Cyclistic Bike-Share Case Study

Brunda Suresh

8/2/2021

This case study is the capstone for my Google Data Analytics Certification aimed at introducing the tasks of a junior data analyst.I will be following a roadmap provided by the certification to complete the case study.

The case study involves a fictional company 'Cyclistic'.

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.

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

My Assignement: "How do annual members and casual riders use Cyclistic bikes differently?"

Business task : Understand how casual riders and members use cyclistic bikes differently to design new marketing strategy to convert casual riders into annual members, thus maximizing the number of annual memberships

I have the cyclistic's historical data between 2013 to 2021. I will be using part of the provided data to analyze and identify trends.The data is made available by Motivate International Inc.

The data is provided in csv format.Since the data is large I have chosen to use R studio to analyze the data at hand.I will hence be downloading the required packages in R.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(lubridate)
```

```
##
## Attaching package: ’lubridate’

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

```r
library(ggplot2)
```

I have chosen the historical data of 12 months between April 2020 to March 2021.

The data of respective 12 months is imported below as follows:

```r
apr_20<- read_csv("202004-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```r
may_20<- read_csv("202005-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
jun_20<- read_csv("202006-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
jul_20<- read_csv("202007-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
aug_20<- read_csv("202008-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
sep_20<- read_csv("202009-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
oct_20<- read_csv("202010-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
nov_20<- read_csv("202011-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_double(),
##   end_station_name = col_character(),
##   end_station_id = col_double(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
dec_20<- read_csv("202012-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_character(),
##   end_station_name = col_character(),
##   end_station_id = col_character(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
jan_21<- read_csv("202101-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_character(),
##   end_station_name = col_character(),
##   end_station_id = col_character(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
feb_21<- read_csv("202102-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_character(),
##   end_station_name = col_character(),
##   end_station_id = col_character(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

```
mar_21<- read_csv("202103-divvy-tripdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ride_id = col_character(),
##   rideable_type = col_character(),
##   started_at = col_datetime(format = ""),
##   ended_at = col_datetime(format = ""),
##   start_station_name = col_character(),
##   start_station_id = col_character(),
##   end_station_name = col_character(),
##   end_station_id = col_character(),
##   start_lat = col_double(),
##   start_lng = col_double(),
##   end_lat = col_double(),
##   end_lng = col_double(),
##   member_casual = col_character()
## )
```

The code chunk below help understand the data with the csv files being used:

```
glimpse(apr_20)
```

```
## Rows: 84,776
## Columns: 13
## $ ride_id            <chr> "A847FADBBC638E45", "5405B80E996FF60D", "5DD24A79A4~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-04-26 17:45:14, 2020-04-17 17:08:54, 2020-04--~
## $ ended_at           <dttm> 2020-04-26 18:12:03, 2020-04-17 17:17:03, 2020-04--~
## $ start_station_name <chr> "Eckhart Park", "Drake Ave & Fullerton Ave", "McClu~
## $ start_station_id   <dbl> 86, 503, 142, 216, 125, 173, 35, 434, 627, 377, 508~
## $ end_station_name   <chr> "Lincoln Ave & Diversey Pkwy", "Kosciuszko Park", "~
## $ end_station_id     <dbl> 152, 499, 255, 657, 323, 35, 635, 382, 359, 508, 37~
## $ start_lat          <dbl> 41.8964, 41.9244, 41.8945, 41.9030, 41.8902, 41.896~
## $ start_lng          <dbl> -87.6610, -87.7154, -87.6179, -87.6975, -87.6262, -~
## $ end_lat            <dbl> 41.9322, 41.9306, 41.8679, 41.8992, 41.9695, 41.892~
## $ end_lng            <dbl> -87.6586, -87.7238, -87.6230, -87.6722, -87.6547, -~
## $ member_casual      <chr> "member", "member", "member", "member", "casual", "~
```

```
glimpse(may_20)
```

```
## Rows: 200,274
## Columns: 13
## $ ride_id            <chr> "02668AD35674B983", "7A50CCAF1EDDB28F", "2FFCDFDB91~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-05-27 10:03:52, 2020-05-25 10:47:11, 2020-05--~
## $ ended_at           <dttm> 2020-05-27 10:16:49, 2020-05-25 11:05:40, 2020-05--~
## $ start_station_name <chr> "Franklin St & Jackson Blvd", "Clark St & Wrightwoo~
## $ start_station_id   <dbl> 36, 340, 260, 251, 261, 206, 261, 180, 331, 219, 24~
## $ end_station_name   <chr> "Wabash Ave & Grand Ave", "Clark St & Leland Ave", ~
## $ end_station_id     <dbl> 199, 326, 260, 157, 206, 22, 261, 180, 300, 305, 14~
## $ start_lat          <dbl> 41.8777, 41.9295, 41.9296, 41.9680, 41.8715, 41.847~
## $ start_lng          <dbl> -87.6353, -87.6431, -87.7079, -87.6500, -87.6699, -~
## $ end_lat            <dbl> 41.8915, 41.9671, 41.9296, 41.9367, 41.8472, 41.869~
## $ end_lng            <dbl> -87.6268, -87.6674, -87.7079, -87.6368, -87.6468, -~
## $ member_casual      <chr> "member", "casual", "casual", "casual", "member", "~
```

