Case study: How Does a Bike-Share Navigate Speedy Success?

2022-05-29

Introduction

This is capstone project for Cyclistic bike-share analysis case study.

The director of marketing in Cyclistic believes the company's future success depends on maximizing the number of annual memberships. Therefore, as a data analyst working in marketing team, the business tasks are to analyze how casual riders and annual members use Cyclistic bikes differently and prodvide recommendations from data insights and data visualizations.

The process of data analysis follows the steps: **Ask**, **Prepare**, **Process**, **Analyze**, **Share**, and **Act**.

1. Ask

Bussiness Task

- Analyze how casual riders and annual members use Cyclistic bikes differently.
- Prodvide recommendations from data insights and data visualizations.

Skateholder

- Lily Moreno - The director of marketing
- Cyclistic marketing analytics team
- Cyclistic executive team

2. Prepare

The previous 12 months of Cyclistic trip data, which corresponding the period from May 2021 to April 2022, are used for working. The datasets are downloaded from Divvy. Each dataset contains the different types of bike available for rental, the date and time for bike rental and return, the started and ended station name and more.

3. Process

Load Libraries

```
library(tidyverse)
library(janitor)
library(readr)
library(tidyr)
library(dplyr)
library(lubridate)
library(data.table)
library(scales)
```

Load datasets

```
df1 <- read_csv("data/tripdata_202105.csv")
df2 <- read_csv("data/tripdata_202106.csv")
df3 <- read_csv("data/tripdata_202107.csv")
df4 <- read_csv("data/tripdata_202108.csv")
df5 <- read_csv("data/tripdata_202109.csv")
df6 <- read_csv("data/tripdata_202110.csv")
df7 <- read_csv("data/tripdata_202111.csv")
df8 <- read_csv("data/tripdata_202112.csv")
df9 <- read_csv("data/tripdata_202201.csv")
```

```
df10 <- read_csv("data/tripdata_202202.csv")
df11 <- read_csv("data/tripdata_202203.csv")
df12 <- read_csv("data/tripdata_202204.csv")
```

Combine all datasets into a single dataframe

```
tripdata <- bind_rows(df1,df2,df3,df4,df5,df6,df7,df8,df9,df10,df11,df12)
```

Check data structures and types

```
str(tripdata)
```

```
## spec_tbl_df [5,757,551 × 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:5757551] "C809ED75D6160B2A" "DD59FDCE0ACACAF3" "0AB8
3CB88C43EFC2" "7881AC6D39110C60" ...
##  $ rideable_type     : chr [1:5757551] "electric_bike" "electric_bike" "electric_b
ike" "electric_bike" ...
##  $ started_at        : POSIXct[1:5757551], format: "2021-05-30 11:58:15" "2021-05-
30 11:29:14" ...
##  $ ended_at          : POSIXct[1:5757551], format: "2021-05-30 12:10:39" "2021-05-
30 12:14:09" ...
##  $ start_station_name: chr [1:5757551] NA NA NA NA ...
##  $ start_station_id  : chr [1:5757551] NA NA NA NA ...
##  $ end_station_name  : chr [1:5757551] NA NA NA NA ...
##  $ end_station_id    : chr [1:5757551] NA NA NA NA ...
##  $ start_lat         : num [1:5757551] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:5757551] -87.6 -87.6 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:5757551] 41.9 41.8 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:5757551] -87.6 -87.6 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:5757551] "casual" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. 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()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Data cleaning and transformation

Remove NA in rows

```
tripdata <- drop_na(tripdata)
```

Remove duplicates on ride_id

```
tripdata <- distinct(tripdata, ride_id, .keep_all = TRUE)
```

Make sure the rows that the started time is not larger than ended time

```
tripdata <- subset(tripdata, started_at <= ended_at)
```

Set languages to en_us for showing day of week

```
Sys.setlocale("LC_TIME", "en_us")
```

```
## [1] "en_us"
```

Calculate the duration of ride in Minutes and determine the day of week of ride started

```
tripdata <- tripdata %>%
  mutate(ride_length = round(difftime(ended_at, started_at, units = "mins"), 2)) %>%
  mutate(day_of_week = weekdays(started_at))
```

Sort the dataframe with started_at in descending order

