Bike Share Case Analysis Findings

Hasan

2022-10-05

Dataset

The data is taken from https://divvy-tripdata.s3.amazonaws.com/index.html where the months are between August 2021 and August 2022. The data has been made available by Motivate International Inc. under this license.)

Prepare for analysis

This section is for creating loading libraries, creating the data frame and viewing the structure of the data. Loaded the 'tidyverse', 'here', 'skimr', 'janitor' and 'lubridate' packages.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(here) #library for path
```

```
## here() starts at /home/hasan/cyclist data
```

```
library(skimr)
library(janitor) #library for cleaning
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(lubridate) #library for date functions
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

Importing the data:

```r
Aug2021 <- read_csv('Aug-2021.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.
```

```r
Sep2021 <- read_csv('Sep-2021.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.
```

```r
Oct2021 <- read_csv('Oct-2021.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.
```

```r
Nov2021 <- read_csv('Nov-2021.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.
```

```r
Dec2021 <- read_csv('Dec-2021.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.
```

```
Jan2022 <- read_csv('Jan-2022.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.
```

```
Feb2022 <- read_csv('Feb-2022.csv')
```

```
## Rows: 115609 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.
```

```
Mar2022 <- read_csv('Mar-2022.csv')
```

```
## Rows: 284042 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.
```

```
Apr2022 <- read_csv('Apr-2022.csv')
```

```
## Rows: 371249 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.
```

```
May2022 <- read_csv('May-2022.csv')
```

```
## Rows: 634858 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.
```

```
Jun2022 <- read_csv('Jun-2022.csv')
```

```
## Rows: 769204 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.
```

```
Jul2022 <- read_csv('Jul-2022.csv')
```

```
## Rows: 823488 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.
```

```
Aug2022 <- read_csv('Aug-2022.csv')
```

```
## Rows: 785932 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.
```

Checking column names to see if there are differences in naming:

```
colnames(Aug2021)
```

```
##  [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(Sep2021)
```

```
##  [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(Oct2021)
```

```
##  [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(Nov2021)
```

```
##  [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(Dec2021)
```

```
##  [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(Jan2022)
```

```
##  [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(Feb2022)
```

```
##  [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(Mar2022)
```

```
##  [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(Apr2022)
```

```
##  [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(May2022)
```

```
##  [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(Jun2022)
```

```
##  [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(Jul2022)
```

```
##  [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(Aug2022)
```

```
##  [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"
```

Combining the data into a single data frame:

```
all <- bind_rows(Aug2021, Sep2021, Oct2021, Nov2021, Dec2021, Jan2022, Feb2022, Mar2022, Apr2022, May20
```

Deleting unnecessary columns:

```
all <- all %>%
  select(-c(start_lat, start_lng, end_lat, end_lng))
```

