Cyclistic bike-share analysis

Yehao Zheng

2022-06-10

About the conpany

Cyclistic is a fictional bike-share company launched in 2016. The shareable bikes 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. Cyclistic offers flexible 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.

Introduction

As a junior data analyst working in the marketing team of a bike-share company Cyclistic in Chicago, I'm responsible for collecting, analyzing, and reporting data that helps guide Cyclistic marketing strategy. In order to get the Cyclistic executives approval for my recommendations, I will come up with compelling data insights and professional data visualizations. The project follow the data analysis process: ask, prepare, process, analyze, share, and act.

Ask

To improve the company's performance, the director of marketing believes the company needs to maximize the number of annual merberships. Therefore, I want to understand How do annual members and casual riders use Cyclistic bikes differently, and designs a new marketing strategy to convert casual riders into annual members.

Prepare

I will be using the Cyclistic's historical trip data from May 2021 to April 2022 to analyze and identify trends. The data has been made available by Motivate International Inc. under this license. It is public data that you can use to explore how different customer types are using Cyclistic bikes.

Process

Load necessary libraries

```
library(tidyverse) # needed for read_csv()
library(janitor) # needed for compare_df_cols()
library(dplyr) # needed for bind_rows()
library(hms) # change difftime to HHMMSS
library(scales) # scale_y_continuous(labels = comma)
```

Load the previous 12 months of cyclistic trip data

```
data_202105 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202106 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
```

```
data_202107 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202108 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202109 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202110 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202111 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202112 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202201 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202202 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202203 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
data_202204 <- read_csv("/Users/yehao/Desktop/Coursera/Google Data Analytics Professional Certification/
```

Examine the data types for the columns of each dataset, Ensure merge can be successful

```
compare_df_cols(data_202105,data_202106,data_202107,data_202108,data_202109,data_202110,data_202111,data
```

```
##           column_name     data_202105     data_202106     data_202107
## 1             end_lat         numeric         numeric         numeric
## 2             end_lng         numeric         numeric         numeric
## 3      end_station_id       character       character       character
## 4    end_station_name       character       character       character
## 5            ended_at POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
## 6       member_casual       character       character       character
## 7             ride_id       character       character       character
## 8       rideable_type       character       character       character
## 9           start_lat         numeric         numeric         numeric
## 10          start_lng         numeric         numeric         numeric
## 11   start_station_id       character       character       character
## 12 start_station_name       character       character       character
## 13         started_at POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
##        data_202108     data_202109     data_202110     data_202111
## 1          numeric         numeric         numeric         numeric
## 2          numeric         numeric         numeric         numeric
## 3        character       character       character       character
## 4        character       character       character       character
## 5  POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
## 6        character       character       character       character
## 7        character       character       character       character
## 8        character       character       character       character
## 9          numeric         numeric         numeric         numeric
## 10         numeric         numeric         numeric         numeric
## 11       character       character       character       character
## 12       character       character       character       character
## 13 POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
##        data_202112     data_202201     data_202202     data_202203
## 1          numeric         numeric         numeric         numeric
## 2          numeric         numeric         numeric         numeric
## 3        character       character       character       character
## 4        character       character       character       character
## 5  POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
## 6        character       character       character       character
## 7        character       character       character       character
## 8        character       character       character       character
## 9          numeric         numeric         numeric         numeric
```

```
## 10         numeric         numeric         numeric         numeric
## 11       character       character       character       character
## 12       character       character       character       character
## 13 POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt POSIXct, POSIXt
##        data_202204
## 1          numeric
## 2          numeric
## 3        character
## 4        character
## 5  POSIXct, POSIXt
## 6        character
## 7        character
## 8        character
## 9          numeric
## 10         numeric
## 11       character
## 12       character
## 13 POSIXct, POSIXt
```

Merge the previous 12 months of data

```
data_12months <- bind_rows(data_202105,data_202106,data_202107,data_202108,data_202109,data_202110,data_
```

