

# Google Data Analytics Specialization: Capstone Project - Cyclistic Bike-share

Adarsh K  
17/08/2021

## Step 0: Importing Necessary Libraries

```
library(tidyverse, quietly = TRUE)
library(lubridate, quietly = TRUE)
library(ggplot2, quietly = TRUE)
#library(dplyr, quietly = TRUE)
```

---

## Step 1: Importing and Wrangling the Data

1. starting\_station\_name and starting\_station\_id are missing for some rows, ignore them.
2. Cyclistic collected some bikes to maintain them while these data were recorded. This is indicated with a negative or zero trip\_duration value. Ignore them too.

```
df1 <- read.csv("202004-divvy-tripdata.csv")
df2 <- read.csv("202005-divvy-tripdata.csv")
df3 <- read.csv("202006-divvy-tripdata.csv")
df4 <- read.csv("202007-divvy-tripdata.csv")
df5 <- read.csv("202008-divvy-tripdata.csv")
df6 <- read.csv("202009-divvy-tripdata.csv")
df7 <- read.csv("202010-divvy-tripdata.csv")
df8 <- read.csv("202011-divvy-tripdata.csv")
df9 <- read.csv("202012-divvy-tripdata.csv")
df10 <- read.csv("202101-divvy-tripdata.csv")
df11 <- read.csv("202102-divvy-tripdata.csv")
df12 <- read.csv("202103-divvy-tripdata.csv")

bike_rides <- rbind(df1, df2, df3, df4, df5, df6, df7, df8, df9, df10,
df11, df12)

bike_rides <- janitor::remove_empty(bike_rides, which = c("cols"))
bike_rides <- janitor::remove_empty(bike_rides, which = c("rows"))
bike_rides <- bike_rides %>%
  filter(start_station_name != "")

bike_rides$Ymd <- as.Date(bike_rides$started_at)

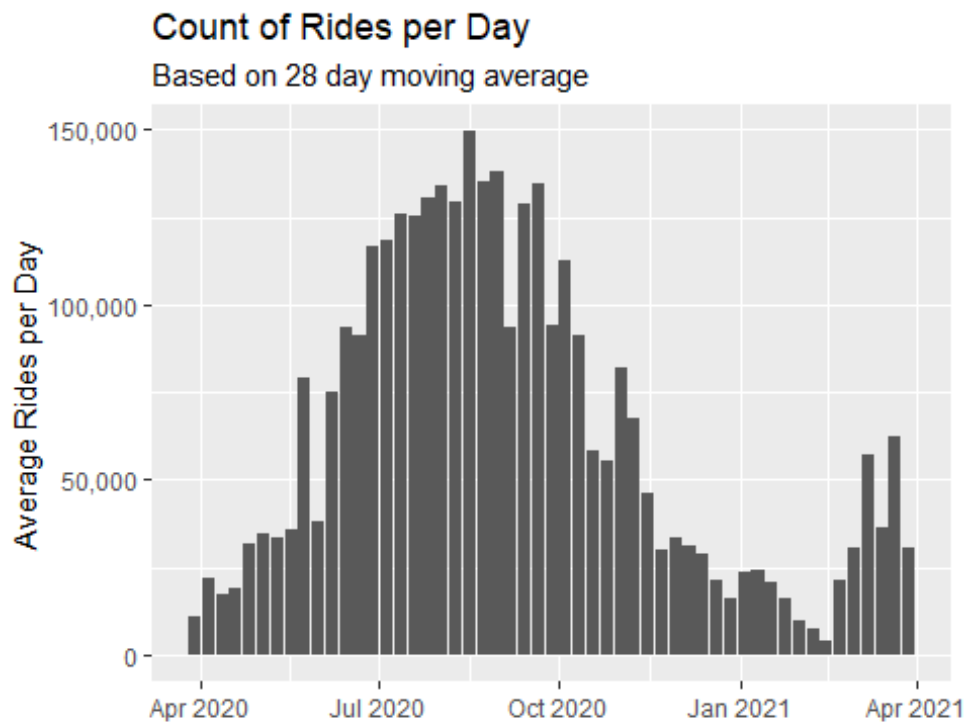
bike_rides$started_at <- lubridate::ymd_hms(bike_rides$started_at)
bike_rides$ended_at <- lubridate::ymd_hms(bike_rides$ended_at)
```



