Cyclistic Bike-Share

Volkan

2024-08-16

Introduction

In this report, I will walk you through the steps of importing, cleaning, transforming, and visualizing data. I will also share insights and address questions about annual members and casual riders. The data comes from the Google Data Analytics Professional Certificate program and represents a fictional bike rental company that offers annual memberships. Let's begin by loading the necessary packages.

```
library('tidyverse') # Helps to transform and better present data
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library('conflicted') # To use filter() from dplyr package
library('scales') # Provides the internal scaling infrastructure used by ggplot2
library('patchwork') # Designed to combine plots
```

Data Collection

After loading necessary packages, we import our data.

- cyclistic_2023_01 <- read_csv('cyclistic_2023_01.csv')
- cyclistic_2023_02 <- read_csv('cyclistic_2023_02.csv')
- cyclistic_2023_03 <- read_csv('cyclistic_2023_03.csv')
- cyclistic_2023_04 <- read_csv('cyclistic_2023_04.csv')
- cyclistic_2023_05 <- read_csv('cyclistic_2023_05.csv')
- cyclistic_2023_06 <- read_csv('cyclistic_2023_06.csv')
- cyclistic_2024_01 <- read_csv('cyclistic_2024_01.csv')
- cyclistic_2024_02 <- read_csv('cyclistic_2024_02.csv')
- cyclistic_2024_03 <- read_csv('cyclistic_2024_03.csv')
- cyclistic_2024_04 <- read_csv('cyclistic_2024_04.csv')
- cyclistic_2024_05 <- read_csv('cyclistic_2024_05.csv')
- cyclistic_2024_06 <- read_csv('cyclistic_2024_06.csv')

There are twelve csv files that are consist of first six months of 2023 and 2024. I could have added them into only one file using excel but I preferred doing it with bind_rows() function.

```
all_rides <- bind_rows(cyclistic_2023_01, cyclistic_2023_02, cyclistic_2023_03,
                       cyclistic_2023_04, cyclistic_2023_05, cyclistic_2023_06,
                       cyclistic_2024_01, cyclistic_2024_02, cyclistic_2024_03,
                       cyclistic_2024_04, cyclistic_2024_05, cyclistic_2024_06)
```

Now we have our data in our hands to be looked, cleaned and visualized. **Our purpose with this data is to compare member riders and casual riders, checking total numbers for the first half of 2023 and 2024 if it's increased or decreased. To see popular stations among riders, most used rideable type etc.**

First Look At Data

```
dim(all_rides)
```

```
## [1] 4795422      13
```

There are 4795422 rows and 13 columns. Let's take a look at them.

```
colnames(all_rides)
```

```
##  [1] "ride_id"            "rideable_type"      "started_at"
##  [4] "ended_at"           "start_station_name" "start_station_id"
##  [7] "end_station_name"   "end_station_id"     "start_lat"
## [10] "start_lng"          "end_lat"            "end_lng"
## [13] "member_casual"
```

We don't need "start_lat","start_lng", "end_lat", "end_lng" since they are in no use to us.

```
all_rides <- select(all_rides, -c(start_lat, start_lng, end_lat, end_lng))
```

```
head(all_rides)
```

```
## # A tibble: 6 x 9
##   ride_id          rideable_type started_at          ended_at
##   <chr>            <chr>         <dttm>              <dttm>
## 1 F96D5A74A3E41399 electric_bike 2023-01-21 20:05:42 2023-01-21 20:16:33
## 2 13CB7EB698CEDB88 classic_bike  2023-01-10 15:37:36 2023-01-10 15:46:05
## 3 BD88A2E670661CE5 electric_bike 2023-01-02 07:51:57 2023-01-02 08:05:11
## 4 C90792D034FED968 classic_bike  2023-01-22 10:52:58 2023-01-22 11:01:44
## 5 3397017529188E8A classic_bike  2023-01-12 13:58:01 2023-01-12 14:13:20
## 6 58E68156DAE3E311 electric_bike 2023-01-31 07:18:03 2023-01-31 07:21:16
## # i 5 more variables: start_station_name <chr>, start_station_id <chr>,
## #   end_station_name <chr>, end_station_id <chr>, member_casual <chr>
```

```r
#More detailed view
glimpse(all_rides)
summary(all_rides)
str(all_rides)
```

```r
# Checking null values
colSums(is.na(all_rides))
```

```r
# Checking for duplicates
all_rides %>%
distinct(ride_id) %>%
nrow() # 479521 rows. But our total was 4795422. So there are 211 duplicate values

# Shows the duplicate values
view(all_rides %>%
group_by(ride_id) %>%
filter(n() > 1))
```

This data needs to be modified to be more representable. Here is the list that we need to do.

To Do:

- Eliminate duplicate entries from the dataset.
- Add columns for date, month, day, and year to make data aggregation easier.
- Introduce a column for ride_length to calculate the duration of each ride in the all_rides dataset.
- Address the issue of negative values in the ride_length column. These negative values may result from data errors or quality control activities where bikes are taken from service. It's best to remove these rows to keep the data accurate.