See if the merge is successful

```
str(data_12months)
```

```
## spec_tbl_df [5,757,551 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:5757551] "C809ED75D6160B2A" "DD59FDCE0ACACAF3" "0AB83CB88C43EFC2" "788
##  $ rideable_type     : chr [1:5757551] "electric_bike" "electric_bike" "electric_bike" "electric_bik
##  $ 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>
```

Calculate the length of each ride in secs

```
ride_length <- difftime(data_12months$ended_at, data_12months$started_at, units = "secs")
```

Change the length of each ride to the format of HHMMSS and store it in a new column ride_length

```
# x <- abs(as.numeric(ride_length))
# data_12months$ride_length <- sprintf("%02d:%02d:%02d",  x %% 86400 %/% 3600, x %% 3600 %/% 60,  x %%
# data_12months$day_of_week <- weekdays(data_12months$started_at)

data_12months$ride_length <- as_hms(ride_length)
```

Filter out the rows with ride_length <= 0

```
data_12months <- filter(data_12months, ride_length > 0)
str(data_12months)
```

```
## spec_tbl_df [5,756,899 x 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:5756899] "C809ED75D6160B2A" "DD59FDCE0ACACAF3" "0AB83CB88C43EFC2" "788
##  $ rideable_type     : chr [1:5756899] "electric_bike" "electric_bike" "electric_bike" "electric_bik
##  $ started_at        : POSIXct[1:5756899], format: "2021-05-30 11:58:15" "2021-05-30 11:29:14" ...
##  $ ended_at          : POSIXct[1:5756899], format: "2021-05-30 12:10:39" "2021-05-30 12:14:09" ...
##  $ start_station_name: chr [1:5756899] NA NA NA NA ...
##  $ start_station_id  : chr [1:5756899] NA NA NA NA ...
##  $ end_station_name  : chr [1:5756899] NA NA NA NA ...
##  $ end_station_id    : chr [1:5756899] NA NA NA NA ...
##  $ start_lat         : num [1:5756899] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:5756899] -87.6 -87.6 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:5756899] 41.9 41.8 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:5756899] -87.6 -87.6 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:5756899] "casual" "casual" "casual" "casual" ...
##  $ ride_length       : 'hms' num [1:5756899] 00:12:24 00:44:55 00:01:12 00:15:13 ...
##   ..- attr(*, "units")= chr "secs"
##  - 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>
```

Create a column "day_of_week" and calculate the day of the week that each ride started and select the necessary columns for analysis

```
data_12months$day_of_week <- weekdays(data_12months$started_at)
data_12months <- data_12months %>%
  select(ride_id, rideable_type, started_at, ended_at, member_casual, ride_length, day_of_week)
str(data_12months)
```

```
## tibble [5,756,899 x 7] (S3: tbl_df/tbl/data.frame)
##  $ ride_id      : chr [1:5756899] "C809ED75D6160B2A" "DD59FDCE0ACACAF3" "0AB83CB88C43EFC2" "7881AC6D3
##  $ rideable_type: chr [1:5756899] "electric_bike" "electric_bike" "electric_bike" "electric_bike" ..
##  $ started_at   : POSIXct[1:5756899], format: "2021-05-30 11:58:15" "2021-05-30 11:29:14" ...
##  $ ended_at     : POSIXct[1:5756899], format: "2021-05-30 12:10:39" "2021-05-30 12:14:09" ...
##  $ member_casual: chr [1:5756899] "casual" "casual" "casual" "casual" ...
##  $ ride_length  : 'hms' num [1:5756899] 00:12:24 00:44:55 00:01:12 00:15:13 ...
##   ..- attr(*, "units")= chr "secs"
##  $ day_of_week  : chr [1:5756899] "Sunday" "Sunday" "Sunday" "Sunday" ...
```

Analyze

Calculate the mean of ride_length

```
length_secs <- as.numeric(data_12months$ride_length)
cat('Mean of ride_length is', mean(length_secs), "\n")
```

```
## Mean of ride_length is 1268.466
```

```
cat('Max ride_length is', max(length_secs), "\n")
```

```
## Max ride_length is 3356649
```

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
cat('Mode of day_of_week is', getmode(data_12months$day_of_week))
```

```
## Mode of day_of_week is Saturday
```

Calculate the average ride_length for members and casual riders.

