Cyclistic Bike Share Analysis Q1 2019

Toochukwu Okutalukwe

2024-06-08

Table of content

Introduction

DATA SOURCE:

The dataset used for this analysis was obtained from the **Cyclistics** company database for the year 2019 provided by **Google/Coursera** to solve real-world case studies.
Here is the dataset for your reference Click Dataset

Purpose of the Project

The business task is to maximize the number of annual members by converting **casual riders** into **annual members**.

Problem Statement

The object is to compare the usage patterns of **annual members and casual riders** of Cyclistic bike-sharing service. By analyzing metrics like trip duration, frequency, and usage times, I aim to identify key factors that can help convert casual riders into annual members, thereby increasing annual memberships. This analysis will provide insights for strategies to boost membership conversion and retention.

Data Cleaning and Transformation

Setting up my environment

Note: Setting up my R environment by loading `tidyverse`, `skimr`, `janitor`, `dplyr`, `dplR`, `ggplot2`, `lubridate`, and `readr` packages

```r
library(tidyverse)
library(skimr)
library(janitor)
library(ggplot2)
library(lubridate)
library(dplyr)
library(dplR)
library(readxl)
library(readr)
```

Load the data using the readr function

Loading and Exploring the Data: Understand the structure and content of the dataset.

```
## Rows: 365069 Columns: 12
## -- Column specification ---------------------------------------------------------
## Delimiter: ","
## chr (6): start_time, end_time, from_station_name, to_station_name, usertype,...
## dbl (5): trip_id, bikeid, from_station_id, to_station_id, birthyear
## num (1): tripduration
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Explore the data using the `head` and `summary` function

```r
head(Cyclistic_data, n=10)
```

```
## # A tibble: 10 x 12
##     trip_id start_time    end_time      bikeid tripduration from_station_id
##       <dbl> <chr>         <chr>          <dbl>        <dbl>           <dbl>
##  1 21742443 1/1/2019 0:04 1/1/2019 0:11   2167          390             199
##  2 21742444 1/1/2019 0:08 1/1/2019 0:15   4386          441              44
##  3 21742445 1/1/2019 0:13 1/1/2019 0:27   1524          829              15
##  4 21742446 1/1/2019 0:13 1/1/2019 0:43    252         1783             123
##  5 21742447 1/1/2019 0:14 1/1/2019 0:20   1170          364             173
##  6 21742448 1/1/2019 0:15 1/1/2019 0:19   2437          216              98
##  7 21742449 1/1/2019 0:16 1/1/2019 0:19   2708          177              98
##  8 21742450 1/1/2019 0:18 1/1/2019 0:20   2796          100             211
##  9 21742451 1/1/2019 0:18 1/1/2019 0:47   6205         1727             150
## 10 21742452 1/1/2019 0:19 1/1/2019 0:24   3939          336             268
## # i 6 more variables: from_station_name <chr>, to_station_id <dbl>,
## #   to_station_name <chr>, usertype <chr>, gender <chr>, birthyear <dbl>
```

```r
summary(Cyclistic_data)
```

```
##     trip_id          start_time          end_time             bikeid
##  Min.   :21742443   Length:365069      Length:365069      Min.   :   1
##  1st Qu.:21848765   Class :character   Class :character   1st Qu.:1777
##  Median :21961829   Mode  :character   Mode  :character   Median :3489
##  Mean   :21960872                                         Mean   :3429
##  3rd Qu.:22071823                                         3rd Qu.:5157
##  Max.   :22178528                                         Max.   :6471
##   tripduration      from_station_id from_station_name  to_station_id
##  Min.   :      61   Min.   :  2.0   Length:365069      Min.   :  2.0  
##  1st Qu.:     326   1st Qu.: 76.0   Class  :character  1st Qu.: 76.0  
##  Median :     524   Median :170.0   Mode   :character  Median :168.0  
##  Mean   :    1016   Mean   :198.1                      Mean   :198.6  
##  3rd Qu.:     866   3rd Qu.:287.0                      3rd Qu.:287.0  
##  Max.   :10628400   Max.   :665.0                      Max.   :665.0  
##  to_station_name      usertype            gender            birthyear   
##  Length:365069      Length:365069      Length:365069      Min.   :1900  
##  Class :character   Class :character   Class :character   1st Qu.:1976  
##  Mode  :character   Mode  :character   Mode  :character   Median :1985  
##                                                           Mean   :1982  
##                                                           3rd Qu.:1991  
##                                                           Max.   :2003  
```

Clean the data

Note: Check for missing values. Here i found out there is an NA value at the **gender** column