Getting initial info about the data.

```
colnames(all)
```

```
## [1] "ride_id"            "rideable_type"      "started_at"
## [4] "ended_at"           "start_station_name" "start_station_id"
## [7] "end_station_name"   "end_station_id"     "member_casual"
```

```
nrow(all)
```

```
## [1] 6687395
```

```
dim(all)
```

```
## [1] 6687395       9
```

```
head(all)
```

```
## # A tibble: 6 x 9
##   ride_id        ridea~1 started_at          ended_at            start~2 start~3
##   <chr>          <chr>   <dttm>              <dttm>              <chr>   <chr>
## 1 99103BB87CC6C~ electr~ 2021-08-10 17:15:49 2021-08-10 17:22:44 <NA>    <NA>
## 2 EAFCCCFB0A3FC~ electr~ 2021-08-10 17:23:14 2021-08-10 17:39:24 <NA>    <NA>
## 3 9EF4F46C57AD2~ electr~ 2021-08-21 02:34:23 2021-08-21 02:50:36 <NA>    <NA>
```

```
## 4 5834D3208BFAF~ electr~ 2021-08-21 06:52:55 2021-08-21 07:08:13 <NA>    <NA>
## 5 CD825CB87ED1D~ electr~ 2021-08-19 11:55:29 2021-08-19 12:04:11 <NA>    <NA>
## 6 612F12C94A964~ electr~ 2021-08-19 12:41:12 2021-08-19 12:47:47 <NA>    <NA>
## # ... with 3 more variables: end_station_name <chr>, end_station_id <chr>,
## #   member_casual <chr>, and abbreviated variable names 1: rideable_type,
## #   2: start_station_name, 3: start_station_id
```

```
tail(all)
```

```
## # A tibble: 6 x 9
##   ride_id        ridea~1 started_at          ended_at            start~2 start~3
##   <chr>          <chr>   <dttm>              <dttm>              <chr>   <chr>
## 1 3A50755D86939~ electr~ 2022-08-09 06:41:21 2022-08-09 06:45:22 Ashlan~ 13269
## 2 EC17BE8AB1D73~ electr~ 2022-08-12 08:28:26 2022-08-12 08:42:26 Paulin~ TA1309~
## 3 57BAC0EA3A067~ electr~ 2022-08-22 12:30:19 2022-08-22 12:36:32 Califo~ 13096
## 4 9B97FAE30276C~ electr~ 2022-08-09 07:43:31 2022-08-09 07:50:50 Clinto~ WL-012
## 5 2B4A9BA1E9AA9~ electr~ 2022-08-24 14:31:58 2022-08-24 14:40:32 Ashlan~ 13269
## 6 7F0D048AC7C3B~ electr~ 2022-08-24 09:45:04 2022-08-24 09:45:19 Clark ~ 13179
## # ... with 3 more variables: end_station_name <chr>, end_station_id <chr>,
## #   member_casual <chr>, and abbreviated variable names 1: rideable_type,
## #   2: start_station_name, 3: start_station_id
```

```
str(all)
```

```
## tibble [6,687,395 x 9] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:6687395] "99103BB87CC6C1BB" "EAFCCCFB0A3FC5A1" "9EF4F46C57AD234D" "583
##  $ rideable_type     : chr [1:6687395] "electric_bike" "electric_bike" "electric_bike" "electric_bik
##  $ started_at        : POSIXct[1:6687395], format: "2021-08-10 17:15:49" "2021-08-10 17:23:14" ...
##  $ ended_at          : POSIXct[1:6687395], format: "2021-08-10 17:22:44" "2021-08-10 17:39:24" ...
##  $ start_station_name: chr [1:6687395] NA NA NA NA ...
##  $ start_station_id  : chr [1:6687395] NA NA NA NA ...
##  $ end_station_name  : chr [1:6687395] NA NA NA NA ...
##  $ end_station_id    : chr [1:6687395] NA NA NA NA ...
##  $ member_casual     : chr [1:6687395] "member" "member" "member" "member" ...
```

```
summary(all)
```

```
##    ride_id          rideable_type        started_at
##  Length:6687395     Length:6687395     Min.   :2021-08-01 00:00:04
##  Class :character   Class :character   1st Qu.:2021-10-05 23:53:29
##  Mode  :character   Mode  :character   Median :2022-04-05 13:03:11
##                                        Mean   :2022-02-24 01:00:11
##                                        3rd Qu.:2022-06-28 18:21:44
##                                        Max.   :2022-08-31 23:59:39
##     ended_at                   start_station_name start_station_id
##  Min.   :2021-08-01 00:03:11   Length:6687395     Length:6687395
##  1st Qu.:2021-10-06 00:23:57   Class :character   Class :character
##  Median :2022-04-05 13:15:03   Mode  :character   Mode  :character
##  Mean   :2022-02-24 01:20:10
##  3rd Qu.:2022-06-28 18:39:52
##  Max.   :2022-09-06 21:49:04
##  end_station_name   end_station_id     member_casual
##  Length:6687395     Length:6687395     Length:6687395
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
```

```
##
##
```

Getting some info about the number of casual customers and members:

```
table(all$member_casual) # to see the number of members and casuals
```

```
##
##  casual  member
## 2881150 3806245
```

To analyze in more depth, created new columns about date.

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

library(hydroTSM) # a library to get seasons from dates
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: xts
```

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
##
## Attaching package: 'hydroTSM'
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
all$season <- time2season(all$date, out.fmt = "seasons") #creating a column for seasons
table(all$season)
```

```
##
##  autumn  spring  summer  winter
## 1747351 1290149 3182976  466919
```

Creating a new column for ride length by subtracting the start date from the end date.

```
all$ride_length <- difftime(all$ended_at,all$started_at) # get the difference using difftime()

#ride length should be numeric so we check

is.factor(all$ride_length)
```

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

```
all$ride_length <- as.numeric(as.character(all$ride_length)) # change the type of ride length as numeri
is.numeric(all$ride_length) # to check again
```