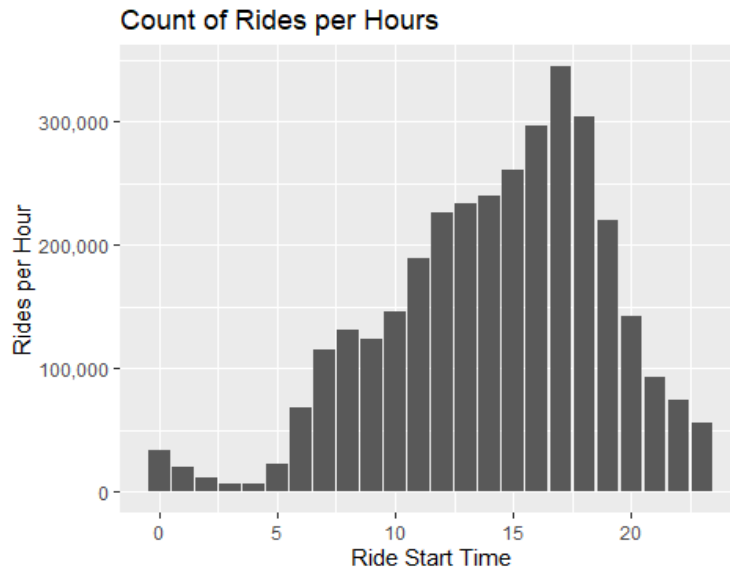
### Step 3: Plots

```
bike_rides2$Monthly <- lubridate::month(bike_rides2$weekly)
```

```
ggplot(data = bike_rides2) + geom_col(aes(x = weekly, y = Count)) + scale_y_continuous(labels = scales::comma) + labs(title = "Count of Rides per Day", subtitle = "Based on 28 day moving average", y = "Average Rides per Day", x = "")
```



```
ggplot(data = bike_rides2) + geom_col(aes(x = start_hour, y = Count)) + scale_y_continuous(labels = scales::comma) + labs(title = "Count of Rides per Hours", y = "Rides per Hour", x = "Ride Start Time")
```



### Count of Rides by Bike Type (rideable\_type)

#### Summary of Bike Types

```
bike_types <- bike_rides %>%
  group_by(member_casual, rideable_type, weekly = floor_date(Ymd, "week")) %>%
  summarize(
    Minutes = sum(Minutes),
    mean = mean(Minutes),
    Median = median(Minutes),
    Max = max(Minutes),
    Min = min(Minutes),
    Count = n()
  ) %>% ungroup()

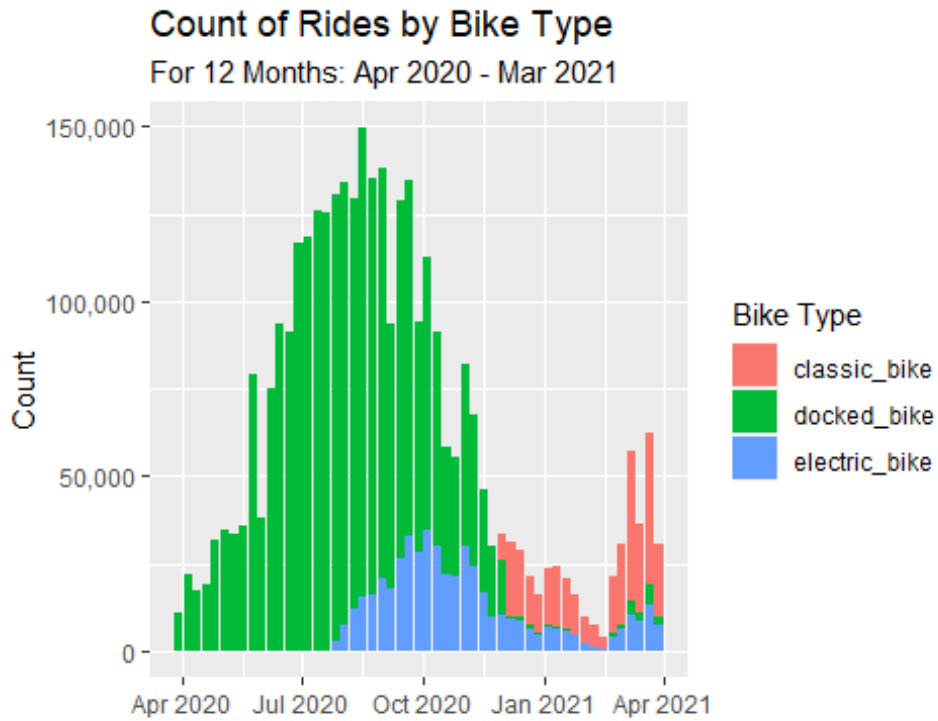
## `summarise()` has grouped output by 'member_casual', 'rideable_type'.
## You can override using the `.groups` argument.
```