```
colSums(is.na(Cyclistic_data))
```

```
##           trip_id        start_time          end_time            bikeid 
##                 0                 0                 0                 0 
##      tripduration   from_station_id from_station_name     to_station_id 
##                 0                 0                 0                 0 
##   to_station_name          usertype            gender         birthyear 
##                 0                 0             19711                 0 
```

Replace NA values with “Not applicable” in **gender** column

```
Cyclistic_data <- Cyclistic_data %>%
  mutate(gender = ifelse(is.na(gender), "Not applicable", gender))
```

Re-check for NA values

```
colSums(is.na(Cyclistic_data))
```

```
##           trip_id        start_time          end_time            bikeid 
##                 0                 0                 0                 0 
##      tripduration   from_station_id from_station_name     to_station_id 
##                 0                 0                 0                 0 
##   to_station_name          usertype            gender         birthyear 
##                 0                 0                 0                 0 
```

Convert date columns to date type

```
Cyclistic_data$start_time <- as.POSIXct(Cyclistic_data$start_time, format="%m/%d/%Y %H:%M")
Cyclistic_data$end_time <- as.POSIXct(Cyclistic_data$end_time, format="%m/%d/%Y %H:%M")
```

Transform the data

Note: I create new variables: **rider_length**, **day_of_week** and **hour_of_day**

```
Cyclistic_data <- Cyclistic_data %>%
  mutate(rider_length = as.numeric(difftime(end_time, start_time, units = "mins")),
         hour_of_day = hour(start_time),
         day_of_week = wday(start_time, label = TRUE),
         start_month = month(start_time, label = TRUE))
print(Cyclistic_data, n=10)
```

```
## # A tibble: 365,069 x 16
##     trip_id start_time          end_time            bikeid tripduration
##       <dbl> <dttm>              <dttm>               <dbl>        <dbl>
##  1 21742443 2019-01-01 00:04:00 2019-01-01 00:11:00   2167          390
##  2 21742444 2019-01-01 00:08:00 2019-01-01 00:15:00   4386          441
##  3 21742445 2019-01-01 00:13:00 2019-01-01 00:27:00   1524          829
##  4 21742446 2019-01-01 00:13:00 2019-01-01 00:43:00    252         1783
##  5 21742447 2019-01-01 00:14:00 2019-01-01 00:20:00   1170          364
##  6 21742448 2019-01-01 00:15:00 2019-01-01 00:19:00   2437          216
##  7 21742449 2019-01-01 00:16:00 2019-01-01 00:19:00   2708          177
##  8 21742450 2019-01-01 00:18:00 2019-01-01 00:20:00   2796          100
##  9 21742451 2019-01-01 00:18:00 2019-01-01 00:47:00   6205         1727
## 10 21742452 2019-01-01 00:19:00 2019-01-01 00:24:00   3939          336
## # i 365,059 more rows
## # i 11 more variables: from_station_id <dbl>, from_station_name <chr>,
## #   to_station_id <dbl>, to_station_name <chr>, usertype <chr>, gender <chr>,
## #   birthyear <dbl>, rider_length <dbl>, hour_of_day <int>, day_of_week <ord>,
## #   start_month <ord>
```

Analysis Breakdown

Statistical analysis: I generated descriptive statistics and look for interesting patterns.