Data Cleaning, Manipulation, Transformation

```r
# 1 211 Duplicates removes
all_rides <- all_rides %>%
distinct(ride_id, .keep_all = TRUE)
```

```r
# 2 Adding columns such as date, year, month, day, day_of_week
all_rides_2 <- all_rides %>%  mutate(date = as.Date(started_at))
all_rides_2 <- all_rides_2 %>% mutate(year = format(as.Date(date), "%Y"))
all_rides_2 <- all_rides_2 %>% mutate(month = format(as.Date(date), "%m"))
all_rides_2 <- all_rides_2 %>% mutate(day = format(as.Date(date), "%d"))
all_rides_2 <- all_rides_2 %>% mutate(day_of_week = format(as.Date(date), "%A"))
```

```r
# 3 Calculating the trip duration in seconds
all_rides_2 <- all_rides_2 %>%
mutate(ride_length = difftime(ended_at, started_at))
```

```r
#Converting ride_length from difftime to numeric value type to perform calculations
all_rides_2$ride_length <- as.numeric(all_rides_2$ride_length)
```

```r
# 4 Removing the bad data. 206 data to be removed.
bad_data <- all_rides_2 %>%
filter(ride_length < 0)
```

```r
bad_data %>% group_by(rideable_type) %>%
summarise(count_rideable = n())
```

```
## # A tibble: 2 x 2
##   rideable_type count_rideable
##   <chr>                  <int>
## 1 classic_bike               9
## 2 electric_bike            197
```

We need to inform the company that most of the bad data caused by electric bikes.

```r
# Bad data removed
all_rides_2 <- all_rides_2 %>%
filter(!(ride_length < 0))
```

With the data now cleaned, we will proceed to extract meaningful insights and visualize the results. Please note that I've included detailed code to clarify how the charts were created. Although I could have just shown the charts, I wanted to provide the code for those who haven't seen the original R script.

Chart 1: Total Of Casual And Member Riders

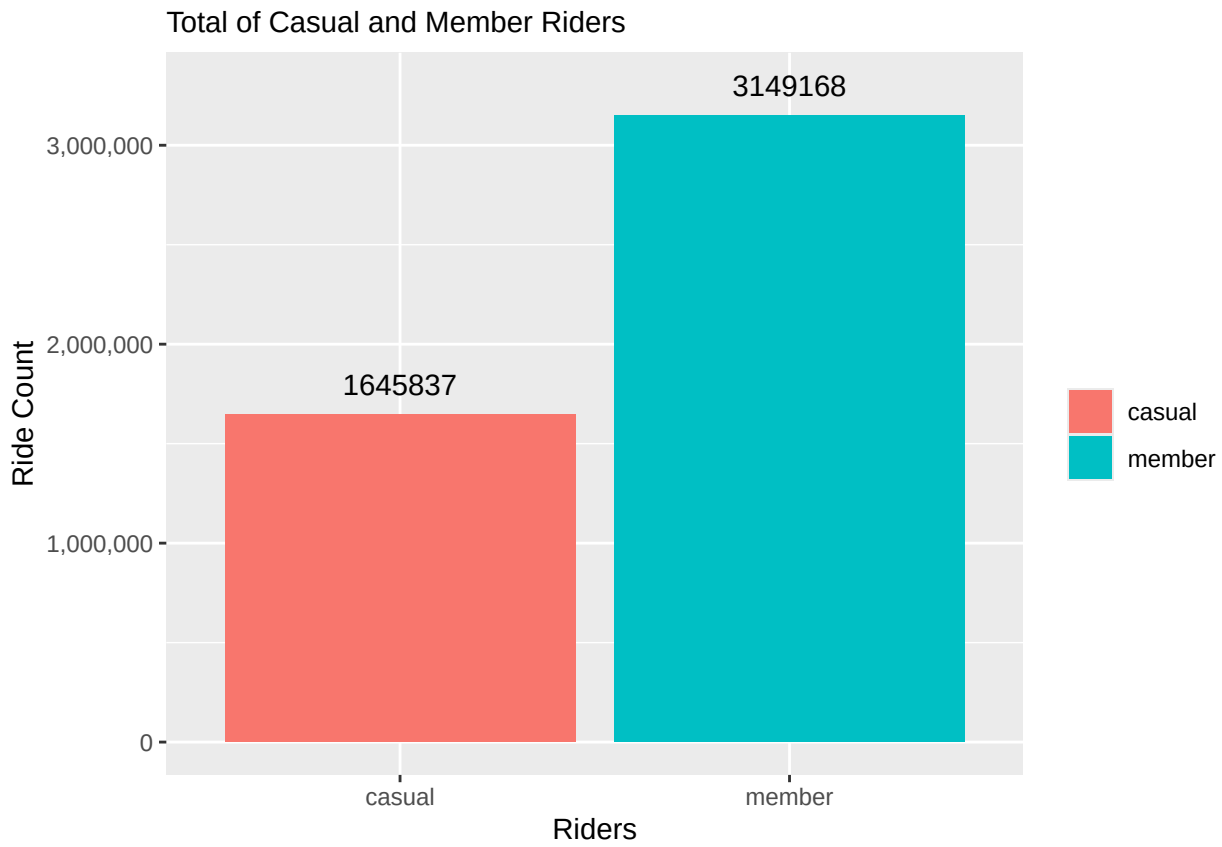
Below code chunks calculate how much of the rides are casual or member.

```r
member_casual_count <- all_rides_2 %>%
group_by(member_casual) %>%
summarise(person_count = n())
member_casual_count
```

```
## # A tibble: 2 x 2
##   member_casual person_count
##   <chr>                <int>
## 1 casual             1645837
## 2 member             3149168
```

```r
total_casual_rides <- member_casual_count %>%
 filter(member_casual == "casual") %>% pull(person_count)
total_member_rides <- member_casual_count %>%
  filter(member_casual == "member") %>% pull(person_count)
```

```r
member_casual_count %>%
ggplot(aes(x = member_casual, y = person_count, fill = member_casual)) +
geom_bar(stat = "identity") +
labs(Title ="Total Rides",
subtitle="Total of Casual and Member Riders",x="Riders",y="Ride Count",fill="")+
scale_y_continuous(labels = comma) +
annotate("text", x=1, y=1800000, label=total_casual_rides, size=4)+
annotate("text", x=2, y=3300000, label=total_member_rides, size=4)
```



