Google Data Analytics Capstone Project: Cyclistic Bike-Share Case Study

Akshita Agrawal

2023-07-10

Introduction

About the Company

In 2016, Cyclistic launched a successful bike-share offering. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system anytime.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes, and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

Cyclistic's finance analysts have concluded that annual members are much more profitable than casual riders. Although the pricing flexibility helps Cyclistic attract more customers, Moreno believes that maximizing the number of annual members will be key to future growth. Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a very good chance to convert casual riders into members. She notes that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.

Moreno has set a clear goal: 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. Moreno and her team are interested in analyzing the Cyclistic historical bike trip data to identify trends.

The Scenario

The director of marketing believes the company's future success depends on maximizing the number of annual memberships. The marketing analyst team wants to understand **how casual riders and annual members use Cyclistic bikes differently**. From these insights, the team will design a new marketing strategy to convert casual riders into annual members.

Assumptions

I have assumed that the data provided in the source is the only available data on bike_share user patterns. I have also assumed that the future trends are going to be based on the historical trends.

Ask

Guiding questions

- **What is the problem you are trying to solve?**

The main objective is to determine a way to build a profile for annual members and identify the best marketing strategies to turn casual bike riders into members.

- **How can your insights drive business decisions?**

The insights will help the marketing team in increasing the members.

Key tasks

- Identify the business task
- Consider key stakeholders

Deliverable A clear statement of the business task: **Find insights in the usage patterns of the riders to design a marketing campaign for membership conversion.**

Tools: Analyzed and cleaned the monthly data separately in Excel, then used R to analyze the data as a whole.

Data Source: Trip_Data

Data Set: Trip Data January2022-December2022

Key Stakeholders *Lily Moreno* The Cyclistic marketing analytics team *Casual Riders* Members

Prepare

Guiding questions

- **Where is your data located?**

The data is located in a kaggle dataset.

- **How is the data organized?**

There is separate csv. file for each month.

- **Are there issues with bias or credibility in this data? Does your data ROCCC?**

The population of the dataset is it's own clients as bike riders thus, have full credibility.It is ROCCC because it's reliable, original, comprehensive, current and cited.

- **How are you addressing licensing, privacy, security, and accessibility?**

The company has their own licence over the dataset andthe dataset doesn't have any personal information about the riders.It is from an open source.

- **How did you verify the data's integrity?**

All the files have consistent columns and each column has the correct type of data.

- **How does it help you answer your question?**

Relationship between the type of riders and their patterns could let us identify insights for our marketing campaign. * **Are there any problems with the data?** Many columns in the data have missing information. As riders' personal identifiable information is hidden, it is not possible to connect pass purchases to credit cards numbers to determine if casual riders live in the Cyclistic service area or if they have purchased multiple single passes.

Key Tasks

- Download the data and save it with correct naming convention.
- Organize the data
- Determine it's credibility

Deliverable

- A description of all data sources used The data source consists of 12 CSV files. There is one file for each month. The period starts at January 2020 and runs until December 2020.

Process & Analyse

Guiding questions

- **What tools are chosen and why?** I started by compiling and cleaning data in Excel then loaded that data to R to manipulate it further as it will allow indepth analysis.
- **Have you ensured the data's integrity?** I examined the columns to check the consistency after manipulation.
- **What steps have you taken to ensure that your data is clean?** The null values and duplicates were removed, the time and dates were formatted.
- **How can you verify that your data is clean and ready to analyze?** The steps have been shown below.
- **Have you documented your cleaning process so you can review and share those results?** The cleaning process has been documented throughout.

Key Tasks

- Check the data for errors
- Choose tools
- Transform the data
- Document the cleaning process

Deliverables Documentation:

1. To start, I downloaded the data from the source and turning the .csv files into excel spreadsheets. The recent data included: June 2022 July 2022 August 2022 September 2022 October 2022 November 2022 December 2022 January 2023 February 2023 March 2023 April 2023 May 2023
2. I removed duplicates using the excel feature.
3. Then removed all rows with any null values using the filter function.
4. Then we move to R where I started by loading the relevant packages:

```
#loading packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(skimr)
library(janitor)
```

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

```
library(ggplot2)
library(readxl)
library(tinytex)
```