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

Click here: R markdown

Cleaning

This section is for cleaning the data. A clean data is a must for a good analysis. Used this website to learn how to drop rows with certaing conditions in R. Also created a new data frame due to removing some of the data.

```
all_v2 <- drop_na(all) # dropping all of the NA values
all_v2 <-  all_v2[!(all_v2$ride_length<0 | all_v2$start_station_name == "HQ QR"), ] # to remove bad dat
```

Analysis

To get a glimpse about the new data frame:

```
glimpse(all_v2)
```

```
## Rows: 5,234,432
## Columns: 16
## $ ride_id            <chr> "DD06751C6019D865", "79973DC3B232048F", "0249AD4B25~
## $ rideable_type      <chr> "classic_bike", "classic_bike", "classic_bike", "cl~
## $ started_at         <dttm> 2021-08-08 17:21:26, 2021-08-27 08:53:52, 2021-08-~
## $ ended_at           <dttm> 2021-08-08 17:25:37, 2021-08-27 09:18:29, 2021-08-~
## $ start_station_name <chr> "Desplaines St & Kinzie St", "Larrabee St & Armitag~
## $ start_station_id   <chr> "TA1306000003", "TA1309000006", "13157", "13042", "~
## $ end_station_name   <chr> "Kingsbury St & Kinzie St", "Michigan Ave & Oak St"~
## $ end_station_id     <chr> "KA1503000043", "13042", "13157", "13042", "13042",~
## $ member_casual      <chr> "member", "member", "member", "casual", "casual", "~
## $ date               <date> 2021-08-08, 2021-08-27, 2021-08-08, 2021-08-12, 20~
## $ month              <chr> "08", "08", "08", "08", "08", "08", "08", "08", "08~
## $ day                <chr> "08", "27", "08", "12", "23", "23", "28", "20", "09~
## $ year               <chr> "2021", "2021", "2021", "2021", "2021", "2021", "20~
## $ day_of_week        <chr> "Sunday", "Friday", "Sunday", "Thursday", "Monday",~
## $ season             <chr> "summer", "summer", "summer", "summer", "summer", "~
## $ ride_length        <dbl> 251, 1477, 37, 282, 2156, 2402, 85, 3245, 3538, 123~
```

```
mean(all_v2$ride_length) #straight average (total ride length / rides)
```

```
## [1] 1104.369
```

```
median(all_v2$ride_length) #midpoint number in the ascending array of ride lengths
```

```
## [1] 673
```

```
max(all_v2$ride_length) #longest ride
```

```
## [1] 2497750
```

```
min(all_v2$ride_length) #shortest ride
```

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

```
#above can be get by summary(all_v2$ride_length)

aggregate(all_v2$ride_length ~ all_v2$member_casual, FUN = mean)
```

```
##   all_v2$member_casual all_v2$ride_length
## 1               casual          1572.9577
## 2               member           759.4304
```

```
aggregate(all_v2$ride_length ~ all_v2$member_casual, FUN = median)
```

```
##   all_v2$member_casual all_v2$ride_length
## 1               casual                896
## 2               member                554
```

```
aggregate(all_v2$ride_length ~ all_v2$member_casual, FUN = max)
```

```
##   all_v2$member_casual all_v2$ride_length
## 1               casual            2497750
## 2               member              89575
```

```
aggregate(all_v2$ride_length ~ all_v2$member_casual, FUN = min)
```

```
##   all_v2$member_casual all_v2$ride_length
## 1               casual                  0
## 2               member                  0
```

```
aggregate(all_v2$ride_length ~ all_v2$member_casual + all_v2$day_of_week, FUN = mean)  # to see the ave
```

```
##    all_v2$member_casual all_v2$day_of_week all_v2$ride_length
## 1                casual             Friday          1470.8695
## 2                member             Friday           742.1041
## 3                casual             Monday          1621.4174
## 4                member             Monday           734.2547
## 5                casual           Saturday          1719.4897
## 6                member           Saturday           854.1877
## 7                casual             Sunday          1812.4474
## 8                member             Sunday           859.5762
## 9                casual           Thursday          1391.0267
## 10               member           Thursday           727.7317
## 11               casual            Tuesday          1384.4727
## 12               member            Tuesday           713.3100
## 13               casual          Wednesday          1348.6163
## 14               member          Wednesday           719.8033
```

```
# to order the days

all_v2$day_of_week <- ordered(all_v2$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "Wednesday",

aggregate(all_v2$ride_length ~ all_v2$member_casual + all_v2$day_of_week, FUN = mean)
```

```
##    all_v2$member_casual all_v2$day_of_week all_v2$ride_length
## 1                casual             Sunday          1812.4474
## 2                member             Sunday           859.5762
## 3                casual             Monday          1621.4174
## 4                member             Monday           734.2547
## 5                casual            Tuesday          1384.4727
## 6                member            Tuesday           713.3100
```

```
## 7               casual          Wednesday          1348.6163
## 8               member          Wednesday           719.8033
## 9               casual           Thursday          1391.0267
## 10              member           Thursday           727.7317
## 11              casual             Friday          1470.8695
## 12              member             Friday           742.1041
## 13              casual           Saturday          1719.4897
## 14              member           Saturday           854.1877
```

```
aggregate(all_v2$ride_length ~ all_v2$member_casual + all_v2$season, FUN = mean) # to see the average r
```

```
##   all_v2$member_casual all_v2$season all_v2$ride_length
## 1               casual        autumn          1606.1883
## 2               member        autumn           733.4841
## 3               casual        spring          1644.5130
## 4               member        spring           746.1221
## 5               casual        summer          1538.9187
## 6               member        summer           807.5994
## 7               casual        winter          1516.4300
## 8               member        winter           631.2849
```

