Cyclistic Bikes Case Study

Meena Rahmanzai

2024-12-14

Introduction

I will be conducting an analysis to answer the question "How do annual members and casual riders use Cyclistic bikes differently?" The main objective of this analysis is to help conduct a strategy on how to convert casual riders into annual members.

First step that needs to be done is to import the data set.

```r
{
  install.packages("readr")
}

# Load the package
library(readr)

# Read the CSV file
my_data <- read_csv("cyclistic_bike_data.csv")
```

Now let's preview the data set to see what we're working with.

```r
# Let's get a shorter and simpler view of the data
head(my_data)
```

```
## # A tibble: 6 x 13
##   ride_id          rideable_type started_at          ended_at
##   <chr>            <chr>         <dttm>              <dttm>
## 1 578DDD7CE1771FFA classic_bike  2024-11-07 19:21:58 2024-11-07 19:28:57
## 2 78B141C50102ABA6 classic_bike  2024-11-22 14:49:00 2024-11-22 14:56:15
## 3 1E794CF36394E2D7 classic_bike  2024-11-08 09:24:00 2024-11-08 09:28:33
## 4 E5DD2CAB58D73F98 classic_bike  2024-11-24 17:51:14 2024-11-24 18:05:32
## 5 57F9878BC8C765F1 classic_bike  2024-11-04 14:59:16 2024-11-04 15:41:02
## 6 631261306B4ADFBF classic_bike  2024-11-03 11:24:46 2024-11-03 11:32:38
## # 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>
```

Create new columns

Now that we have our data set imported, we will create new columns that will help us in our analysis. I will take the started_at and ended_at column to create a new column labeled ride_length, which will give us the total length of time for each ride taken.

```r
install.packages("dplyr")

# Load the packages
library(dplyr)

# Add the new column
my_data <- my_data %>%
  mutate(ride_length = (ended_at - started_at))

# Convert "ride_length" from factor to numeric so we can run calculations on the data
is.factor(my_data$ride_length)
```

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

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

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

Now I will create the second column which will be labeled day_of_week. This column will calculate the day of the week that each ride started. Each day will be associated with a number. For example, Sunday = 1, Monday = 2, etc. We will create this column using the started_at column.

```r
install.packages("lubridate")

# Let's load the package
library(lubridate)

# Add the day of week column
my_data <- my_data %>%
  mutate(day_of_week = wday(started_at, label = TRUE))

# Convert the days into numbers
my_data <- my_data %>%
  mutate(day_of_week = match(day_of_week, c("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat")))
```

Let's check to make sure the new columns were added

```r
head(my_data)
```

```
## # A tibble: 6 x 15
##   ride_id          rideable_type started_at          ended_at
##   <chr>            <chr>         <dttm>              <dttm>
## 1 578DDD7CE1771FFA classic_bike  2024-11-07 19:21:58 2024-11-07 19:28:57
## 2 78B141C50102ABA6 classic_bike  2024-11-22 14:49:00 2024-11-22 14:56:15
## 3 1E794CF36394E2D7 classic_bike  2024-11-08 09:24:00 2024-11-08 09:28:33
## 4 E5DD2CAB58D73F98 classic_bike  2024-11-24 17:51:14 2024-11-24 18:05:32
## 5 57F9878BC8C765F1 classic_bike  2024-11-04 14:59:16 2024-11-04 15:41:02
## 6 631261306B4ADFBF classic_bike  2024-11-03 11:24:46 2024-11-03 11:32:38
## # i 11 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>,
## #   ride_length <dbl>, day_of_week <int>
```

Let’s see how many observations fall under each user type

```r
table(my_data$member_casual)
```

```
##
## casual member
##  93078 241997
```

Remove “bad” data

The data frame includes a few hundred entries when bikes were taken out of docks and checked for quality by Divvy or ride_length was negative. We will create a new version of the data frame (v2) since data is being removed

```r
my_data_v2 <- my_data[!(my_data$start_station_name == "HQ QR" | my_data$ride_length<0),]

# Remove all rows that contains at least one NA value
my_data_v2 <- na.omit(my_data)
```

Descriptive analysis on ride_length (all figures in seconds)

```r
summary(my_data_v2$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3325.5   299.1   495.6   773.2   846.5 88690.1
```

Compare members and casual users

```r
aggregate(my_data_v2$ride_length ~ my_data_v2$member_casual, FUN = mean)
```

```
##   my_data_v2$member_casual my_data_v2$ride_length
## 1                   casual              1071.0313
## 2                   member               657.4732
```

```r
aggregate(my_data_v2$ride_length ~ my_data_v2$member_casual, FUN = median)
```

```
##   my_data_v2$member_casual my_data_v2$ride_length
## 1                   casual               585.3820
## 2                   member               465.0845
```

```r
aggregate(my_data_v2$ride_length ~ my_data_v2$member_casual, FUN = max)
```

```
##   my_data_v2$member_casual my_data_v2$ride_length
## 1                   casual               88690.11
## 2                   member               87475.87
```

```r
aggregate(my_data_v2$ride_length ~ my_data_v2$member_casual, FUN = min)
```

```
##   my_data_v2$member_casual my_data_v2$ride_length
## 1                   casual              -3286.839
## 2                   member              -3325.542
```