```
glimpse(jun_20)
```

```
## Rows: 343,005
## Columns: 13
## $ ride_id            <chr> "8CD5DE2C2B6C4CFC", "9A191EB2C751D85D", "F37D14B0B5~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-06-13 23:24:48, 2020-06-26 07:26:10, 2020-06--~
## $ ended_at           <dttm> 2020-06-13 23:36:55, 2020-06-26 07:31:58, 2020-06--~
## $ start_station_name <chr> "Wilton Ave & Belmont Ave", "Federal St & Polk St",~
## $ start_station_id   <dbl> 117, 41, 81, 303, 327, 327, 41, 115, 338, 84, 317, ~
## $ end_station_name   <chr> "Damen Ave & Clybourn Ave", "Daley Center Plaza", "~
## $ end_station_id     <dbl> 163, 81, 5, 294, 117, 117, 81, 303, 164, 53, 168, 1~
## $ start_lat          <dbl> 41.94018, 41.87208, 41.88424, 41.94553, 41.92154, 4~
## $ start_lng          <dbl> -87.65304, -87.62954, -87.62963, -87.64644, -87.653~
## $ end_lat            <dbl> 41.93193, 41.88424, 41.87405, 41.97835, 41.94018, 4~
## $ end_lng            <dbl> -87.67786, -87.62963, -87.62772, -87.65975, -87.653~
## $ member_casual      <chr> "casual", "member", "member", "casual", "casual", "~
```

```
glimpse(jul_20)
```

```
## Rows: 551,480
## Columns: 13
## $ ride_id            <chr> "762198876D69004D", "BEC9C9FBA0D4CF1B", "D2FD8EA432~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-07-09 15:22:02, 2020-07-24 23:56:30, 2020-07--~
## $ ended_at           <dttm> 2020-07-09 15:25:52, 2020-07-25 00:20:17, 2020-07--~
## $ start_station_name <chr> "Ritchie Ct & Banks St", "Halsted St & Roscoe St", ~
## $ start_station_id   <dbl> 180, 299, 329, 181, 268, 635, 113, 211, 176, 31, 14~
## $ end_station_name   <chr> "Wells St & Evergreen Ave", "Broadway & Ridge Ave",~
## $ end_station_id     <dbl> 291, 461, 156, 94, 301, 289, 140, 31, 191, 142, 31,~
## $ start_lat          <dbl> 41.90687, 41.94367, 41.93259, 41.89076, 41.91172, 4~
## $ start_lng          <dbl> -87.62622, -87.64895, -87.63643, -87.63170, -87.626~
## $ end_lat            <dbl> 41.90672, 41.98404, 41.93650, 41.91831, 41.90799, 4~
## $ end_lng            <dbl> -87.63483, -87.66027, -87.64754, -87.63628, -87.631~
## $ member_casual      <chr> "member", "member", "casual", "casual", "member", "~
```

```
glimpse(aug_20)
```

```
## Rows: 622,361
## Columns: 13
## $ ride_id            <chr> "322BD23D287743ED", "2A3AEF1AB9054D8B", "67DC1D133E~
## $ rideable_type      <chr> "docked_bike", "electric_bike", "electric_bike", "e~
## $ started_at         <dttm> 2020-08-20 18:08:14, 2020-08-27 18:46:04, 2020-08--~
## $ ended_at           <dttm> 2020-08-20 18:17:51, 2020-08-27 19:54:51, 2020-08--~
## $ start_station_name <chr> "Lake Shore Dr & Diversey Pkwy", "Michigan Ave & 14~
## $ start_station_id   <dbl> 329, 168, 195, 81, 658, 658, 196, 67, 153, 177, 313~
## $ end_station_name   <chr> "Clark St & Lincoln Ave", "Michigan Ave & 14th St",~
## $ end_station_id     <dbl> 141, 168, 44, 47, 658, 658, 49, 229, 225, 305, 296,~
## $ start_lat          <dbl> 41.93259, 41.86438, 41.88464, 41.88409, 41.90299, 4~
## $ start_lng          <dbl> -87.63643, -87.62368, -87.61955, -87.62964, -87.683~
## $ end_lat            <dbl> 41.91569, 41.86422, 41.88497, 41.88958, 41.90300, 4~
## $ end_lng            <dbl> -87.63460, -87.62344, -87.62757, -87.62754, -87.683~
## $ member_casual      <chr> "member", "casual", "casual", "casual", "casual", "~
```