To see the mean ride length per season, also to compare members and casual customers:

```
all_v2 %>%
  group_by(member_casual, season) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, season)
```

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

```
## # A tibble: 8 x 4
## # Groups:   member_casual [2]
##   member_casual season number_of_rides average_duration
##   <chr>         <chr>            <int>            <dbl>
## 1 casual        autumm          552001            1606.
## 2 casual        spring          379297            1645.
## 3 casual        summer         1215295            1539.
## 4 casual        winter           72825            1516.
## 5 member        autumm          802956             733.
## 6 member        spring          611789             746.
## 7 member        summer         1327417             808.
## 8 member        winter          272852             631.
```

To see the mean ride length per week days, also to compare members and casual customers:

```
all_v2 %>%
  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 and average
            ,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
```

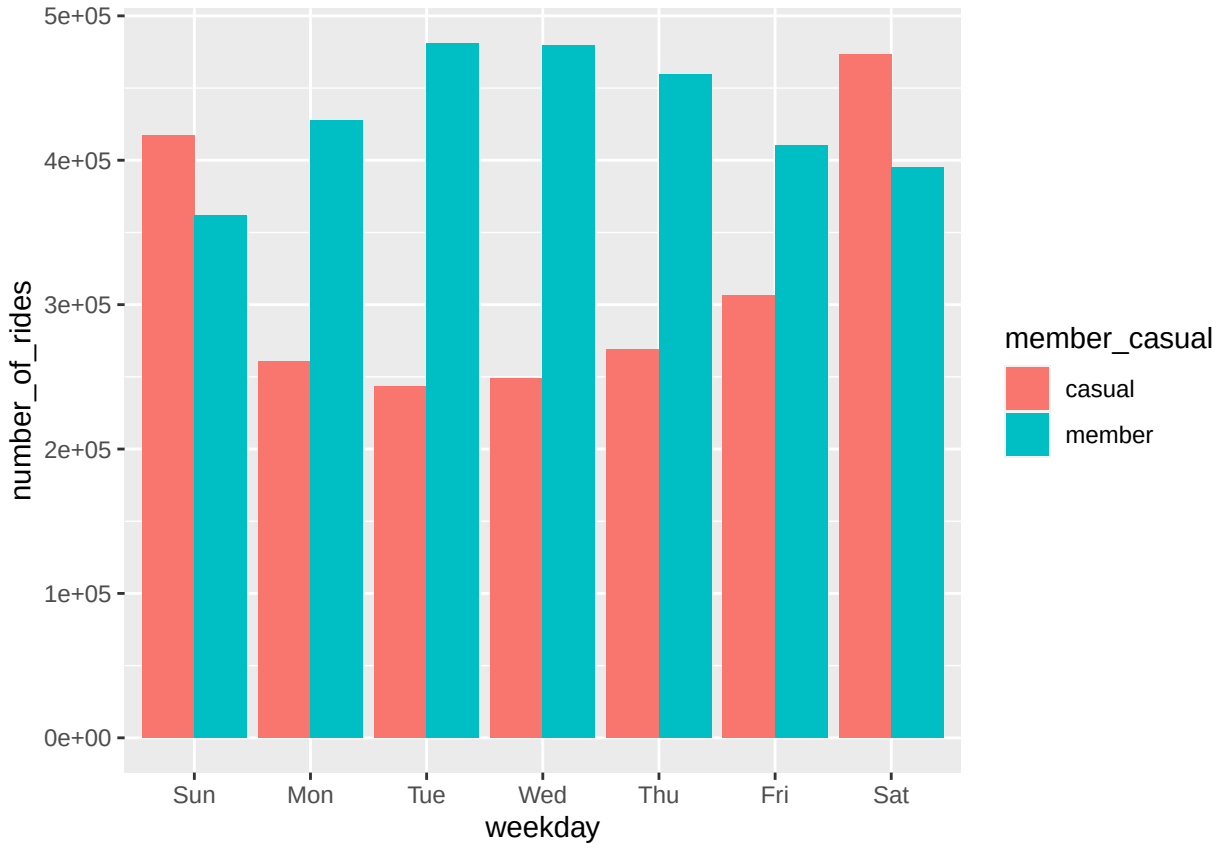
```
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              417389            1812.
##  2 casual        Mon              260931            1621.
##  3 casual        Tue              243155            1384.
##  4 casual        Wed              249207            1349.
##  5 casual        Thu              269208            1391.
##  6 casual        Fri              306150            1471.
##  7 casual        Sat              473378            1719.
##  8 member        Sun              361590             860.
##  9 member        Mon              427709             734.
## 10 member        Tue              481196             713.
## 11 member        Wed              479340             720.
## 12 member        Thu              459754             728.
## 13 member        Fri              410428             742.
## 14 member        Sat              394997             854.
```