Analyze rider data by type and weekday

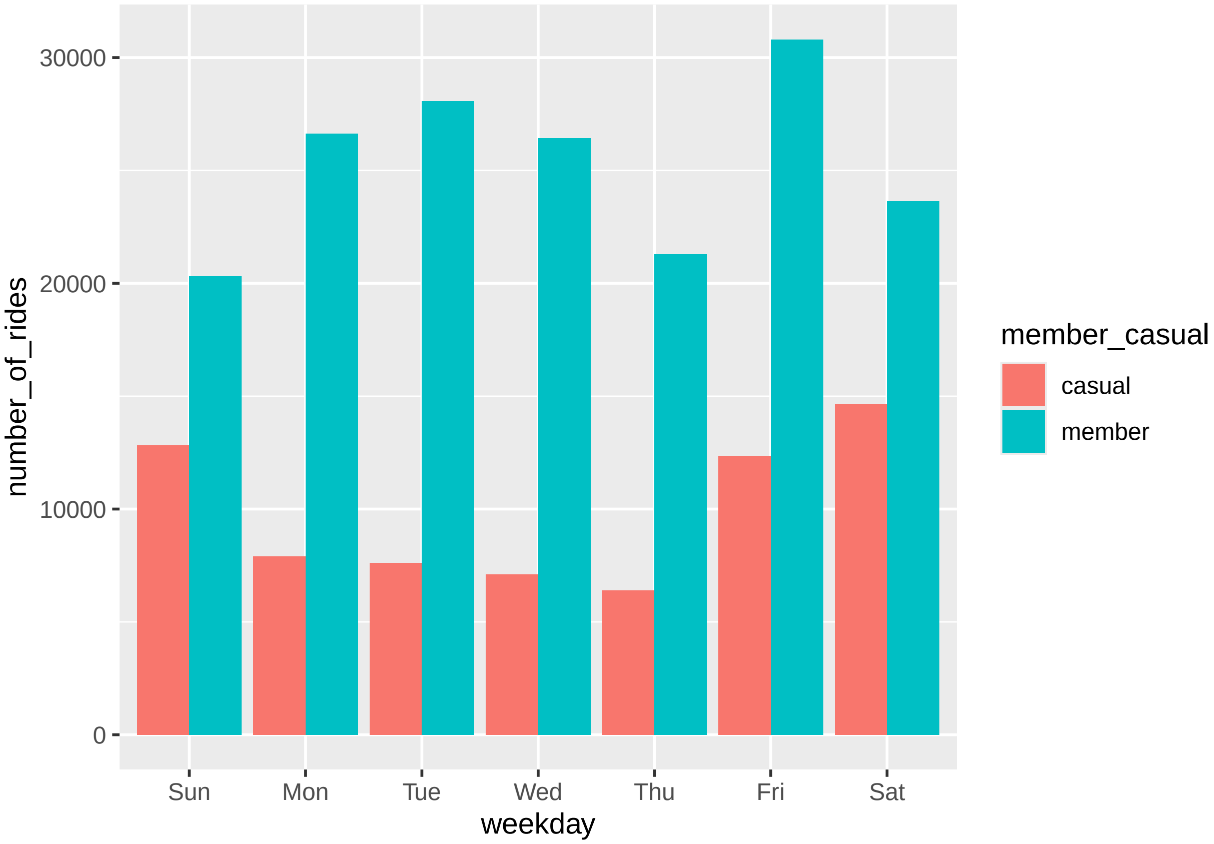
```r
my_data_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE))  %>% #creates weekday field using wday()
  group_by(member_casual, weekday) %>% #groups by user type and weekday
  summarise(
    number_of_rides = n(),
    average_duration = mean(ride_length) #calculates the number of rides and average duration
  ) %>%
  arrange(member_casual, weekday) # sorts
```

