Cyclistic Case Study: A Chicago Bike-share Analysis

Isabella Pelletier

2022-11-16

SCENARIO

In this case study I am a junior data analyst working in the marketing analyst team for the fictional bike-share company, Cyclistic. The Director of Marketing, Lily Moreno, believes that the company's future success depends on maximizing the number of annual memberships. My team wants to understand how casual riders and annual members differ in the way they use Cyclistic bikes. Using these insights my team will design a new marketing strategy to convert casual riders into annual members.

About Cyclistic

Cyclistic is a bike sharing company based out of Chicago, Illinois that launched in 2016. The program has a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across the City.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. There are three different pricing plans which offer flexibility to the companies users. Pricing plans are: single-ride passes, full-day passes, and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders, while customers who purchase annual memberships are annual members.

Cyclistic has set itself apart by offering reclining bikes, hand tricycles, and cargo bikes, making bike-sharing more inclusive to people with disabilities and riders who can't use a standard two-wheel bike. The majority of riders opt for traditional bikes, about 8% of riders use the assistive options. Historically, Cyclistic users are more likely to ride for leisure, but about 30% use them to commute to work each day.

ASK

Business task: To create a marketing campaign designed to convert casual riders to annual members by analyzing the differences between casual and annual Cyclistic users.

PREPARE

Data source: https://divvy-tripdata.s3.amazonaws.com/index.html

Data License: https://ride.divvybikes.com/data-license-agreement

(Note: these data sets have a different name because Cyclistic is a fictional company)

The files were downloaded and unzipped so they could be used in .csv format.

This data is current and credible as well as being original from the source. From this data source I will be using files from the 12 month period between October 2021 and September 2022.

PROCESS

I am using R to process and publish the data set.

The data frames all have 13 columns:

- ride_id (the ID of the rider of the particular ride)
- rideable_type (the type of bike used for the ride)
- started_at (the start date/time of the ride)
- ended_at (the end date/time of the ride)
- start_station_name (the name of the station where the ride started)
- start_station_id (the ID of the station where the ride started)
- end_station_name (the name of the station where the ride ended)
- end_station_id (the ID of the station where the ride ended)
- start_lat (the latitude of the station where the ride started)
- start_lng (the longitude of the station where the ride ended)
- end_lat (the latitude of the station where the ride started)
- end_lng (the longitude of the station where the ride ended)
- member_casual (the type of rider on that particular ride)

I start by loading the necessary packages.

Once I familiarized myself with the data I downloaded the 12 sets into R, from October 2021 through September 2022.

I will now combine the files into one data frame and delete the individual files to clear space. The total number of observations in the combined data frame is 5,828,235.

```
cyclistic_data <- rbind(october_2021, november_2021, december_2021, january_2022, february_2022, march_
remove(october_2021, november_2021, december_2021, january_2022, february_2022, march_2022, april_2022,
```

Now it's time to clean the data set. First I begin by removing the nulls and making sure all rows are distinct.

```
cyclistic_data <- na.omit(cyclistic_data)
cyclistic_data <- distinct(cyclistic_data)
```

Doing this reduced the number of observations from 5,828,235 to 4,474,141 (1,354,094 rows removed).

Next, I filter out the stations where maintenance testing was being conducted.

```
cyclistic_data <- cyclistic_data[!grepl("TEST", cyclistic_data$start_station_id),]
cyclistic_data <- cyclistic_data[!grepl("TEST", cyclistic_data$end_station_id),]
cyclistic_data <- cyclistic_data[!grepl("TEST", cyclistic_data$start_station_name),]
cyclistic_data <- cyclistic_data[!grepl("TEST", cyclistic_data$end_station_name),]
```

The number of observations went from 4,474,141 to 4,472,680 (1,461 rows removed).

Now I want to discover the length of the riders bike rides. To do this I create ride_length, measuring in minutes.

```
ride_length <- difftime(cyclistic_data$ended_at,cyclistic_data$started_at, units="mins")
cyclistic_data$ride_length <- ride_length
```

With this variable I start by deleting all ride lengths that are negative and over 24 hours, to ensure the data is reliable.

```
cyclistic_data <- cyclistic_data[cyclistic_data$ride_length >= 0,]
cyclistic_data <- cyclistic_data[cyclistic_data$ride_length < 1440,]
```

The number of observations went from 4,472,680 to 4,472,340 (340 rows removed).