Visualization

This section is for visualization. The following graphs give visualizations about the analysis.

```
all_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")
```

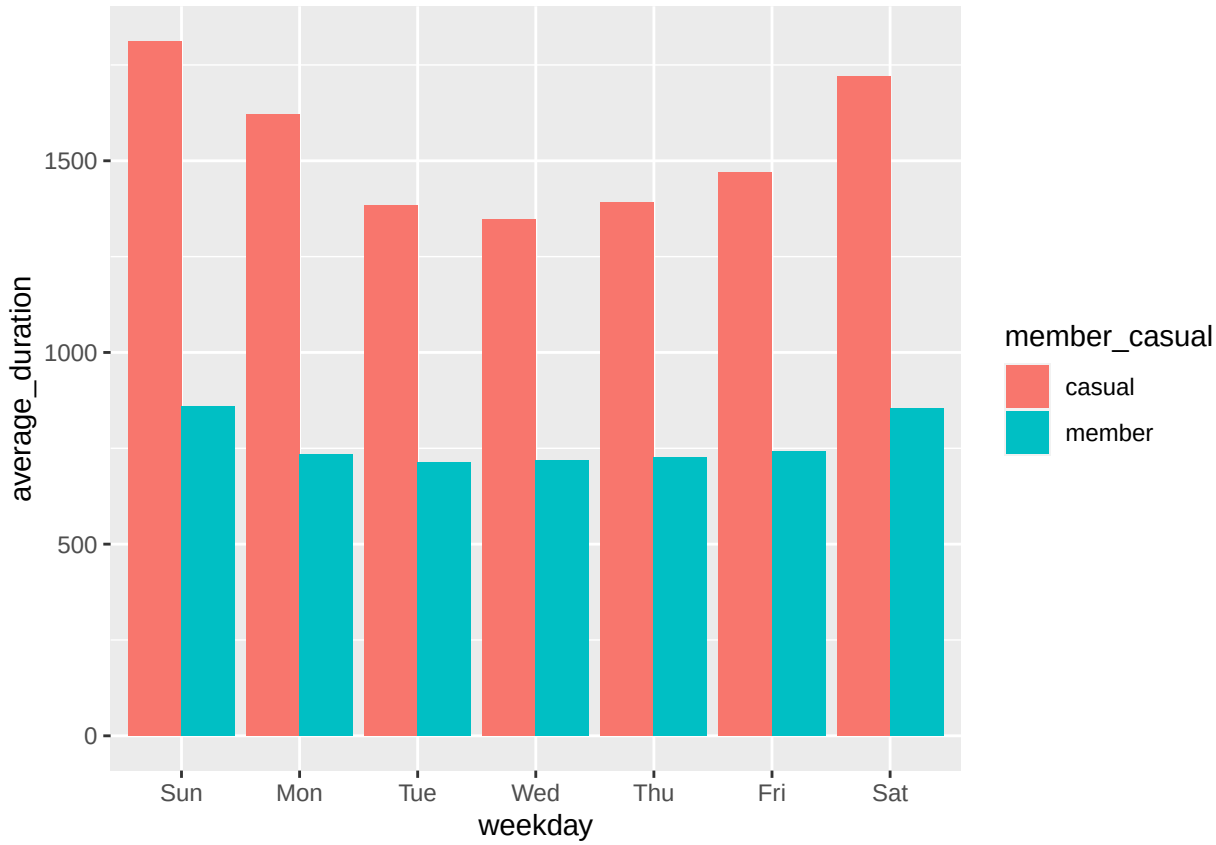
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```




```
#visualization for average duration

all_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")
```

