Cyclistic Case Study

Nikitha Anto

2022-07-07

Cyclistic Bike Share Casestudy

Objective is to identify the differences in the usage of bikes between Casual and Member users

##Install Tidyverse, lubridate and ggplot2

#Prepare Data ##Upload four datasets for quarterly data of cyclistic bike trip data

```
quarter2_2019 <- read_csv("C:/Users/ajumo/OneDrive/Desktop/cyclistic_data/Divvy_Trips_2019_Q2.csv")
```

```
## Rows: 1048575 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): 01 - Rental Details Rental ID, 03 - Rental Start Station Name, 02 ...
## dbl  (5): 01 - Rental Details Bike ID, 01 - Rental Details Duration In Secon...
## dttm (2): 01 - Rental Details Local Start Time, 01 - Rental Details Local En...
##
## 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.
```

```
quarter3_2019 <- read_csv("C:/Users/ajumo/OneDrive/Desktop/cyclistic_data/Divvy_Trips_2019_Q3.csv")
```

```
## Rows: 1640718 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): from_station_name, to_station_name, usertype, gender
## dbl  (5): trip_id, bikeid, from_station_id, to_station_id, birthyear
## dttm (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.
```

```
quarter4_2019 <- read_csv("C:/Users/ajumo/OneDrive/Desktop/cyclistic_data/Divvy_Trips_2019_Q4.csv")
```

```
## Rows: 704054 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): from_station_name, to_station_name, usertype, gender
```

```
## dbl  (5): trip_id, bikeid, from_station_id, to_station_id, birthyear
## dttm (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.
```

```
quarter1_2020 <- read_csv("C:/Users/ajumo/OneDrive/Desktop/cyclistic_data/Divvy_Trips_2020_Q1.csv")
```

```
## Rows: 426887 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## dttm (2): started_at, ended_at
##
## 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.
```

##Rename column names of quarter2_2019, quarter3_2019 and quarter4_2019 to make it consistent with quarter1_2020

##Change datatype of ride_id & rideable_type of quarter2_2019, quarter3_2019 & quarter4_2019 consistent to quarter1_2020

```
quarter4_2019 <- mutate(quarter4_2019, ride_id = as.character(ride_id),
                        rideable_type = as.character(rideable_type))
quarter3_2019 <- mutate(quarter3_2019, ride_id = as.character(ride_id),
                        rideable_type = as.character(rideable_type))
quarter2_2019 <- mutate(quarter2_2019, ride_id = as.character(ride_id),
                        rideable_type = as.character(rideable_type))
```

##Combine all 4 dataframes to one dataframe

```
trip_data <- bind_rows(quarter2_2019, quarter3_2019, quarter4_2019, quarter1_2020)
```

#Clean data ##Remove unwanted columns in trip_data

```
trip_data <- trip_data %>% select(-c("01 - Rental Details Duration In Seconds Uncapped"
                                     ,"Member Gender","05 - Member Details Member Birthday Year"
                                     ,"tripduration","gender","birthyear","start_lat","start_lng"
                                     ,"end_lat","end_lng"))
```

##Change values of member_casual column to make it consistent with quarter1_2020

##Introduce new columns that show date, day, month and year for each trip

```
trip_data$date <- as.Date(trip_data$started_at)
trip_data$months <- format(as.Date(trip_data$date), "%m")
trip_data$day <- format(as.Date(trip_data$date), "%d")
trip_data$year <- format(as.Date(trip_data$date), "%Y")
trip_data$day_of_week <- format(as.Date(trip_data$date), "%A")
```

##Add new column called ride_duration to trip_data

```
trip_data$ride_duration <- difftime(trip_data$ended_at,trip_data$started_at)
```

##Convert datatype of ride_duration column to numeric

```
trip_data$ride_duration <- as.numeric(as.character(trip_data$ride_duration))
```

##Remove rows with negative data from ride_duration column

```
bike_trip_data <- trip_data[!(trip_data$start_station_name == "HQ QR" | trip_data$ride_duration<0),]
```

#Analyse data ##Do calculations on data to find out mean, median,max and min of ride duration

```
bike_trip_data_v1 <- na.omit(bike_trip_data)
mean(bike_trip_data_v1$ride_duration)
```

```
## [1] 1341.07
```

```
median(bike_trip_data_v1$ride_duration)
```

```
## [1] 639
```

```
max(bike_trip_data_v1$ride_duration)
```

```
## [1] 9387024
```

```
min(bike_trip_data_v1$ride_duration)
```

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

```
summary(bike_trip_data_v1$ride_duration)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1     385     639    1341    1078 9387024
```