5. Then I imported the individual sheets.

```
##Importing the CSV files
tripdata1 <- read_excel("tripdata_clean.xlsx",
                        sheet = "June2022")
str(tripdata1)
```

```
## tibble [620,350 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:620350] 6 6 6 6 6 6 6 6 6 6 ...
##  $ ride_id           : chr [1:620350] "B12AD6565494C368" "BAD4CB075003A605" "76DAD9FC95774B53" "47DE
##  $ rideable_type     : chr [1:620350] "classic_bike" "electric_bike" "electric_bike" "electric_bike"
##  $ started_at        : POSIXct[1:620350], format: "2022-06-09 22:28:32" "2022-06-19 17:08:23" ...
##  $ ended_at          : POSIXct[1:620350], format: "2022-06-09 22:52:17" "2022-06-19 17:08:25" ...
##  $ start_station_name: chr [1:620350] "California Ave & Milwaukee Ave" "California Ave & Milwaukee Av
##  $ start_station_id  : chr [1:620350] "084" "084" "222" "637" ...
##  $ end_station_name  : chr [1:620350] "California Ave & Milwaukee Ave" "California Ave & Milwaukee Av
##  $ end_station_id    : chr [1:620350] "084" "084" "222" "256" ...
##  $ start_lat         : num [1:620350] 41.9 41.9 41.7 41.9 41.9 ...
##  $ start_lng         : num [1:620350] -87.7 -87.7 -87.5 -87.7 -87.7 ...
##  $ end_lat           : num [1:620350] 41.9 41.9 41.7 41.9 41.9 ...
##  $ end_lng           : num [1:620350] -87.7 -87.7 -87.5 -87.7 -87.7 ...
##  $ member_casual     : chr [1:620350] "casual" "casual" "casual" "casual" ...
```

```
tripdata2 <- read_excel("tripdata_clean.xlsx",
                        sheet = "July2022")
str(tripdata2)
```

```
## tibble [642,680 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:642680] 7 7 7 7 7 7 7 7 7 7 ...
##  $ ride_id           : chr [1:642680] "954144C2F67B1932" "292E027607D218B6" "57765852588AD6E0" "B5B6E
##  $ rideable_type     : chr [1:642680] "classic_bike" "classic_bike" "classic_bike" "classic_bike" ..
##  $ started_at        : POSIXct[1:642680], format: "2022-07-05 08:12:47" "2022-07-26 12:53:38" ...
##  $ ended_at          : POSIXct[1:642680], format: "2022-07-05 08:24:32" "2022-07-26 12:55:31" ...
##  $ start_station_name: chr [1:642680] "Ashland Ave & Blackhawk St" "Buckingham Fountain (Temp)" "Buck
##  $ start_station_id  : chr [1:642680] "224" "541" "541" "541" ...
##  $ end_station_name  : chr [1:642680] "Kingsbury St & Kinzie St" "Michigan Ave & 8th St" "Michigan Av
##  $ end_station_id    : chr [1:642680] "043" "623" "623" "164" ...
##  $ start_lat         : num [1:642680] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:642680] -87.7 -87.6 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num [1:642680] 41.9 41.9 41.9 41.8 41.9 ...
##  $ end_lng           : num [1:642680] -87.6 -87.6 -87.6 -87.6 -87.7 ...
##  $ member_casual     : chr [1:642680] "member" "casual" "casual" "casual" ...
```

```
tripdata3 <- read_excel("tripdata_clean.xlsx",
                        sheet = "August2022")
str(tripdata3)
```

```
## tibble [605,325 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:605325] 8 8 8 8 8 8 8 8 8 8 ...
##  $ ride_id           : chr [1:605325] "241C440C74CB31BB" "53A7590B28ED25E2" "C34EE790A58C0434" "4925
##  $ rideable_type     : chr [1:605325] "classic_bike" "classic_bike" "classic_bike" "electric_bike" .
##  $ started_at        : POSIXct[1:605325], format: "2022-08-05 16:13:36" "2022-08-11 23:30:11" ...
##  $ ended_at          : POSIXct[1:605325], format: "2022-08-05 16:22:40" "2022-08-11 23:30:56" ...
##  $ start_station_name: chr [1:605325] "DuSable Museum" "California Ave & Milwaukee Ave" "California 
##  $ start_station_id  : chr [1:605325] "075" "084" "256" "637" ...
##  $ end_station_name  : chr [1:605325] "Cottage Grove Ave & 51st St" "California Ave & Milwaukee Ave"
##  $ end_station_id    : chr [1:605325] "067" "084" "256" "637" ...
##  $ start_lat         : num [1:605325] 41.8 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:605325] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:605325] 41.8 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:605325] -87.6 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:605325] "casual" "casual" "casual" "casual" ...
```

```r
tripdata4 <- read_excel("tripdata_clean.xlsx",
                        sheet = "September2022")
str(tripdata4)
```

```
## tibble [535,145 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:535145] 9 9 9 9 9 9 9 9 9 9 ...
##  $ ride_id           : chr [1:535145] "8C94D4680E7B1DBC" "7FD174E61EBA0B99" "B23F6CCA8AC89F52" "F715
##  $ rideable_type     : chr [1:535145] "electric_bike" "electric_bike" "electric_bike" "classic_bike"
##  $ started_at        : POSIXct[1:535145], format: "2022-09-10 22:52:54" "2022-09-10 22:58:27" ...
##  $ ended_at          : POSIXct[1:535145], format: "2022-09-10 22:53:43" "2022-09-10 22:59:20" ...
##  $ start_station_name: chr [1:535145] "California Ave & Milwaukee Ave" "California Ave & Milwaukee A
##  $ start_station_id  : chr [1:535145] "084" "084" "637" "222" ...
##  $ end_station_name  : chr [1:535145] "California Ave & Milwaukee Ave" "California Ave & Milwaukee A
##  $ end_station_id    : chr [1:535145] "084" "084" "637" "222" ...
##  $ start_lat         : num [1:535145] 41.9 41.9 41.9 41.7 41.9 ...
##  $ start_lng         : num [1:535145] -87.7 -87.7 -87.7 -87.5 -87.7 ...
##  $ end_lat           : num [1:535145] 41.9 41.9 41.9 41.7 41.9 ...
##  $ end_lng           : num [1:535145] -87.7 -87.7 -87.7 -87.5 -87.7 ...
##  $ member_casual     : chr [1:535145] "casual" "casual" "casual" "casual" ...
```

```r
tripdata5 <- read_excel("tripdata_clean.xlsx",
                        sheet = "October2022")
str(tripdata5)
```

```
## tibble [414,269 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:414269] 10 10 10 10 10 10 10 10 10 10 ...
##  $ ride_id           : chr [1:414269] "A50255C1E17942AB" "DB692A70BD2DD4E3" "3C02727AAF60F873" "47E65
##  $ rideable_type     : chr [1:414269] "classic_bike" "electric_bike" "electric_bike" "electric_bike"
##  $ started_at        : POSIXct[1:414269], format: "2022-10-14 17:13:30" "2022-10-01 16:29:26" ...
##  $ ended_at          : POSIXct[1:414269], format: "2022-10-14 17:19:39" "2022-10-01 16:49:06" ...
##  $ start_station_name: chr [1:414269] "Noble St & Milwaukee Ave" "Damen Ave & Charleston St" "Hoyne A
##  $ start_station_id  : chr [1:414269] "290" "288" "655" "133" ...
##  $ end_station_name  : chr [1:414269] "Larrabee St & Division St" "Damen Ave & Cullerton St" "Wester
##  $ end_station_id    : chr [1:414269] "079" "089" "140" "620" ...
##  $ start_lat         : num [1:414269] 41.9 41.9 42 41.9 41.9 ...
##  $ start_lng         : num [1:414269] -87.7 -87.7 -87.7 -87.6 -87.6 ...
##  $ end_lat           : num [1:414269] 41.9 41.9 42 41.9 41.9 ...
##  $ end_lng           : num [1:414269] -87.6 -87.7 -87.7 -87.6 -87.6 ...
##  $ member_casual     : chr [1:414269] "member" "casual" "member" "member" ...
```

```r
tripdata6 <- read_excel("tripdata_clean.xlsx",
                        sheet = "November2022")
str(tripdata6)
```

```
## tibble [255,794 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:255794] 11 11 11 11 11 11 11 11 11 11 ...
##  $ ride_id           : chr [1:255794] "BCC66FC6FAB27CC7" "772AB67E902C180F" "585EAD07FDEC0152" "91C4
##  $ rideable_type     : chr [1:255794] "electric_bike" "classic_bike" "classic_bike" "classic_bike" .
##  $ started_at        : POSIXct[1:255794], format: "2022-11-10 06:21:55" "2022-11-04 07:31:55" ...
##  $ ended_at          : POSIXct[1:255794], format: "2022-11-10 06:31:27" "2022-11-04 07:46:25" ...
##  $ start_station_name: chr [1:255794] "Canal St & Adams St" "Canal St & Adams St" "Indiana Ave & Roos
##  $ start_station_id  : chr [1:255794] "011" "011" "005" "005" ...
##  $ end_station_name  : chr [1:255794] "St. Clair St & Erie St" "St. Clair St & Erie St" "St. Clair S
##  $ end_station_id    : chr [1:255794] "016" "016" "016" "016" ...
##  $ start_lat         : num [1:255794] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:255794] -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num [1:255794] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:255794] -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ member_casual     : chr [1:255794] "member" "member" "member" "member" ...
```

