Case study of a Cyclistic (Divvy) bike share data

Shibin Shahid K V

2022-11-06

This is a case study done as part of Google Data Analytics Capstone: Case study of a Cyclistic (Divvy) bike share data. The bike share data from October 2021 to September 2022 is used for analysing the trend.

Installing the required packages

```
install.packages("tidyverse", repos = "http://cran.us.r-project.org ")  # tidyverse for data import and
install.packages("lubridate", repos = "http://cran.us.r-project.org ")  # lubridate for date functions
install.packages("ggplot2", repos = "http://cran.us.r-project.org ")    # ggplot for visualization
install.packages("readxl", repos = "http://cran.us.r-project.org ")     # readxl for uploading excel fi
install.packages("janitor", repos = "http://cran.us.r-project.org ")    # for cleaning
```

Load the packages

```
library(tidyverse)  #helps wrangle data
library(lubridate)  #helps wrangle date attributes
library(ggplot2)    #helps visualize data
library(readxl)     #helps to read excel file
library(janitor)
```

STEP 1: COLLECT DATA The source for downloading the data is from the link. *I have done the analysis from October 2021 to September 2022.* Upload Divvy datasets (xlsx files) here. *Here I have added columns for ride_length and day_of_week from excel sheet itself.

```
tripdata_2021_oct <- read_excel('202110-divvy-tripdata.xlsx')
tripdata_2021_nov <- read_excel('202111-divvy-tripdata.xlsx')
tripdata_2021_dec <- read_excel('202112-divvy-tripdata.xlsx')
tripdata_2022_jan <- read_excel('202201-divvy-tripdata.xlsx')
tripdata_2022_feb <- read_excel('202202-divvy-tripdata.xlsx')
tripdata_2022_mar <- read_excel('202203-divvy-tripdata.xlsx')
tripdata_2022_apr <- read_excel('202204-divvy-tripdata.xlsx')
tripdata_2022_may <- read_excel('202205-divvy-tripdata.xlsx')
tripdata_2022_jun <- read_excel('202206-divvy-tripdata.xlsx')
tripdata_2022_jul <- read_excel('202207-divvy-tripdata.xlsx')
tripdata_2022_aug <- read_excel('202208-divvy-tripdata.xlsx')
tripdata_2022_sep <- read_excel('202209-divvy-tripdata.xlsx')
```

STEP 2: WRANGLE DATA AND COMBINE INTO A SINGLE FILE *Compare column names of each of the files.* checking the datatypes are consistent in all sheets. For example, the first and last data is shown.

```
str(tripdata_2021_oct)
```

```
## tibble [631,226 x 15] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:631226] "620BC6107255BF4C" "4471C70731AB2E45" "26CA69D43D15EE14" "3629
##  $ rideable_type     : chr [1:631226] "electric_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:631226], format: "2021-10-22 12:46:42" "2021-10-21 09:12:37" ...
##  $ ended_at          : POSIXct[1:631226], format: "2021-10-22 12:49:50" "2021-10-21 09:14:14" ...
##  $ start_station_name: chr [1:631226] "Kingsbury St & Kinzie St" NA NA NA ...
##  $ start_station_id  : chr [1:631226] "KA1503000043" NA NA NA ...
##  $ end_station_name  : chr [1:631226] NA NA NA NA ...
##  $ end_station_id    : chr [1:631226] NA NA NA NA ...
##  $ start_lat         : num [1:631226] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:631226] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:631226] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:631226] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:631226] "member" "member" "member" "member" ...
##  $ ride_length       : POSIXct[1:631226], format: "1899-12-31 00:03:08" "1899-12-31 00:01:37" ...
##  $ day_of_week       : num [1:631226] 6 5 7 7 4 5 5 4 5 4 ...
```

```
str(tripdata_2021_oct)
str(tripdata_2021_nov)
str(tripdata_2021_dec)
str(tripdata_2022_jan)
str(tripdata_2022_feb)
str(tripdata_2022_mar)
str(tripdata_2022_apr)
str(tripdata_2022_may)
str(tripdata_2022_jun)
str(tripdata_2022_jul)
str(tripdata_2022_aug)
str(tripdata_2022_sep) # end_station_id was found to be in num data type. so it has to be converted to
```

```
str(tripdata_2022_sep)
```

```
## tibble [701,339 x 15] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:701339] "5156990AC19CA285" "E12D4A16BF51C274" "A02B53CD7DB72DD7" "C82E
##  $ rideable_type     : chr [1:701339] "electric_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:701339], format: "2022-09-01 08:36:22" "2022-09-01 17:11:29" ...
##  $ ended_at          : POSIXct[1:701339], format: "2022-09-01 08:39:05" "2022-09-01 17:14:45" ...
##  $ start_station_name: chr [1:701339] NA NA NA NA ...
##  $ start_station_id  : chr [1:701339] NA NA NA NA ...
##  $ end_station_name  : chr [1:701339] "California Ave & Milwaukee Ave" NA NA NA ...
##  $ end_station_id    : num [1:701339] 13084 NA NA NA NA ...
##  $ start_lat         : num [1:701339] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:701339] -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num [1:701339] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:701339] -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:701339] "casual" "casual" "casual" "casual" ...
##  $ ride_length       : POSIXct[1:701339], format: "1899-12-31 00:02:43" "1899-12-31 00:03:16" ...
##  $ day_of_week       : num [1:701339] 5 5 5 5 5 5 5 5 5 5 ...
```