Now I will create new columns in the cyclistic_data data frame for month, day, year, time and day of the week to help with later calculations.

```
cyclistic_data$date <- as.Date(cyclistic_data$started_at)
cyclistic_data$weekday <- format(as.Date(cyclistic_data$date), "%A")
cyclistic_data$month <- format(as.Date(cyclistic_data$date), "%b")
cyclistic_data$day <- format(as.Date(cyclistic_data$date), "%d")
cyclistic_data$year <- format(as.Date(cyclistic_data$date), "%Y")
cyclistic_data$time <- as_hms((cyclistic_data$started_at))
cyclistic_data$hour <- hour(cyclistic_data$time)
```

ANALYZE & SHARE

To begin I check the total number of Cyclistic rides between October 2021 and September 2022. This is found to be 4,472,340, the same as the number of rows.

```
nrow(cyclistic_data)
```

```
## [1] 4472340
```

The average length of ride for all riders is 17.22373 minutes.

```
mean(cyclistic_data$ride_length)
```

```
## Time difference of 17.22373 mins
```

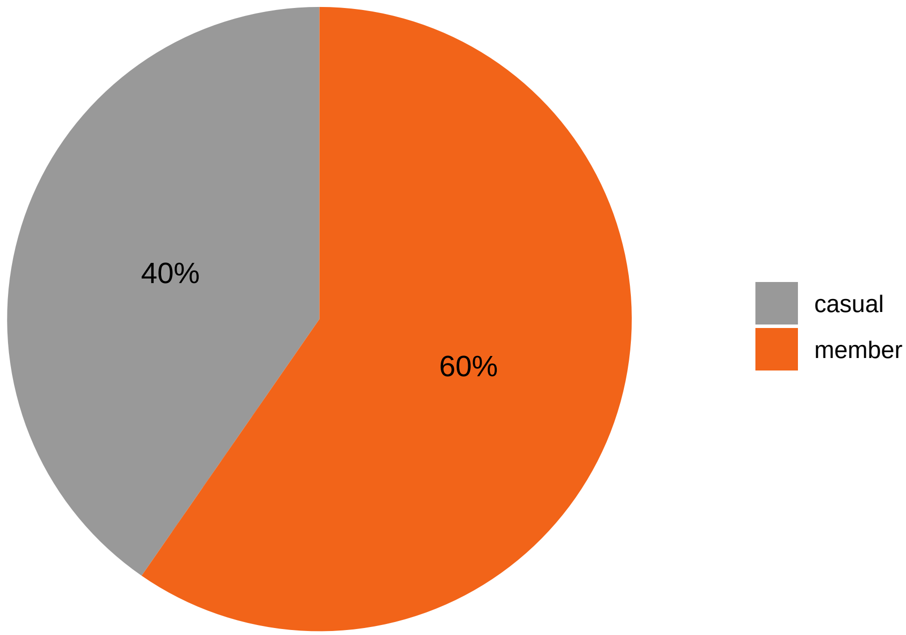
To delve into the differences between the two member types I will to group a few variables by rider type. From this, I can see that 1,804,992 rides (*40%*) were casual riders and 2,667,348 rides (*60%*) were annual members. The average length of ride for casual riders was *24.24666 minutes*, while the average ride length for annual members was *12.47131 minutes*, approximately half that of casual riders. The maximum ride length was very similar between the two groups, at *1439.367 minutes* for casual riders and *1435.467 minutes* for annual members. The minimum ride length for both groups was zero.

```
cyclistic_data %>%
  group_by(member_casual) %>%
  summarise(number_members=(n = n()),average_ride_length=mean(ride_length), max_ride_length=max(ride_le
```

```
## # A tibble: 2 x 5
##   member_casual number_members average_ride_length max_ride_length min_ride_le~1
##   <chr>                  <int> <drtn>              <drtn>          <drtn>
## 1 casual               1804992 24.24666 mins       1439.367 mins   0 mins
## 2 member               2667348 12.47131 mins       1435.467 mins   0 mins
## # ... with abbreviated variable name 1: min_ride_length
```

```
ggplot(percent_members, aes(x = "", y = percent_riders, fill=member_casual)) + geom_bar(stat="identity"
          axis.text = element_blank(),
          axis.ticks = element_blank(),
          plot.title = element_text(hjust = 0.5, color = "#666666"))
```