Here we see around 65% of rides are done by members.

Chart 2: Casual And Member Count 2023-2024

Keep in mind that 2023 and 2024 datas include only first six months.

```
casual_2023 <- all_rides_2 %>% filter(year == 2023 & member_casual == "casual") %>% nrow()
member_2023 <- all_rides_2 %>% filter(year == 2023 & member_casual == "member") %>% nrow()

casual_2024 <- all_rides_2 %>% filter(year == 2024 & member_casual == "casual") %>% nrow()
member_2024 <- all_rides_2 %>% filter(year == 2024 & member_casual == "member") %>% nrow()

#Casual and member count 2023-2024
print(paste(casual_2023, member_2023, casual_2024, member_2024))
```

```
## [1] "827913 1562524 817924 1586644"
```

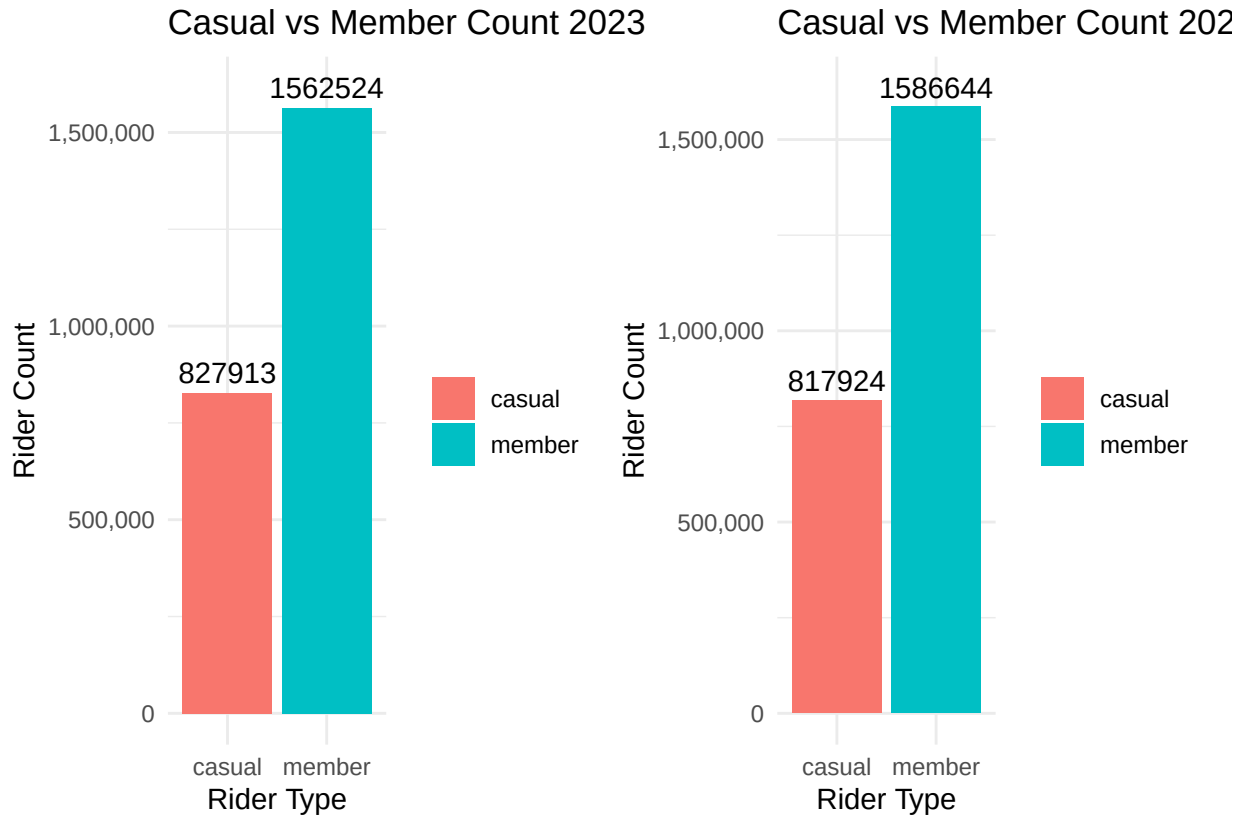
```
#Creating casual_vs_member_count_2023 plot
casual_vs_member_count_2023 <- all_rides_2 %>% filter(year == 2023) %>%
group_by(member_casual) %>%
summarise(rider_count = n()) %>%
ggplot(aes(x = member_casual, y = rider_count, fill = member_casual)) +
geom_bar(stat = "identity") +
labs(title="Casual vs Member Count 2023", x="Rider Type", y = "Rider Count", fill = "")+
scale_y_continuous(labels = comma) + theme_minimal() +
```

```
annotate("text", x = 1, y = 880000, label = casual_2023, size = 4) +
annotate("text", x = 2, y = 1615000, label = member_2023, size = 4)
```

```
#Creating casual_vs_member_count_2024 plot
casual_vs_member_count_2024 <- all_rides_2 %>% filter(year == 2024) %>%
group_by(member_casual) %>%
summarise(rider_count = n()) %>%
ggplot(aes(x = member_casual, y = rider_count, fill = member_casual)) +
geom_bar(stat = "identity") +
labs(title="Casual vs Member Count 2024", x="Rider Type", y="Rider Count",fill = "")+
scale_y_continuous(labels = comma) + theme_minimal() +
annotate("text", x = 1, y = 870000, label = casual_2024, size = 4) +
annotate("text", x = 2, y = 1635000, label = member_2024, size = 4)
```

```
#Combine these 2 plots with patchwork package
combined_plot <- casual_vs_member_count_2023 + casual_vs_member_count_2024
```