```
glimpse(sep_20)
```

```
## Rows: 532,958
## Columns: 13
## $ ride_id            <chr> "2B22BD5F95FB2629", "A7FB70B4AFC6CAF2", "86057FA01B~
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", ~
## $ started_at         <dttm> 2020-09-17 14:27:11, 2020-09-17 15:07:31, 2020-09--~
## $ ended_at           <dttm> 2020-09-17 14:44:24, 2020-09-17 15:07:45, 2020-09--~
## $ start_station_name <chr> "Michigan Ave & Lake St", "W Oakdale Ave & N Broadw~
## $ start_station_id   <dbl> 52, NA, NA, 246, 24, 94, 291, NA, NA, NA, 273, 145,~
## $ end_station_name   <chr> "Green St & Randolph St", "W Oakdale Ave & N Broadw~
## $ end_station_id     <dbl> 112, NA, NA, 249, 24, NA, 256, NA, NA, NA, 273, NA,~
## $ start_lat          <dbl> 41.88669, 41.94000, 41.94000, 41.95606, 41.89186, 4~
## $ start_lng          <dbl> -87.62356, -87.64000, -87.64000, -87.66892, -87.621~
## $ end_lat            <dbl> 41.88357, 41.94000, 41.94000, 41.96398, 41.89135, 4~
## $ end_lng            <dbl> -87.64873, -87.64000, -87.64000, -87.63822, -87.620~
## $ member_casual      <chr> "casual", "casual", "casual", "casual", "casual", "~
```

```
glimpse(oct_20)
```

```
## Rows: 388,653
## Columns: 13
## $ ride_id            <chr> "ACB6B40CF5B9044C", "DF450C72FD109C01", "B6396B54A1~
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", ~
## $ started_at         <dttm> 2020-10-31 19:39:43, 2020-10-31 23:50:08, 2020-10-~
## $ ended_at           <dttm> 2020-10-31 19:57:12, 2020-11-01 00:04:16, 2020-10-~
## $ start_station_name <chr> "Lakeview Ave & Fullerton Pkwy", "Southport Ave & W~
## $ start_station_id   <dbl> 313, 227, 102, 165, 190, 359, 313, 125, NA, 174, 11~
## $ end_station_name   <chr> "Rush St & Hubbard St", "Kedzie Ave & Milwaukee Ave~
## $ end_station_id     <dbl> 125, 260, 423, 256, 185, 53, 125, 313, 199, 635, 30~
## $ start_lat          <dbl> 41.92610, 41.94817, 41.77346, 41.95085, 41.92886, 4~
## $ start_lng          <dbl> -87.63898, -87.66391, -87.58537, -87.65924, -87.663~
## $ end_lat            <dbl> 41.89035, 41.92953, 41.79145, 41.95281, 41.91778, 4~
## $ end_lng            <dbl> -87.62607, -87.70782, -87.60005, -87.65010, -87.691~
## $ member_casual      <chr> "casual", "casual", "casual", "casual", "casual", "~
```

```
glimpse(nov_20)
```

```
## Rows: 259,716
## Columns: 13
## $ ride_id            <chr> "BD0A6FF6FFF9B921", "96A7A7A4BDE4F82D", "C61526D065~
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", ~
## $ started_at         <dttm> 2020-11-01 13:36:00, 2020-11-01 10:03:26, 2020-11-~
## $ ended_at           <dttm> 2020-11-01 13:45:40, 2020-11-01 10:14:45, 2020-11-~
## $ start_station_name <chr> "Dearborn St & Erie St", "Franklin St & Illinois St~
## $ start_station_id   <dbl> 110, 672, 76, 659, 2, 72, 76, NA, 58, 394, 623, NA,~
## $ end_station_name   <chr> "St. Clair St & Erie St", "Noble St & Milwaukee Ave~
## $ end_station_id     <dbl> 211, 29, 41, 185, 2, 76, 72, NA, 288, 273, 2, 506, ~
## $ start_lat          <dbl> 41.89418, 41.89096, 41.88098, 41.89550, 41.87650, 4~
## $ start_lng          <dbl> -87.62913, -87.63534, -87.61675, -87.68201, -87.620~
## $ end_lat            <dbl> 41.89443, 41.90067, 41.87205, 41.91774, 41.87645, 4~
## $ end_lng            <dbl> -87.62338, -87.66248, -87.62955, -87.69139, -87.620~
## $ member_casual      <chr> "casual", "casual", "casual", "casual", "casual", "~
```

```
glimpse(dec_20)
```

```
## Rows: 131,573
## Columns: 13
## $ ride_id            <chr> "70B6A9A437D4C30D", "158A465D4E74C54A", "5262016E0F~
## $ rideable_type      <chr> "classic_bike", "electric_bike", "electric_bike", "~
## $ started_at         <dttm> 2020-12-27 12:44:29, 2020-12-18 17:37:15, 2020-12-~
## $ ended_at           <dttm> 2020-12-27 12:55:06, 2020-12-18 17:44:19, 2020-12-~
## $ start_station_name <chr> "Aberdeen St & Jackson Blvd", NA, NA, NA, NA, NA, N~
## $ start_station_id   <chr> "13157", NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ end_station_name   <chr> "Desplaines St & Kinzie St", NA, NA, NA, NA, NA, NA~
## $ end_station_id     <chr> "TA1306000003", NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ start_lat          <dbl> 41.87773, 41.93000, 41.91000, 41.92000, 41.80000, 4~
## $ start_lng          <dbl> -87.65479, -87.70000, -87.69000, -87.70000, -87.590~
## $ end_lat            <dbl> 41.88872, 41.91000, 41.93000, 41.91000, 41.80000, 4~
## $ end_lng            <dbl> -87.64445, -87.70000, -87.70000, -87.70000, -87.590~
## $ member_casual      <chr> "member", "member", "member", "member", "member", "~
```