Total Cyclistic rides from Oct 2021 to Sept 2022



Next, I look to see if there is a difference in the type of bike preferred by each user type. From the summary below I discover that the most used bike type over the year was the classic bike, used a total of 2,735,294 times. 1,797,020 times by annual members and 938,274 times by casual members. The second most used bike type was the electric bike, used a total of 1,547,375 times. 870,328 times by annual members and 677,047 times by casual members. The last bike type was the docked bike which was used 189,671 times by only casual members.

```
bike_by_type <- cyclistic_data %>%
  group_by(member_casual, rideable_type) %>%
  summarise(n = n())
```

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

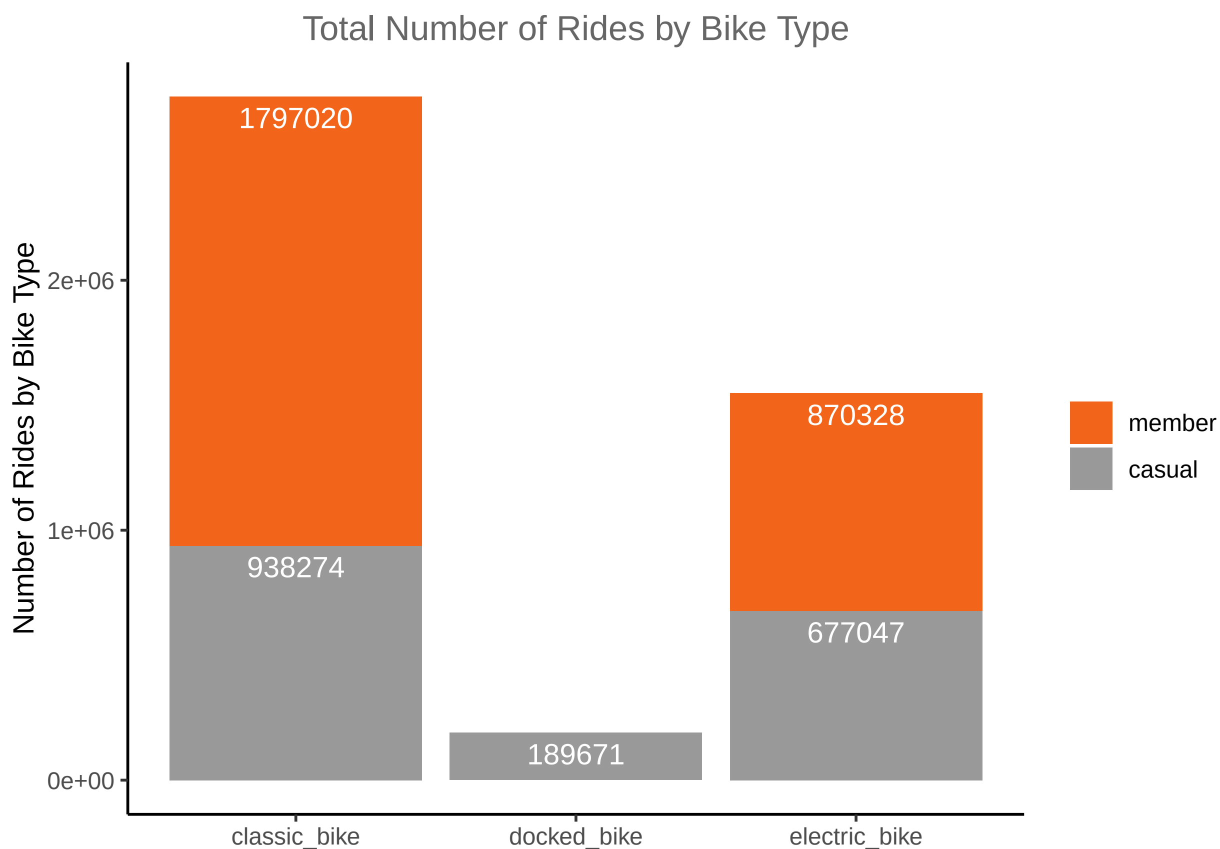
```
bike_by_type
```

```
## # A tibble: 5 x 3
## # Groups:   member_casual [2]
##   member_casual rideable_type       n
##   <chr>         <chr>           <int>
## 1 casual        classic_bike   938274
## 2 casual        docked_bike    189671
## 3 casual        electric_bike  677047
## 4 member        classic_bike  1797020
## 5 member        electric_bike  870328
```

```
bike_by_type <- bike_by_type %>%
  group_by(rideable_type) %>%
  mutate(label_y = cumsum(n))
```

```
ggplot(bike_by_type, aes(x = rideable_type, y = n, fill = member_casual)) +
  geom_col(position = position_stack(reverse = TRUE)) +
```

```
  guides(fill = guide_legend(reverse = TRUE)) +
  labs(x = NULL, y="Number of Rides by Bike Type", fill = NULL, title = "Total Number of Rides by Bike '
  geom_text(aes(y = label_y, label = n), vjust = 1.5, colour = "white") +
  scale_fill_manual(values = c("#999999", "#F26419")) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, color = "#666666"))
```