```
tripdata <- tripdata %>%
  arrange(desc(started_at))
```

Add the column for month and year

```
tripdata <- tripdata %>%
  dplyr::mutate(month = lubridate::month(started_at), year = lubridate::year(started_
at)) %>%
  unite(col = "month_year", c("month", "year"), sep = "-")
```

Remove columns of start_lat, start_lng, end_lat, end_lng that not used

```
tripdata <- tripdata %>%
  select(-c(start_lat:end_lng))
str(tripdata)
```

```
## tibble [4,615,636 × 12] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:4615636] "376DDF4CD6EB0F0D" "D50575DD96056AAF" "FCA7
BD10C05FC4F9" "C41246F2C4F65308" ...
##  $ rideable_type     : chr [1:4615636] "classic_bike" "electric_bike" "electric_bi
ke" "classic_bike" ...
##  $ started_at        : POSIXct[1:4615636], format: "2022-04-30 23:59:54" "2022-04-
30 23:59:52" ...
##  $ ended_at          : POSIXct[1:4615636], format: "2022-05-01 00:00:02" "2022-05-
01 00:09:50" ...
##  $ start_station_name: chr [1:4615636] "Federal St & Polk St" "Indiana Ave & Roose
velt Rd" "Greenview Ave & Diversey Pkwy" "Clark St & Armitage Ave" ...
##  $ start_station_id  : chr [1:4615636] "SL-008" "SL-005" "13294" "13146" ...
##  $ end_station_name  : chr [1:4615636] "Federal St & Polk St" "Indiana Ave & Roose
velt Rd" "Ashland Ave & Division St" "Clark St & Wellington Ave" ...
```

```
##  $ end_station_id    : chr [1:4615636] "SL-008" "SL-005" "13061" "TA1307000136" ..
.
##  $ member_casual     : chr [1:4615636] "member" "casual" "casual" "casual" ...
##  $ ride_length       : 'difftime' num [1:4615636] 0.13 9.97 19.08 21.58 ...
##   ..- attr(*, "units")= chr "mins"
##  $ day_of_week       : chr [1:4615636] "Saturday" "Saturday" "Saturday" "Saturday"
...
##  $ month_year        : chr [1:4615636] "4-2022" "4-2022" "4-2022" "4-2022" ...
```

Convert ride_length to numeric

```
tripdata$ride_length <- as.numeric(tripdata$ride_length)
is.numeric(tripdata$ride_length)
```

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

Check counts of customer types

```
table(tripdata$member_casual)
```

```
## 
##  casual  member 
## 2015771 2599865
```

Aggregate total ride duration by customer types

```
setNames(aggregate(tripdata$ride_length ~ tripdata$member_casual, tripdata, sum), c("
Customer Type", "Total Ride Duration"))
```

```
##   Customer Type Total Ride Duration
## 1        casual            63272314
## 2        member            33170320
```

4. Analyze

Statistical summary of ride_length for all trips

```
summary(tripdata$ride_length)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##     0.00     6.68    11.77    20.89    21.32 55944.15
```

Summarize ride_length by customer types

```
tripdata %>%
  group_by(member_casual) %>%
  summarise(min_ride_length = min(ride_length), max_ride_length = max(ride_length),
            median_ride_length = median(ride_length), average_ride_length = mean(ride
_length))
```

```
## # A tibble: 2 × 5
##   member_casual min_ride_length max_ride_length median_ride_length
##   <chr>                   <dbl>           <dbl>              <dbl>
## 1 casual                      0          55944.              16.3 
## 2 member                      0           1496.               9.33
## # … with 1 more variable: average_ride_length <dbl>
```

Total number of rides and average duration by customer type and day of week

and day of week

Fix the order for day of week variable to show same sequence in output table and visualization

```
tripdata$day_of_week <- ordered(tripdata$day_of_week, levels = c("Sunday", "Monday", 
"Tuesday", 
                                "Wednesday", "Thursday", "Friday", "Saturday"))
```

Calculate the average duration and total number of rides by day of week