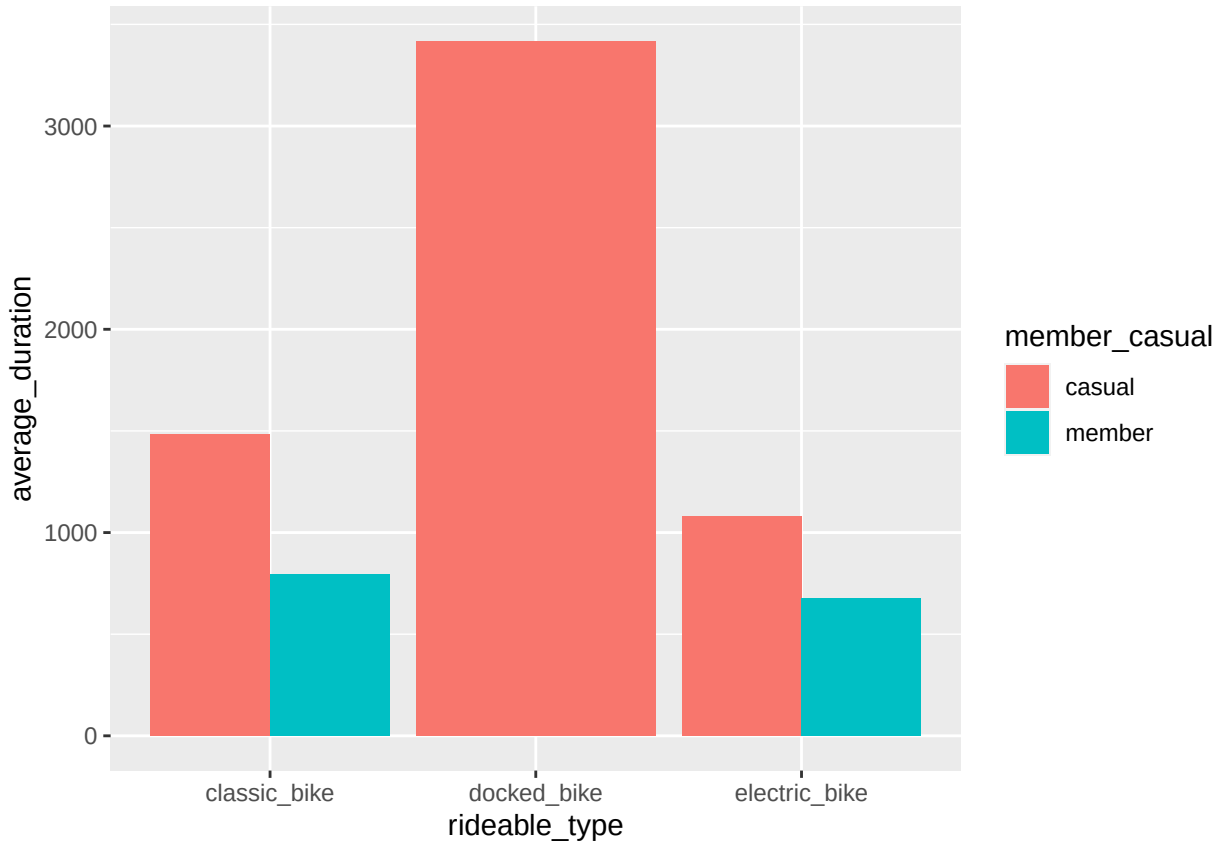
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```




```
#visualization for rideable type

all_v2 %>%
  group_by(member_casual, rideable_type) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, rideable_type) %>%
  ggplot(aes(x = rideable_type, y = average_duration, fill = member_casual)) +
    geom_col(position = "dodge")
```

