Analysis on Cyclistic Data

CHANDRU.M

2024-07-03

Introduction

This documentation is done for the Capstone Project given in Google Data Analytics 2023. The objective is to provide conclusion using analysis on the given scenario in order to convert casuals into members.

Scenario

You are a junior data analyst working on the marketing analyst team at Cyclistic, a bike-share company in Chicago. The director of marketing believes the company's future success depends on maximizing the number of annual memberships. Therefore, your team wants to understand how casual riders and annual members use Cyclistic bikes differently. From these insights, your team will design a new marketing strategy to convert casual riders into annual members. But first, Cyclistic executives must approve your recommendations, so they must be backed up with compelling data insights and professional data visualizations.

Data

The data is chosen from [here] (https://divvy-tripdata.s3.amazonaws.com/index.html (https://divvy-tripdata.s3.amazonaws.com/index.html)). Recent 12 months data are chosen. Data follows ROCCC (Reliable, Original, Comprehensive, Current, and Cited).

Data Processing

Data from the source is downloaded. R code is written to precise the data for analysis and all 12 csv files are combined for plotting.The following lines are codes and are excluded from code chunks (code blocks) to avoid complexity.

Code Starts

library(readr) library(dplyr) library(lubridate) library(ggplot2) csv_files <- list.files(path = "C:\Users\sys\Desktop\Track A\Copy", pattern = "*.csv", full.names = TRUE) ## Files are extracted dir.create("C:\Users\sys\Desktop\Track A", recursive = TRUE, showWarnings = FALSE) columns_to_remove <- c(1,5,6,7,8,9,10,11,12) # Columns to be removed for (file in csv_files) { b = read_csv(file) b = b[, -columns_to_remove] $bstart_date = as.Date(b$started_at) #Starting Date $bstart_time = format(b$started_at,"%H:%M:%S") #Starting Time b

started_at = NULL #Coulmn Strated_at is removed
b

$end_date = as.Date(b$ended_at) $bend_time = format(b$ended_at,"

b ended_at = NULL #As the type of time varibale is "doube", it is changed to type "POSIXct" to find the difference between starting and ending. time1_posix = as.POSIXct(b

$start_time, format ="

$time2_posix = as.POSIXct(b$end_time, format = "%H:%M:%S") #A day(86400 secs) is added to end_time if it is lesser thanstart_time. time2_posix[time2_posix < time1_posix] = time2_posix[time2_posix < time1_posix] + 86400 b

$usage_time = difftime(time2_posix, time1_posix, units = " mins ")$*b$start_day = weekdays(b$start_date) # Day is calculated output_file = file.path("C:\Users\sys\Desktop\Track A", basename(file)) write_csv(b, path = output_file) #Saved in the provided path } csv_files <- list.files(path = "C:\Users\sys\Desktop\Track A", pattern = "*.csv", full.names = TRUE) #Files are combined and stored into a single file as combined_data <- lapply(csv_files, read_csv) %>%bind_rows() write_csv(combined_data, "C:\Users\sys\Desktop\Track A\final_data.csv") ### Code Ends

Plots

Visualization of the processed data is made to draw conclusions.

```
library(tibble)
library(readr)
a = read_csv("C:\\Users\\sys\\Desktop\\Track A\\final_data.csv")
```

```
## Rows: 5743278 Columns: 8
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (3): rideable_type, member_casual, start_day
## dbl  (1): usage_time
## date (2): start_date, end_date
## time (2): start_time, end_time
##
## 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.
```

```
# Reading the csv file
as_tibble(a)
```

```
## # A tibble: 5,743,278 × 8
##    rideable_type member_casual start_date start_time end_date   end_time
##    <chr>         <chr>         <date>     <time>     <date>     <time>
##  1 electric_bike member        2023-06-05 13:34:12   2023-06-05 14:31:56
##  2 electric_bike member        2023-06-05 01:30:22   2023-06-05 01:33:06
##  3 electric_bike member        2023-06-20 18:15:49   2023-06-20 18:32:05
##  4 electric_bike member        2023-06-19 14:56:00   2023-06-19 15:00:35
##  5 electric_bike member        2023-06-19 15:03:34   2023-06-19 15:07:16
##  6 electric_bike member        2023-06-09 21:30:25   2023-06-09 21:49:52
##  7 electric_bike member        2023-06-03 13:34:09   2023-06-03 13:34:28
##  8 electric_bike member        2023-06-03 13:34:46   2023-06-03 13:35:00
##  9 electric_bike member        2023-06-02 22:27:35   2023-06-02 22:35:26
## 10 electric_bike member        2023-06-02 21:18:31   2023-06-03 01:27:19
## # ℹ 5,743,268 more rows
## # ℹ 2 more variables: usage_time <dbl>, start_day <chr>
```