```
tripdata7 <- read_excel("tripdata_clean.xlsx",
                        sheet = "December2022")
str(tripdata7)
```

```
## tibble [135,403 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:135403] 12 12 12 12 12 12 12 12 12 12 ...
##  $ ride_id           : chr [1:135403] "65DBD2F447EC51C2" "0C201AA7EA0EA1AD" "E0B148CCB358A49D" "54C5
##  $ rideable_type     : chr [1:135403] "electric_bike" "classic_bike" "electric_bike" "classic_bike"
##  $ started_at        : POSIXct[1:135403], format: "2022-12-05 10:47:18" "2022-12-18 06:42:33" ...
##  $ ended_at          : POSIXct[1:135403], format: "2022-12-05 10:56:34" "2022-12-18 07:08:44" ...
##  $ start_station_name: chr [1:135403] "Clifton Ave & Armitage Ave" "Broadway & Belmont Ave" "Sangamo
##  $ start_station_id  : chr [1:135403] "163" "277" "015" "038" ...
##  $ end_station_name  : chr [1:135403] "Sedgwick St & Webster Ave" "Sedgwick St & Webster Ave" "St. Cl
##  $ end_station_id    : chr [1:135403] "191" "191" "016" "134" ...
##  $ start_lat         : num [1:135403] 41.9 41.9 41.9 41.8 41.9 ...
##  $ start_lng         : num [1:135403] -87.7 -87.6 -87.7 -87.6 -87.7 ...
##  $ end_lat           : num [1:135403] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:135403] -87.6 -87.6 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:135403] "member" "casual" "member" "member" ...
```

```
tripdata8 <- read_excel("tripdata_clean.xlsx",
                        sheet = "January2023")
str(tripdata8)
```

```
## tibble [148,284 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:148284] 1 1 1 1 1 1 1 1 1 1 ...
##  $ ride_id           : chr [1:148284] "F96D5A74A3E41399" "13CB7EB698CEDB88" "BD88A2E670661CE5" "C907
##  $ rideable_type     : chr [1:148284] "electric_bike" "classic_bike" "electric_bike" "classic_bike"
##  $ started_at        : POSIXct[1:148284], format: "2023-01-21 20:05:42" "2023-01-10 15:37:36" ...
##  $ ended_at          : POSIXct[1:148284], format: "2023-01-21 20:16:33" "2023-01-10 15:46:05" ...
##  $ start_station_name: chr [1:148284] "Lincoln Ave & Fullerton Ave" "Kimbark Ave & 53rd St" "Western
##  $ start_station_id  : chr [1:148284] "058" "037" "005" "037" ...
##  $ end_station_name  : chr [1:148284] "Hampden Ct & Diversey Ave" "Greenwood Ave & 47th St" "Valli P
##  $ end_station_id    : chr [1:148284] "480" "002" "599" "002" ...
##  $ start_lat         : num [1:148284] 41.9 41.8 42 41.8 41.8 ...
##  $ start_lng         : num [1:148284] -87.6 -87.6 -87.7 -87.6 -87.6 ...
##  $ end_lat           : num [1:148284] 41.9 41.8 42 41.8 41.8 ...
##  $ end_lng           : num [1:148284] -87.6 -87.6 -87.7 -87.6 -87.6 ...
##  $ member_casual     : chr [1:148284] "member" "member" "casual" "member" ...
```

```
tripdata9 <- read_excel("tripdata_clean.xlsx",
                        sheet = "Febuary2023")
str(tripdata9)
```

```
## tibble [190,445 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:190445] 2 2 2 2 2 2 2 2 2 2 ...
##  $ ride_id           : chr [1:190445] "CBCD0D7777F0E45F" "F3EC5FCE5FF39DE9" "E54C1F27FA9354FF" "3D56
##  $ rideable_type     : chr [1:190445] "classic_bike" "electric_bike" "classic_bike" "electric_bike"
##  $ started_at        : POSIXct[1:190445], format: "2023-02-14 11:59:42" "2023-02-15 13:53:48" ...
##  $ ended_at          : POSIXct[1:190445], format: "2023-02-14 12:13:38" "2023-02-15 13:59:08" ...
##  $ start_station_name: chr [1:190445] "Southport Ave & Clybourn Ave" "Clarendon Ave & Gordon Ter" "S
##  $ start_station_id  : chr [1:190445] "030" "379" "030" "030" ...
##  $ end_station_name  : chr [1:190445] "Clark St & Schiller St" "Sheridan Rd & Lawrence Ave" "Aberdee
##  $ end_station_id    : chr [1:190445] "024" "041" "156" "008" ...
##  $ start_lat         : num [1:190445] 41.9 42 41.9 41.9 41.8 ...
##  $ start_lng         : num [1:190445] -87.7 -87.6 -87.7 -87.7 -87.6 ...
##  $ end_lat           : num [1:190445] 41.9 42 41.9 41.9 41.8 ...
##  $ end_lng           : num [1:190445] -87.6 -87.7 -87.7 -87.6 -87.6 ...
##  $ member_casual     : chr [1:190445] "casual" "casual" "member" "member" ...
```

```
tripdata10 <- read_excel("tripdata_clean.xlsx",
                        sheet = "March2023")
str(tripdata10)
```

```
## tibble [200,447 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:200447] 3 3 3 3 3 3 3 3 3 3 ...
##  $ ride_id           : chr [1:200447] "6842AA605EE9FBB3" "FF7CF57CFE026D02" "6B61B916032CB6D6" "E55E
##  $ rideable_type     : chr [1:200447] "electric_bike" "classic_bike" "classic_bike" "electric_bike"
##  $ started_at        : POSIXct[1:200447], format: "2023-03-16 08:20:34" "2023-03-31 12:28:09" ...
##  $ ended_at          : POSIXct[1:200447], format: "2023-03-16 08:22:52" "2023-03-31 12:38:47" ...
##  $ start_station_name: chr [1:200447] "Clark St & Armitage Ave" "Orleans St & Chestnut St (NEXT Apts
##  $ start_station_id  : chr [1:200447] "146" "620" "003" "067" ...
##  $ end_station_name  : chr [1:200447] "Larrabee St & Webster Ave" "Clark St & Randolph St" "Sheffiel
##  $ end_station_id    : chr [1:200447] "193" "030" "154" "015" ...
##  $ start_lat         : num [1:200447] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:200447] -87.6 -87.6 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:200447] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:200447] -87.6 -87.6 -87.7 -87.7 -87.6 ...
##  $ member_casual     : chr [1:200447] "member" "member" "member" "member" ...
```

```
tripdata11 <- read_excel("tripdata_clean.xlsx",
                         sheet = "April2023")
str(tripdata11)
```

```
## tibble [324,197 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:324197] 4 4 4 4 4 4 4 4 4 4 ...
##  $ ride_id           : chr [1:324197] "5B6500E1E58655C0" "AA65D25D69AF771F" "079FB2C196414482" "5996
##  $ rideable_type     : chr [1:324197] "classic_bike" "classic_bike" "electric_bike" "classic_bike" .
##  $ started_at        : POSIXct[1:324197], format: "2023-04-10 17:34:35" "2023-04-12 12:29:46" ...
##  $ ended_at          : POSIXct[1:324197], format: "2023-04-10 18:02:36" "2023-04-12 12:54:00" ...
##  $ start_station_name: chr [1:324197] "Avenue O & 134th St" "Cottage Grove Ave & 51st St" "Morgan Av
##  $ start_station_id  : chr [1:324197] "214" "067" "002" "067" ...
##  $ end_station_name  : chr [1:324197] "Avenue O & 134th St" "Cottage Grove Ave & 51st St" "Morgan Av
##  $ end_station_id    : chr [1:324197] "214" "067" "002" "067" ...
##  $ start_lat         : num [1:324197] 41.7 41.8 41.9 41.8 41.9 ...
##  $ start_lng         : num [1:324197] -87.5 -87.6 -87.7 -87.6 -87.7 ...
##  $ end_lat           : num [1:324197] 41.7 41.8 41.9 41.8 41.9 ...
##  $ end_lng           : num [1:324197] -87.5 -87.6 -87.7 -87.6 -87.7 ...
##  $ member_casual     : chr [1:324197] "member" "member" "member" "member" ...
```

```
tripdata12 <- read_excel("tripdata_clean.xlsx",
                         sheet = "May2023")
str(tripdata12)
```

```
## tibble [463,227 x 14] (S3: tbl_df/tbl/data.frame)
##  $ month             : num [1:463227] 5 5 5 5 5 5 5 5 5 5 ...
##  $ ride_id           : chr [1:463227] "DDEB93BC2CE9AA77" "C07B70172FC92F59" "2BA66385DF8F815A" "31EF
##  $ rideable_type     : chr [1:463227] "classic_bike" "classic_bike" "classic_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:463227], format: "2023-05-10 16:47:01" "2023-05-09 18:30:34" ...
##  $ ended_at          : POSIXct[1:463227], format: "2023-05-10 16:59:52" "2023-05-09 18:39:28" ...
##  $ start_station_name: chr [1:463227] "Carpenter St & Huron St" "Southport Ave & Clark St" "Clinton
##  $ start_station_id  : chr [1:463227] "196" "047" "032" "300" ...
##  $ end_station_name  : chr [1:463227] "Damen Ave & Cortland St" "Southport Ave & Belmont Ave" "McClu
##  $ end_station_id    : chr [1:463227] "133" "229" "029" "431" ...
##  $ start_lat         : num [1:463227] 41.9 42 41.9 41.9 41.8 ...
##  $ start_lng         : num [1:463227] -87.7 -87.7 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num [1:463227] 41.9 41.9 41.9 41.9 41.8 ...
##  $ end_lng           : num [1:463227] -87.7 -87.7 -87.6 -87.6 -87.6 ...
##  $ member_casual     : chr [1:463227] "member" "member" "member" "casual" ...
```