```
tripdata_weekly <- tripdata %>%
  group_by(member_casual, day_of_week) %>%
  summarise(number_of_rides = n(), average_duration = mean(ride_length)) %>%
  arrange(member_casual, day_of_week)
```

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

```
head(tripdata_weekly)
```

```
## # A tibble: 6 × 4
## # Groups:   member_casual [1]
##   member_casual day_of_week number_of_rides average_duration
##   <chr>         <ord>                 <int>            <dbl>
## 1 casual        Sunday               393831             36.6
## 2 casual        Monday               225975             31.3
## 3 casual        Tuesday              206251             26.7
## 4 casual        Wednesday            218605             27.4
## 5 casual        Thursday             229681             27.9
## 6 casual        Friday               280154             29.4
```

Total number of rides and average duration by customer type and month

Fix the order for month_year variable to show same sequence in output table and visualization

```
tripdata$month_year <- ordered(tripdata$month_year, levels = c("5-2021", "6-2021", "7
-2021", "8-2021", 
                                                               "9-2021", "10-2021", "
11-2021", "12-2021", 
                                                               "1-2022", "2-2022", "3
-2022", "4-2022"))
```

Calculate the average duration and total number of rides by month

```
tripdata_monthly <- tripdata %>%
  group_by(member_casual, month_year) %>%
  summarise(number_of_rides = n(), average_duration = mean(ride_length)) %>%
  arrange(member_casual, month_year)
```

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

```
head(tripdata_monthly)
```

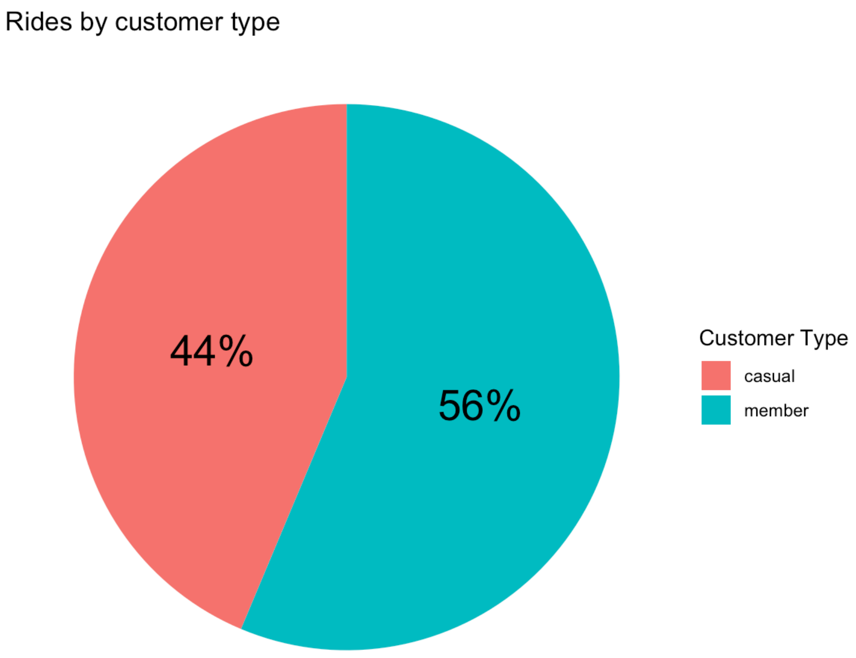
```
## # A tibble: 6 × 4
## # Groups:   member_casual [1]
##   member_casual month_year number_of_rides average_duration
##   <chr>         <ord>                <int>            <dbl>
## 1 casual        5-2021              216829             39.6
## 2 casual        6-2021              304189             38.5
## 3 casual        7-2021              369407             33.3
## 4 casual        8-2021              341469             28.6
## 5 casual        9-2021              292926             28.1
## 6 casual        10-2021             189117             26.3
```

5. Share

Visualizing

Rides by customer type