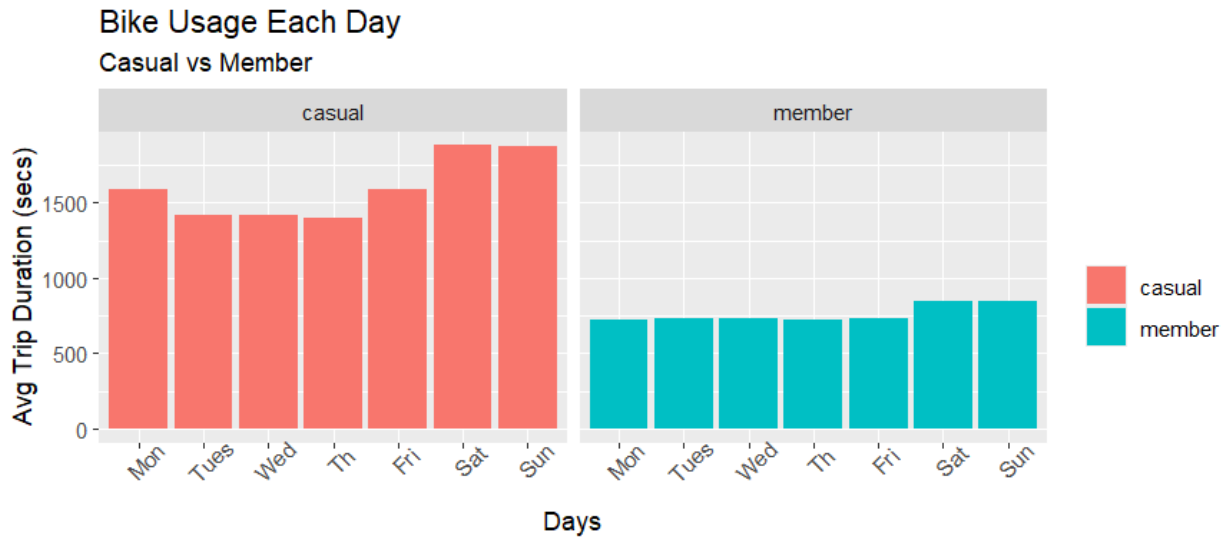
```
combined_plot
```



If we compare these values we can understand that member riders count has increased by 24120 which corresponds to 0.5% change. On the other hand casual riders count decreased by 9989, around 0.2% change. These numbers are little in comparison to total values but still is a good change.