```
glimpse(jan_21)
```

```
## Rows: 96,834
## Columns: 13
## $ ride_id            <chr> "E19E6F1B8D4C42ED", "DC88F20C2C55F27F", "EC45C94683~
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", ~
## $ started_at         <dttm> 2021-01-23 16:14:19, 2021-01-27 18:43:08, 2021-01-~
## $ ended_at           <dttm> 2021-01-23 16:24:44, 2021-01-27 18:47:12, 2021-01-~
## $ start_station_name <chr> "California Ave & Cortez St", "California Ave & Cor~
## $ start_station_id   <chr> "17660", "17660", "17660", "17660", "17660", "17660~
## $ end_station_name   <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, "Wood St & Augu~
## $ end_station_id     <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, "657", "13258",~
## $ start_lat          <dbl> 41.90034, 41.90033, 41.90031, 41.90040, 41.90033, 4~
## $ start_lng          <dbl> -87.69674, -87.69671, -87.69664, -87.69666, -87.696~
## $ end_lat            <dbl> 41.89000, 41.90000, 41.90000, 41.92000, 41.90000, 4~
## $ end_lng            <dbl> -87.72000, -87.69000, -87.70000, -87.69000, -87.700~
## $ member_casual      <chr> "member", "member", "member", "member", "casual", "~
```

```
glimpse(feb_21)
```

```
## Rows: 49,622
## Columns: 13
## $ ride_id            <chr> "89E7AA6C29227EFF", "0FEFDE2603568365", "E6159D746B~
## $ rideable_type      <chr> "classic_bike", "classic_bike", "electric_bike", "c~
## $ started_at         <dttm> 2021-02-12 16:14:56, 2021-02-14 17:52:38, 2021-02-~
## $ ended_at           <dttm> 2021-02-12 16:21:43, 2021-02-14 18:12:09, 2021-02-~
## $ start_station_name <chr> "Glenwood Ave & Touhy Ave", "Glenwood Ave & Touhy A~
## $ start_station_id   <chr> "525", "525", "KA1503000012", "637", "13216", "1800~
## $ end_station_name   <chr> "Sheridan Rd & Columbia Ave", "Bosworth Ave & Howar~
## $ end_station_id     <chr> "660", "16806", "TA1305000029", "TA1305000034", "TA~
## $ start_lat          <dbl> 42.01270, 42.01270, 41.88579, 41.89563, 41.83473, 4~
## $ start_lng          <dbl> -87.66606, -87.66606, -87.63110, -87.67207, -87.625~
## $ end_lat            <dbl> 42.00458, 42.01954, 41.88487, 41.90312, 41.83816, 4~
## $ end_lng            <dbl> -87.66141, -87.66956, -87.62750, -87.67394, -87.645~
## $ member_casual      <chr> "member", "casual", "member", "member", "member", "~
```

```
glimpse(mar_21)
```

```
## Rows: 228,496
## Columns: 13
## $ ride_id            <chr> "CFA86D4455AA1030", "30D9DC61227D1AF3", "846D87A156~
## $ rideable_type      <chr> "classic_bike", "classic_bike", "classic_bike", "cl~
## $ started_at         <dttm> 2021-03-16 08:32:30, 2021-03-28 01:26:28, 2021-03-~
## $ ended_at           <dttm> 2021-03-16 08:36:34, 2021-03-28 01:36:55, 2021-03-~
## $ start_station_name <chr> "Humboldt Blvd & Armitage Ave", "Humboldt Blvd & Ar~
## $ start_station_id   <chr> "15651", "15651", "15443", "TA1308000021", "525", "~
## $ end_station_name   <chr> "Stave St & Armitage Ave", "Central Park Ave & Bloo~
## $ end_station_id     <chr> "13266", "18017", "TA1308000043", "13323", "E008", ~
## $ start_lat          <dbl> 41.91751, 41.91751, 41.84273, 41.96881, 42.01270, 4~
## $ start_lng          <dbl> -87.70181, -87.70181, -87.63549, -87.65766, -87.666~
## $ end_lat            <dbl> 41.91774, 41.91417, 41.83066, 41.95283, 42.05049, 4~
## $ end_lng            <dbl> -87.69139, -87.71676, -87.64717, -87.64999, -87.677~
## $ member_casual      <chr> "casual", "casual", "casual", "casual", "casual", "~
```

After observing the 12 data frames above the following columns needs to have data types aligned. 1.ride_id into character 2.rideable_type into character 3.start_station_id into character 4.end_station_id into character

```
apr_20<-mutate(apr_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
may_20<-mutate(may_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
jun_20<-mutate(jun_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
jul_20<-mutate(jul_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
aug_20<-mutate(aug_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
sep_20<-mutate(sep_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
oct_20<-mutate(oct_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
nov_20<-mutate(nov_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
dec_20<-mutate(dec_20,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
jan_21<-mutate(jan_21,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
feb_21<-mutate(feb_21,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
mar_21<-mutate(mar_21,ride_id=as.character(ride_id),rideable_type=as.character(rideable_type),start_sta
```

After aligning all the data types for every attribute, I am combining the 12 months data into one data frame as one_year_data

```
one_year_data<-bind_rows(apr_20,may_20,jun_20,jul_20,aug_20,sep_20,oct_20,nov_20,dec_20,jan_21,feb_21,ma
```

Since we will not be using the columns start_lat,start_lng,end_lat,end_lng the following columns are dropped from the data frame.

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

The following code chunks is to understand the data frame at hand. We will bee looking at the columns within the data frame, number of columns,dimensions, first few rows of data using head(), a list of all columns and their data types and finally a summary of the data frame.