##compare all mean, median and min and max values

```
aggregate(bike_trip_data_v1$ride_duration ~ bike_trip_data_v1$member_casual, FUN = mean)
```

```
##   bike_trip_data_v1$member_casual bike_trip_data_v1$ride_duration
## 1                          casual                       3544.5127
## 2                          member                        775.6158
```

```
aggregate(bike_trip_data_v1$ride_duration ~ bike_trip_data_v1$member_casual, FUN = median)
```

```
##   bike_trip_data_v1$member_casual bike_trip_data_v1$ride_duration
## 1                          casual                            1383
## 2                          member                             553
```

```
aggregate(bike_trip_data_v1$ride_duration ~ bike_trip_data_v1$member_casual, FUN = max)
```

```
##   bike_trip_data_v1$member_casual bike_trip_data_v1$ride_duration
## 1                          casual                         9387024
## 2                          member                         9056634
```

```
aggregate(bike_trip_data_v1$ride_duration ~ bike_trip_data_v1$member_casual, FUN = min)
```

```
##   bike_trip_data_v1$member_casual bike_trip_data_v1$ride_duration
## 1                          casual                               2
## 2                          member                               1
```

##Organize day_of_week in correct order

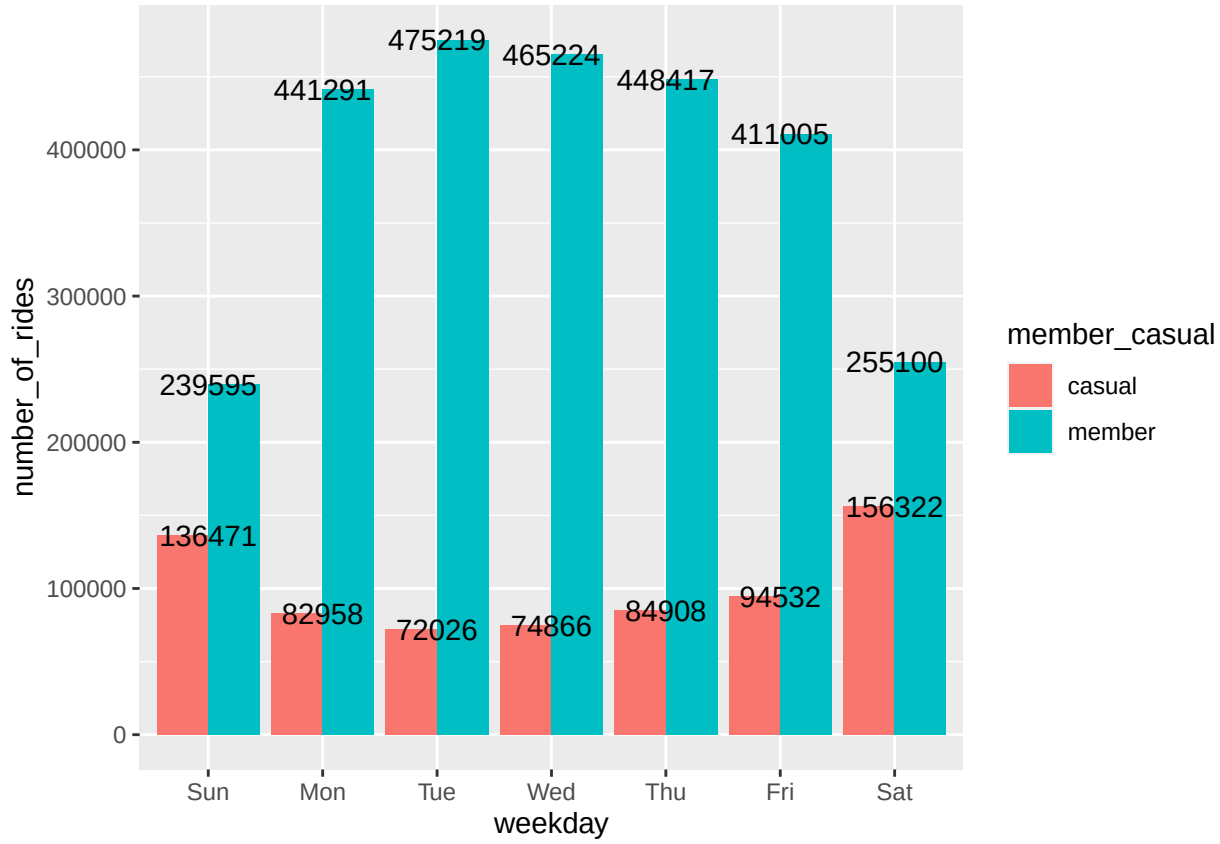
```
bike_trip_data_v1$day_of_week <- ordered(bike_trip_data_v1$day_of_week, levels=
c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"))
aggregate(bike_trip_data_v1$ride_duration ~ bike_trip_data_v1$member_casual +
          bike_trip_data_v1$day_of_week, FUN = mean)
```

```
##    bike_trip_data_v1$member_casual bike_trip_data_v1$day_of_week
## 1                           casual                         Sunday
## 2                           member                         Sunday
## 3                           casual                         Monday
## 4                           member                         Monday
## 5                           casual                        Tuesday
## 6                           member                        Tuesday
## 7                           casual                      Wednesday
## 8                           member                      Wednesday
## 9                           casual                       Thursday
## 10                          member                       Thursday
## 11                          casual                         Friday
## 12                          member                         Friday
## 13                          casual                       Saturday
## 14                          member                       Saturday
##    bike_trip_data_v1$ride_duration
## 1                        3624.8708
## 2                         831.8936
## 3                        3371.5806
## 4                         768.1699
## 5                        3555.8366
## 6                         763.3817
## 7                        3679.4305
## 8                         748.0193
## 9                        3658.3209
## 10                        751.5553
## 11                       3844.7670
## 12                        750.3009
## 13                       3252.9116
## 14                        891.8368
```