Chart 3: Bike Usage Each Day

```
all_rides_2 %>%
mutate(ride_length = as.numeric(ride_length, units = "secs"),
day_of_week = factor(day_of_week,
levels=c("Monday","Tuesday","Wednesday","Thursday","Friday","Saturday","Sunday")))%>%
group_by(day_of_week, member_casual) %>%
summarise(mean_ride_length = mean(ride_length), .groups = "drop") %>%
ggplot(aes(x = day_of_week, y = mean_ride_length, fill = member_casual)) +
geom_bar(stat = "identity", position = "dodge") +
facet_wrap(~member_casual) +
theme(axis.text.x = element_text(angle = 45)) +
labs(title = "Bike Usage Each Day", subtitle = "Casual vs Member",
x = "Days", y = "Avg Trip Duration (secs)", fill = "") +
scale_x_discrete(labels = c("Monday" = "Mon", "Tuesday" = "Tues", "Wednesday" = "Wed",
"Thursday" = "Th", "Friday" = "Fri", "Saturday" = "Sat", "Sunday" = "Sun"))
```

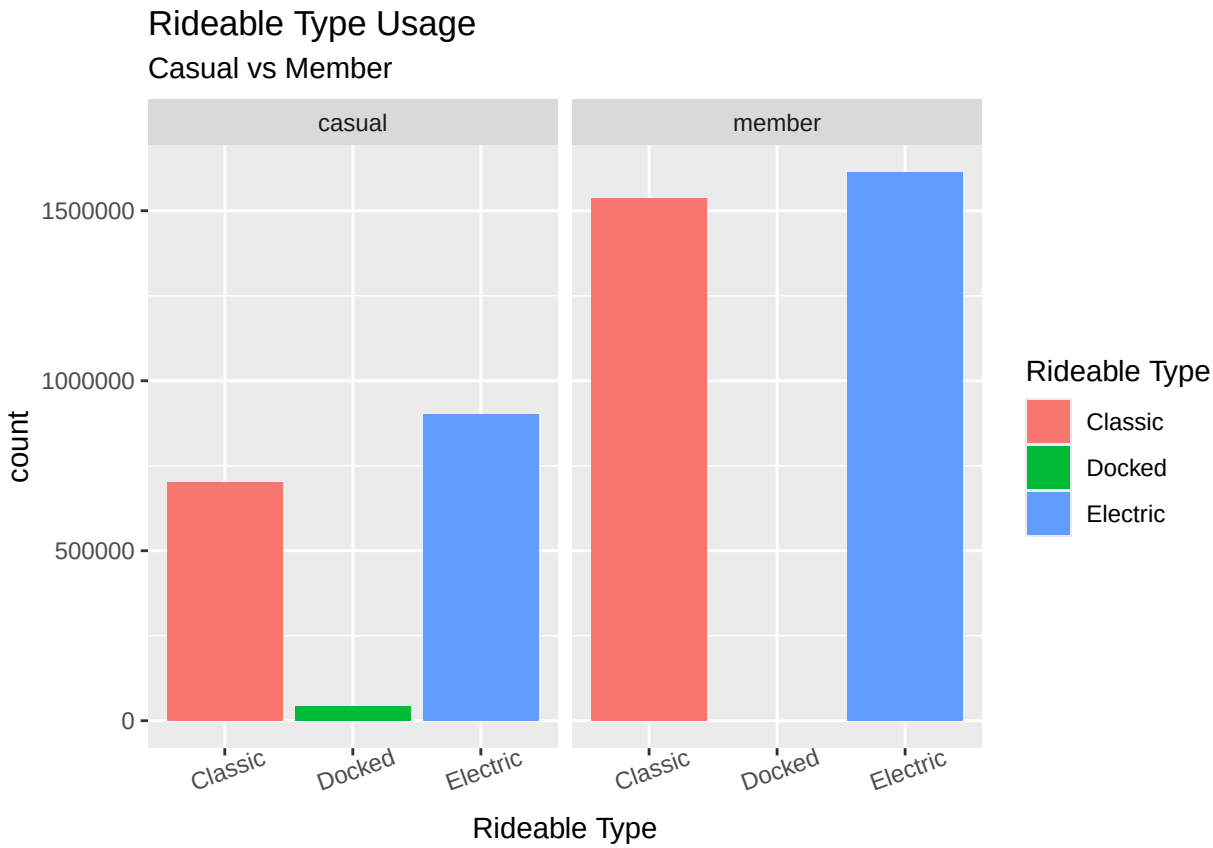


Member riders have a much consistent ride journey than casual riders. Riders trip duration is higher in both charts. By these charts it is understandable that member riders might use bikes to go to work since they are much more consistent.

Chart 4: Rideable Type Usage

```
all_rides_2 %>%
ggplot(aes(x=rideable_type,fill=rideable_type))+
geom_bar() + facet_wrap(~member_casual) +
theme(axis.text.x = element_text(angle = 20)) +
labs(
  title="Rideable Type Usage",
  subtitle="Casual vs Member",
  x="Rideable Type",fill="Rideable Type") +
scale_x_discrete(labels = c(
  "classic_bike" = "Classic",
```

```
  "docked_bike" = "Docked",
  "electric_bike" = "Electric")) +
scale_fill_discrete(labels = c(
  "classic_bike" = "Classic",
  "docked_bike" = "Docked",
  "electric_bike" = "Electric"))
```