First 10 rows are displayed to show how the data looks after data processing ## Total Users by User Type

```
library(dplyr)
```

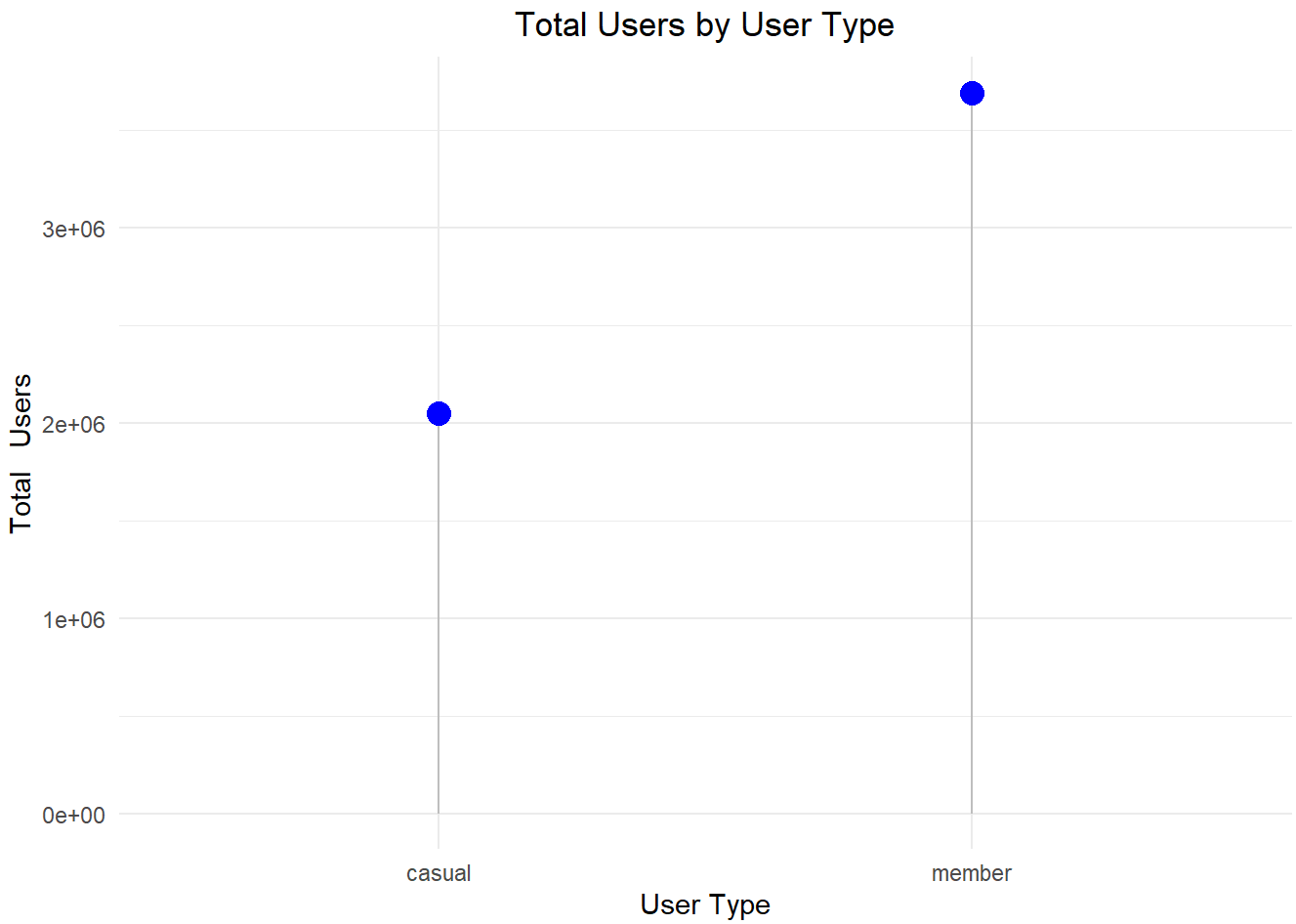
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)