converting end_station_id of sep_2022 to chr

```
tripdata_2022_sep <- mutate(tripdata_2022_sep,
  end_station_id = as.character(end_station_id))
```

Combining the 12 months data into 1 dataframe

```
tripdata_combined <- bind_rows(
                              tripdata_2021_oct,
                              tripdata_2021_nov,
                              tripdata_2021_dec,
                              tripdata_2022_jan,
                              tripdata_2022_feb,
                              tripdata_2022_mar,
                              tripdata_2022_apr,
                              tripdata_2022_may,
                              tripdata_2022_jun,
                              tripdata_2022_jul,
                              tripdata_2022_aug,
                              tripdata_2022_sep
                              )
```

STEP 3: CLEAN UP AND ADD DATA TO PREPARE FOR ANALYSIS Inspect the new table that has been created

```
colnames(tripdata_combined) #List of column names
```

```
##  [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"      "ride_length"        "day_of_week"
```

```
nrow(tripdata_combined)     #How many rows are in data frame?
```

```
## [1] 5828235
```

```
dim(tripdata_combined)      #Dimensions of the data frame?
```

```
## [1] 5828235      15
```

```
head(tripdata_combined)     #See the first 6 rows of data frame.  Also tail(all_trips)
```

```
## # A tibble: 6 x 15
##   ride_id        ridea~1 started_at          ended_at            start~2 start~3
##   <chr>          <chr>   <dttm>              <dttm>              <chr>   <chr>
## 1 620BC6107255B~ electr~ 2021-10-22 12:46:42 2021-10-22 12:49:50 Kingsb~ KA1503~
## 2 4471C70731AB2~ electr~ 2021-10-21 09:12:37 2021-10-21 09:14:14 <NA>    <NA>
## 3 26CA69D43D15E~ electr~ 2021-10-16 16:28:39 2021-10-16 16:36:26 <NA>    <NA>
## 4 362947F0437E1~ electr~ 2021-10-16 16:17:48 2021-10-16 16:19:03 <NA>    <NA>
## 5 BB731DE2F2EC5~ electr~ 2021-10-20 23:17:54 2021-10-20 23:26:10 <NA>    <NA>
```

```
## 6 7176307BBC097~ electr~ 2021-10-21 16:57:37 2021-10-21 17:11:58 <NA>    <NA>
## # ... with 9 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, ride_length <dttm>, day_of_week <dbl>, and abbreviated
## #   variable names 1: rideable_type, 2: start_station_name, 3: start_station_id
```

```
str(tripdata_combined)      #See list of columns and data types (numeric, character, etc)
```

```
## tibble [5,828,235 x 15] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:5828235] "620BC6107255BF4C" "4471C70731AB2E45" "26CA69D43D15EE14" "362
##  $ rideable_type     : chr [1:5828235] "electric_bike" "electric_bike" "electric_bike" "electric_bik
##  $ started_at        : POSIXct[1:5828235], format: "2021-10-22 12:46:42" "2021-10-21 09:12:37" ...
##  $ ended_at          : POSIXct[1:5828235], format: "2021-10-22 12:49:50" "2021-10-21 09:14:14" ...
##  $ start_station_name: chr [1:5828235] "Kingsbury St & Kinzie St" NA NA NA ...
##  $ start_station_id  : chr [1:5828235] "KA1503000043" NA NA NA ...
##  $ end_station_name  : chr [1:5828235] NA NA NA NA ...
##  $ end_station_id    : chr [1:5828235] NA NA NA NA ...
##  $ start_lat         : num [1:5828235] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:5828235] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:5828235] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:5828235] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:5828235] "member" "member" "member" "member" ...
##  $ ride_length       : POSIXct[1:5828235], format: "1899-12-31 00:03:08" "1899-12-31 00:01:37" ...
##  $ day_of_week       : num [1:5828235] 6 5 7 7 4 5 5 4 5 4 ...
```

```
summary(tripdata_combined)  #Statistical summary of data. Mainly for numerics
```

```
##    ride_id          rideable_type        started_at
##  Length:5828235     Length:5828235     Min.   :2021-10-01 00:00:09.00
##  Class :character   Class :character   1st Qu.:2022-02-28 19:21:08.50
##  Mode  :character   Mode  :character   Median :2022-06-08 06:41:28.00
##                                        Mean   :2022-05-06 21:39:18.18
##                                        3rd Qu.:2022-08-02 11:26:01.00
##                                        Max.   :2022-09-30 23:59:56.00
##
##     ended_at                      start_station_name start_station_id
##  Min.   :2021-10-01 00:03:11.0   Length:5828235     Length:5828235
##  1st Qu.:2022-02-28 19:34:02.5   Class :character   Class :character
##  Median :2022-06-08 06:55:07.0   Mode  :character   Mode  :character
##  Mean   :2022-05-06 21:58:54.2
##  3rd Qu.:2022-08-02 11:46:26.0
##  Max.   :2022-10-05 19:53:11.0
##
##  end_station_name   end_station_id       start_lat       start_lng
##  Length:5828235     Length:5828235     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:5828235
```

```
##  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.   :42.37   Max.   :-87.30
##  NA's   :5844    NA's   :5844
##   ride_length                      day_of_week
##  Min.   :1899-12-30 21:42:35.00   Min.   :1.000
##  1st Qu.:1899-12-31 00:05:56.00   1st Qu.:2.000
##  Median :1899-12-31 00:10:29.00   Median :4.000
##  Mean   :1899-12-31 00:19:37.23   Mean   :4.117
##  3rd Qu.:1899-12-31 00:18:51.00   3rd Qu.:6.000
##  Max.   :1900-01-28 06:25:01.00   Max.   :7.000
##  NA's   :1
```