```
colnames(one_year_data)
```

```
## [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(one_year_data)
```

```
## [1] 3489748
```

```
dim(one_year_data)
```

```
## [1] 3489748       9
```

```
head(one_year_data)
```

```
## # A tibble: 6 x 9
##   ride_id rideable_type started_at          ended_at            start_station_n~
##   <chr>   <chr>         <dttm>              <dttm>              <chr>
## 1 A847FA~ docked_bike   2020-04-26 17:45:14 2020-04-26 18:12:03 Eckhart Park
## 2 5405B8~ docked_bike   2020-04-17 17:08:54 2020-04-17 17:17:03 Drake Ave & Ful~
## 3 5DD24A~ docked_bike   2020-04-01 17:54:13 2020-04-01 18:08:36 McClurg Ct & Er~
## 4 2A59BB~ docked_bike   2020-04-07 12:50:19 2020-04-07 13:02:31 California Ave ~
## 5 27AD30~ docked_bike   2020-04-18 10:22:59 2020-04-18 11:15:54 Rush St & Hubba~
## 6 356216~ docked_bike   2020-04-30 17:55:47 2020-04-30 18:01:11 Mies van der Ro~
## # ... with 4 more variables: start_station_id <chr>, end_station_name <chr>,
## #   end_station_id <chr>, member_casual <chr>
```

```
tail(one_year_data)
```

```
## # A tibble: 6 x 9
##   ride_id rideable_type started_at          ended_at            start_station_n~
##   <chr>   <chr>         <dttm>              <dttm>              <chr>
## 1 081549~ electric_bike 2021-03-14 01:59:38 2021-03-14 03:13:09 Larrabee St & A~
## 2 9397BD~ docked_bike   2021-03-20 14:58:56 2021-03-20 17:22:47 Michigan Ave & ~
## 3 BBBEB8~ classic_bike  2021-03-02 11:35:10 2021-03-02 11:43:37 Kingsbury St & ~
## 4 637FF7~ classic_bike  2021-03-09 11:07:36 2021-03-09 11:49:11 Michigan Ave & ~
## 5 F8F43A~ classic_bike  2021-03-01 18:11:57 2021-03-01 18:18:37 Kingsbury St & ~
## 6 3AE64E~ electric_bike 2021-03-26 17:58:14 2021-03-26 18:06:43 <NA>
## # ... with 4 more variables: start_station_id <chr>, end_station_name <chr>,
## #   end_station_id <chr>, member_casual <chr>
```

```
str(one_year_data)
```

```
## tibble [3,489,748 x 9] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:3489748] "A847FADBBC638E45" "5405B80E996FF60D" "5DD24A79A4E006F4" "2A59
##  $ rideable_type     : chr [1:3489748] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:3489748], format: "2020-04-26 17:45:14" "2020-04-17 17:08:54" ...
##  $ ended_at          : POSIXct[1:3489748], format: "2020-04-26 18:12:03" "2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:3489748] "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie
##  $ start_station_id  : chr [1:3489748] "86" "503" "142" "216" ...
##  $ end_station_name  : chr [1:3489748] "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave &
##  $ end_station_id    : chr [1:3489748] "152" "499" "255" "657" ...
##  $ member_casual     : chr [1:3489748] "member" "member" "member" "member" ...
```

```
summary(one_year_data)
```

```
##    ride_id          rideable_type        started_at
##  Length:3489748     Length:3489748     Min.   :2020-04-01 00:00:30
##  Class :character   Class :character   1st Qu.:2020-07-14 19:38:28
##  Mode  :character   Mode  :character   Median :2020-08-29 14:50:36
##                                        Mean   :2020-09-10 01:21:45
##                                        3rd Qu.:2020-10-20 18:14:13
##                                        Max.   :2021-03-31 23:59:08
##     ended_at                  start_station_name start_station_id
##  Min.   :2020-04-01 00:10:45   Length:3489748     Length:3489748
##  1st Qu.:2020-07-14 20:13:07   Class :character   Class :character
##  Median :2020-08-29 15:21:13   Mode  :character   Mode  :character
##  Mean   :2020-09-10 01:46:31
##  3rd Qu.:2020-10-20 18:28:46
##  Max.   :2021-04-06 11:00:11
##  end_station_name   end_station_id     member_casual
##  Length:3489748     Length:3489748     Length:3489748
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
```

We are using the started_at column to obtain the start date which would at a later point in time help us visulize data with respect day, month , year and so on.

To achieve the same, I am creating three columns, date, month,day and year within the data frame

```
one_year_data$date <- as.Date(one_year_data$started_at)
one_year_data$month <- format(as.Date(one_year_data$date),"%m")
one_year_data$day <- format(as.Date(one_year_data$date),"%d")
one_year_data$year <- format(as.Date(one_year_data$date),"%Y")
```

The following chunk of code creates yet another attribute that stores the day of the week.