I can now look at how the time of day, day of the week, and month differentiate between the rider types.

Both rider types are busiest in the late afternoon, and slowest in the early morning between 2-4 am. Casual riders appear to have a slow increase in the number of riders throughout the day, with a *minimum* of 4,949 at *4 am* and a *maximum* of 172,797 at *5 pm*. Annual members show a *minimum* of 5,371 at *4 am* and then a spike in the morning of 170,659 at 8 am during the morning commute. The number of riders then declines until approximately 10 am before increasing into the afternoon to reach a *maximum* of 286,859 at *5 pm*.

```
hour_count <- cyclistic_data %>%
  group_by(hour, member_casual) %>%
  summarise(twelve_hour_count = n())
```

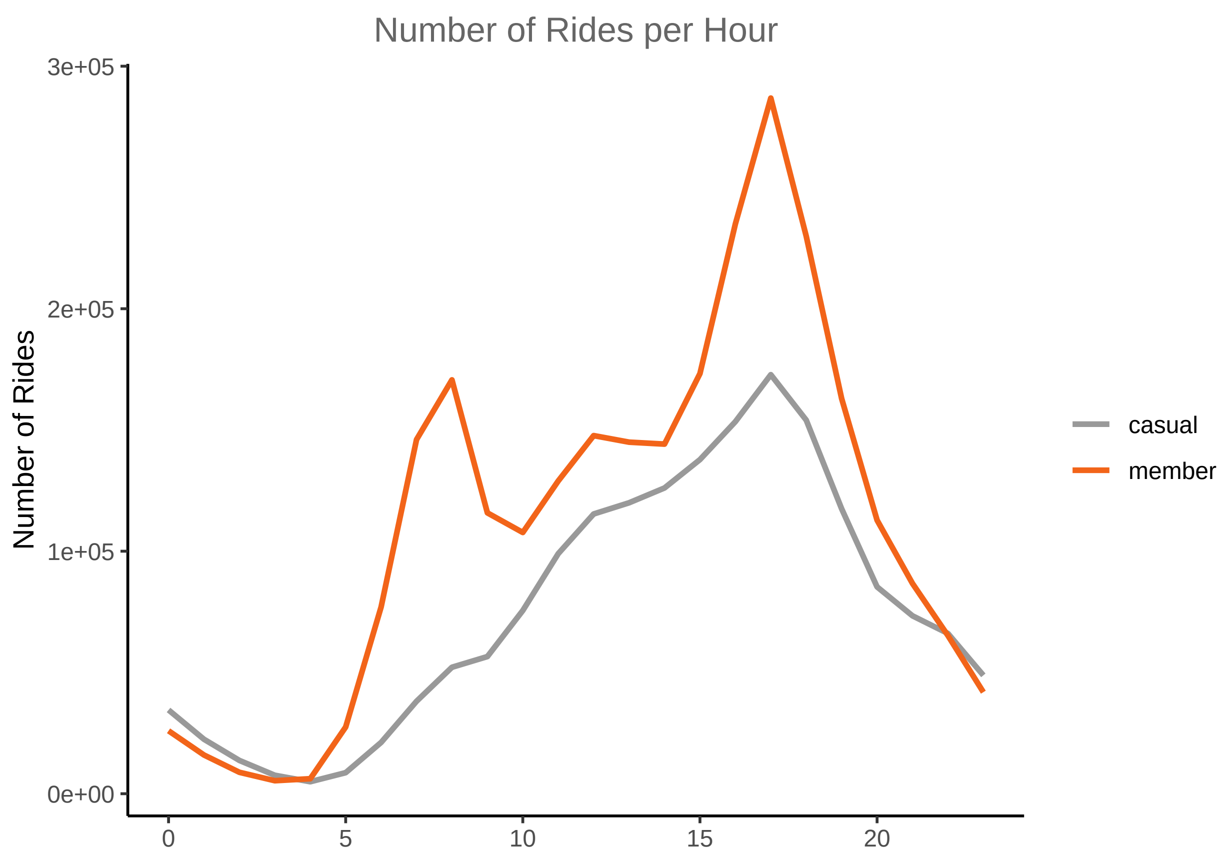
```
## `summarise()` has grouped output by 'hour'. You can override using the
## `.groups` argument.
```

```
hour_count
```

```
## # A tibble: 48 x 3
## # Groups:   hour [24]
##     hour member_casual twelve_hour_count
##    <int> <chr>                     <int>
##  1     0 casual                    34544
```

```
## 2      0 member                    25952
## 3      1 casual                    22456
## 4      1 member                    15971
## 5      2 casual                    13719
## 6      2 member                     8827
## 7      3 casual                     7606
## 8      3 member                     5371
## 9      4 casual                     4949
## 10     4 member                     6181
## # ... with 38 more rows
## # i Use `print(n = ...)` to see more rows
```

```
ggplot(hour_count, aes(x = hour, y = twelve_hour_count)) +
  geom_line(aes(color = member_casual), size = 1) +
  scale_color_manual(values = c("#999999", "#F26419")) +
  labs(x = NULL, y="Number of Rides", title = "Number of Rides per Hour") +
  theme_classic()+
  theme(legend.title=element_blank()) +
  theme(plot.title = element_text(hjust = 0.5, color = "#666666"))
```