Finding the ride_length and converting into seconds

```
tripdata_combined$ride_length <- difftime(
                                        tripdata_combined$ended_at,
                                        tripdata_combined$started_at,
                                        units = "secs"
                                        )
```

converting ride_length data type from 'difftime' num to num

```
tripdata_combined$ride_length <- as.numeric(tripdata_combined$ride_length)
```

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

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

Remove "bad" data The dataframe includes a few entries when bikes were taken out of docks and checked for quality by Divvy or ride_length was negative We will create a new version of the dataframe (all_data) since data is being removed

```
all_trips <- tripdata_combined %>% filter(ride_length>=0) %>% filter(start_station_name != "HQ QR")
```

STEP 4: CONDUCT DESCRIPTIVE ANALYSIS Descriptive analysis on ride_length (all figures in seconds)

```
mean(all_trips$ride_length) #straight average (total ride length / rides)
```

```
## [1] 1243.658
```

```
median(all_trips$ride_length) #midpoint number in the ascending array of ride lengths
```

```
## [1] 644
```

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

```
## [1] 2442301
```

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

```
## [1] 0
```

Compare members and casual users

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual             1915.3672
## 2                  member              775.0878
```

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual                   840
## 2                  member                   542
```

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual               2442301
## 2                  member                 93594
```

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

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

See the average ride time by each day for members vs casual users

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

```
##    all_trips$member_casual all_trips$day_of_week all_trips$ride_length
## 1                   casual                Friday             1836.1176
## 2                   member                Friday              759.4998
## 3                   casual                Monday             1943.0161
## 4                   member                Monday              748.7428
```

```
## 5                  casual             Saturday            2118.7804
## 6                  member             Saturday             869.4681
## 7                  casual               Sunday            2230.3099
## 8                  member               Sunday             866.0173
## 9                  casual             Thursday            1676.9442
## 10                 member             Thursday             746.5041
## 11                 casual              Tuesday            1686.0902
## 12                 member              Tuesday             737.0860
## 13                 casual            Wednesday            1632.7380
## 14                 member            Wednesday             734.2675
```

Notice that the days of the week are out of order. Let's fix that.