On both charts electric bikes > classic bikes > docked bikes

Chart 5: Top Five Popular Stations

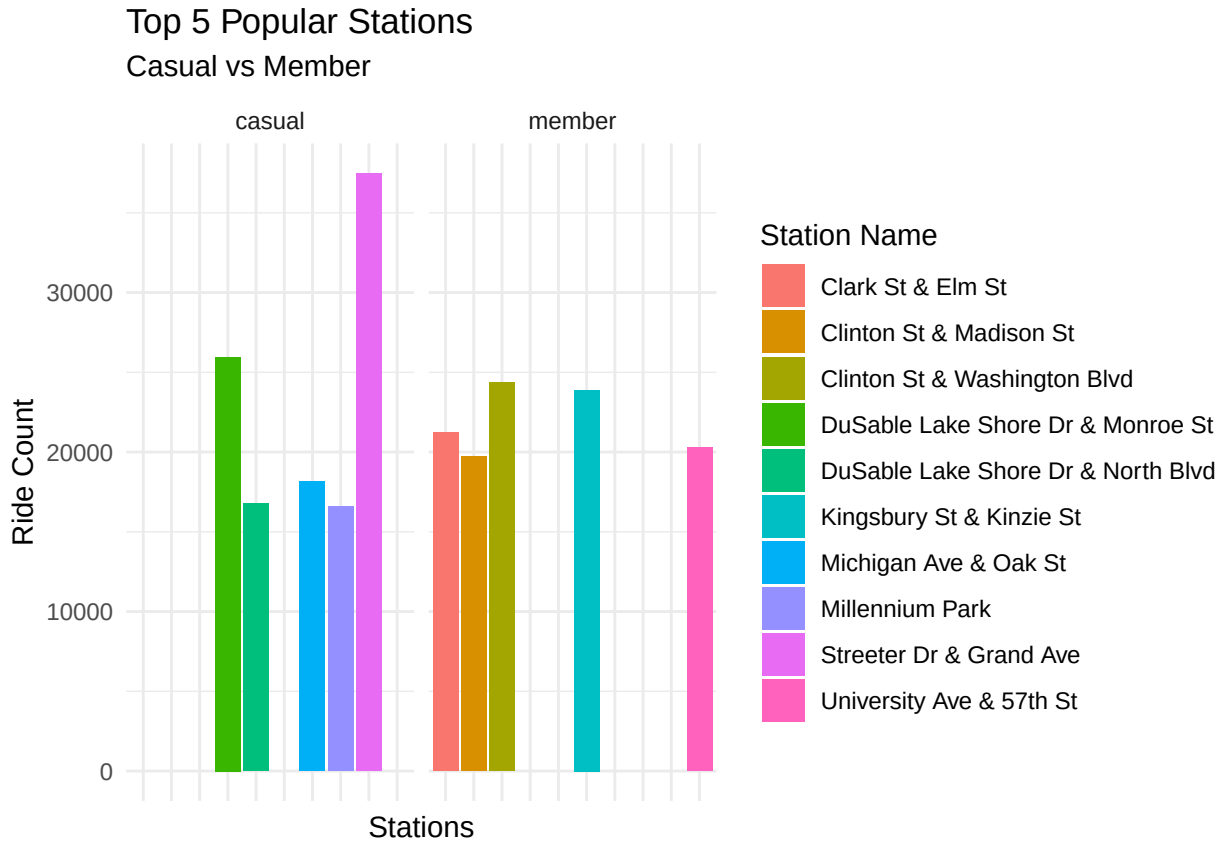
```
#Calculates top 5 stations for member riders
member_station <- all_rides_2 %>% drop_na(start_station_name) %>%
filter(member_casual == "member") %>%
group_by(start_station_name) %>%
summarise(each_station_ride_count = n()) %>%
arrange(-each_station_ride_count) %>%
slice_head(n=5)

member_station
```

```
casual_station <- casual_station %>% mutate(member_casual = "casual")
member_station <- member_station %>% mutate(member_casual = "member")

member_casual_station <- bind_rows(casual_station, member_station)
```

```
member_casual_station %>% arrange(each_station_ride_count) %>%
ggplot(aes(
  x = start_station_name,
  y = each_station_ride_count,
  fill= start_station_name))+
geom_bar(stat = "identity")+ facet_wrap(~member_casual) +
theme_minimal() + theme(axis.text.x = element_blank()) +
labs(
  title = "Top 5 Popular Stations",
  subtitle = "Casual vs Member",
  x = "Stations",y = "Ride Count",
  fill = "Station Name")
```



If this chart shows us top five stations it can show us the least favorite stations. Maybe we can do something to increase the popularity such as organizing events, discount on rides etc.