a %>%
  group_by(member_casual) %>%
  summarise(total = n(), .groups = 'drop') %>%
  ggplot(aes(x = member_casual, y = total)) +
  geom_segment(aes(x = member_casual, xend = member_casual, y = 0,
  yend = total), color = "grey") +geom_point(size = 4, color = "blue") +
  labs(title = "Total Users by User Type", x = "User Type",   y = "Total   Users") + theme_minim
al() +
  theme(plot.title = element_text(hjust = 0.5))
```

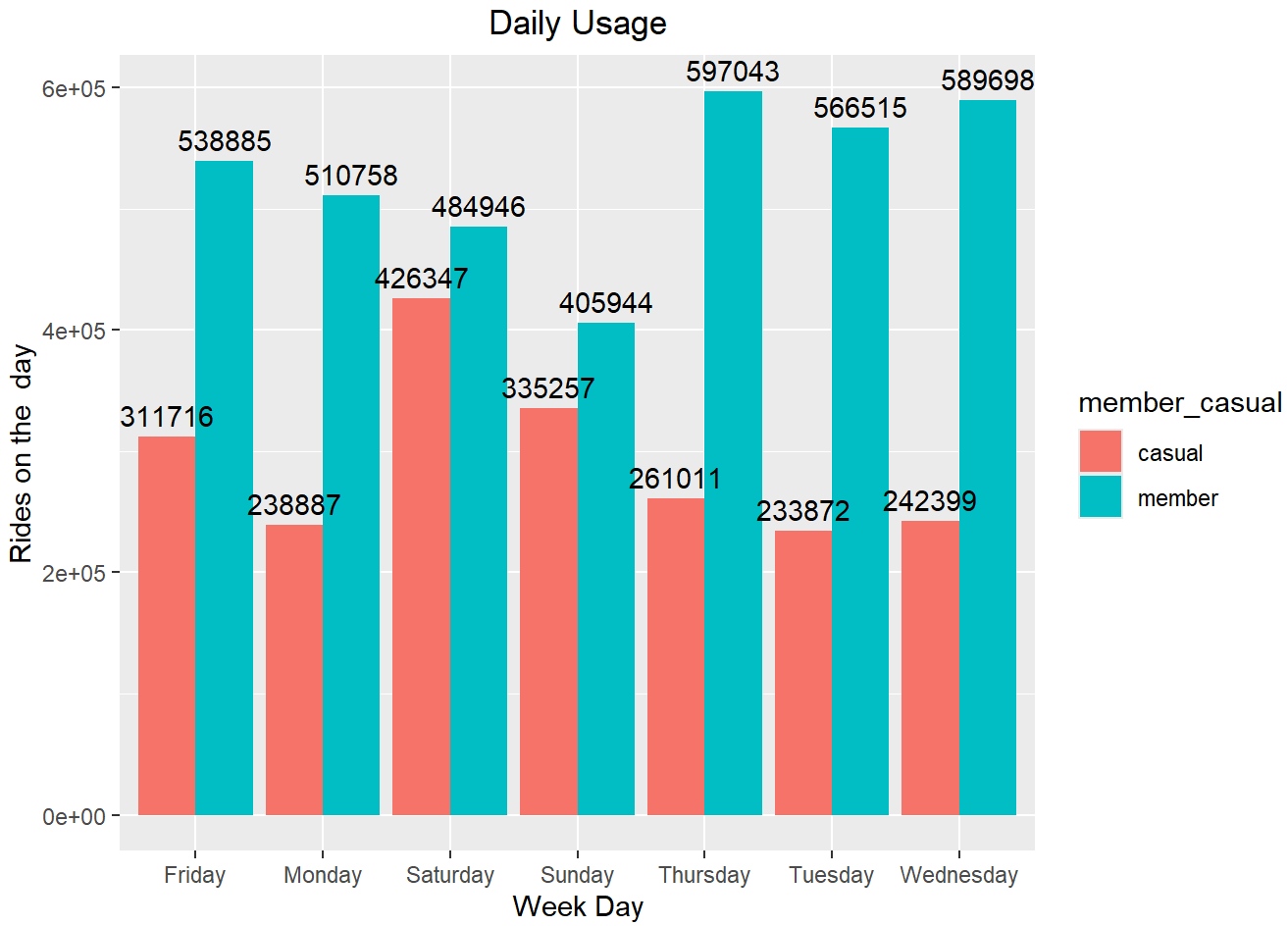


There are more number of members using bikes.Still, there are about 2.5 million casual users using bikes. So, making them member will be profitable for the company.

Total Users By Days