```
tripdata %>%
  group_by(member_casual) %>%
  summarise(number_of_rides = n()) %>%
  mutate(portion = number_of_rides / sum(number_of_rides)) %>%
  ggplot(aes(x ="", y=number_of_rides, fill=member_casual, label = scales::percent(po
rtion))) +
  geom_bar(stat = "identity", size = 1) +
  coord_polar("y", start = 0)+
  theme_void() +
  labs(title = "Rides by customer type", fill="Customer Type")+
  geom_text(position = position_stack(vjust = 0.5), size = 7.5)
```

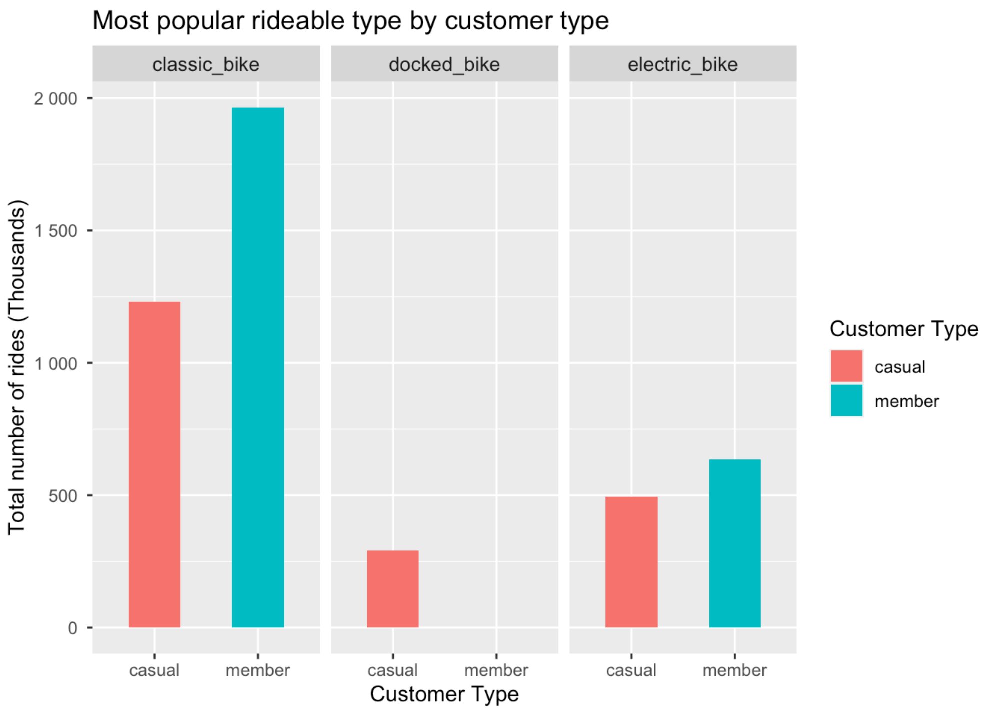


The pie chart shows that the annual members are 56% of the total users.

Most popular rideable type by customer type