```
summary(Cyclistic_data$rider_length)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
##      1.00      5.00      9.00     16.94     14.00 177200.00
```

```
summary(Cyclistic_data$day_of_week)
```

```
##   Sun   Mon   Tue   Wed   Thu   Fri   Sat 
## 27999 50399 61005 60414 66903 63047 35302
```

```
summary(Cyclistic_data$hour_of_day)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00    9.00   14.00   13.34   17.00   23.00
```

```
summary(Cyclistic_data$start_month, 4)
```

```
##     Mar     Jan     Feb (Other) 
##  165611  103272   96186       0
```

Top 5 start stations

```
top_start_stations <- Cyclistic_data %>%
  count(from_station_name, sort = TRUE) %>%
  top_n(5)
```

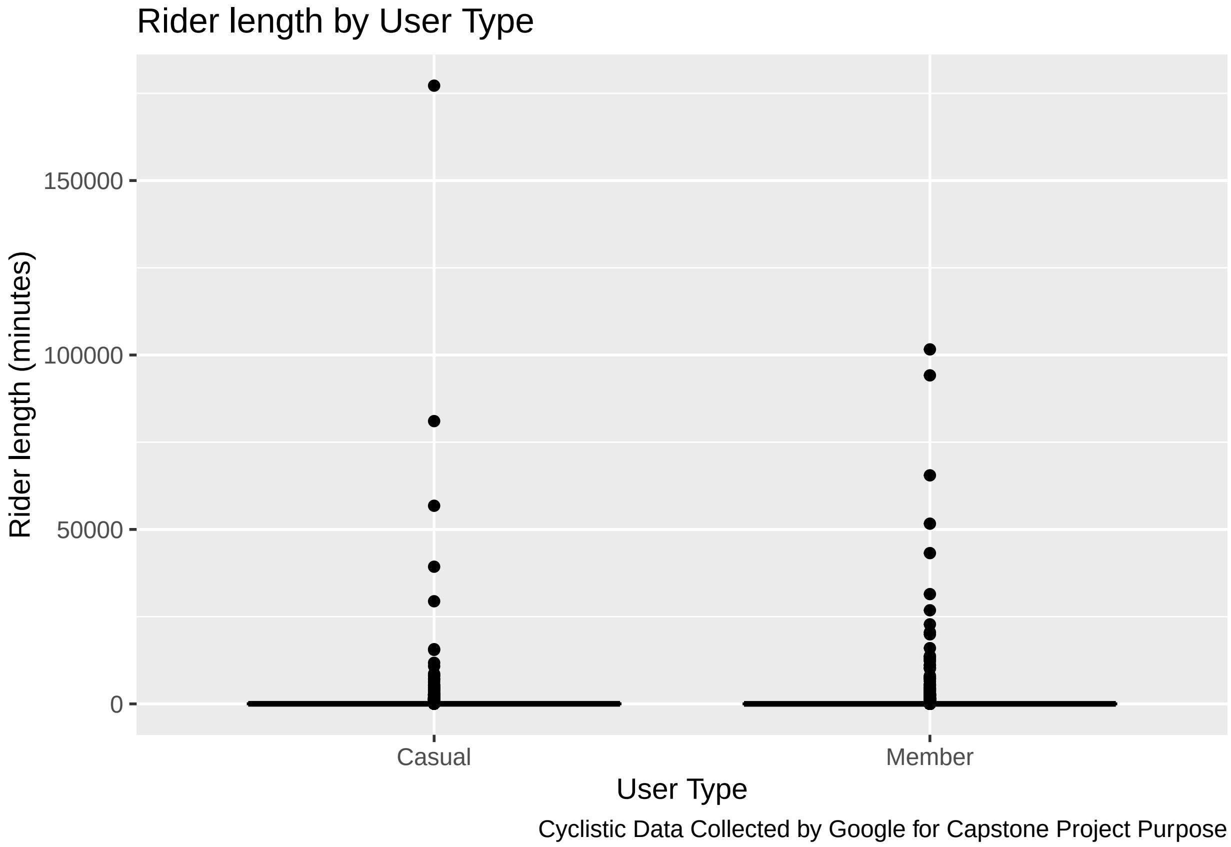
```
## Selecting by n
```