```
all_trips$day_of_week <- ordered(all_trips$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "Wednes
```

Now, let's run the average ride time by each day for members vs casual users

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

```
##    all_trips$member_casual all_trips$day_of_week all_trips$ride_length
## 1                  casual                 Sunday            2230.3099
## 2                  member                 Sunday             866.0173
## 3                  casual                 Monday            1943.0161
## 4                  member                 Monday             748.7428
## 5                  casual                Tuesday            1686.0902
## 6                  member                Tuesday             737.0860
## 7                  casual              Wednesday            1632.7380
## 8                  member              Wednesday             734.2675
## 9                  casual               Thursday            1676.9442
## 10                 member               Thursday             746.5041
## 11                 casual                 Friday            1836.1176
## 12                 member                 Friday             759.4998
## 13                 casual               Saturday            2118.7804
## 14                 member               Saturday             869.4681
```

analyze ridership data by type and weekday

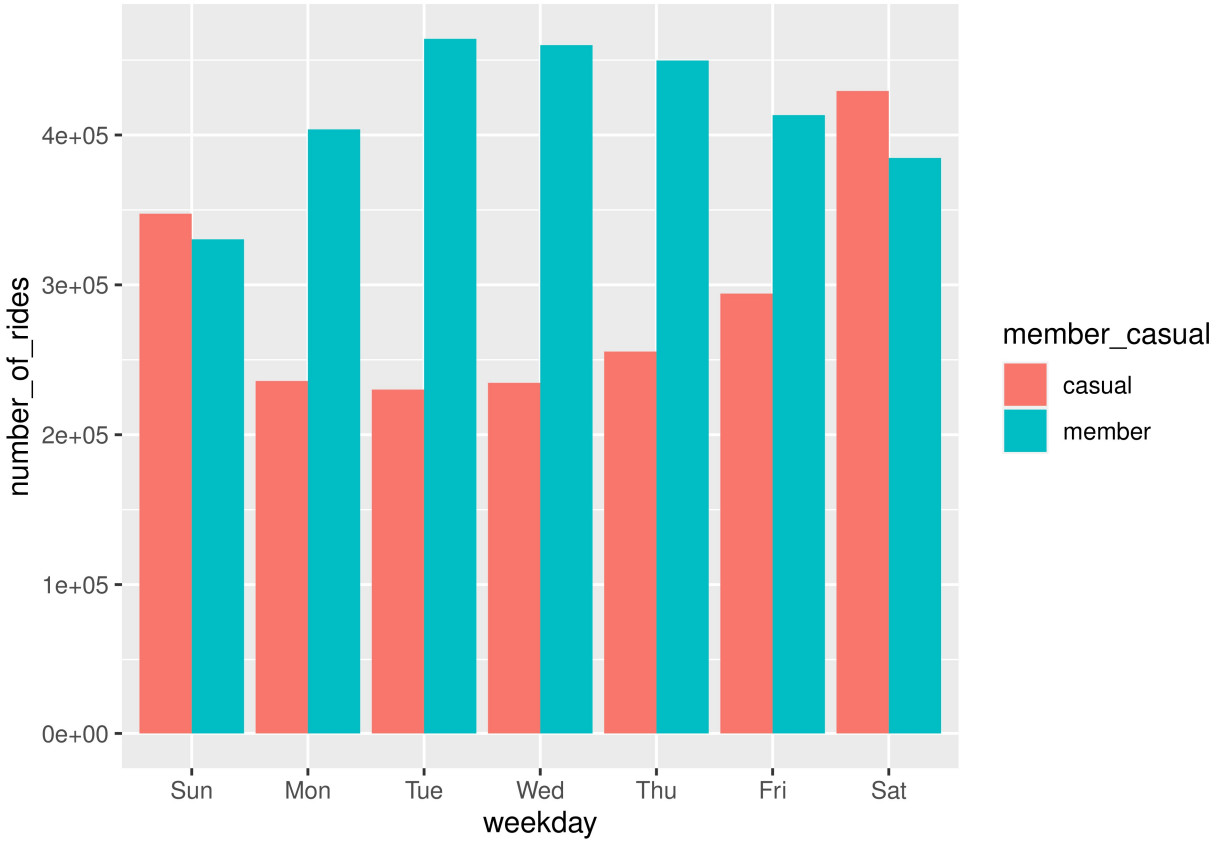
```
all_trips %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%  # creates weekday field using wday()
  group_by(member_casual, weekday) %>%                  # groups by usertype and weekday
  summarise(number_of_rides = n()                                # calculates the number of rides an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