6. Then I combined the individual files to one:

```
## Combining the individual files into one; code learnt from kaggle.com
tripdata_2023 <- rbind(tripdata1,tripdata2,tripdata3,tripdata4,tripdata5,tripdata6,
                       tripdata7,tripdata8,tripdata9,tripdata10,tripdata11,tripdata12)

## Then I checked the completeness of the data:
rowtotal <- sum(
  nrow(tripdata1),
  nrow(tripdata2),
  nrow(tripdata3),
  nrow(tripdata4),
  nrow(tripdata5),
  nrow(tripdata6),
  nrow(tripdata7),
  nrow(tripdata8),
  nrow(tripdata9),
  nrow(tripdata10),
  nrow(tripdata11),
  nrow(tripdata12))
print(rowtotal)
```

```
## [1] 4535566
```

```
print(nrow(tripdata_2023))
```

```
## [1] 4535566
```

7. Examin and correct the data:

```r
#examining the combined dataset
head(tripdata_2023)
```

```
## # A tibble: 6 x 14
##   month ride_id          rideable_type started_at          ended_at
##   <dbl> <chr>            <chr>         <dttm>              <dttm>
## 1     6 B12AD6565494C368 classic_bike  2022-06-09 22:28:32 2022-06-09 22:52:17
## 2     6 BAD4CB075003A605 electric_bike 2022-06-19 17:08:23 2022-06-19 17:08:25
## 3     6 76DAD9FC95774B53 electric_bike 2022-06-26 23:59:44 2022-06-27 00:25:26
## 4     6 47DE68ACCA138C13 electric_bike 2022-06-27 11:40:53 2022-06-27 11:50:16
## 5     6 5D899636D3334ED5 classic_bike  2022-06-27 16:01:13 2022-06-27 16:35:56
## 6     6 7A163D957F8CF0DD classic_bike  2022-06-19 22:29:14 2022-06-19 22:29:57
## # i 9 more variables: start_station_name <chr>, start_station_id <chr>,
## #   end_station_name <chr>, end_station_id <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, member_casual <chr>
```

```r
#cleaning

tripdata_2023$date <- as.Date(tripdata_2023$started_at)
tripdata_2023$month <- format(as.Date(tripdata_2023$date), "%b")
tripdata_2023$day <- format(as.Date(tripdata_2023$date), "%d")
tripdata_2023$year <- format(as.Date(tripdata_2023$date), "%Y")
tripdata_2023$day_of_week <- format(as.Date(tripdata_2023$date), "%A")
head(tripdata_2023)
```

```
## # A tibble: 6 x 18
##   month ride_id          rideable_type started_at          ended_at
##   <chr> <chr>            <chr>         <dttm>              <dttm>
## 1 Jun   B12AD6565494C368 classic_bike  2022-06-09 22:28:32 2022-06-09 22:52:17
## 2 Jun   BAD4CB075003A605 electric_bike 2022-06-19 17:08:23 2022-06-19 17:08:25
## 3 Jun   76DAD9FC95774B53 electric_bike 2022-06-26 23:59:44 2022-06-27 00:25:26
## 4 Jun   47DE68ACCA138C13 electric_bike 2022-06-27 11:40:53 2022-06-27 11:50:16
## 5 Jun   5D899636D3334ED5 classic_bike  2022-06-27 16:01:13 2022-06-27 16:35:56
## 6 Jun   7A163D957F8CF0DD classic_bike  2022-06-19 22:29:14 2022-06-19 22:29:57
## # i 13 more variables: start_station_name <chr>, start_station_id <chr>,
## #   end_station_name <chr>, end_station_id <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, member_casual <chr>,
## #   date <date>, day <chr>, year <chr>, day_of_week <chr>
```

```r
#drop null values
tripdata_2023 <- drop_na(tripdata_2023)
#remove duplicates
tripdata_2023_no_duplicates <- tripdata_2023[!duplicated(tripdata_2023$ride_id), ]
print(paste("Removed", nrow(tripdata_2023) - nrow(tripdata_2023_no_duplicates), "duplicate rows"))
```

```
## [1] "Removed 0 duplicate rows"
```