```
top_start_stations
```

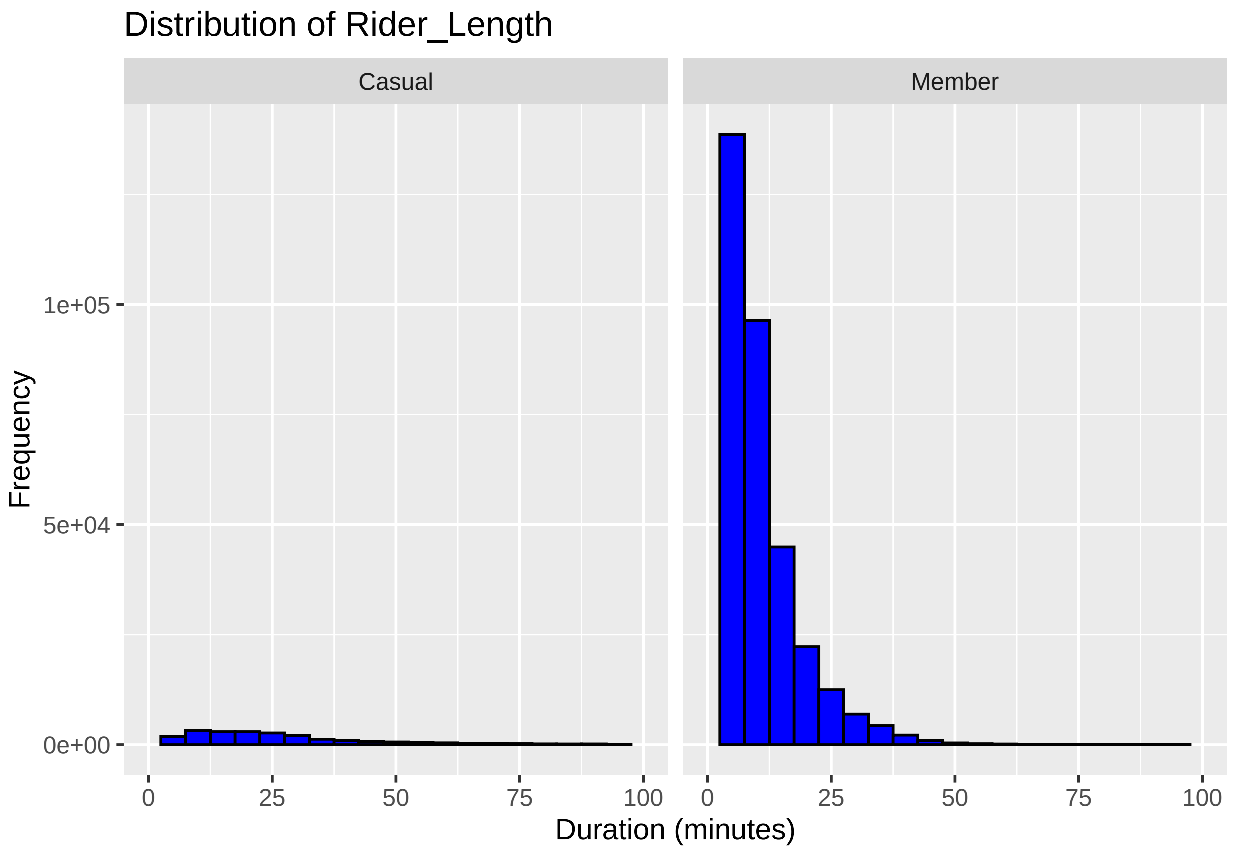
```
## # A tibble: 5 x 2
##   from_station_name                n
##   <chr>                        <int>
## 1 Clinton St & Washington Blvd  7699
## 2 Clinton St & Madison St       6565
## 3 Canal St & Adams St           6342
## 4 Columbus Dr & Randolph St     4655
## 5 Canal St & Madison St         4571
```

Create visuals

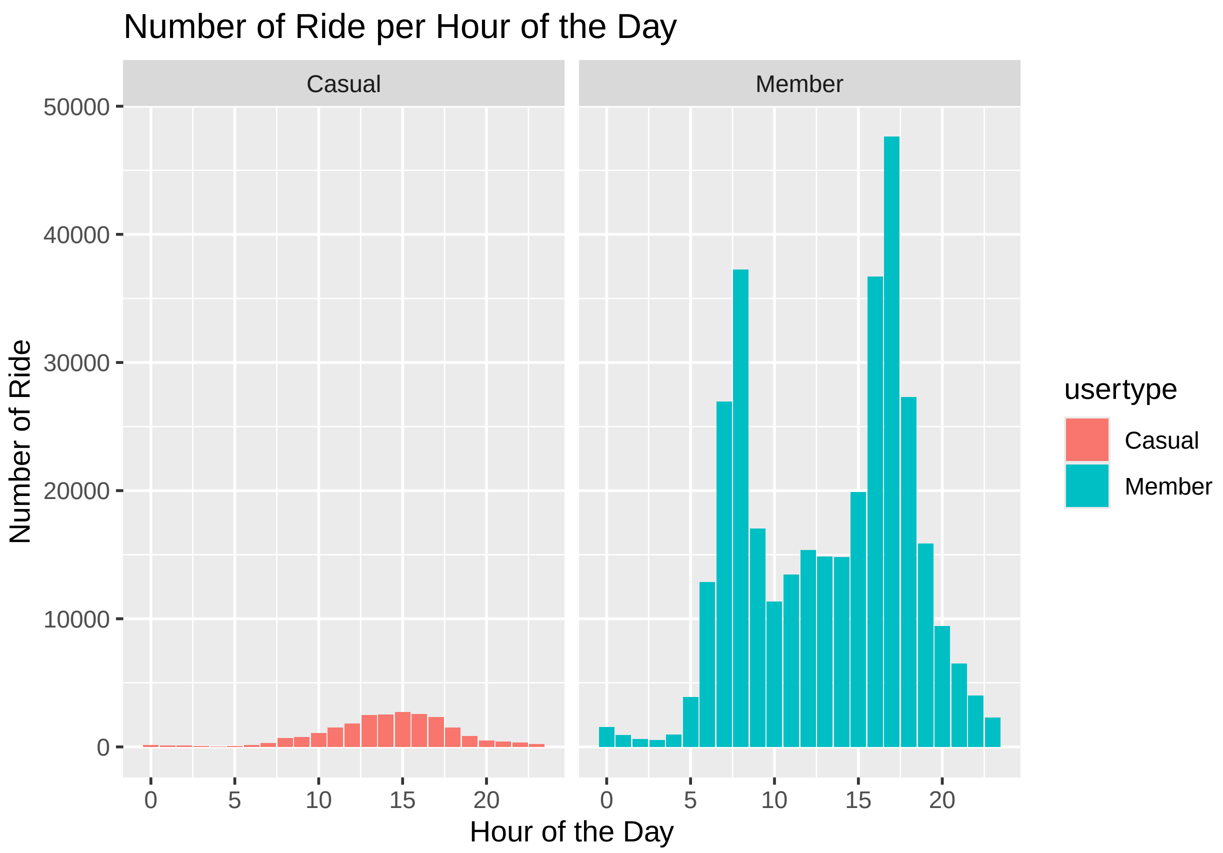
Rider length by user type: This plot compares trip durations across user types, revealing that casual riders generally have longer trip durations compared to members.



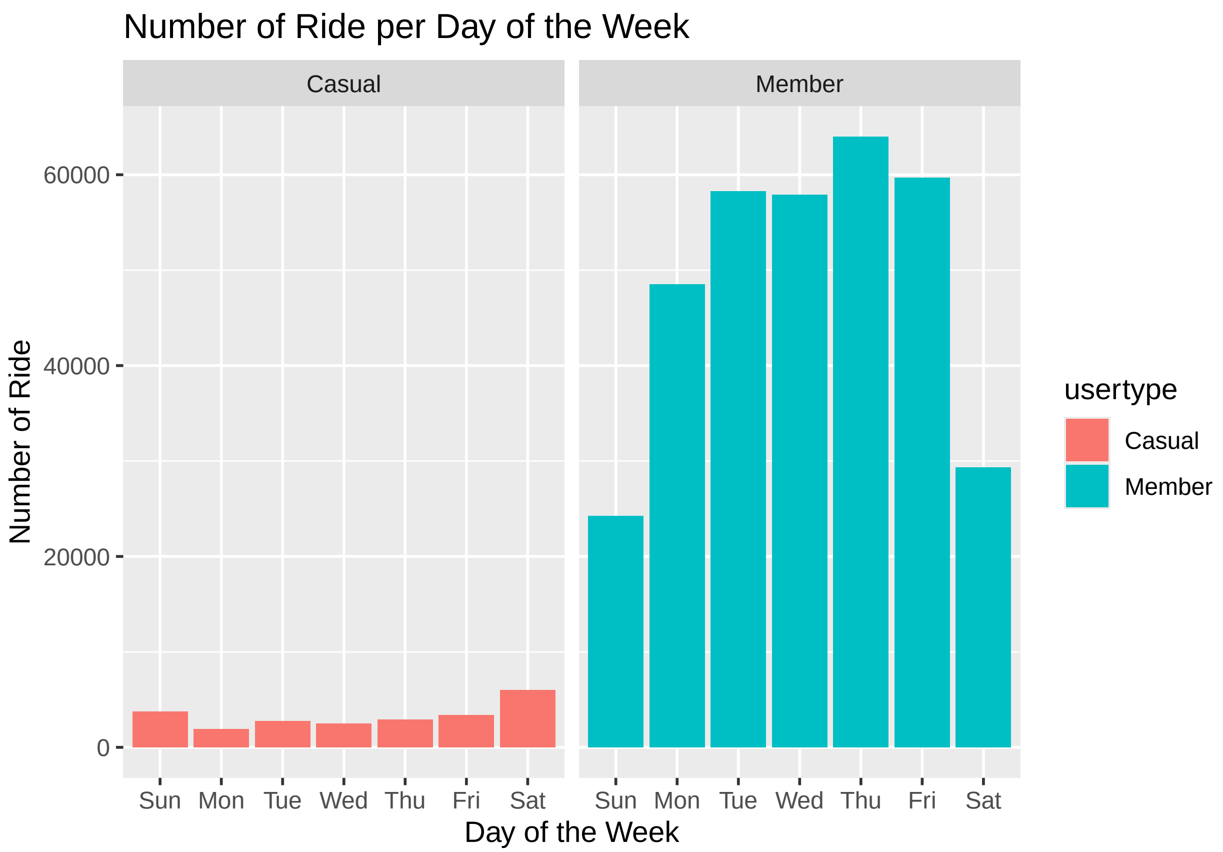
Distribution of rider_length: This histogram displays the distribution of trip durations, indicating that the majority of trips for both user types are short, with most lasting less than 30 minutes.



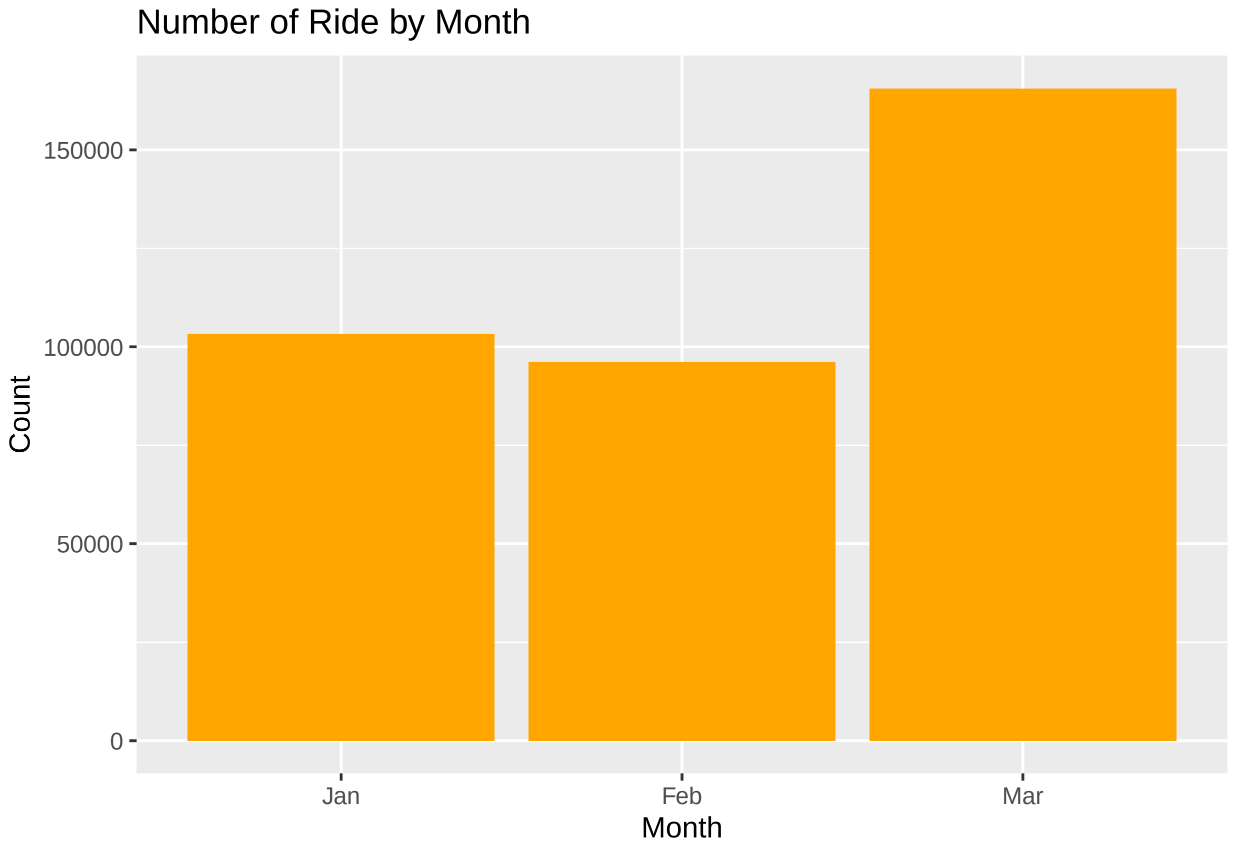
Number of ride per hour of the day: This plot illustrates the peak usage hours for riders, showing a higher frequency of trips during afternoon and evening hours for **casual riders**, whereas **members** exhibit peak usage in the morning and evening.