```
tripdata %>%
  group_by(member_casual, rideable_type) %>%
  summarise(number_of_rides = n()) %>%
  ggplot(aes(x = member_casual, y = number_of_rides, fill=member_casual)) +
  geom_col(width = 0.5, position = position_dodge(0.5)) +
  labs(title = "Most popular rideable type by customer type", fill = "Customer Type")
+
  xlab("Customer Type") + ylab("Total number of rides (Thousands)") +
  scale_y_continuous(labels = unit_format(unit ="", scale = 1e-3)) +
  facet_wrap(~rideable_type)+
  theme(strip.text = element_text(size = 10))
```

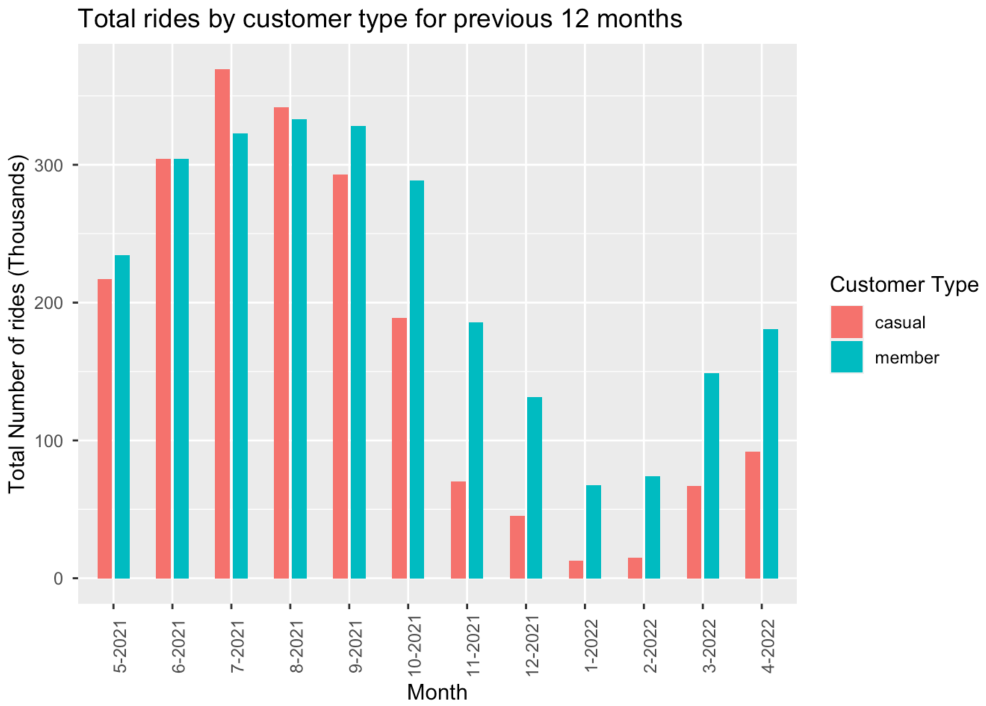
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```



According to the dataset, Cyclistic's bikes are classified as classic bike, docked bike and electric bike. The bar chart shows that the most popular bike is classic bike for annual members and casual users. However, there are no docked bikes rented by annual members after double check all the datasets.

Total number of rides by customer type for previous 12 months

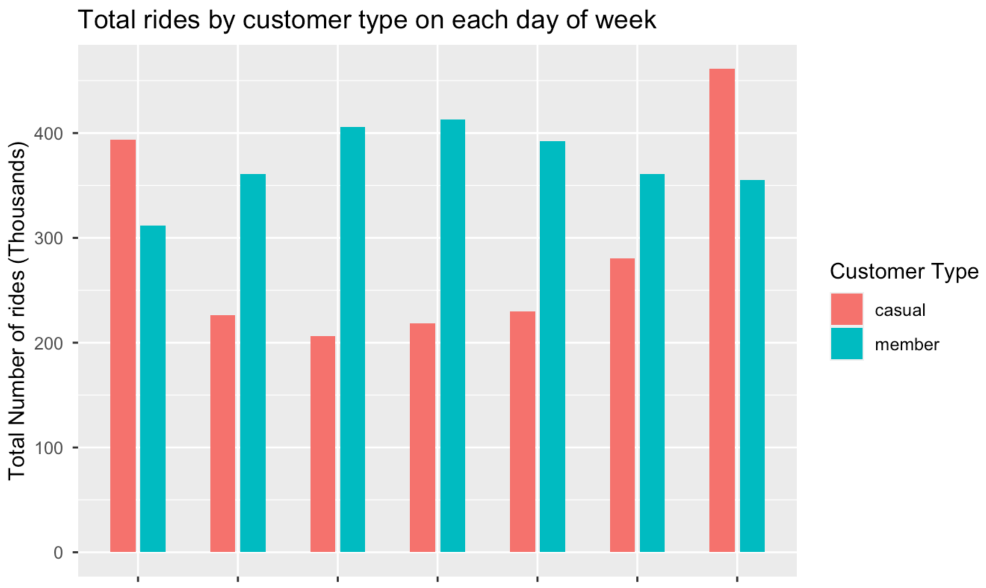
```
tripdata_monthly %>%
  ggplot(aes(x = month_year, y = number_of_rides, fill = member_casual)) +
           geom_col(width = 0.5, position = position_dodge(0.6))+
  labs(title = "Total rides by customer type for previous 12 months", fill = "Custome
r Type")+
  theme(axis.text.x = element_text(angle = 90)) +
  xlab("Month") + ylab("Total Number of rides (Thousands)") +
  scale_y_continuous(labels = unit_format(unit = "", scale = 1e-3))
```