```
a %>% group_by(member_casual,start_day) %>%
  summarise(total_users = n()) %>%
  ggplot(aes(x = start_day,y= total_users, fill = member_casual)) +
  geom_col(position = "dodge") + labs(x = "Week Day",
  y ="Rides on the  day", title ="Daily Usage") +  geom_text(aes(
  label = total_users),
  position = position_dodge(width = 0.9) , vjust = -0.5) +
  theme(plot.title = element_text(hjust = 0.5))
```

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

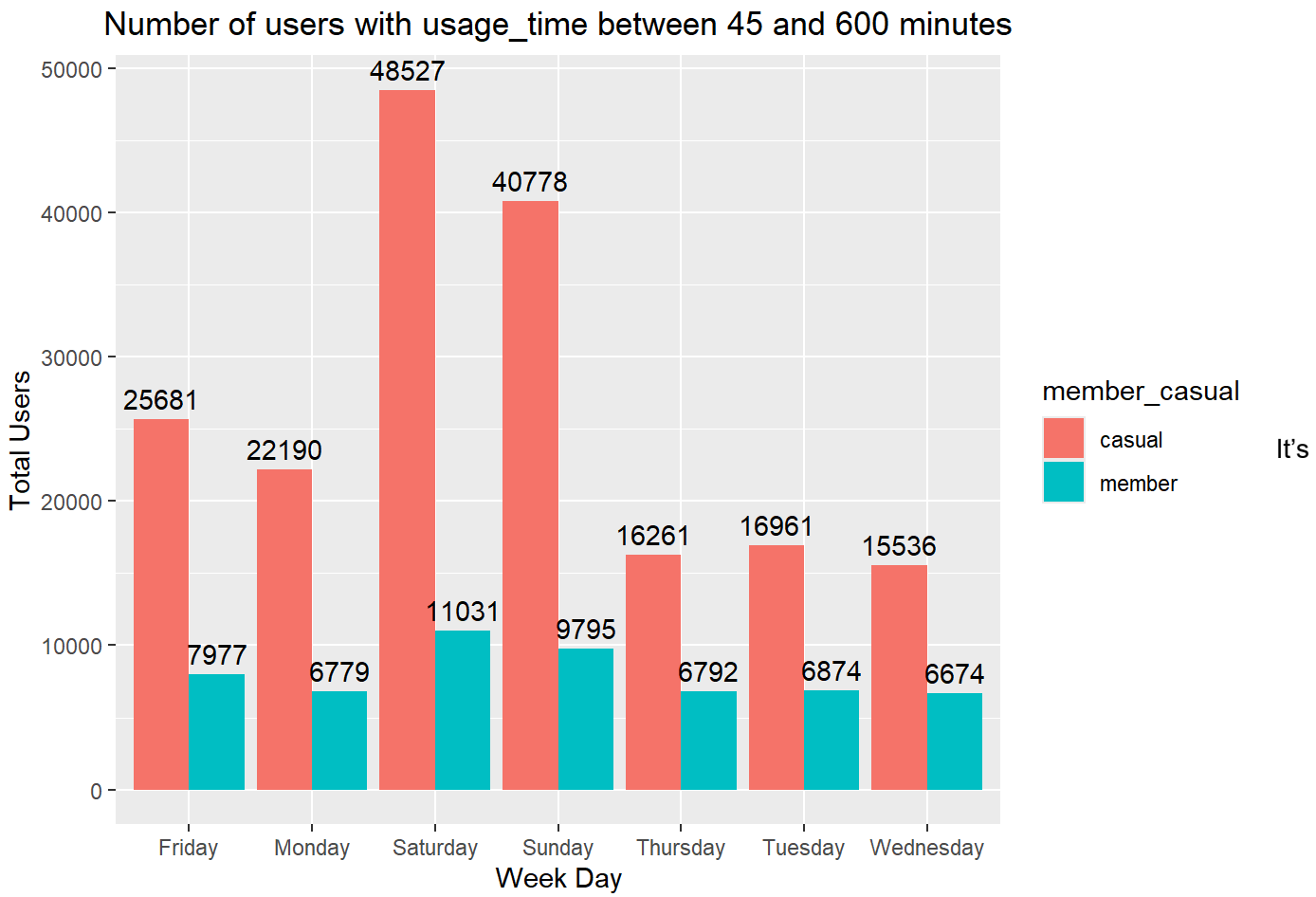


Members are the leading users of bikes. But during weekends, casuals are also looking for bikes as much as members do.

Users with usage_time between 45 and 600 minutes