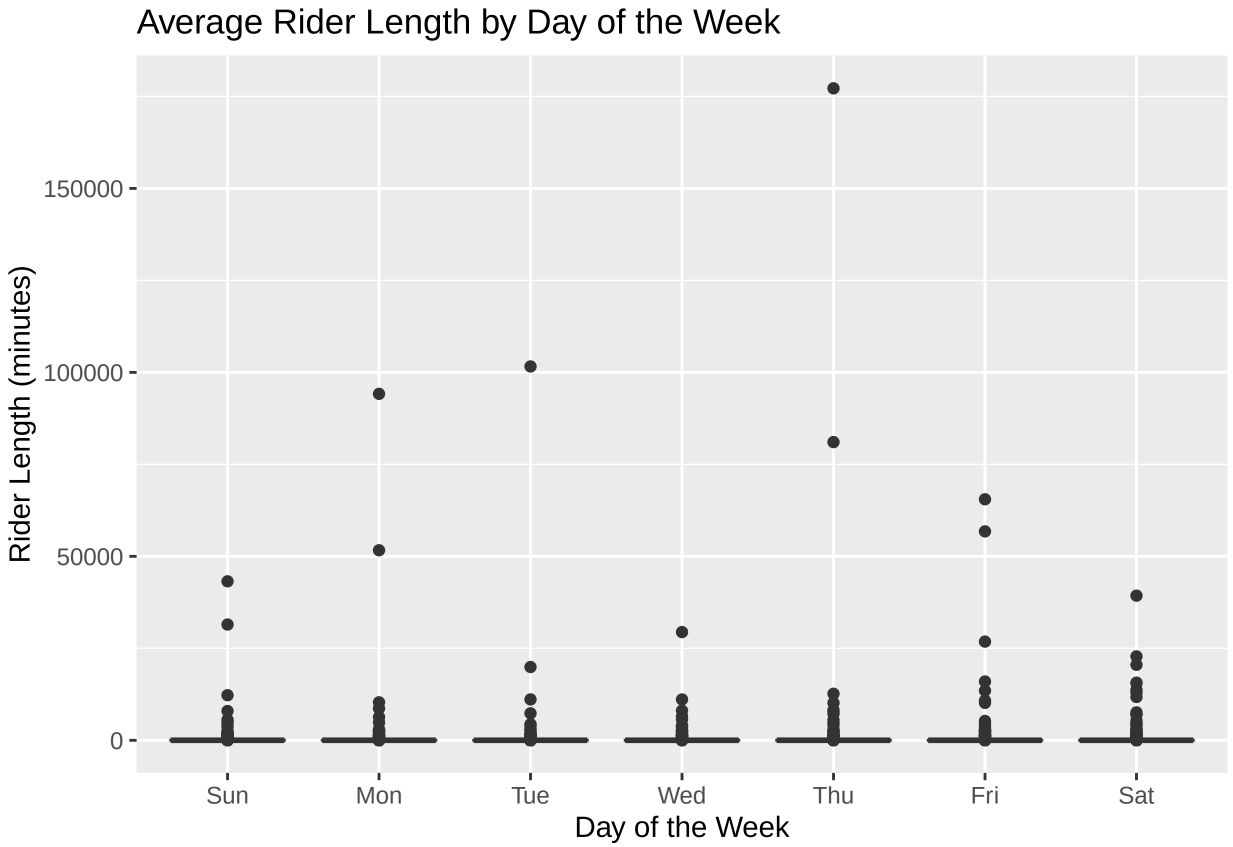
Number of ride per day of the week: This bar chart depicts the weekly riding frequency by user type, highlighting that **casual riders** predominantly ride on Saturdays and Sundays, while **members** have a higher riding frequency on weekdays.



Number of ride by month: This bar chart illustrates a seasonal trend for the first quartile, showing a decline in rider activity in **February** and an increment in **March**.



Average ride duration by day of the week: This plot presents the average trip duration by day of the week, indicating that Thursday has the highest average trip duration.



Summary

Analysis of trip durations and riding patterns reveals distinct user behavior between casual riders and members. While short trips (under 30 minutes) dominate both groups, casual riders demonstrate a propensity for longer journeys. Peak usage times further differentiate the user base. Casual riders exhibit a preference for afternoons and evenings, whereas members favor mornings and