```
mean_rl_mc <- data_12months %>%
  group_by(member_casual) %>%
  summarise("Average of ride_length" = round(mean(ride_length), 2)) %>%
  rename("member or casual"= member_casual)
mean_rl_mc
```

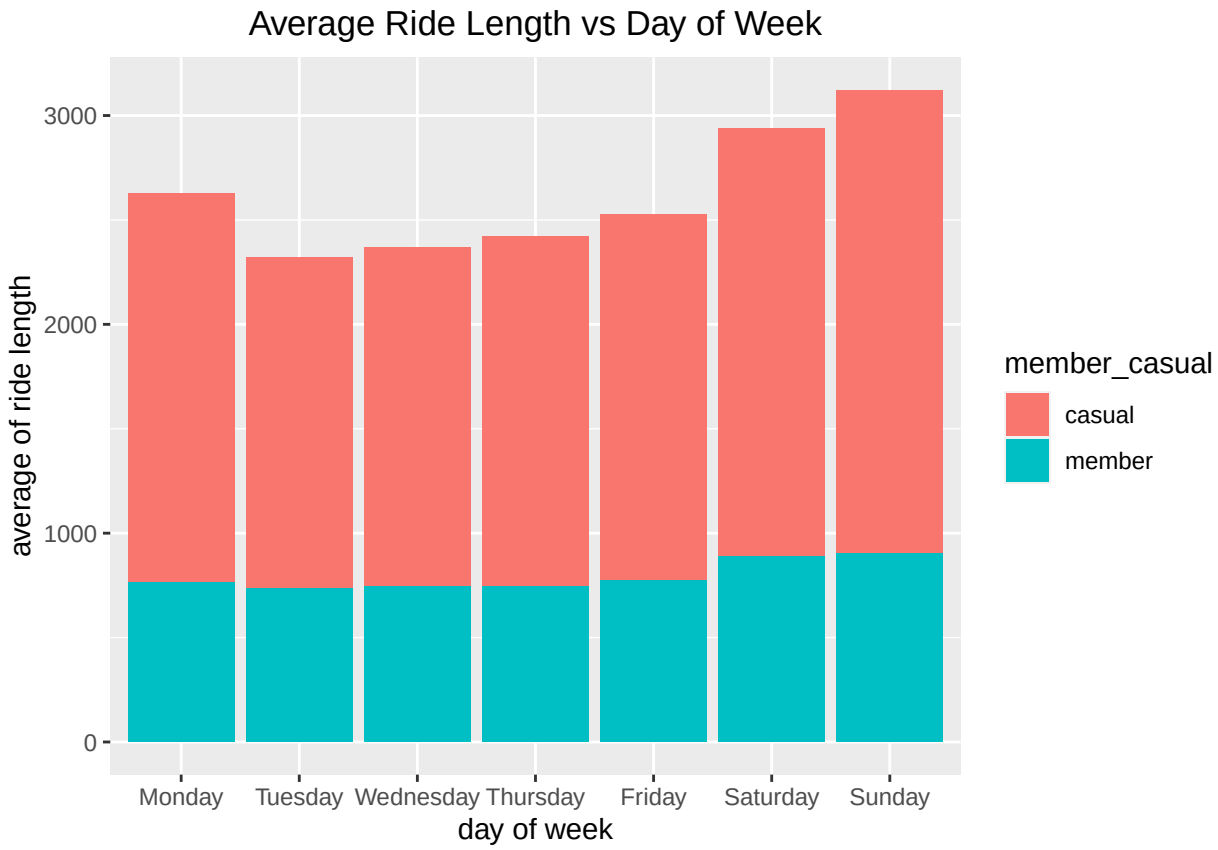
```
## # A tibble: 2 x 2
##   `member or casual` `Average of ride_length`
##   <chr>              <drtn>
## 1 casual             1877.74 secs
## 2 member              788.75 secs
```

Sort the data from Monday to Sunday and then calculate the average ride_length for users by day_of_week.