During the week annual members reach a *minimum* on *Sunday's* of 302,440 and a *maximum* of 428,300 on *Tuesday's*. For annual members the amount of rides is higher during the work week and low on weekends. The opposite is true for casual riders. The number of casual riders go up during the weekend and decrease during the work week. Casual riders reach a minimum of 202,780 on *Tuesday's* and a *maximum* of 388014 on *Saturday's*.

```
week_count <- cyclistic_data %>%
  group_by(weekday, member_casual) %>%
```

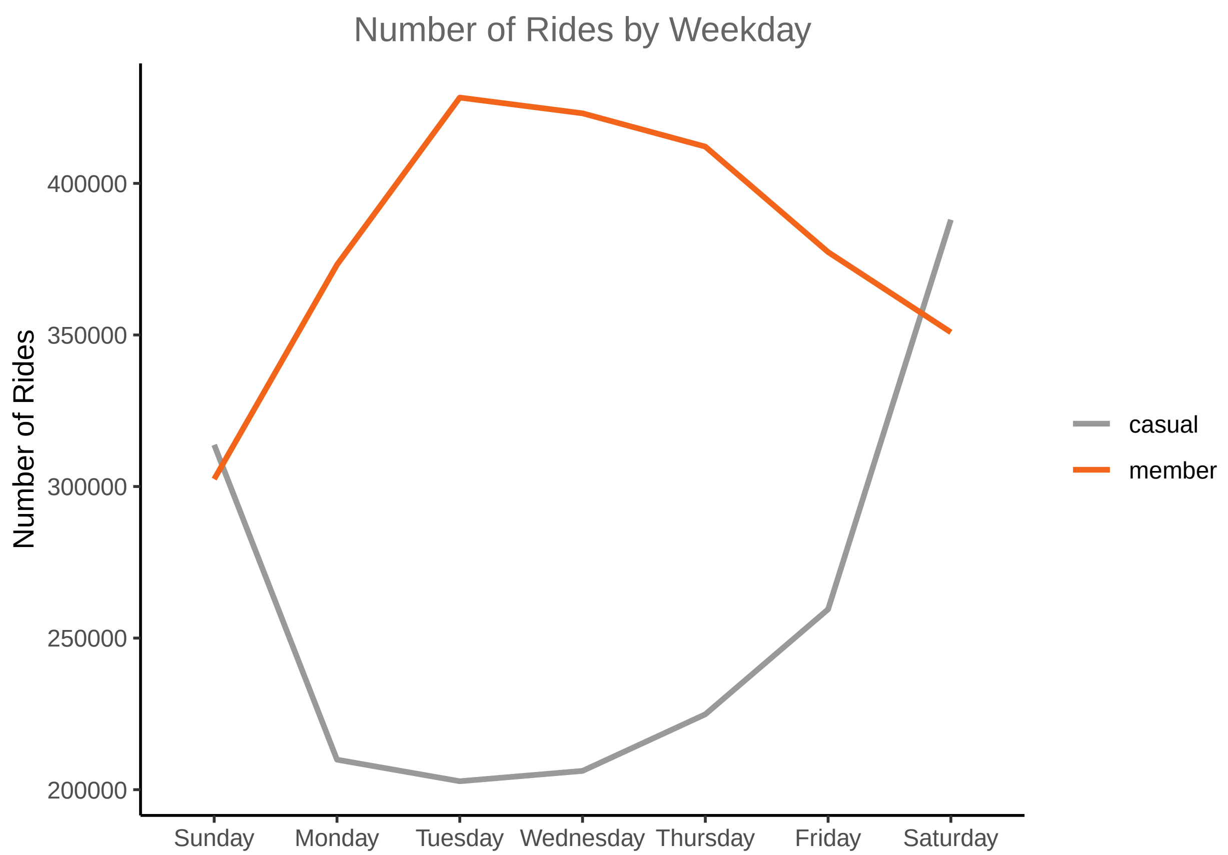
```
  summarise(weekday_count = n())
```

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

```
week_count
```

```
## # A tibble: 14 x 3
## # Groups:   weekday [7]
##    weekday   member_casual weekday_count
##    <chr>     <chr>                 <int>
##  1 Friday    casual               259502
##  2 Friday    member               377337
##  3 Monday    casual               209894
##  4 Monday    member               373160
##  5 Saturday  casual               388014
##  6 Saturday  member               350881
##  7 Sunday    casual               313748
##  8 Sunday    member               302440
##  9 Thursday  casual               224863
## 10 Thursday  member               412121
## 11 Tuesday   casual               202780
## 12 Tuesday   member               428300
## 13 Wednesday casual               206191
## 14 Wednesday member               423109
```

```
ggplot(week_count, aes(x = factor(week_count$weekday, levels = c("Sunday", "Monday", "Tuesday", "Wednes
  geom_line(aes(color = member_casual), size = 1) +
  scale_color_manual(values = c("#999999", "#F26419")) +
  labs(x = NULL, y="Number of Rides", title = "Number of Rides by Weekday") +
  theme_classic() +
  theme(legend.title=element_blank()) +
  theme(plot.title = element_text(hjust = 0.5, color = "#666666"))
```



Over the year the highest number of rides were recorded during the summer months while the lowest were in the cold winter months. Casual riders reached a *minimum* number of rides in *January* of 12,587 and a *maximum* in *July* of 311,488. Annual members reached a *minimum* number of rides in January of *67,512* and a *maximum* in *August* of 335,064.

```
month_count <- cyclistic_data %>%
  group_by(month, member_casual) %>%
  summarise(twelve_month_count = n())
```