evenings. Weekly riding patterns diverge as well, with casual riders gravitating towards weekends, especially Saturdays and Sundays. Conversely, members tend to ride more frequently on weekdays. Seasonal trends highlight a decrease in activity during February, followed by a subsequent rise in March. Notably, average trip durations reach a peak on Thursdays, suggesting specific mid-week riding habits. These insights present valuable opportunities for targeted service and engagement strategies to enhance user satisfaction and optimize operational efficiency for both casual riders and members.

Recommendation

A Data-Driven Approach to Converting Casual Riders into Annual Members
This report outlines a multi-faceted marketing strategy designed to leverage user data and convert casual riders into annual members. Our analysis of trip durations and riding patterns has yielded several key insights that inform these targeted initiatives:

- **Cost Savings for Frequent Riders**: We have observed that casual riders often take longer trips. Capitalizing on this data point, we will emphasize the significant cost benefits an annual membership offers for frequent long-distance riders. Comparative cost analyses will be presented to clearly demonstrate potential savings.

- **Peak Hour Promotions**: Our data suggests casual riders primarily utilize the service during afternoon and evening hours. To capture their attention during peak usage times, promotions and advertisements for membership plans will be strategically placed during these periods.
- **Weekend Membership Campaigns**: Casual riders exhibit a clear preference for riding on Saturdays and Sundays. Weekend-only promotions or special offers for annual memberships will be launched to capitalize on this trend and entice riders during their peak usage days.
- **Seasonal Incentives**: Recognizing a decline in rider activity during February and a subsequent increase in March, we propose offering limited-time discounts on annual memberships at the end of winter. This incentivizes casual riders to commit to a membership as the weather improves and riding frequency rises.
- **Promoting Weekday Benefits**: As members tend to ride more on weekdays, we will highlight the advantages of membership for those considering increased riding during the workweek. Campaigns showcasing the convenience and cost savings for regular commuters will be implemented.
- **Leveraging High Trip Duration Days**: With Thursdays exhibiting the highest average trip duration, we will create targeted campaigns specifically tailored to this day. “Thursday Membership Deals” will incentivize casual riders taking longer trips to convert to annual memberships.
- **The Power of Social Proof**: Customer testimonials and success stories from current members who transitioned from being casual riders will be shared. These narratives will highlight member satisfaction with the program’s benefits, emphasizing cost savings, convenience, and overall value.

By implementing these data-driven strategies, the company can effectively target casual riders, addressing their specific usage patterns and motivations. This comprehensive approach will ultimately lead to a significant increase in annual membership conversions.