```
data_12months$day_of_week <- ordered(data_12months$day_of_week, levels=c("Monday", "Tuesday", "Wednesda
mean_rl_mc_wday <- data_12months %>%
  group_by(day_of_week, member_casual) %>%
  summarise("Average_of_ride_length" = round(mean(ride_length), 2))
mean_rl_mc_wday
```

```
## # A tibble: 14 x 3
## # Groups:   day_of_week [7]
##    day_of_week member_casual Average_of_ride_length
##    <ord>       <chr>         <drtn>
##  1 Monday      casual        1864.19 secs
##  2 Monday      member         762.79 secs
##  3 Tuesday     casual        1588.05 secs
##  4 Tuesday     member         735.63 secs
##  5 Wednesday   casual        1625.75 secs
##  6 Wednesday   member         745.01 secs
##  7 Thursday    casual        1673.42 secs
##  8 Thursday    member         746.33 secs
##  9 Friday      casual        1752.74 secs
## 10 Friday      member         772.74 secs
## 11 Saturday    casual        2051.84 secs
## 12 Saturday    member         886.90 secs
## 13 Sunday      casual        2218.41 secs
## 14 Sunday      member         903.78 secs
```

```
ggplot(data = mean_rl_mc_wday) +
  geom_bar(mapping = aes(x = day_of_week, y = as.numeric(Average_of_ride_length), fill = member_casual)
  labs(x = "day of week", y = "average of ride length", title = "Average Ride Length vs Day of Week") +
  theme(plot.title = element_text(hjust = 0.5))
```

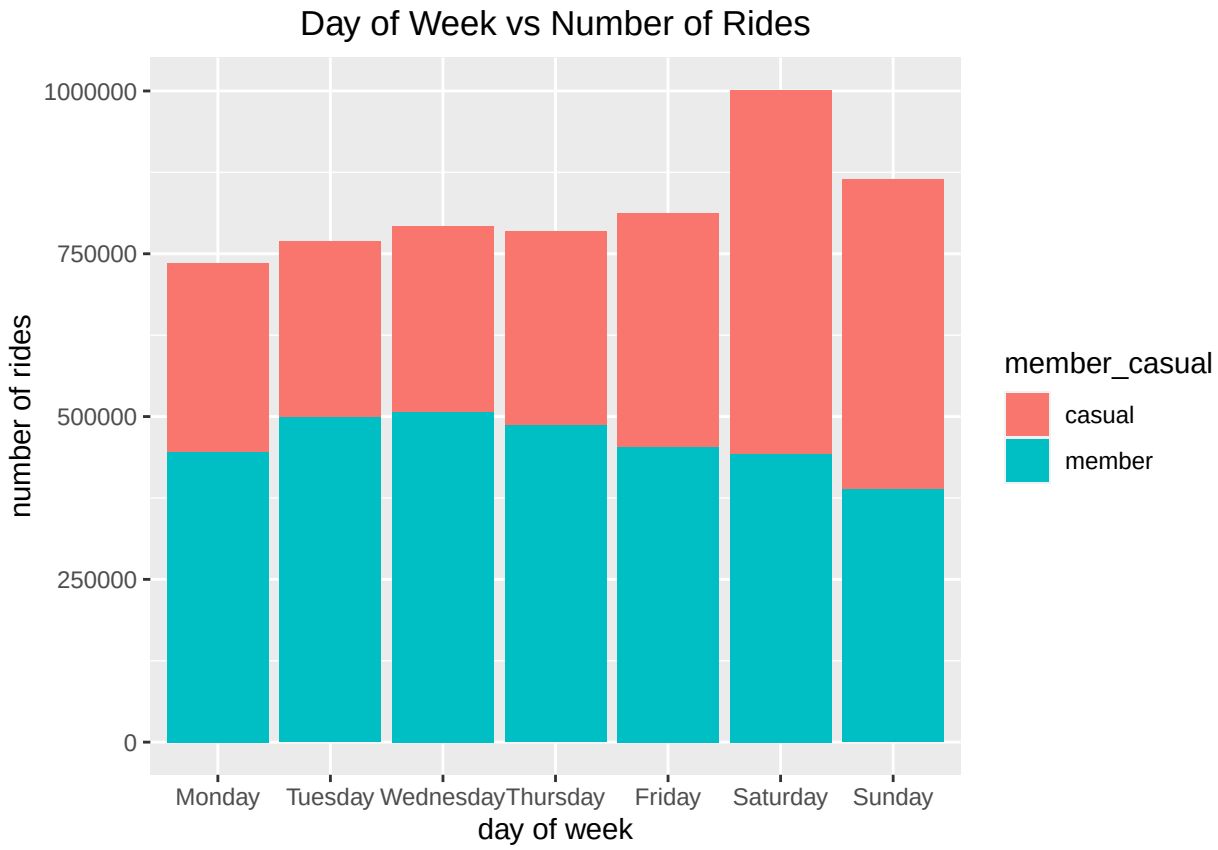


Calculate the number of rides for users by day_of_week by adding Count of trip_id to Values.