8. Then I started the analysis, it was done with the help of this notebook. firstly I observed the trends of riders, casual vs the members

```r
#ridelength
tripdata_2023_v2 <- mutate(tripdata_2023_no_duplicates, ride_length = difftime(ended_at, started_at, uni
str(tripdata_2023_v2)
```

```
## tibble [4,494,681 x 19] (S3: tbl_df/tbl/data.frame)
##  $ month             : chr [1:4494681] "Jun" "Jun" "Jun" "Jun" ...
##  $ ride_id           : chr [1:4494681] "B12AD6565494C368" "BAD4CB075003A605" "76DAD9FC95774B53" "47D
##  $ rideable_type     : chr [1:4494681] "classic_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:4494681], format: "2022-06-09 22:28:32" "2022-06-19 17:08:23" ...
##  $ ended_at          : POSIXct[1:4494681], format: "2022-06-09 22:52:17" "2022-06-19 17:08:25" ...
##  $ start_station_name: chr [1:4494681] "California Ave & Milwaukee Ave" "California Ave & Milwaukee
##  $ start_station_id  : chr [1:4494681] "084" "084" "222" "637" ...
##  $ end_station_name  : chr [1:4494681] "California Ave & Milwaukee Ave" "California Ave & Milwaukee
##  $ end_station_id    : chr [1:4494681] "084" "084" "222" "256" ...
##  $ start_lat         : num [1:4494681] 41.9 41.9 41.7 41.9 41.9 ...
##  $ start_lng         : num [1:4494681] -87.7 -87.7 -87.5 -87.7 -87.7 ...
##  $ end_lat           : num [1:4494681] 41.9 41.9 41.7 41.9 41.9 ...
##  $ end_lng           : num [1:4494681] -87.7 -87.7 -87.5 -87.7 -87.7 ...
##  $ member_casual     : chr [1:4494681] "casual" "casual" "casual" "casual" ...
##  $ date              : Date[1:4494681], format: "2022-06-09" "2022-06-19" ...
##  $ day               : chr [1:4494681] "09" "19" "26" "27" ...
##  $ year              : chr [1:4494681] "2022" "2022" "2022" "2022" ...
##  $ day_of_week       : chr [1:4494681] "Thursday" "Sunday" "Sunday" "Monday" ...
##  $ ride_length       : 'difftime' num [1:4494681] 23.75 0.0333333333333333 25.7 9.38333333333333 ...
##   ..- attr(*, "units")= chr "mins"
```

```r
##amount of members vs casual riders
nrow(tripdata_2023_v2[tripdata_2023_v2$ride_length < 0,])
```

```
## [1] 75
```

```r
tripdata_2023_v3 <- tripdata_2023_v2[!tripdata_2023_v2$ride_length <0,]
glimpse(tripdata_2023_v3)
```

```
## Rows: 4,494,606
## Columns: 19
## $ month              <chr> "Jun", "Jun", "Jun", "Jun", "Jun", "Jun", "Jun", "J~
## $ ride_id            <chr> "B12AD6565494C368", "BAD4CB075003A605", "76DAD9FC95~
## $ rideable_type      <chr> "classic_bike", "electric_bike", "electric_bike", "~
## $ started_at         <dttm> 2022-06-09 22:28:32, 2022-06-19 17:08:23, 2022-06-~
## $ ended_at           <dttm> 2022-06-09 22:52:17, 2022-06-19 17:08:25, 2022-06-~
## $ start_station_name <chr> "California Ave & Milwaukee Ave", "California Ave &~
## $ start_station_id   <chr> "084", "084", "222", "637", "256", "084", "256", "0~
## $ end_station_name   <chr> "California Ave & Milwaukee Ave", "California Ave &~
## $ end_station_id     <chr> "084", "084", "222", "256", "256", "084", "084", "0~
## $ start_lat          <dbl> 41.92269, 41.92261, 41.70463, 41.89561, 41.90303, 4~
## $ start_lng          <dbl> -87.69715, -87.69715, -87.52841, -87.67210, -87.697~
## $ end_lat            <dbl> 41.92269, 41.92269, 41.70458, 41.90303, 41.90303, 4~
## $ end_lng            <dbl> -87.69715, -87.69715, -87.52823, -87.69747, -87.697~
## $ member_casual      <chr> "casual", "casual", "casual", "casual", "casual", "~
## $ date               <date> 2022-06-09, 2022-06-19, 2022-06-26, 2022-06-27, 20~
## $ day                <chr> "09", "19", "26", "27", "27", "19", "20", "20", "20~
## $ year                <chr> "2022", "2022", "2022", "2022", "2022", "2022", "20~
## $ day_of_week         <chr> "Thursday", "Sunday", "Sunday", "Monday", "Monday",~
## $ ride_length         <drtn> 23.75000000 mins, 0.03333333 mins, 25.70000000 min~
```

```
rider_type_total <- table(tripdata_2023_v3$member_casual)
head(rider_type_total)
```

```
##
##  casual  member
## 1747867 2746739
```

```
member_riders <- tripdata_2023_v3 %>%
  filter(member_casual == 'member') %>%
  summarise(count_riders = n())
head(member_riders)
```

```
## # A tibble: 1 x 1
##   count_riders
##          <int>
## 1      2746739
```