```
a %>% filter(round(usage_time) > 45 & round(usage_time) < 600) %>%
  group_by(member_casual, start_day) %>%
  summarise(num = n()) %>% ggplot(aes(x=start_day, y=num,
  fill = member_casual )) + geom_col(position = "dodge") +
  geom_text(aes(label = num), position = position_dodge(width = 0.9)
  , vjust = -0.5) + labs(x = "Week Day", y ="Total Users",
  title ="Number of users with usage_time between 45 and 600 minutes") +   theme(plot.title = el
ement_text(hjust = 0.5))
```

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

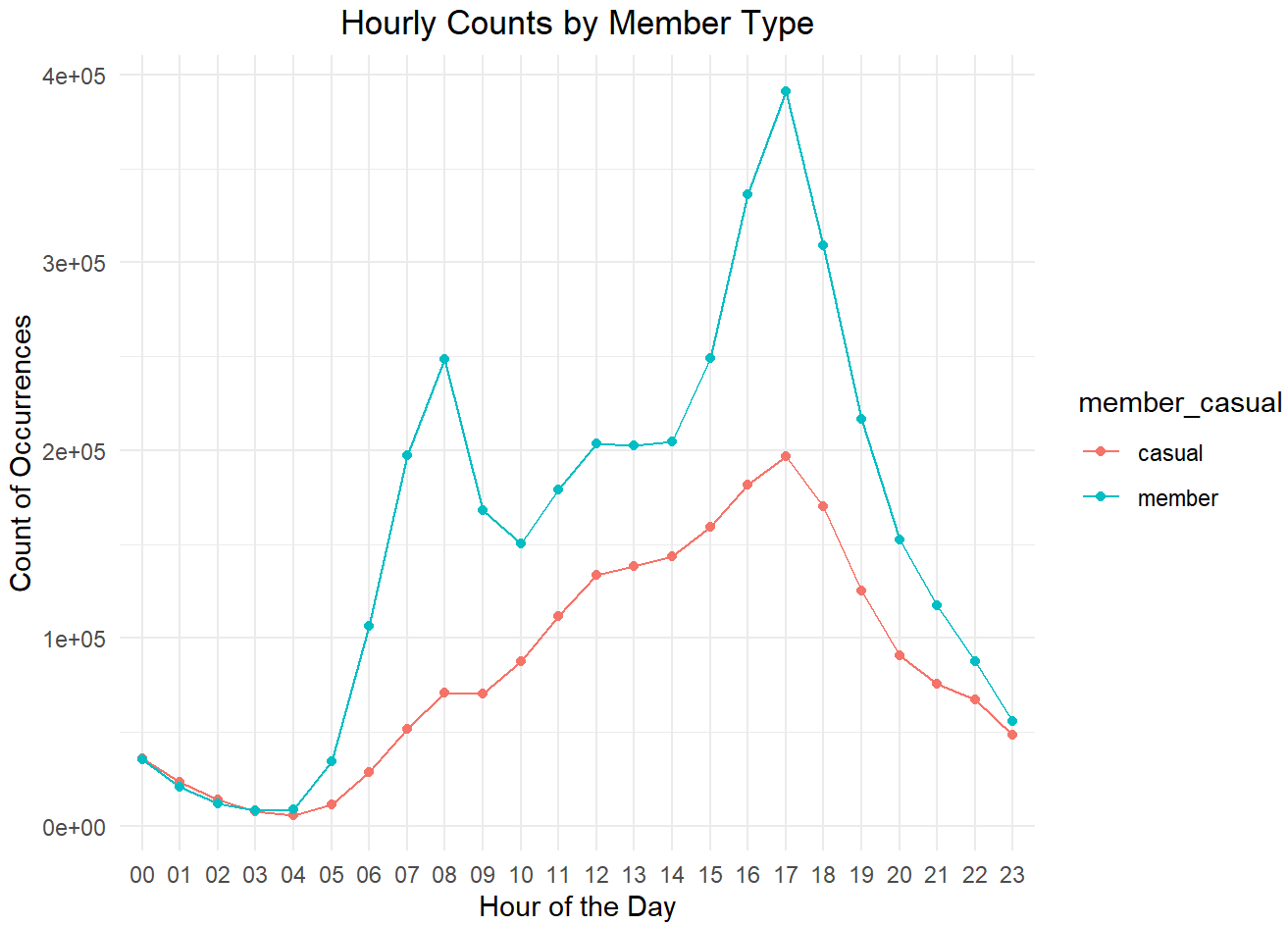


It's clear that, casuals are using bikes in higher range with an usage_time between 45 and 600 minutes. New membership schemes can be implemented based on hours.

Usage Hour