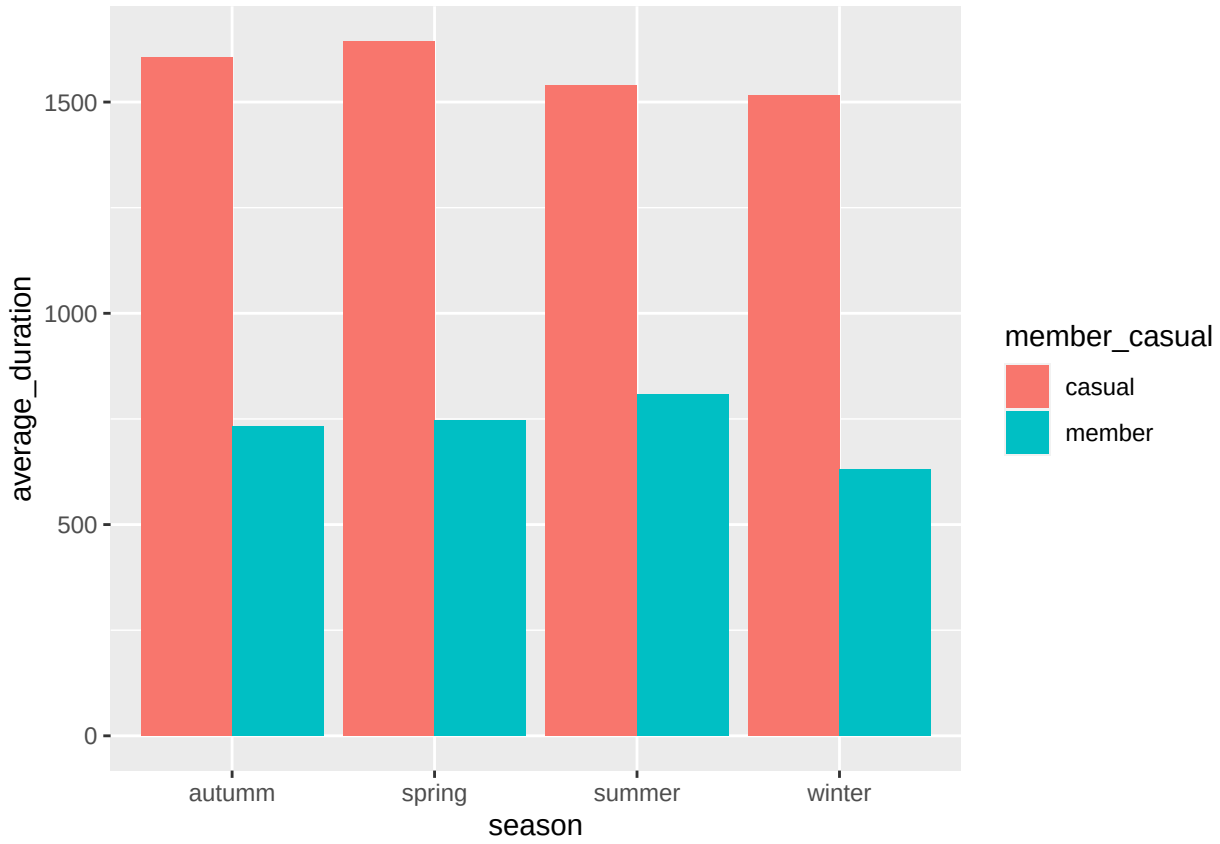
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```




```
#visualization for seasons

all_v2 %>%
  group_by(member_casual, season) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, season) %>%
  ggplot(aes(x = season, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```

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



Findings

First of all, casual customers ride more than members in weekends in terms of number of rides. For this reason, if the company wants to get more members from casual customers, adding special events or packets for weekends could be useful. A membership packet for weekends could attract casual customers. Members ride more than casual customers in weekdays. This could be due to using bikes for transportation.

Secondly, the average duration of ride length of casual customers is more than the average duration of the members. I believe this could be due to members being busy but additional data is required.

Additionally, none of the members use docked bike type. For this reason, a discount or a special packet for classic and electric bike types could attract more members from the casual customers.

Finally, I checked the average duration per seasons, and the average ride length of casual customers is more than that of members. There is a slight decrease in winters but this is probably due to weather being cold.