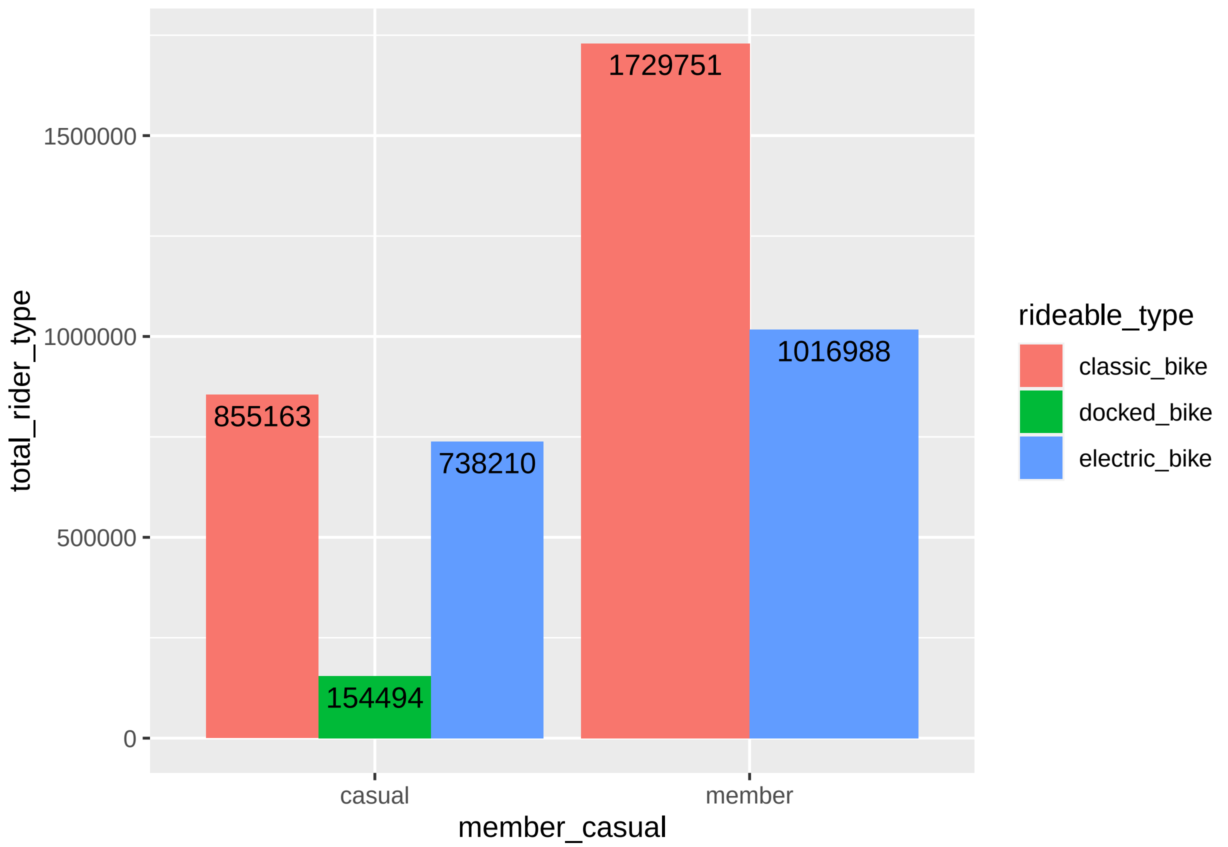
```
casual_riders <- tripdata_2023_v3 %>%
  filter(member_casual == 'casual') %>%
  summarise(count_riders = n())
head(casual_riders)
```

```
## # A tibble: 1 x 1
##   count_riders
##          <int>
## 1      1747867
```

```
diff_riders<- member_riders -casual_riders
```

```
#visualisation of usage
tripdata_2023_v3 %>%
  group_by(member_casual,rideable_type) %>%
  summarise(total_rider_type = n()) %>%
  ggplot(aes(x = member_casual, y = total_rider_type, fill = rideable_type)) +
  geom_col(position = "dodge") + geom_text(aes(label = total_rider_type),vjust = 1.5,position = positio
```

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



9. Then I tried to observe the monthly trends:

```
#statistics on lengths
trip_stats <- tripdata_2023_v3 %>%
  group_by(member_casual) %>%
  summarise(average_ride_length = mean(ride_length), standard_deviation = sd(ride_length), median_ride_l
head(trip_stats)
```

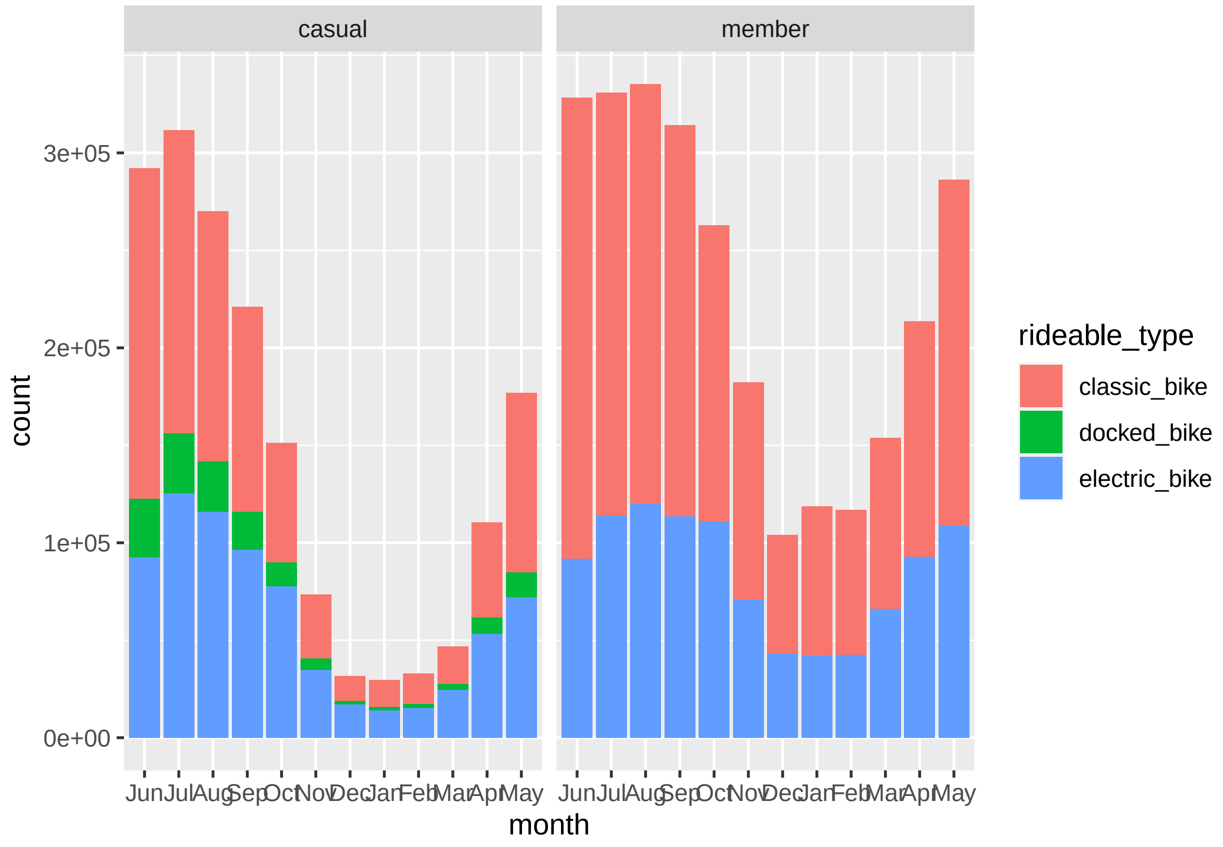
```
## # A tibble: 2 x 6
##   member_casual average_ride_length standard_deviation median_ride_length
##   <chr>         <drtn>                           <dbl> <drtn>
## 1 casual        22.71984 mins                     52.1 13.08333 mins
## 2 member        12.19365 mins                     19.5  8.75000 mins
## # i 2 more variables: min_ride_length <drtn>, max_ride_length <drtn>
```

```
#monthly trends
tripdata_2023_v3$month <- ordered(tripdata_2023_v3$month, levels = c("Jun","Jul","Aug","Sep","Oct","Nov

tripdata_2023_v3 %>%
  group_by(member_casual, month) %>%
  summarise(rider_type_total = n(), average_ride_length = mean(ride_length)) %>%
  arrange(member_casual, month)
```