```
a %>% mutate(start_time = as.POSIXct(start_time, format="%Y-%m-%d %H:%M:%S"),
hour = format(start_time, "%H")) %>% group_by(member_casual,hour) %>%
summarise( count = n()) %>% ggplot( aes(x = hour, y = count, group = member_casual, color = memb
er_casual)) +geom_line() +geom_point() +
labs(title = "Hourly Counts by Member Type",x = "Hour of the Day",
y = "Count of Occurrences") + theme_minimal() +
theme(plot.title = element_text(hjust = 0.5))
```

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



People are showing interests to make use of bikes during day time. So, membership scheme based on hours must be a good choice.

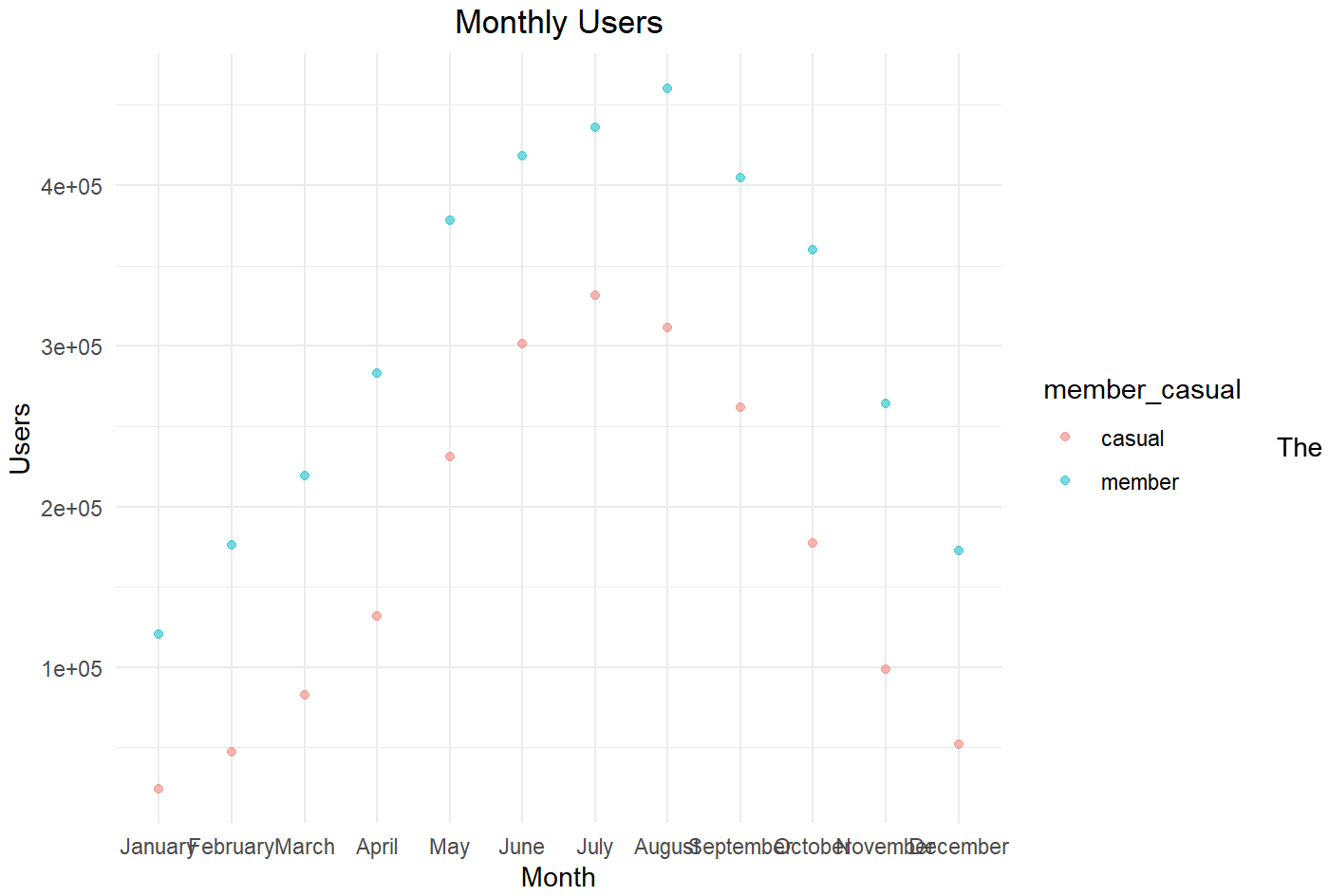
Monthly Users

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
a %>% mutate(mon = month(start_date,label = TRUE, abbr = FALSE)) %>% group_by(member_casual, mo
n) %>% summarise(total = n(),.groups = 'drop') %>% ggplot(aes(x= mon, y=total,group = member_cas
ual, color = member_casual)) + geom_point(alpha = 0.5) +labs(title = "Monthly Users",x = "Mont
h",y = "Users") +theme_minimal() +
theme(plot.title = element_text(hjust = 0.5))
```