```
one_year_data$day_of_week <- format(as.Date(one_year_data$date),"%A")
```

We will create another column called ride_length that consist of the time frame between start and end time of a particular ride in seconds.

```
one_year_data$ride_length <- difftime(one_year_data$ended_at,one_year_data$started_at)
```

We are now trying to summarize all the columns within the dataframe and the respective datatypes.

```
str(one_year_data)
```

```
## tibble [3,489,748 x 15] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:3489748] "A847FADBBC638E45" "5405B80E996FF60D" "5DD24A79A4E006F4" "2A5
##  $ rideable_type     : chr [1:3489748] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:3489748], format: "2020-04-26 17:45:14" "2020-04-17 17:08:54" ...
##  $ ended_at          : POSIXct[1:3489748], format: "2020-04-26 18:12:03" "2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:3489748] "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie
##  $ start_station_id  : chr [1:3489748] "86" "503" "142" "216" ...
##  $ end_station_name  : chr [1:3489748] "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave &
##  $ end_station_id    : chr [1:3489748] "152" "499" "255" "657" ...
##  $ member_casual     : chr [1:3489748] "member" "member" "member" "member" ...
##  $ date              : Date[1:3489748], format: "2020-04-26" "2020-04-17" ...
##  $ month             : chr [1:3489748] "04" "04" "04" "04" ...
##  $ day               : chr [1:3489748] "26" "17" "01" "07" ...
##  $ year              : chr [1:3489748] "2020" "2020" "2020" "2020" ...
##  $ day_of_week       : chr [1:3489748] "Sunday" "Friday" "Wednesday" "Tuesday" ...
##  $ ride_length       : 'difftime' num [1:3489748] 1609 489 863 732 ...
##   ..- attr(*, "units")= chr "secs"
```

In the output above , we can see that the ride_length is not numeric data:

```
is.numeric(one_year_data$ride_length)
```

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

We need the ride_length to be of type numeric to be able to perform required calculations on them. Hence we perform the required as below:

```
one_year_data$ride_length<-as.numeric(as.character(one_year_data$ride_length))
is.numeric(one_year_data$ride_length)
```

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

In the given data, if the start_station name is headquarters and the ride_length is negative, we remove such data since this is the time the bike was taken to service or other reasons

The all_trips dataframe now consist of all the valid data that will help us obtain the required output.

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

```
## # A tibble: 3,356,684 x 15
##    ride_id          rideable_type started_at          ended_at           
##    <chr>            <chr>         <dttm>              <dttm>             
##  1 A847FADBBC638E45 docked_bike   2020-04-26 17:45:14 2020-04-26 18:12:03
##  2 5405B80E996FF60D docked_bike   2020-04-17 17:08:54 2020-04-17 17:17:03
##  3 5DD24A79A4E006F4 docked_bike   2020-04-01 17:54:13 2020-04-01 18:08:36
##  4 2A59BBDF5CDBA725 docked_bike   2020-04-07 12:50:19 2020-04-07 13:02:31
##  5 27AD306C119C6158 docked_bike   2020-04-18 10:22:59 2020-04-18 11:15:54
##  6 356216E875132F61 docked_bike   2020-04-30 17:55:47 2020-04-30 18:01:11
##  7 A2759CB06A81F2BC docked_bike   2020-04-02 14:47:19 2020-04-02 14:52:32
##  8 FC8BC2E2D54F35ED docked_bike   2020-04-07 12:22:20 2020-04-07 13:38:09
##  9 9EC5648678DE06E6 docked_bike   2020-04-15 10:30:11 2020-04-15 10:35:55
## 10 A8FFF89140C33017 docked_bike   2020-04-04 15:02:28 2020-04-04 15:19:47
## # ... with 3,356,674 more rows, and 11 more variables:
## #   start_station_name <chr>, start_station_id <chr>, end_station_name <chr>,
## #   end_station_id <chr>, member_casual <chr>, date <date>, month <chr>,
## #   day <chr>, year <chr>, day_of_week <chr>, ride_length <dbl>
```

The summary on ride_length column helps us to understand the variations within the ride_length column.

```
summary(all_trips$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       1     484     884    1704    1616 3523202
```

The output above shows the minimum ride_length is 1second and the maximum is 3523202seconds and so on.

The code below uses the aggregate function to calculate the mean of ride_lengths between the member and casual riders respectively

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual             2749.6943
## 2                  member              974.0844
```

The code below uses the aggregate function to calculate the median of the ride_lengths between the member and casual riders respectively.

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual                  1293
## 2                  member                   695
```

The code below uses the aggregate function to calculate the max of the ride_lengths between the member and casual riders respectively.

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

```
##   all_trips$member_casual all_trips$ride_length
## 1                  casual               3341033
## 2                  member               3523202
```

The code below uses the aggregate function to calculate the min of the ride_lengths between the member and casual riders respectively.

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

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

The code below uses the aggregate function to calculate the mean of ride_lengths between the member and casual riders respectively on each day of the week.