Chart 6: Monthly Rides 2023-2024

```
#total rides of 2023 first six months
total_rides_2023_01_06 <- all_rides_2 %>% filter(year == 2023) %>% nrow()
#total rides of 2024 first six months
total_rides_2024_01_06 <- all_rides_2 %>% filter(year == 2024) %>% nrow()
```

```r
plot_2023 <- all_rides_2 %>%
filter(year == 2023) %>%
arrange(started_at) %>%
group_by(month) %>%
summarise(year = "2023", monthly_ride = n())

plot_2024 <- all_rides_2 %>%
filter(year == 2024) %>%
arrange(started_at) %>%
group_by(month) %>%
summarise(year = "2024", monthly_ride = n())
```

```r
plot_2023 <- plot_2023 %>%
ggplot(aes(x = month, y = monthly_ride, group = 1)) +
geom_line() + geom_point(color = "red")+
labs(title= "Monthly Rides 2023", x = "Month", y = "Ride Count")+
scale_x_discrete(labels = c(
  "01" = "Jan",
  "02" = "Feb",
  "03" = "Mar",
  "04" = "Apr",
  "05" = "May",
  "06" = "June")) +
scale_y_continuous(labels = comma) + theme_minimal() +
annotate("text",x=3,y=600000,label="Total Rides =",size=4,color="Red") +
annotate("text",x=3,y=565000,label=total_rides_2023_01_06,size=4,color="Red")

plot_2024 <- plot_2024 %>%
ggplot(aes(x = month, y = monthly_ride, group = 1)) +
geom_line() + geom_point(color = "purple")+
labs(title= "Monthly Rides 2024", x = "Month", y = "Ride Count") +
scale_x_discrete(labels=c("01"="Jan","02"="Feb","03"="Mar","04"="Apr","05"="May","06"= "June")) +
scale_y_continuous(labels = comma) + theme_minimal() +
annotate("text",x = 3,y=600000,label="Total Rides =", size = 4, color = "Purple") +
annotate("text",x=3,y=565000,label=total_rides_2024_01_06,size=4,color="Purple")
```

```r
# Combined this plots with patchwork package
combined_2023_2024 <- plot_2023 + plot_2024
```

```r
combined_2023_2024
```