#Prepare visualizations ##Visualizing number of rides Vs average duration by rider_type

```
bike_trip_data_v1 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n(),options(scipen = 100),average_duration = mean(ride_duration))%>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") + geom_text(aes(label = signif(number_of_rides)))
```

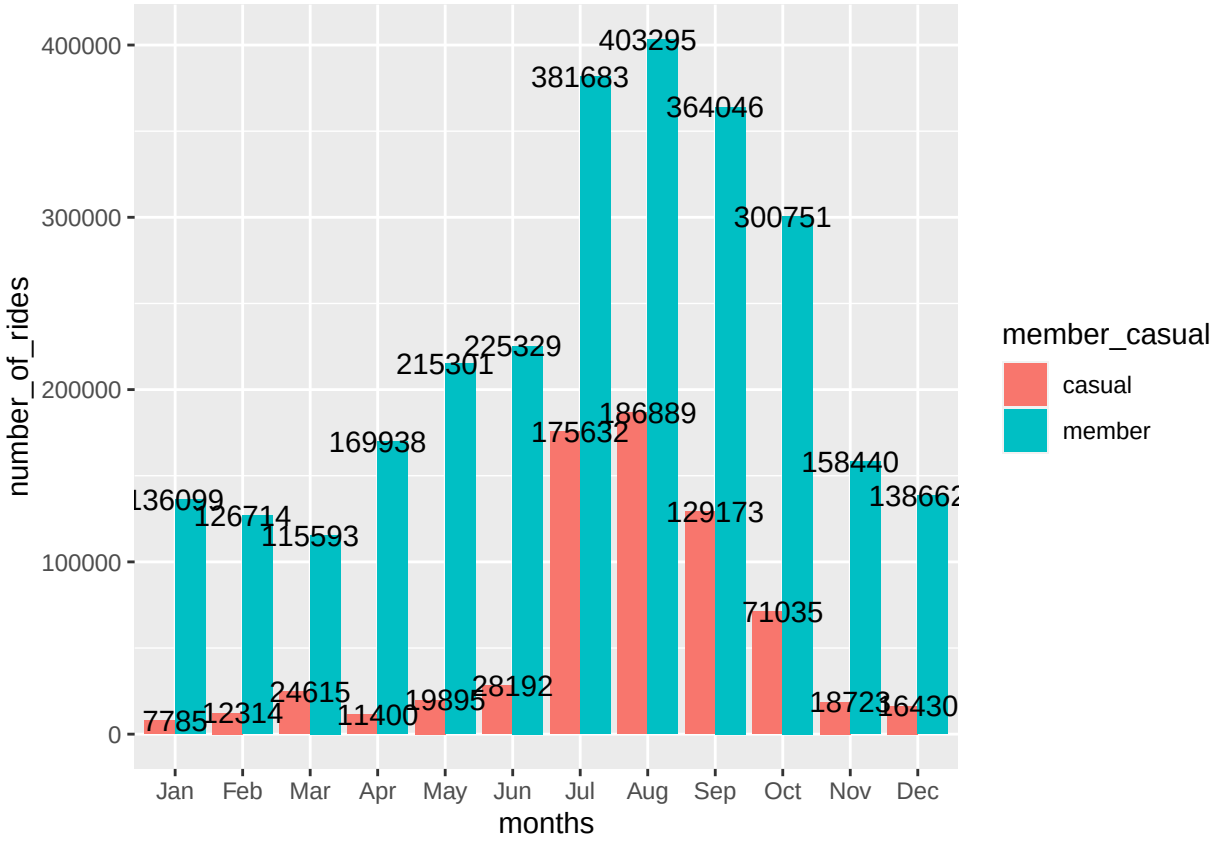
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```



