2022-01-31
##  NA's   :4754    NA's   :4754
##     month              day               year          day_of_week
##  Length:4902928     Length:4902928     Length:4902928     Length:4902928
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   ride_length
##  Min.   :      1
##  1st Qu.:    413
##  Median :    729
##  Mean   :   1369
##  3rd Qu.:   1325
##  Max.   :3356649
##
```

Descriptive analysis of ride length (in seconds)

```
mean(all_trips_v2$ride_length) # average
```

```
## [1] 1369.35
```

```
median(all_trips_v2$ride_length) # midpoint number
```

```
## [1] 729
```

```
max(all_trips_v2$ride_length) # longest ride
```

```
## [1] 3356649
```

```
min(all_trips_v2$ride_length) # shortest ride
```

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

Comparing members and casual users

```
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                2037.2253
## 2                     member                 821.3849
```

```
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                      988
## 2                     member                      581
```

```
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                        1
## 2                     member                        1
```

Average ride time by each day for members vs casual users

```
# Ordering the days of the week
all_trips_v2$day_of_week <- ordered(all_trips_v2$day_of_week,
                                    levels=c("Sunday", "Monday", "Tuesday",
                                             "Wednesday", "Thursday", "Friday",
                                             "Saturday"))

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                2372.4010
## 2                      member                   Sunday                 948.6329
## 3                      casual                   Monday                2034.7344
## 4                      member                   Monday                 796.5779
## 5                      casual                  Tuesday                1778.7680
## 6                      member                  Tuesday                 770.6328
## 7                      casual                Wednesday                1768.9325
```

```
## 8                    member               Wednesday                768.9271
## 9                    casual                Thursday               1772.8194
## 10                   member                Thursday                769.4319
## 11                   casual                  Friday               1939.1870
## 12                   member                  Friday                801.9084
## 13                   casual                Saturday               2191.8425
## 14                   member                Saturday                923.2949
```

Analyze ridership data by type and weekday

```
all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  #creates weekday field using wday()
  group_by(member_casual, weekday) %>%
  #groups by user type and weekday
  summarise(number_of_rides = n()
            #calculates the number of rides and average duration
            ,average_duration = mean(ride_length)) %>%
            # calculates the average duration
  arrange(member_casual, weekday) # sorts
```

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

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              429641            2372.
##  2 casual        Mon              248260            2035.
##  3 casual        Tue              235150            1779.
##  4 casual        Wed              238805            1769.
##  5 casual        Thu              245045            1773.
##  6 casual        Fri              314346            1939.
##  7 casual        Sat              498446            2192.
##  8 member        Sun              329276             949.
##  9 member        Mon              367424             797.
## 10 member        Tue              412556             771.
## 11 member        Wed              421674             769.
## 12 member        Thu              397222             769.
## 13 member        Fri              387684             802.
## 14 member        Sat              377399             923.
```