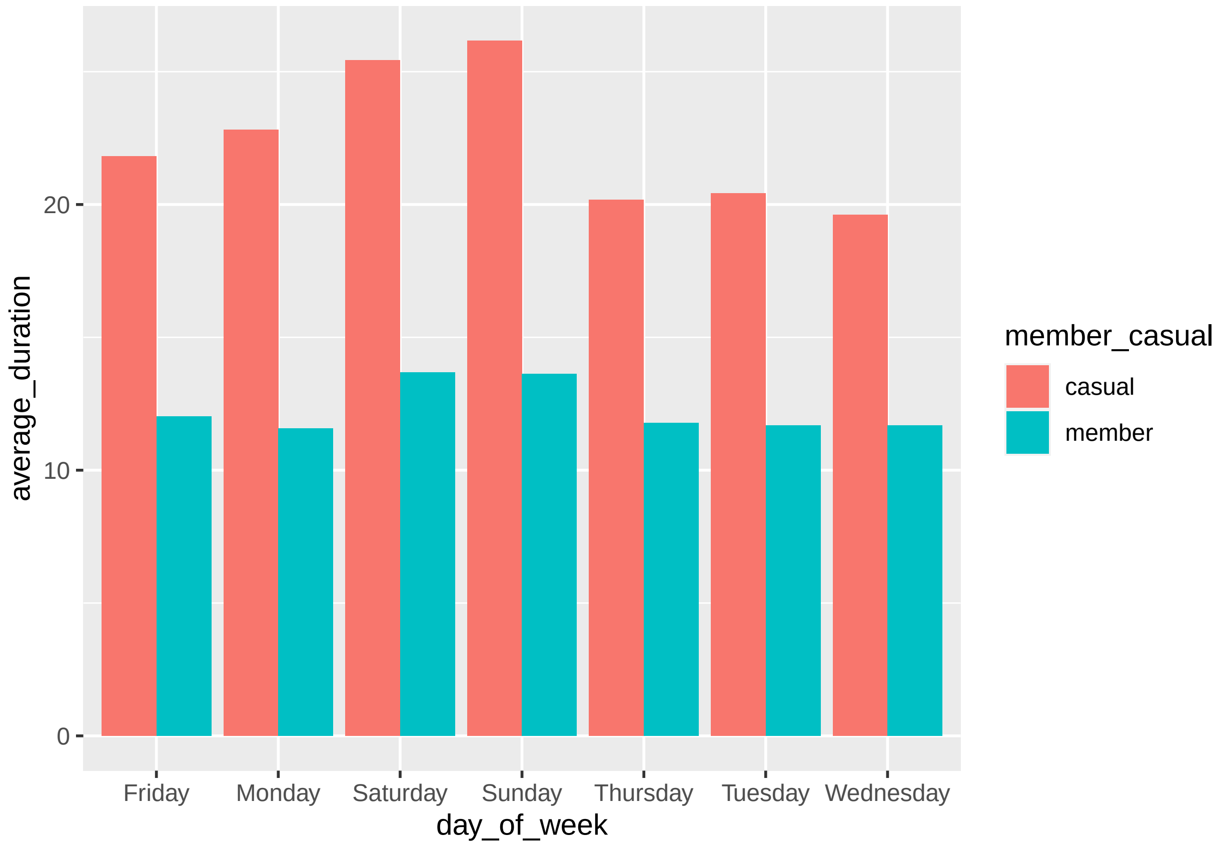
```
## ‘summarise()‘ has grouped output by ’member_casual’. You can override using the
## ‘.groups‘ argument.
## # A tibble: 24 x 4
## # Groups:   member_casual [2]
##    member_casual month rider_type_total average_ride_length
##    <chr>         <ord>            <int> <drtn>
##  1 casual        Jun             292067 25.01903 mins
##  2 casual        Jul             311670 25.09333 mins
##  3 casual        Aug             270089 23.28440 mins
##  4 casual        Sep             220913 21.80400 mins
##  5 casual        Oct             151324 20.46782 mins
##  6 casual        Nov              73536 17.24603 mins
##  7 casual        Dec              31505 14.84043 mins
##  8 casual        Jan              29621 14.87888 mins
##  9 casual        Feb              32776 17.67230 mins
## 10 casual        Mar              46792 16.71918 mins
## # i 14 more rows
```

```
##most popular month
ggplot(data = tripdata_2023_v3)+
  geom_bar(mapping = aes(x=month,fill = rideable_type,))+
  facet_wrap(~member_casual)
```



```
##weekly trends
tripdata_2023_v3 %>%
  group_by(member_casual, day_of_week) %>%
  summarise(number_of_rides = n(),average_duration = mean(ride_length)) %>%
  arrange(member_casual, day_of_week)  %>%
  ggplot(aes(x = day_of_week, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```

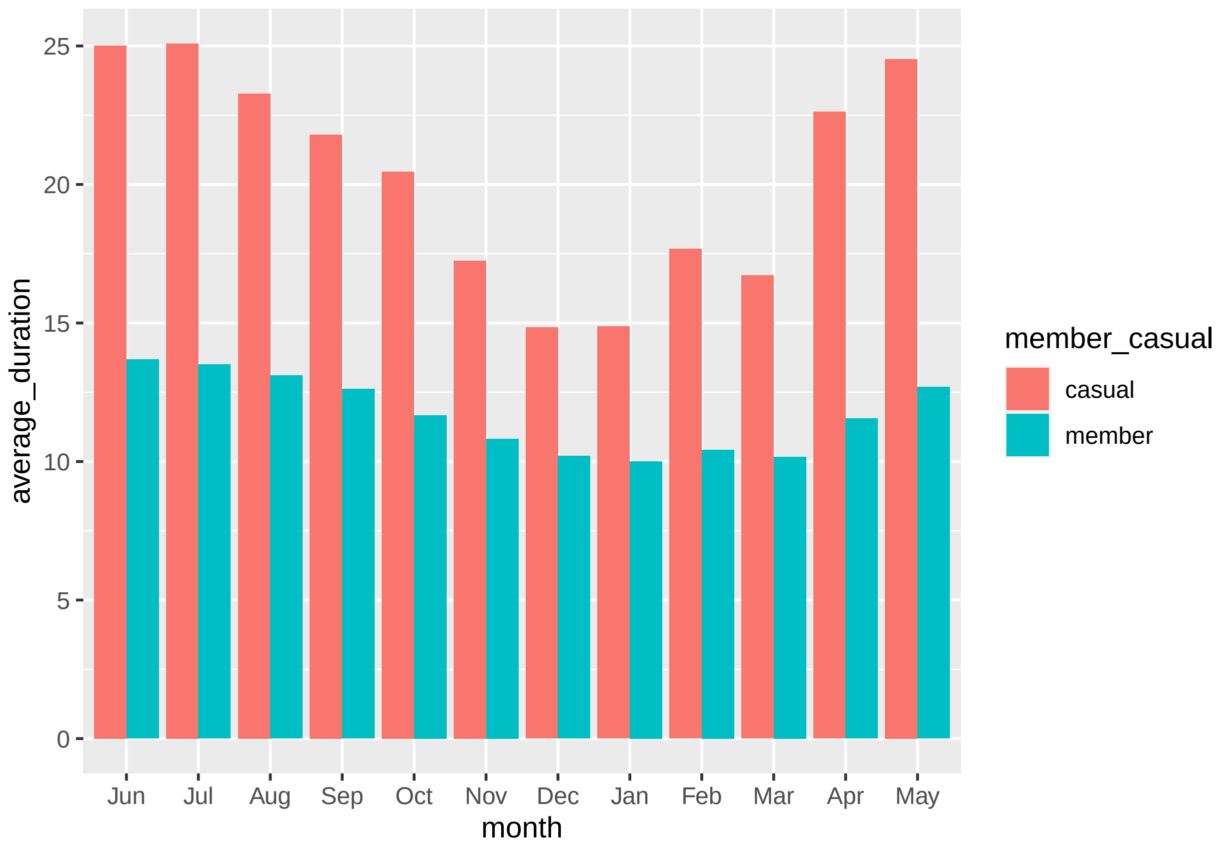
```
## ‘summarise()‘ has grouped output by ’member_casual’. You can override using the
## ‘.groups‘ argument.
## Don’t know how to automatically pick scale for object of type <difftime>.
## Defaulting to continuous.
```