#### Count by Bike Type (Total by Week)

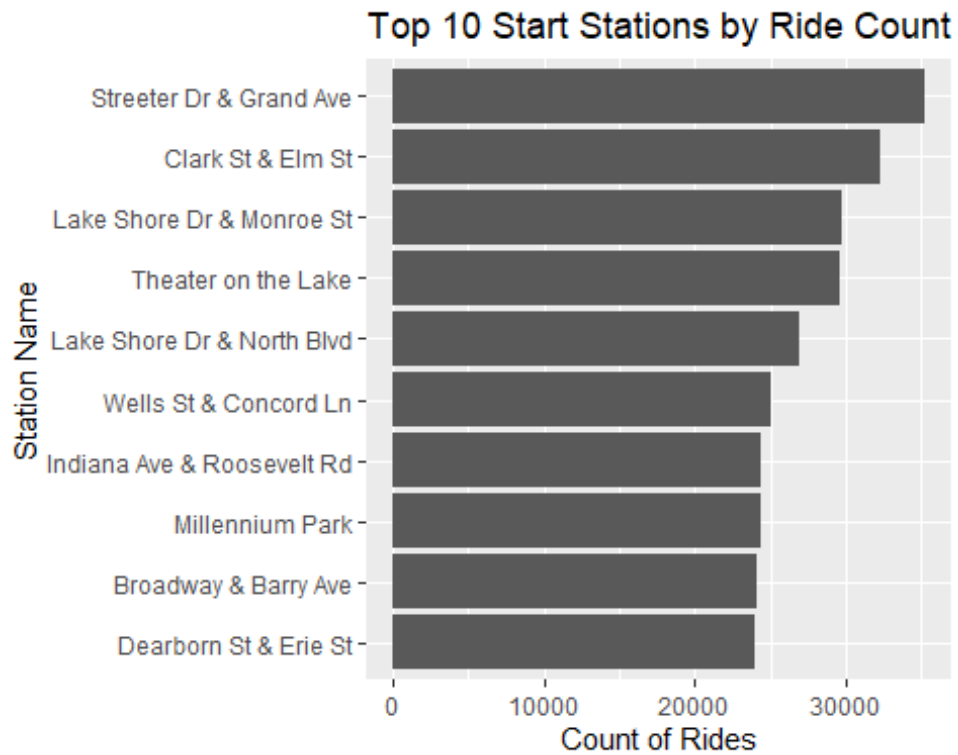
```
table(bike_types$rideable_type)

##
## classic_bike   docked_bike electric_bike
##           36           90           73

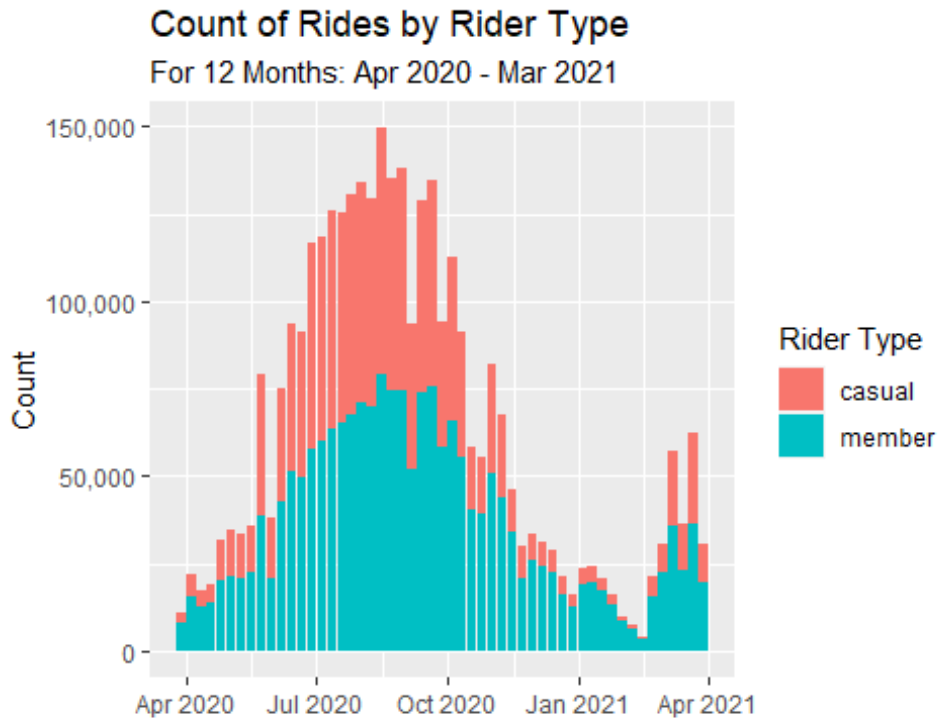
ggplot(data = bike_types) + geom_col(aes(x = weekly, y = Count, fill =
rideable_type)) + scale_y_continuous(labels = scales::comma) + labs(title =
'Count of Rides by Bike Type', subtitle = 'For 12 Months: Apr 20
20 - Mar 2021', fill = "Bike Type", x = "")
```