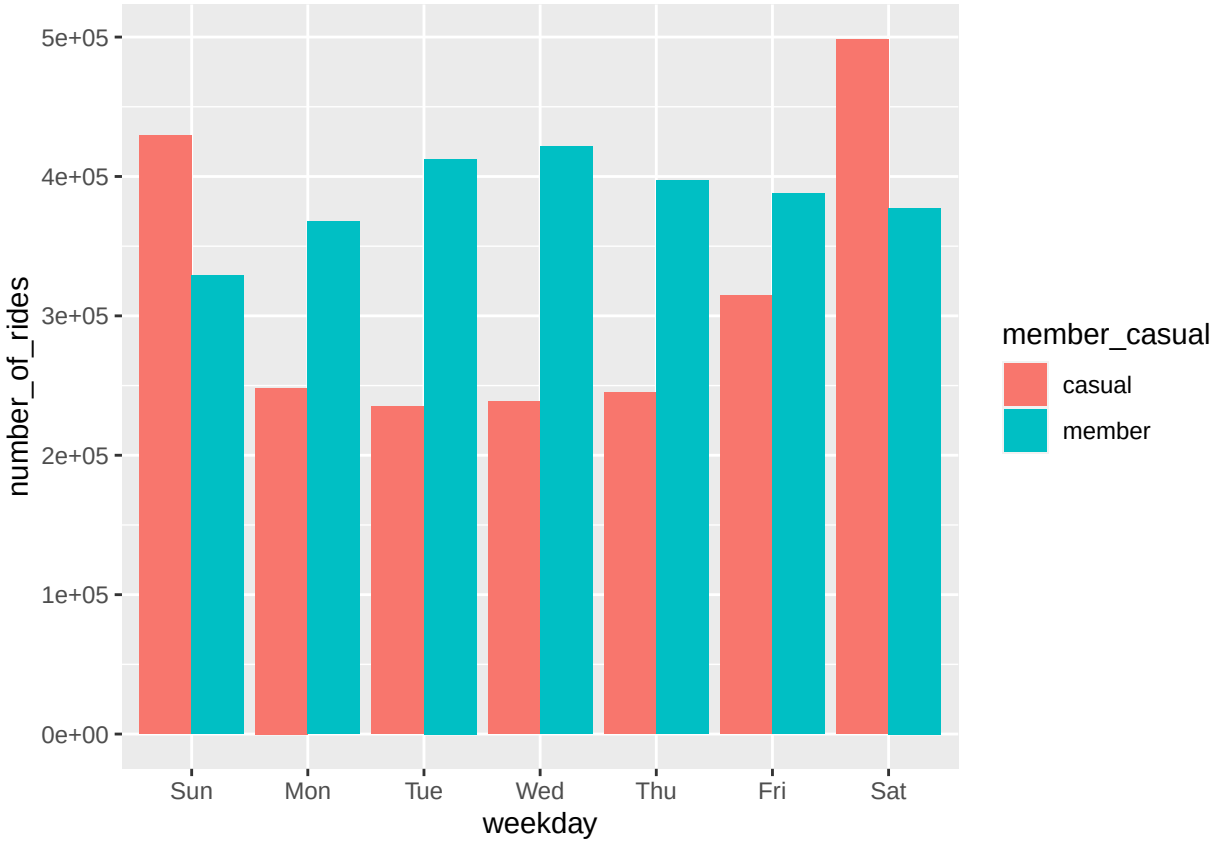
## Visualizations

Let's visualize the number of rides by rider type in order to see how many rides are taken by casual riders and member riders everyday

```
all_trips_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")
```

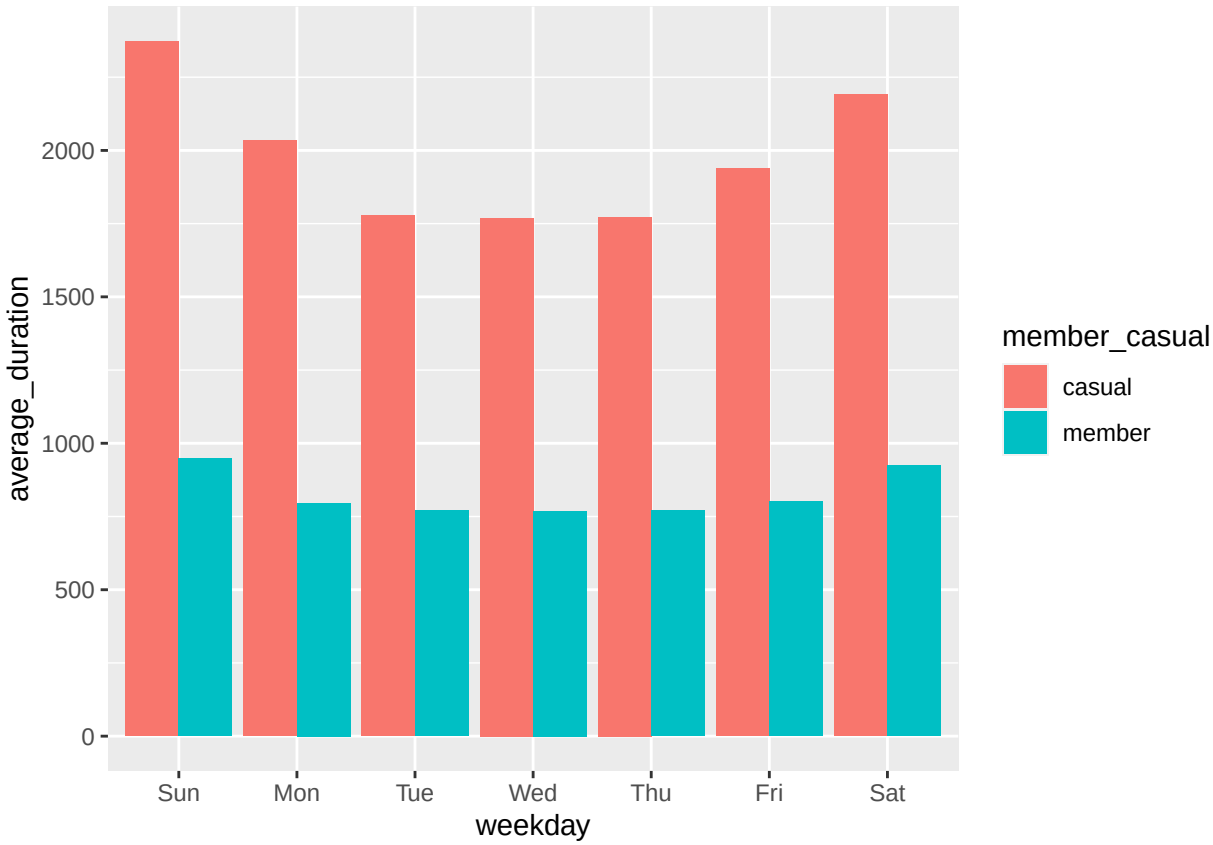
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```