From May 2021 to April 2022, the chart shows that the number of rides is the most in July 2021. The number of rides are gradually decreasing from summer to winter by annual member and casual users. For casual users, the number of rides is dropped significantly from summer to winter, especially the number of casual user is less than that of annual member over this period.

Total number of rides by customer type on each day of week

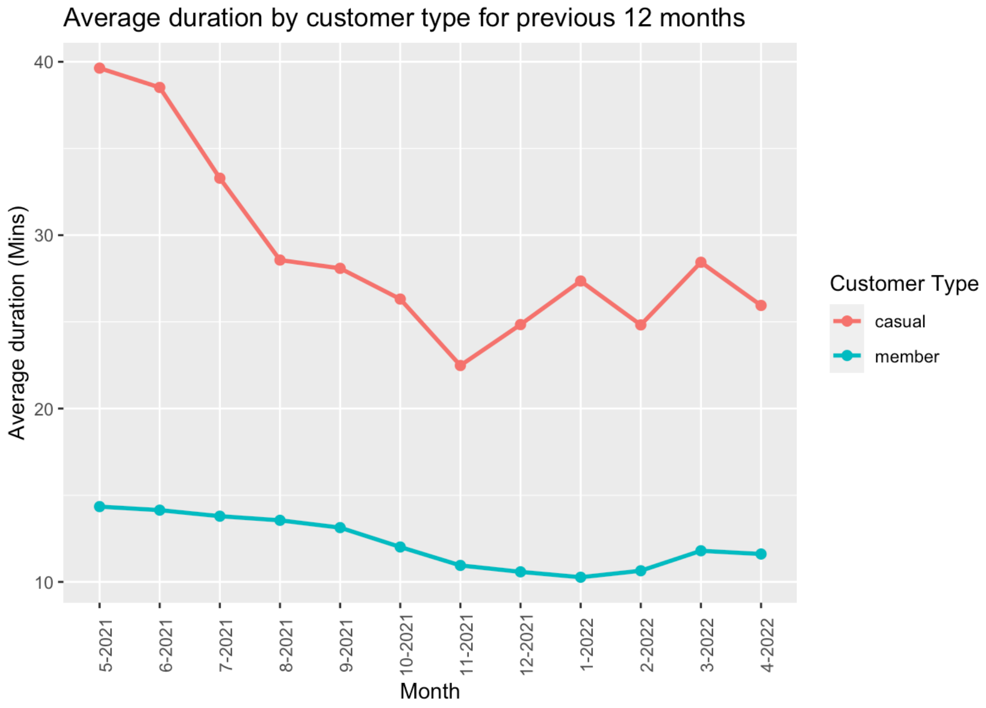
```
tripdata_weekly %>%
  ggplot(aes(x = day_of_week, y = number_of_rides, fill = member_casual)) +
  geom_col(width = 0.5, position = position_dodge(0.6))+
  labs(title = "Total rides by customer type on each day of week", fill = "Customer T
ype")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.25, hjust = 1)) +
  xlab("Day of week") + ylab("Total Number of rides (Thousands)") +
  scale_y_continuous(labels = unit_format(unit = "", scale = 1e-3))
```



From this column chart, Saturday is the busiest day as the increase in the number of rides by casual users. Over the weeks, there are no significant changes for the number of rides by annual member.

The average duration by customer type for previous 12 months

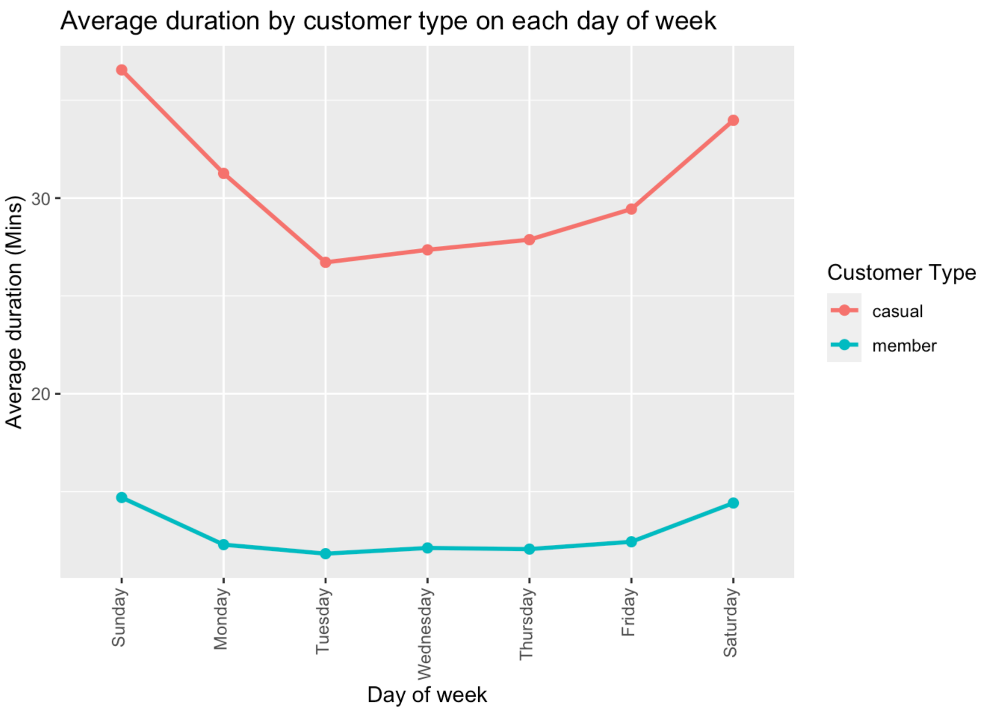
```
tripdata_monthly %>%
  ggplot(aes(x = month_year, y = average_duration, color = member_casual, group = mem
ber_casual)) +
  geom_line(size = 1) + geom_point(size = 2) +
  labs(title = "Average duration by customer type for previous 12 months", color = "C
ustomer Type")+
  theme(axis.text.x = element_text(angle = 90)) +
  xlab("Month") + ylab("Average duration (Mins)")
```



The maximum of average trip duration is close to 40 minutes in May 2021 by casual users. The average duration of casual users is twice or more than that of annual members.

The average duration by customer type on each day of week