```
#Durational trends

tripdata_2023_v3 %>%
  group_by(member_casual, month) %>%
  summarise(number_of_rides = n(),average_duration = mean(ride_length)) %>%
  arrange(member_casual, month)  %>%
  ggplot(aes(x = month, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```

```
## ‘summarise()‘ has grouped output by ’member_casual’. You can override using the
## ‘.groups‘ argument.
## Don’t know how to automatically pick scale for object of type <difftime>.
## Defaulting to continuous.
```



```
#popular stations
popular_start_stations_member <- tripdata_2023_v3 %>%
  filter(member_casual == 'member') %>%
  group_by(start_station_name) %>%
  summarise(number_of_starts = n()) %>%
  arrange(- number_of_starts)

head(popular_start_stations_member)
```

```
## # A tibble: 6 x 2
##   start_station_name           number_of_starts
##   <chr>                                   <int>
## 1 Kingsbury St & Kinzie St                23602
## 2 Clark St & Elm St                       21561
## 3 Clinton St & Washington Blvd            20890
## 4 Wells St & Concord Ln                   19935
## 5 Loomis St & Lexington St                19813
## 6 University Ave & 57th St                19131
```

```
popular_start_stations_casual <- tripdata_2023_v3 %>%
  filter(member_casual == 'casual') %>%
  group_by(start_station_name) %>%
  summarise(number_of_starts = n()) %>%
  arrange(- number_of_starts)

head(popular_start_stations_casual)
```

```
## # A tibble: 6 x 2
##   start_station_name                 number_of_starts
##   <chr>                                         <int>
## 1 Streeter Dr & Grand Ave                       52623
## 2 DuSable Lake Shore Dr & Monroe St             29958
## 3 Michigan Ave & Oak St                         23181
## 4 Millennium Park                               23034
## 5 DuSable Lake Shore Dr & North Blvd            21484
## 6 Shedd Aquarium                                18989
```

Insights:

1. With a strength of 2746739 member riders as compared to 1747867 casual riders, there are 998872 more member riders.
2. Classic bikes are the most popular among both groups.
3. The July 2022 saw the most casual riders while August 2022 saw the most member riders.
4. Sunday is the most prefered week by casual riders, while members have similar count on saturdays too.
5. Kingsbury St & Kinzie St is the most popular station among members.
6. Streeter Dr & Grand Ave is the most popular station among casual riders.
7. Casual members ride bikes for longer durations as compared to members.

Share

Guiding Questions

- **Were you able to answer the problem question?** Yes, I determined differences in riding behaviour and preferences between casual riders and annual members.
- **What story does your data tell?** The data tells us that there are a large number of casual riders who have a higher average of ride duration. This is a potential target for the digital marketing campaign.
- **How do your findings relate to your original question?** The data answered all the original questions.
- **Why would casual riders buy Cyclistic annual memberships?** Casual members would buy the annual membership as they on average use the bikes for longer than members and the longest individual ride was also by a casual member.
- **How can Cyclistic use digital media to influence casual riders to become members?** A digital media campaign with a focus on the benefits of a membership aimed at the casual rider who are using the bikes for long durations.

Key tasks

- Determine the best way to share findings
- Create effective data visualizations
- Present your findings
- Ensure your work is accessible

Deliverable Supporting visualizations and key findings

Act

Guiding Questions

- **What is your final conclusion based on your analysis?** There is an opportunity for Cyclistic to turn casual riders into annual members. There are casual riders who are using the bike sharing longer than the annual members and with a targeted marketing campaign at the popular stations, they can convert them to members.
- **How could your team and business apply your insights?** The team could now work on a digital marketing campaign targeting long use casual riders. The campaign would focus on the benefits of being a member over a casual rider for longer rides.
- **What next steps would you or your stakeholders take based on your findings?** I would recommend a more in-depth analysis on the long use casual riders however there is enough data to support a regional marketing campaign.
- **Is there additional data you could use to expand on your findings?** Additional data that would expand the findings would include: demographic data, climate data, financial data, and marketing campaign history with there ROI.

Key tasks

- Create your portfolio
- Add your case study
- Practice presenting your case study

Deliverable Top three recommendations based on the analysis.

1. Educate casual riders on the perks of memberships as a medium to afford their long rides and how it could be more economical.
2. The popular months could be targeted for the launch of campaigns at the popular stations.
3. Incentives could be provided for using cycles on the non popular months by both members as well as casual riders.

Conclusion

The bike_share is popular among both members as well as casual riders, analysing the usage pattern hints that the casual riders could find the membership useful. By using the popular locations and moths strategically, they could release offers and increase their membership base manifolds.I would like to thank the creators of the Google Data Analytics ProfessionalCertificate for making data analytics seem so easy. I would also like to thank the R community for helping me whenever I got stuck.