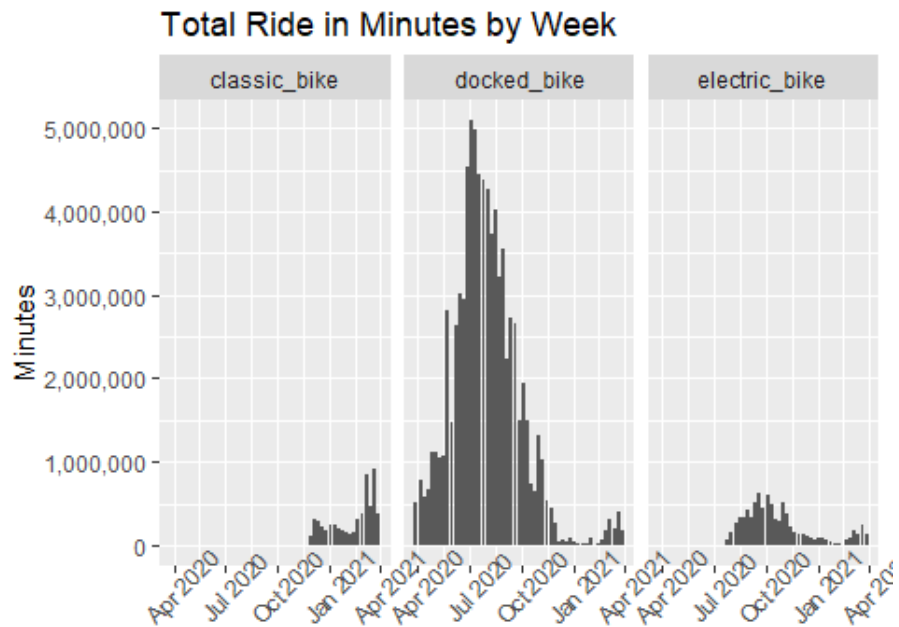
```
bike_rides %>% count(start_station_name, sort = TRUE) %>% top_n(10) %>%
%
  ggplot() + geom_col(aes(x=reorder(start_station_name, n), y=n)) + coord_flip() + labs(x = "Station Name", y = "Count of Rides", title = "Top 10 Start Stations by Ride Count")
## Selecting by n
```



```
ggplot(bike_types) + geom_col(aes(x = weekly, y = Count, fill = member_casual)) + scale_y_continuous(labels = scales::comma) + labs(title = 'Count of Rides by Rider Type', subtitle = "For 12 Months: Apr 2020 - Mar 2021", fill = "Rider Type", x = "")
```



```
ggplot(bike_types) + geom_col(aes(x = weekly, y = Minutes)) + scale_y_
continuous(labels = scales::comma) + facet_wrap(~rideable_type) + labs
(title = "Total Ride in Minutes by Week", x = "") + theme(axis.text.x
= element_text(angle = 45))
```



```
ggplot(bike_types) + geom_col(aes(x = weekly, y = Minutes, fill = rideable_type)) + scale_y_continuous(labels = scales::comma) + labs(title = "Weekly Rides in Minutes by Bike Type", x = "", y = "Bike Trip in Minutes", fill = "Bike Type")
```