```
tripdata_weekly %>%
  ggplot(aes(x = day_of_week, y = average_duration, color = member_casual, group = me
mber_casual)) +
  geom_line(size = 1) + geom_point(size = 2) +
  labs(title = "Average duration by customer type on each day of week", color = "Cust
omer Type")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.25, hjust = 1)) +
  xlab("Day of week") + ylab("Average duration (Mins)")
```



The average duration of casual users are twice or more than that of the annual member. It represents that the casual users are likely spending more time on long bike trip during the weekend. Also, the average duration of annual member keeps consistent over the weekdays.

Save the datasets for futher uses

```
write_csv(tripdata, file = "data\tripdata.csv")
write_csv(tripdata_monthly, file = "data\tripdata_monthly.csv")
write_csv(tripdata_weekly, file = "data\tripdata_weekly.csv")
```

6. Act

Key Findings

- Annual members are occupied 56% of total users.
- Casual users take long time which is close to 40 minutes for the bike trip while annual members take short time bike trip over 12 months.
- Classic bike is the most popular bike type for both groups.
- Saturday is the busiest time as the number of rides is the most by casual users.
- July is the most popular month of 2021. There is the significant increase in number of rides for both groups compared to other months, especially the casual users that they tend to have a long bike trip in summer.
- The average duration for day of week indicates that the annual member use the rental bike for going to work or school while the casual users use for leisure or tourism.

Recommandation

- Offer a special discount of membership during summer for casual users.
- Develop reward point systems for casual users depending on the ride length and the frequency of bike rental. The higher reward points can get a special discount for renewing membership.
- Provide the promotion for the bike rental who willing to join membership for casual users during non-peak days.