Next we will see the average time each usertype(member or casual) is riding a bike

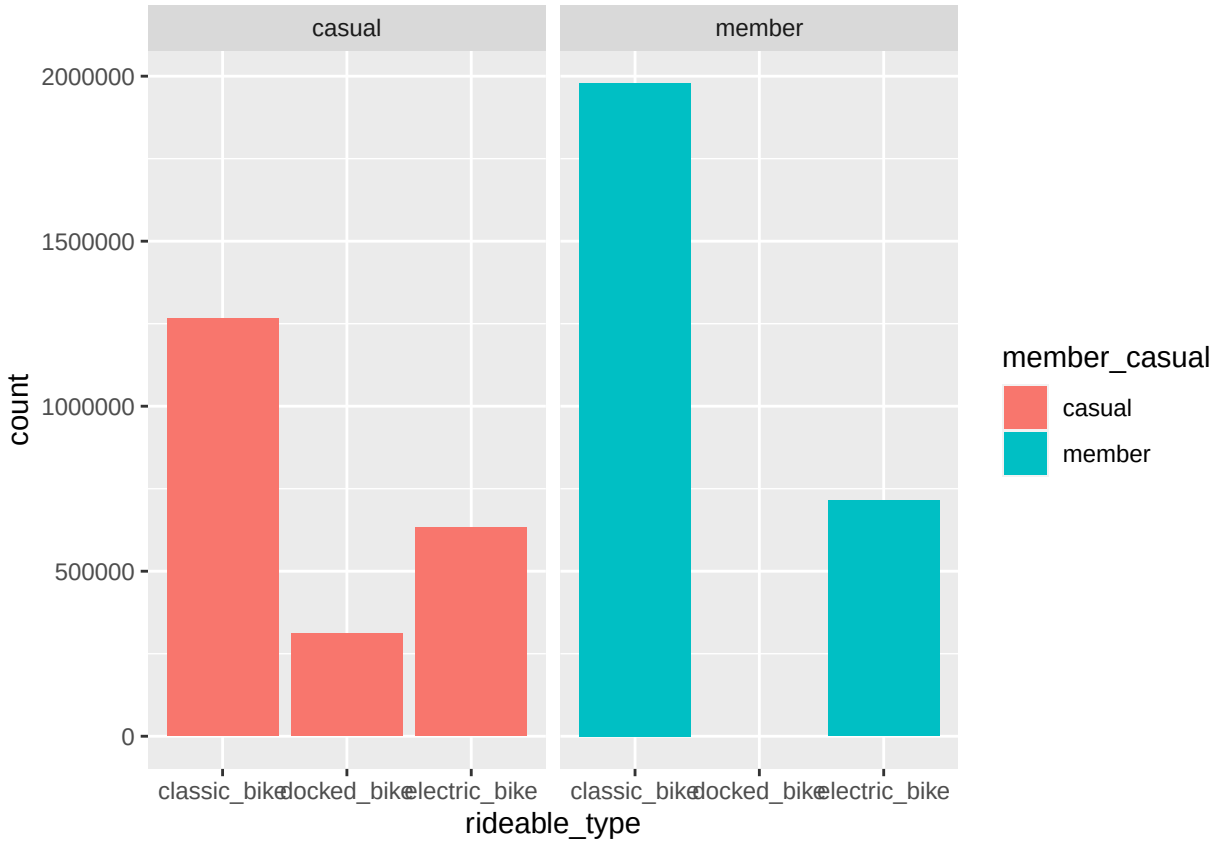
```
all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```

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




We can also see the number of riders using different types of bikes, for the different usertypes

```
ggplot(data=all_trips_v2) + geom_bar(mapping=(aes(x=rideable_type, fill = member_casual))) + facet_wrap
```




## Conclusions

- The annual members are more than the casual members.
- Casual members rent bikes more on weekends. This might be because they rent bikes for leisure or exercise. Annual members rent bikes more during the weekdays. This might be because they rent bikes to go to work or for other professional activities.
- Casual members rent bikes much longer than annual members.
- Annual members make use of classic bikes and electric bikes. Docked bikes are used only by Casual members.

### Recommendations

- Prices could be increased for renting bikes on weekends for Casual members. This would encourage them to become annual members.
- Discounting the cost for becoming an Annual member might encourage Casual members to join, since it is clear that they like renting our bikes.
- Docked bikes are used solely by Casual members. If it is because it is cheaper, raising the price of renting a docked bike might encourage Causal members to upgrade to Annual membership.