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              347491            2230.
##  2 casual        Mon              235875            1943.
##  3 casual        Tue              230115            1686.
##  4 casual        Wed              234628            1633.
```

```
##  5 casual        Thu              255488           1677.
##  6 casual        Fri              294181           1836.
##  7 casual        Sat              429369           2119.
##  8 member        Sun              330398            866.
##  9 member        Mon              403761            749.
## 10 member        Tue              464179            737.
## 11 member        Wed              459975            734.
## 12 member        Thu              449707            747.
## 13 member        Fri              413270            759.
## 14 member        Sat              384684            869.
```

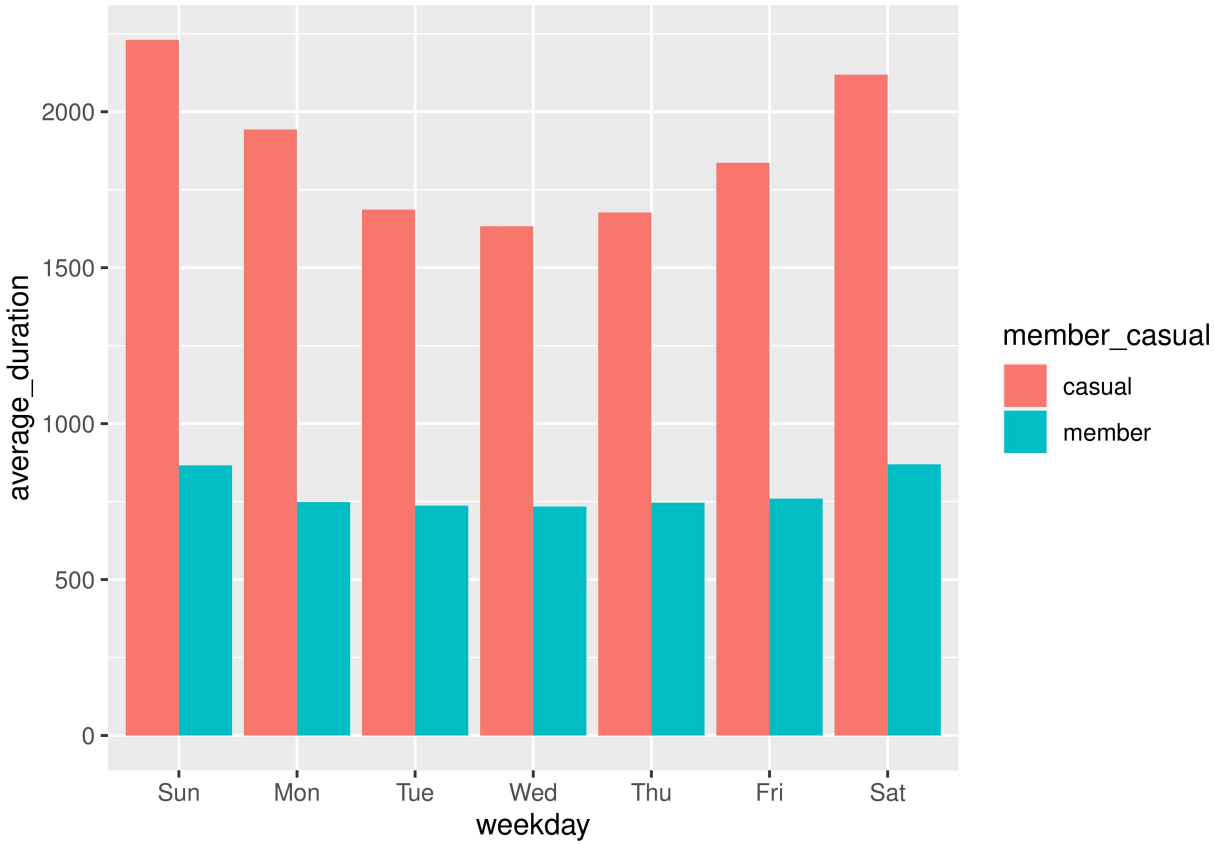
Let's visualize the number of rides by rider type

```
all_trips %>%
  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")
```



Let's create a visualization for average duration

```r
all_trips%>%
  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")
```



Conclusions

- More number of Casual riders are using the Cyclistic on weekends compared to Cyclistic members.
- Everyday the casual riders are riding the cycle for more duration compared to Cyclistic members.