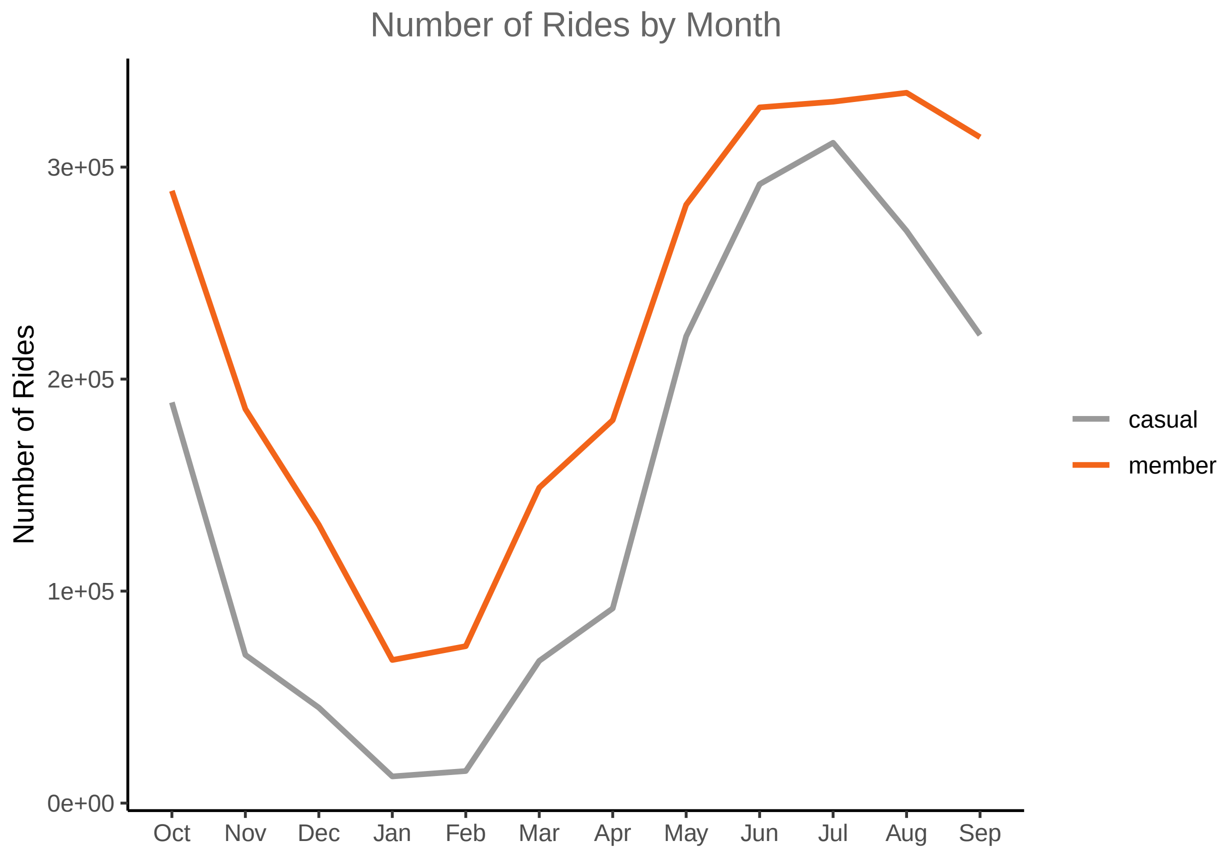
```
## `summarise()` has grouped output by 'month'. You can override using the
## `.groups` argument.
```

```
month_count
```

```
## # A tibble: 24 x 3
## # Groups:   month [12]
##    month member_casual twelve_month_count
##    <chr> <chr>                      <int>
##  1 Apr   casual                     91835
##  2 Apr   member                    180629
##  3 Aug   casual                    269932
##  4 Aug   member                    335064
##  5 Dec   casual                     45038
##  6 Dec   member                    131281
##  7 Feb   casual                     15123
##  8 Feb   member                     74022
##  9 Jan   casual                     12587
## 10 Jan   member                     67512
## # ... with 14 more rows
```

```
## # i Use `print(n = ...)` to see more rows
```

```
ggplot(month_count, aes(x = factor(month_count$month, levels = c("Oct", "Nov", "Dec", "Jan", "Feb", "Ma
  geom_line(aes(color = member_casual), size = 1) +
  scale_color_manual(values = c("#999999", "#F26419")) +
  labs(x = NULL, y="Number of Rides", title = "Number of Rides by Month") +
  theme_classic() +
  theme(legend.title=element_blank()) +
  theme(plot.title = element_text(hjust = 0.5, color = "#666666"))
```



ACT

Based on the analysis I conducted I have three recommendations for the Director of Marketing and the stakeholders involved in Cyclistic.

1. Personalize annual discounts to casual riders based on their riding habits.

2. Offer incentives to purchasing annual memberships on weekends when the majority of casual riders are using Cyclistic.

3. Launch a marketing campaign during the peak season in July and August to reach the maximum number of riders.