```
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             2617.4584
## 2                   member                Friday              955.2019
## 3                   casual                Monday             2756.8318
## 4                   member                Monday              927.2568
## 5                   casual              Saturday             2861.0661
## 6                   member              Saturday             1076.3603
## 7                   casual                Sunday             3094.6639
## 8                   member                Sunday             1103.5590
## 9                   casual              Thursday             2633.3411
## 10                  member              Thursday              918.4713
## 11                  casual               Tuesday             2480.7203
## 12                  member               Tuesday              914.1592
## 13                  casual             Wednesday             2471.6304
## 14                  member             Wednesday              923.9960
```

As we can see in the output above the days of the week is not ordered properly and hence we will order the days of the week as below:

```
all_trips$day_of_week<-ordered(all_trips$day_of_week,level=c("Sunday","Monday","Tuesday","Wednesday","T
```

One aggregating the ride_lengths based on rider types and days of week again , we will receive the days of the week mean ride_length values in order.

```
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             3094.6639
## 2                   member                Sunday             1103.5590
## 3                   casual                Monday             2756.8318
## 4                   member                Monday              927.2568
## 5                   casual               Tuesday             2480.7203
## 6                   member               Tuesday              914.1592
## 7                   casual             Wednesday             2471.6304
## 8                   member             Wednesday              923.9960
## 9                   casual              Thursday             2633.3411
## 10                  member              Thursday              918.4713
## 11                  casual                Friday             2617.4584
## 12                  member                Friday              955.2019
## 13                  casual              Saturday             2861.0661
## 14                  member              Saturday             1076.3603
```

The code chunk below will display the mean ride_length of casual and member riders on each weekday for all the rides within the data frame all_trips

```
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)
```

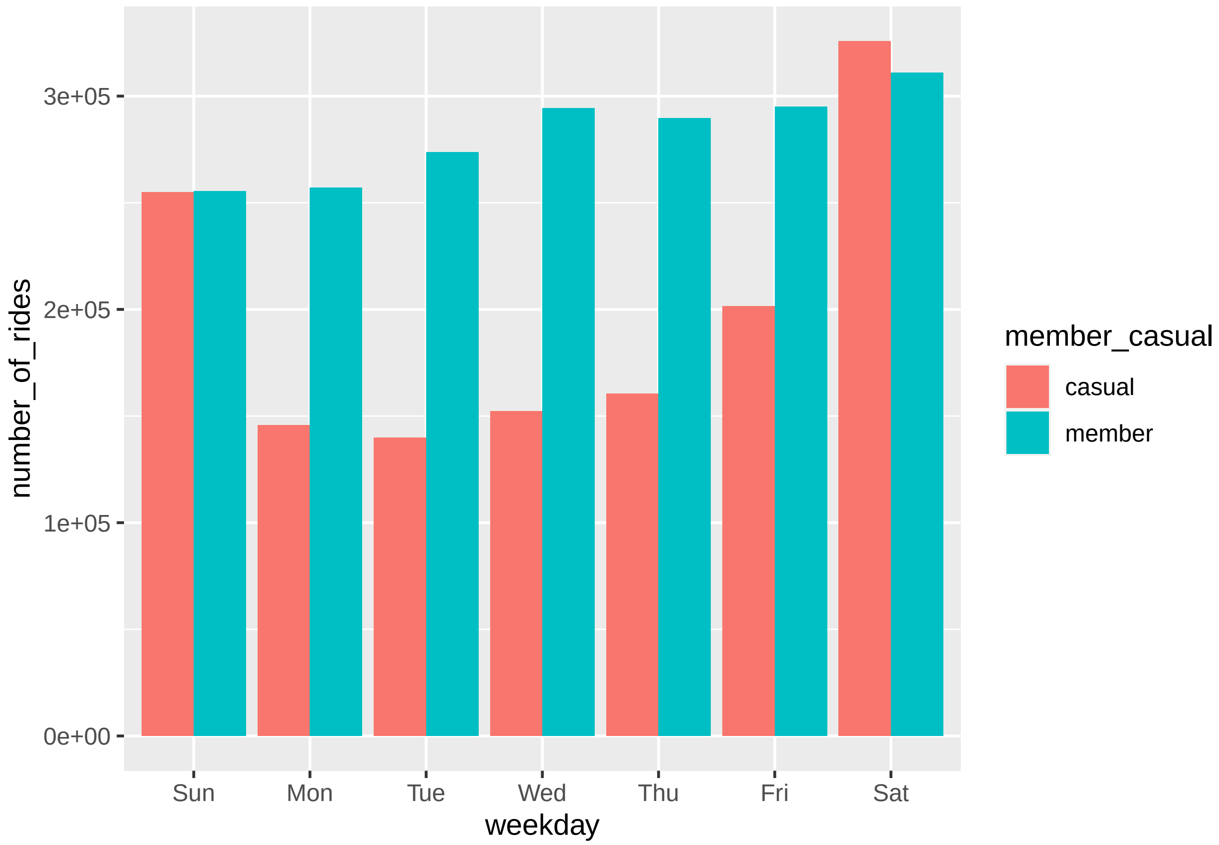
```
## `summarise()` has grouped output by 'member_casual'. You can override using the `.groups` argument.
```

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              254960            3095.
##  2 casual        Mon              145684            2757.
##  3 casual        Tue              139812            2481.
##  4 casual        Wed              152350            2472.
##  5 casual        Thu              160358            2633.
##  6 casual        Fri              201523            2617.
##  7 casual        Sat              325776            2861.
##  8 member        Sun              255355            1104.
##  9 member        Mon              257156             927.
## 10 member        Tue              273750             914.
## 11 member        Wed              294280             924.
## 12 member        Thu               289660            918.
## 13 member        Fri               295050            955.
## 14 member        Sat               310970           1076.
```