```
num_ride_wday <- data_12months %>%
  group_by(day_of_week, member_casual) %>%
  summarise("number_of_rides" = n_distinct(ride_id))
num_ride_wday
```

```
## # A tibble: 14 x 3
## # Groups:   day_of_week [7]
##    day_of_week member_casual number_of_rides
##    <ord>       <chr>                  <int>
##  1 Monday      casual                288991
##  2 Monday      member                445605
##  3 Tuesday     casual                270509
##  4 Tuesday     member                498645
##  5 Wednesday   casual                284833
##  6 Wednesday   member                506899
##  7 Thursday    casual                298033
##  8 Thursday    member                485812
##  9 Friday      casual                358157
## 10 Friday      member                453244
## 11 Saturday    casual                558543
## 12 Saturday    member                442711
## 13 Sunday      casual                476936
## 14 Sunday      member                387981
```

```
ggplot(data = num_ride_wday) +
  geom_bar(mapping = aes(x = day_of_week, y = number_of_rides, fill = member_casual), stat =  "identity
  labs(x = "day of week", y = "number of rides", title = "Day of Week vs Number of Rides") +
  theme(plot.title = element_text(hjust = 0.5))
```

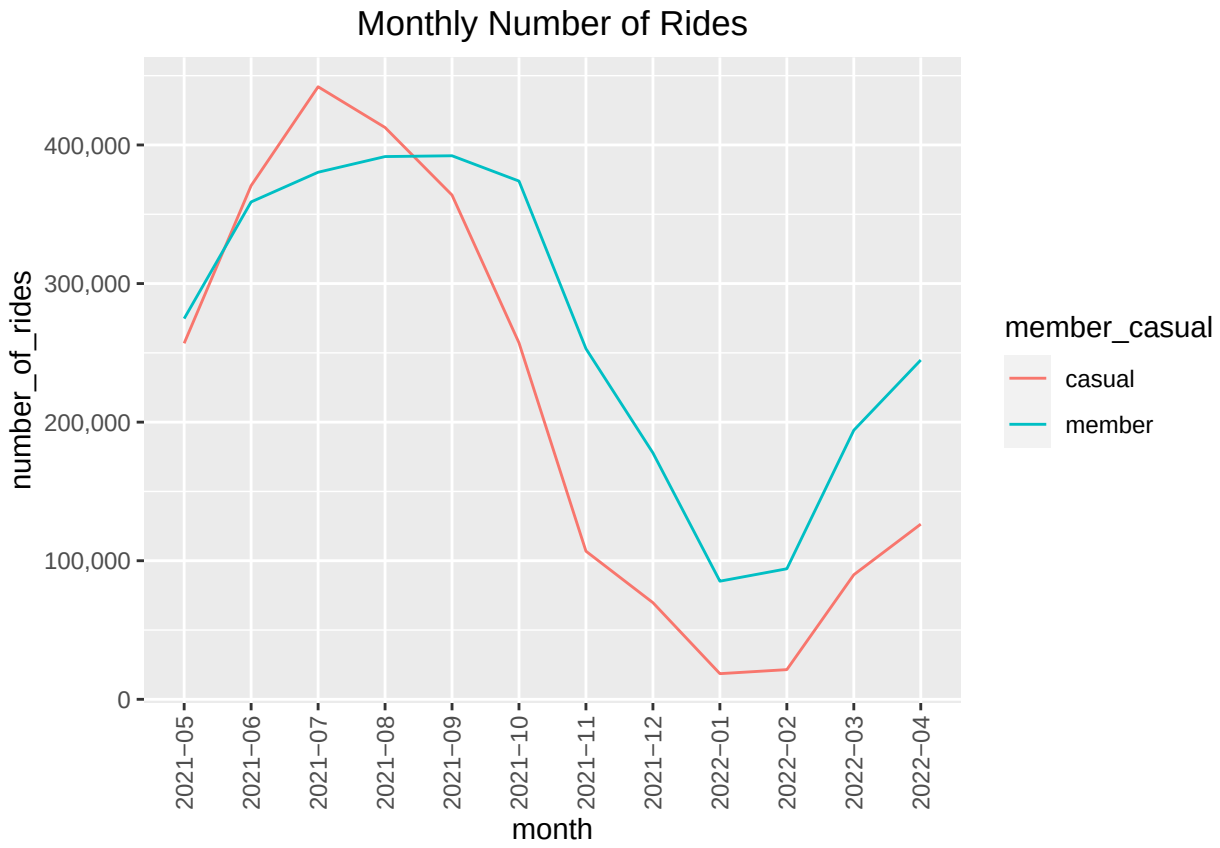


Difference in number of rides for both annual member and casual riders by month

```
#num_ride_per_day <- data_12months %>%
#  group_by("day" = as.Date(started_at)) %>%
#  summarise("number_of_rides" = n_distinct(ride_id))

num_ride_per_month <- data_12months %>%
  group_by("month" = format(as.Date(started_at), "%Y-%m"), member_casual) %>%
  summarise("number_of_rides" = n_distinct(ride_id))