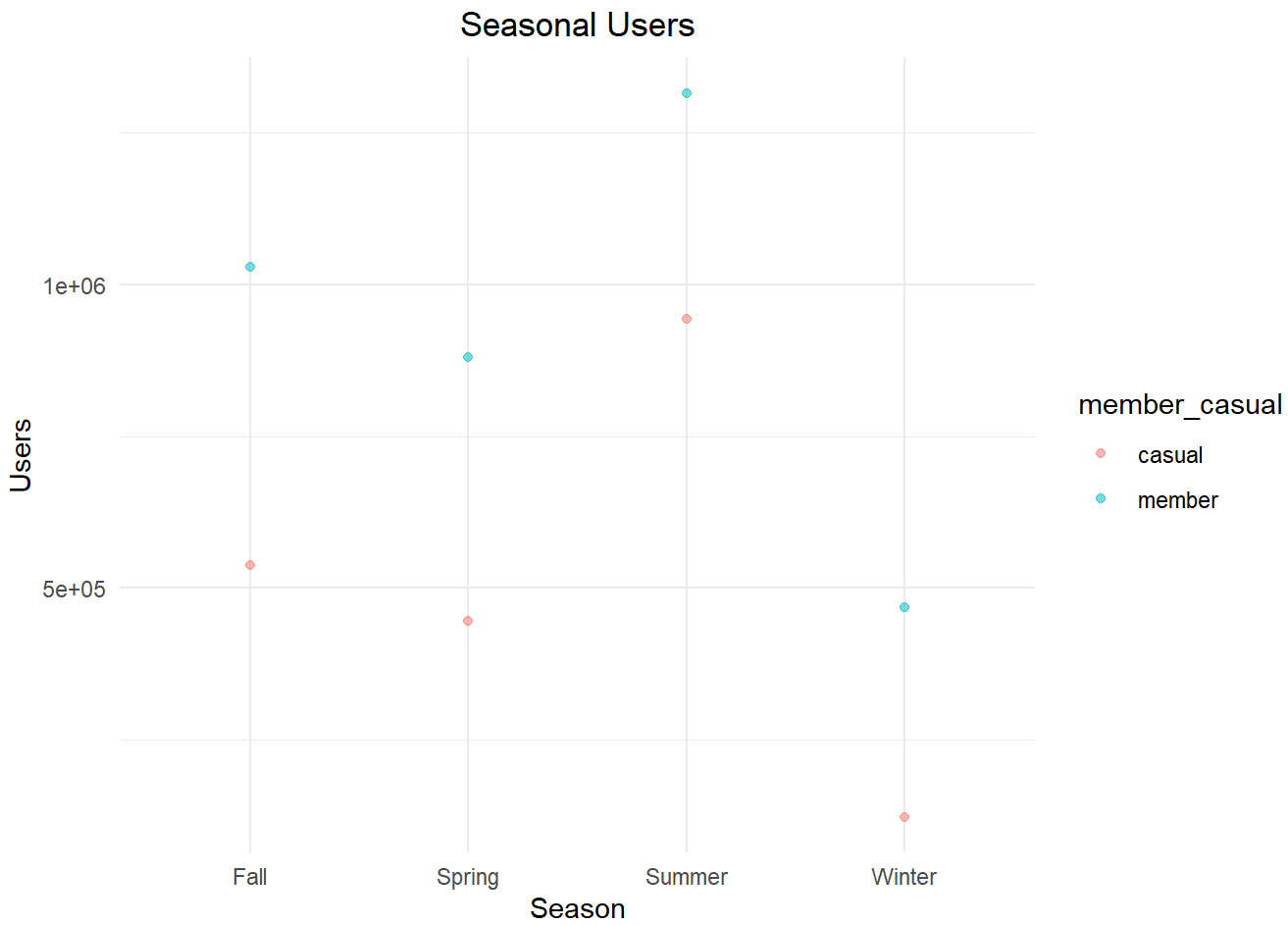


The numbers highly vary among months. So, monthly memberships will play a vital role in attracting casual riders.

Seasonal Users

```
get_season <- function(date) {
  month <- month(date)
  if (month %in% c(12, 1, 2)) {
    return("Winter")
  } else if (month %in% c(3, 4, 5)) {
    return("Spring")
  } else if (month %in% c(6, 7, 8)) {
    return("Summer")
  } else if (month %in% c(9, 10, 11)) {
    return("Fall")
  }
}

a %>%
  mutate(season = sapply(start_date, get_season)) %>%
  group_by(member_casual, season) %>%
  summarise(total = n(), .groups = 'drop') %>%
  ggplot(aes(x = season, y = total, group = member_casual,
  color = member_casual)) + geom_point(alpha = 0.5) +
  labs(title = "Seasonal Users", x = "Season", y = "Users") +
  theme_minimal() + theme(plot.title = element_text(hjust = 0.5))
```



There is a drop in the usage of bikes during Winter season. So annual membership may not be an interest for the casuals. Providing them seasonal pass will be an added advantage for the company

Conclusions

1. High Volume of Casual Riders: The analysis reveals that there are 2.5 million casual riders. This substantial user base represents a significant opportunity for conversion to membership plans.
2. Peak Usage on Weekends: Data indicates that weekends experience the highest usage rates. This trend suggests the potential for introducing a weekend-specific membership plan to cater to this demand. Targeted Annual Membership Plans:
3. Based on hourly usage patterns, two distinct annual membership options can be introduced:

- Option 1: For users with ride durations less than one hour.
- Option 2: For users with ride durations up to five hours.

4. Flexible Membership Options: To attract more casual riders, the introduction of monthly and seasonal memberships is recommended.

These flexible plans can serve as an entry point for casual users to transition to more committed membership plans.