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun               12832            1279.
##  2 casual        Mon                7900             963.
##  3 casual        Tue                7609             854.
##  4 casual        Wed                7102             848.
##  5 casual        Thu                6390             879.
##  6 casual        Fri               12353            1049.
##  7 casual        Sat               14637            1271.
##  8 member        Sun               20316             691.
##  9 member        Mon               26621             621.
## 10 member        Tue               28062             631.
## 11 member        Wed               26430             637.
## 12 member        Thu               21289             670.
## 13 member        Fri               30800             659.
## 14 member        Sat               23630             711.
```

Let's visualize the number of rides by rider type

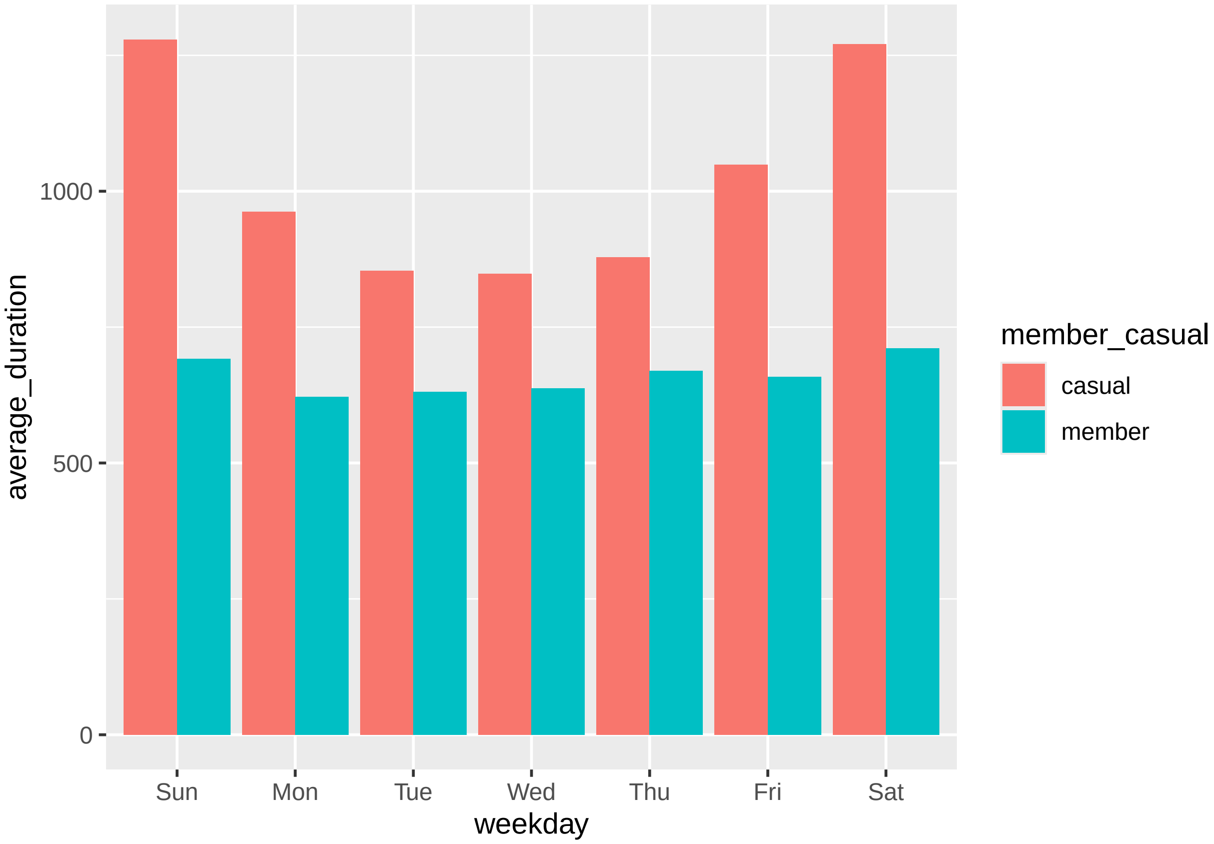
```
install.packages("ggplot2")
library(ggplot2)

my_data_v2 %>%
mutate(weekday = wday(started_at, label = TRUE)) %>%
group_by(member_casual, weekday) %>%
summarise(
  number_of_rides = n(),
  average_duration = mean(ride_length)
) %>%
arrange(member_casual, weekday) %>%
ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
geom_col(position = "dodge")
```



Let's create a visualization for the average duration

```
my_data_v2 %>%
mutate(weekday = wday(started_at, label = TRUE)) %>%
group_by(member_casual, weekday) %>%
summarise(number_of_rides = n(),
  average_duration = mean(ride_length)
) %>%
arrange(member_casual, weekday) %>%
ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
geom_col(position = "dodge")
```



Conclusion

Based on our findings, we can conclude that even though there are more member riders than casual, casual riders use our bikes, on average, for a longer period of time. This leads us to believe that casual riders rent our bikes when they go long distances. Casual riders ride more on weekends, whereas member riders ride more mid-week and on Fridays. Based on our findings, here are our top 3 recommendation on how to convert casual riders into members.

1. Weekend-Specific Discounts: Offer discounts on weekend passes or annual memberships to motivate casual riders to commit to a membership.
2. Weekday Discounts: Offer weekday-specific free ride credits during peak commuting hours as part of the membership to encourage casual riders to use the bikes during the week.
3. Mileage-Based Rewards: Implement a rewards program that rewards riders for the distance they cover.