The summary created above is now visualized below using the ggplot2 package:

```
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")
```

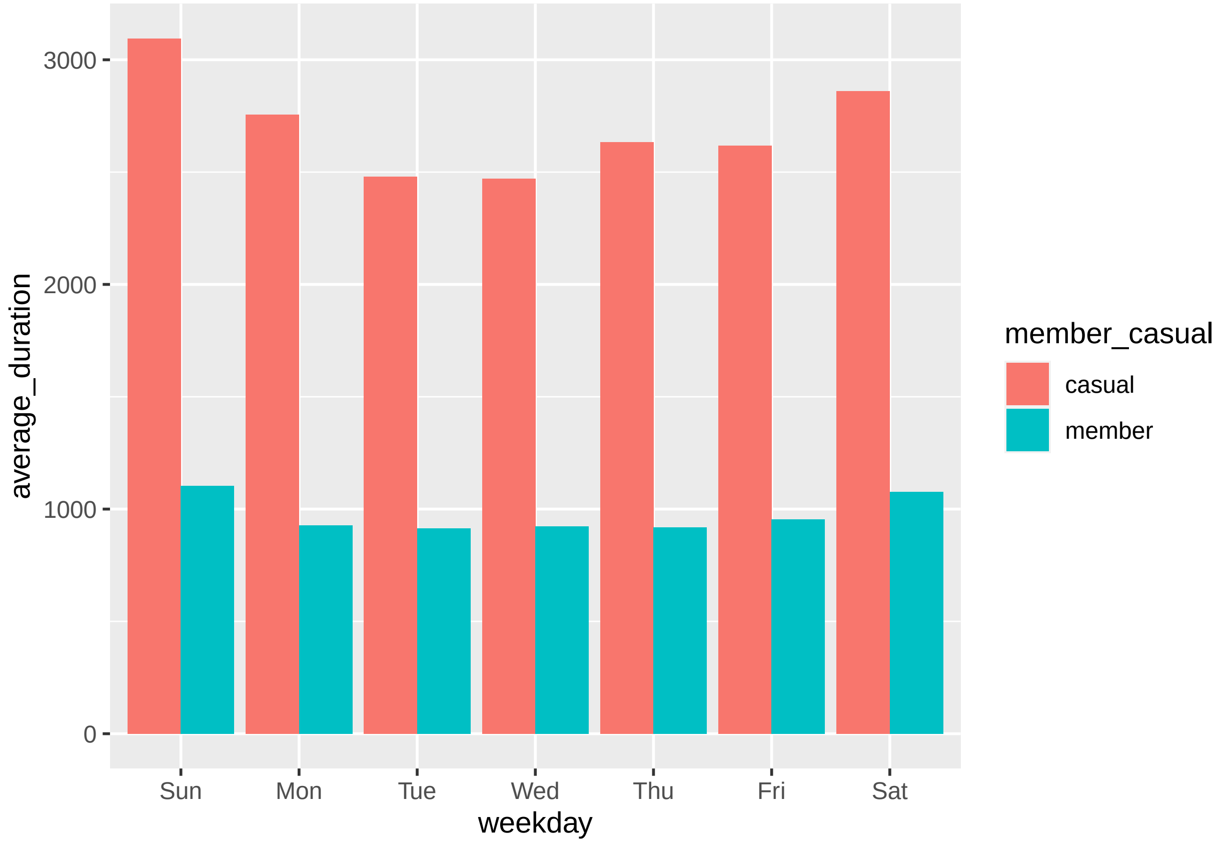
```
## ‘summarise()‘ has grouped output by ’member_casual’. You can override using the ‘.groups‘ argument.
```



The plot above visualizes the number of rides over the y-axis and days of week over the x-axis. We can understand from the visualization that the number of rides by the casual riders is in peak over the weekends while it reduces during the weekdays. The annual members however, use the bikes consistently throughout the week.

```
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")
```

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



The plot above shows that average ride duration on the y-axis and the weekdays over the x-axis. We can see the casual riders tend to travel for a longer duration while annual members use the bike for a smaller period of time, usually around 1000seconds.

Key Findings: Based on he case study performed, 1. The although the number of rides the casual riders take through out the week is lower , the duration of usage is higher. 2. The annual members have higher number of rides where the trip duration for these consistent rides are usually small duration.