ggplot(data = num_ride_per_month) +
  geom_line(mapping = aes(x = month, y = number_of_rides, colour = member_casual, group = member_casual
  scale_y_continuous(labels = comma) +
  labs(title = "Monthly Number of Rides") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, vjust = 0.5, hju
```



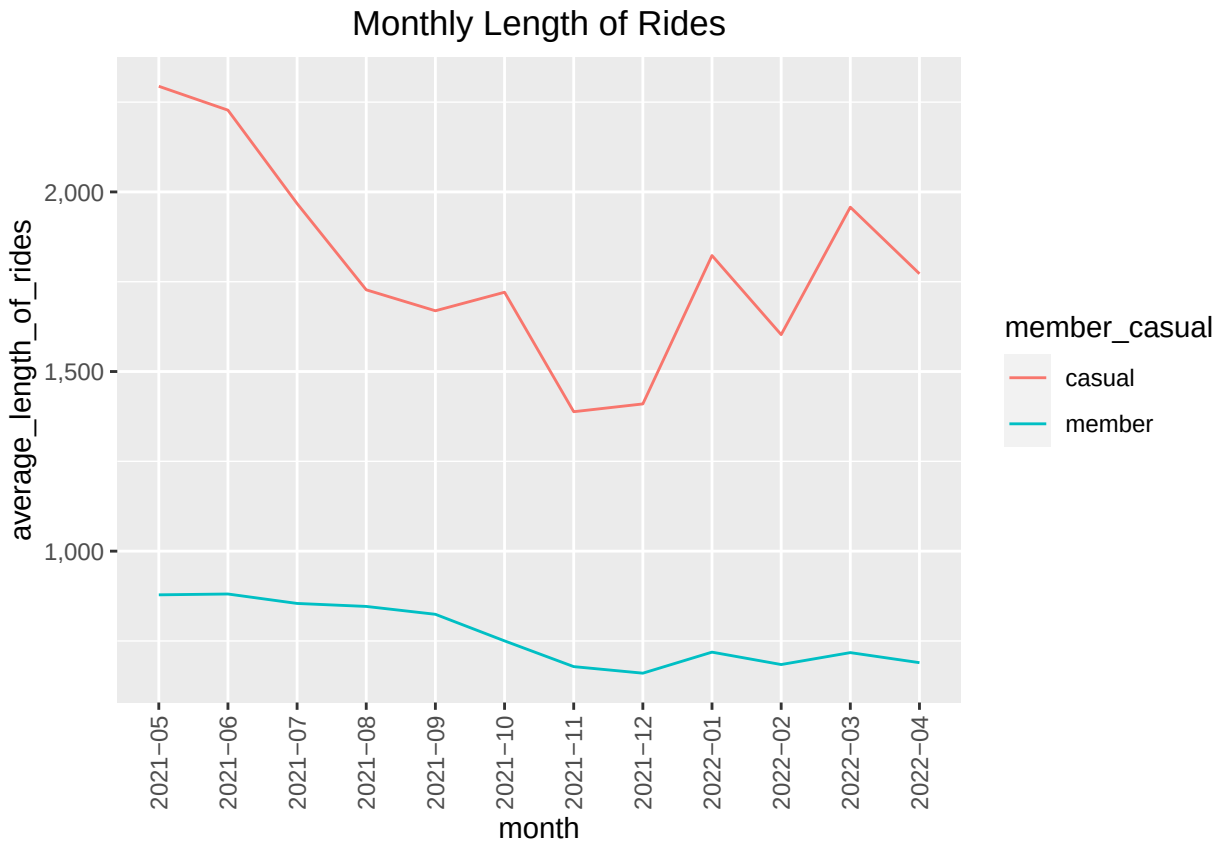
Difference in length of rides for both annual member and casual riders by month

```
ride_length_per_month <- data_12months %>%
  group_by("month" = format(as.Date(started_at), "%Y-%m"), member_casual) %>%
  summarise("average_length_of_rides" = round(mean(ride_length), 2))
ride_length_per_month
```

```
## # A tibble: 24 x 3
## # Groups:   month [12]
##    month   member_casual average_length_of_rides
##    <chr>   <chr>         <drtn>
##  1 2021-05 casual        2294.11 secs
##  2 2021-05 member         878.42 secs
##  3 2021-06 casual        2227.56 secs
##  4 2021-06 member         880.72 secs
##  5 2021-07 casual        1967.61 secs
##  6 2021-07 member         854.44 secs
##  7 2021-08 casual        1727.45 secs
##  8 2021-08 member         846.15 secs
##  9 2021-09 casual        1669.13 secs
## 10 2021-09 member         824.19 secs
## # ... with 14 more rows
```

```
ggplot(data = ride_length_per_month) +
  geom_line(mapping = aes(x = month, y = average_length_of_rides, colour = member_casual, group = membe
  scale_y_continuous(labels = comma) +
  labs(title = "Monthly Length of Rides") +
```

```
theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, vjust = 0.5, hjus
```



Share

Act

From the above tables and graphs, we know that the annual members tend to ride the bikes for commute to work because their usage of bikes are higher during the weekdays. In contrast, the casual riders prefer to ride the bikes for leisure during the weekend. we can also conclude that most of the time annual members have used the shareable bikes more frequently than the casual riders. However,the average ride length of casual riders are significantly longer than the annual members. To improve the company's performance, we can use this conclusion to convince the casual riders that even though they might not use our bikes very often, they would probably save more money by becoming a annual member because each of their ride is expensive based on the length. Then, we can also introduce a new annual weekend plan where the members of this plan can use the bikes freely during all the weekends.