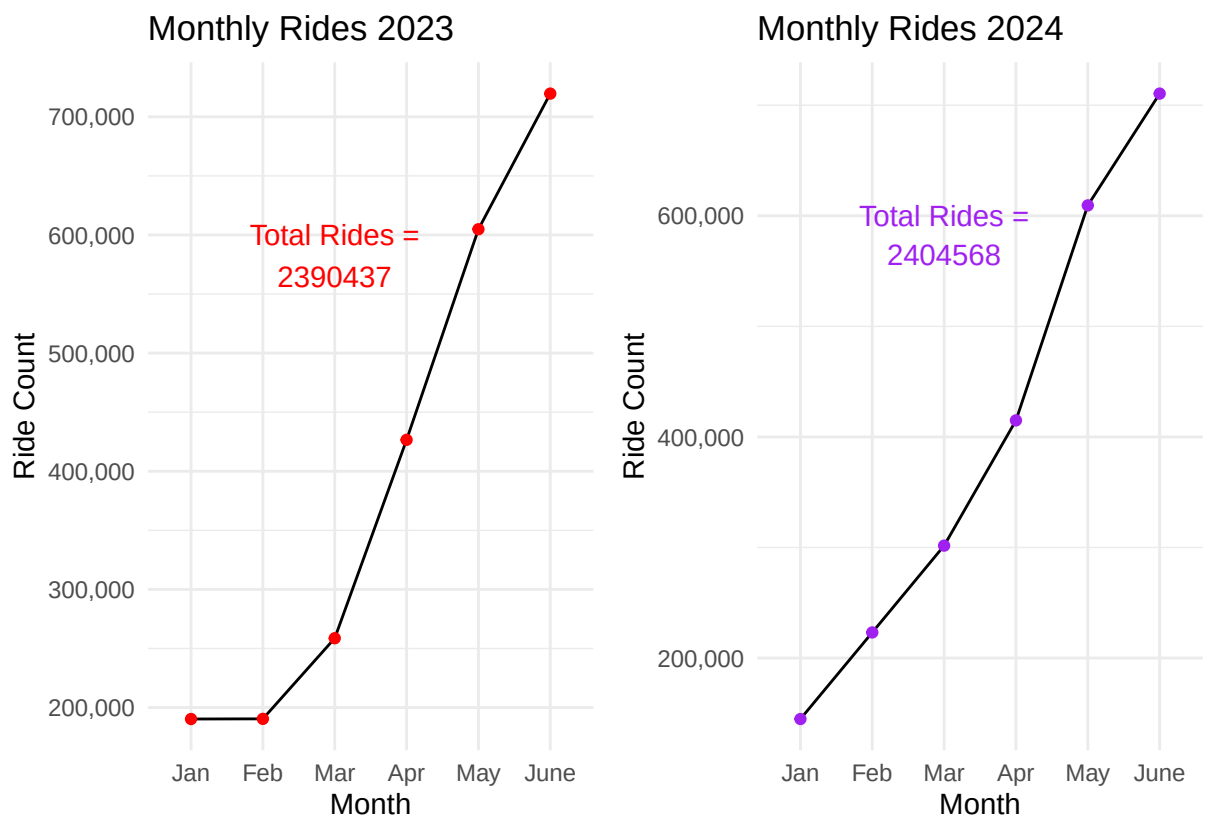
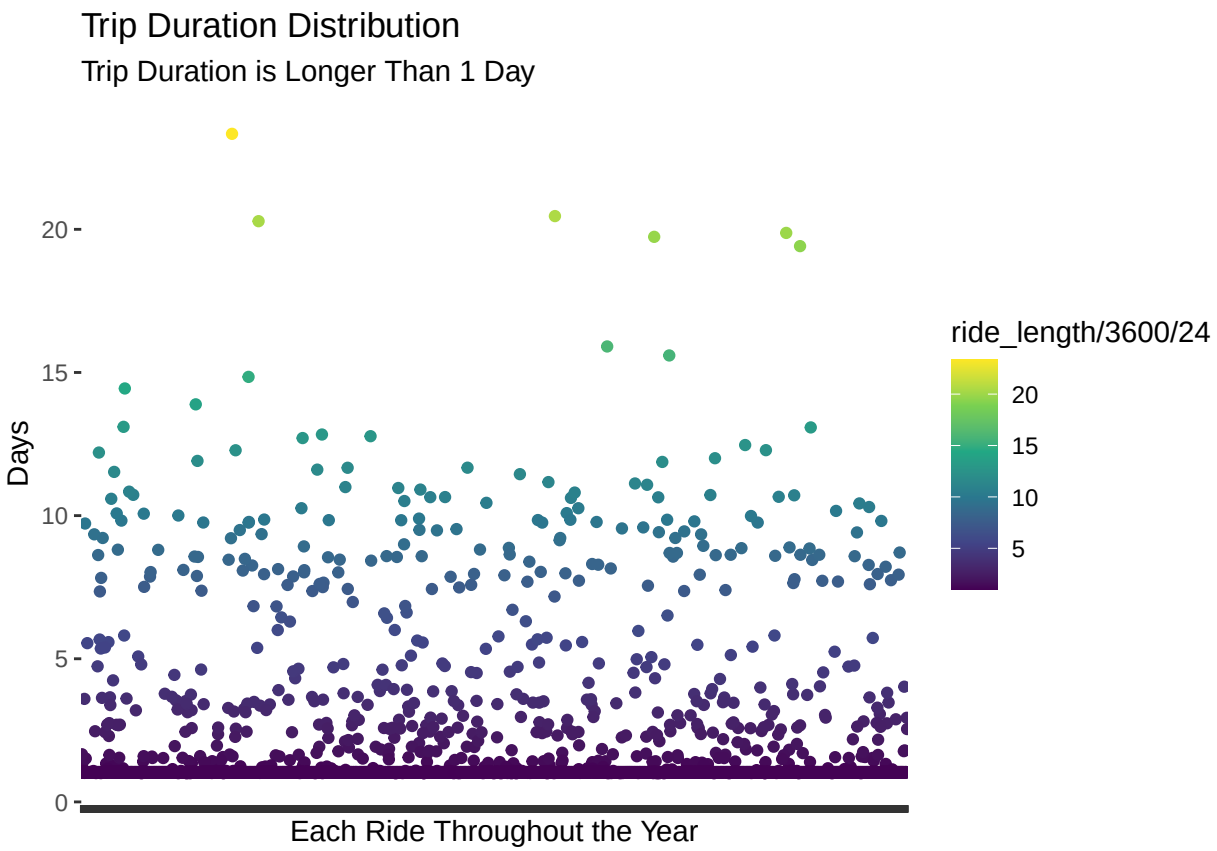


Chart 7: Trip Duration Distribution

```
# Scatterplot for the people who took the bike more than 1 day
all_rides_2 %>%
filter(ride_length >86400) %>%
ggplot(aes(
 x = ride_id,
 y = ride_length/3600/24,
 color = ride_length /3600/24)) +
geom_jitter() + theme(axis.text.x = element_blank()) +
labs(
  title= "Trip Duration Distribution",
  subtitle = "Trip Duration is Longer Than 1 Day",
  x = "Each Ride Throughout the Year", y = "Days") +
scale_color_viridis_c()
```



With this last chart, we have looked through some visualized data to have an idea about what this data is and how it can help to us to determine the future of the company. Now let me share what I learned from these charts.

Key Findings

Data Analysis on Bike Types: The analysis reveals that classic bikes have resulted in 9 instances of bad data, while electric bikes account for 197 instances. The majority of data issues are associated with electric bikes, indicating a need for the company to implement measures to address this.

Rider Demographics and Strategy: Approximately 65% of riders are members. To increase membership, it is recommended that the company identify popular stations frequented by casual riders and focus targeted marketing efforts on these locations. Strategies could include offering discounts, prizes, and organizing events to engage and convert casual riders into members.

Year-over-Year Comparison: Comparing data from 2023 to 2024, there is a noticeable increase in both the total number of riders and the number of member riders. Although the growth is modest, it is a positive trend. If this growth continues over the next decade, it could have a significant and beneficial impact on the company's performance.

Note: The data analyzed in this report covers the first six months of 2023 and 2024.