##Visualizing number of rides vs months by rider type

```
bike_trip_data_v1 %>%
  mutate(months = month(started_at, label = TRUE)) %>%
  group_by(member_casual, months) %>%
  summarise(number_of_rides = n(),options(scipen = 100),average_duration = mean(ride_duration))%>%
  arrange(member_casual, months)  %>%
  ggplot(aes(x = months, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") + geom_text(aes(label = signif(number_of_rides)))
```

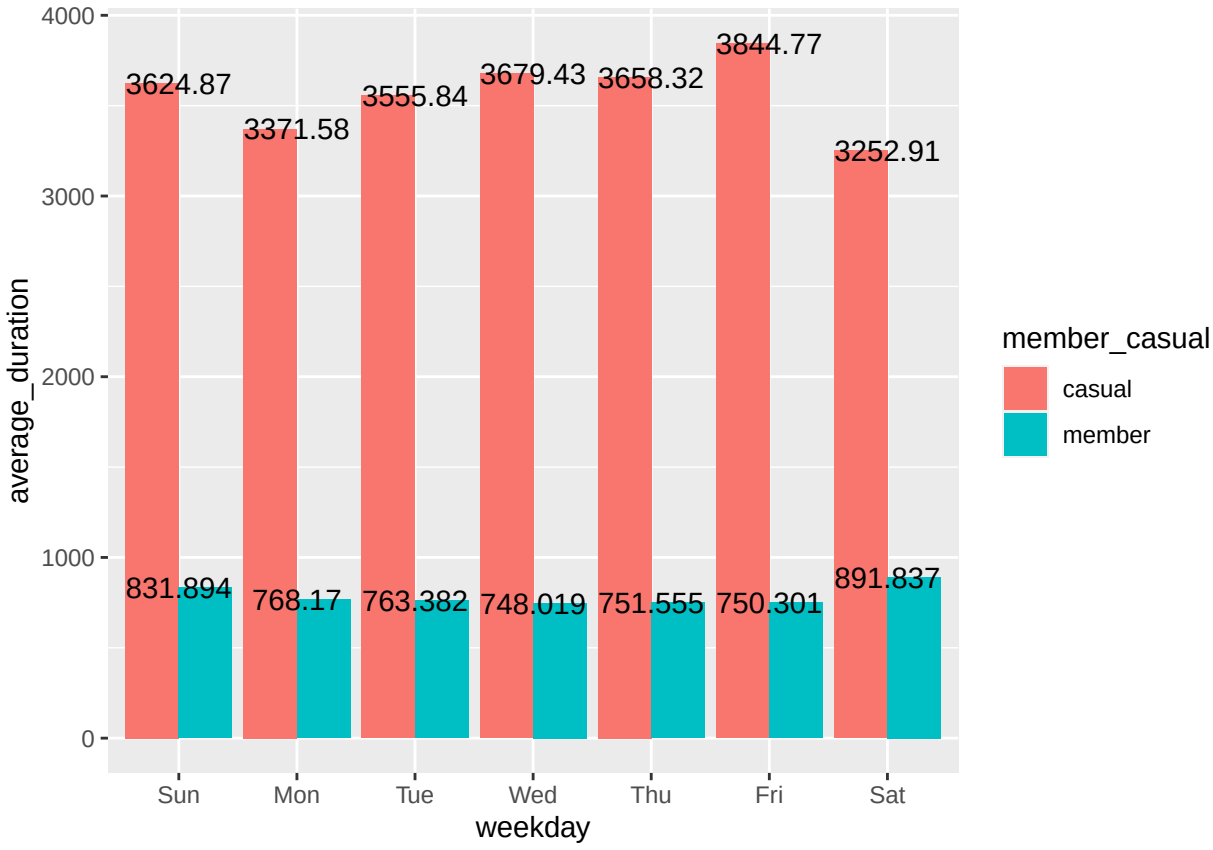
```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```



##Visualization of average ride_duration vs weekday with ridertype

```
bike_trip_data_v1 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n(),average_duration = mean(ride_duration)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge") + geom_text(aes(label = signif(average_duration)))
```

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


4000
3000
2000
1000
0
average_duration
3624.87
3371.58
3555.84
3679.43
3658.32
3844.77
3252.91
831.894
768.17
763.382
748.019
751.555
750.301
891.837
Sun
Mon
Tue
Wed
Thu
Fri
Sat
weekday
member_casual
casual
member