# Divvy Case Study, Data Cleaning Report

Tanner Quesenberry

7/13/2021

## Data Cleaning Steps

- Install and load the `tidyverse`, `lubridate`, `ggplot2` and `dplyr` packages.
- Read in the .csv data files for the months of June 2020 to May 2021.
- Join the tables from June 2020 to Nov 2020 together with `full_join()` to get combined data frames. Example: `apr_may <- full_join(apr_2021_trips, may_2021_trips)`
- Repeat table joining for tables from Dec 2020 to May 2021.
- Resolve the incompatible type difference of `start_station_id` and `end_station_id` columns by casting to characters.
```
jun_nov <- jun_nov %>% mutate(start_station_id = as.character(start_station_id))
jun_nov <- jun_nov %>% mutate(end_station_id = as.character(end_station_id))
```
- Join these two 6 month tables to form a single one year table for analysis.
- Save the table to a .csv file for later reference if needed.
```
write.csv(df, "~/Desktop/Programming/DataAnalyst/Capstone/202006-202105-divvy-tripdata.csv",
row.names = FALSE)
```
- Found 209 duplicate rows exist using
```
df %>% summarise(count = n_distinct(ride_id))
```
- Removed duplicate rows
```
df <- df %>% distinct(ride_id, .keep_all = TRUE)
```
- Verified that member_casual contains correct data
```
View(filter(df, member_casual != "casual", member_casual != "member"))
```
- Verified `start_at` and `ended_at` had times for all rows through viewing and sorting the table.
- Checked that `rideable_type` column contained the correct bike options.
```
df %>% distinct(rideable_type)
```
- Removed the `start_station_id` and `end_station_id` as they serve no purpose in this analysis.
```
df <- subset(df, select = -c(start_station_id,end_station_id))
```
- Similarly removed `start_lat`, `start_lng`, `end_lat`, `end_lng`
```
df <- subset(df, select = -c(start_lat,start_lng,end_lat,end_lng))
```
- Created an additional column `day_of_week` to indicate which weekday the ride starts on.
```
df <- df %>% mutate(day_of_week = wday(started_at))
```
- Created additional columns for date, month, day, and year to aggregate on later.
```
df$date <- as.Date(df$started_at)
df$month <- format(as.Date(df$date), "%m")
df$day <- format(as.Date(df$date), "%d")
df$year <- format(as.Date(df$date), "%Y")
```

- Created an additional column for `ride_length_secs`. Then removed any row with a negative duration, and converted to a time format column `ride_length`.
  ```
  df <- mutate(df, ride_length_secs = (ended_at - started_at))
  df <- subset(df, df$ride_length_secs > 0)
  df <- mutate(df, ride_length = hms::hms(seconds_to_period(df$ride_length_secs)))
  ```
- Saved the cleaned data set as divvy-cleaned-data.csv
  ```
  write.csv(df, "~/Desktop/Programming/DataAnalyst/Capstone/divvy-cleaned-data.csv",
  row.names = FALSE)
  ```

## Descriptive Analysis

### Gathering the descriptive statistics for the user ride lengths.

```
Ride length (seconds)
Min:  1
Max:  3257001
Median:  843
Mean:  1617.189
```

### Comparing the ride lengths of members vs casual riders

```
Mean
   Rider length_secs
1 casual   2562.0662
2 member    930.4542

Median
   Rider length_secs
1 casual        1213
2 member         664

Max
   Rider length_secs
1 casual     3257001
2 member     2476260

Min
   Rider length_secs
1 casual           1
2 member           1
```

### Comparing casual vs member ride lengths by day of week

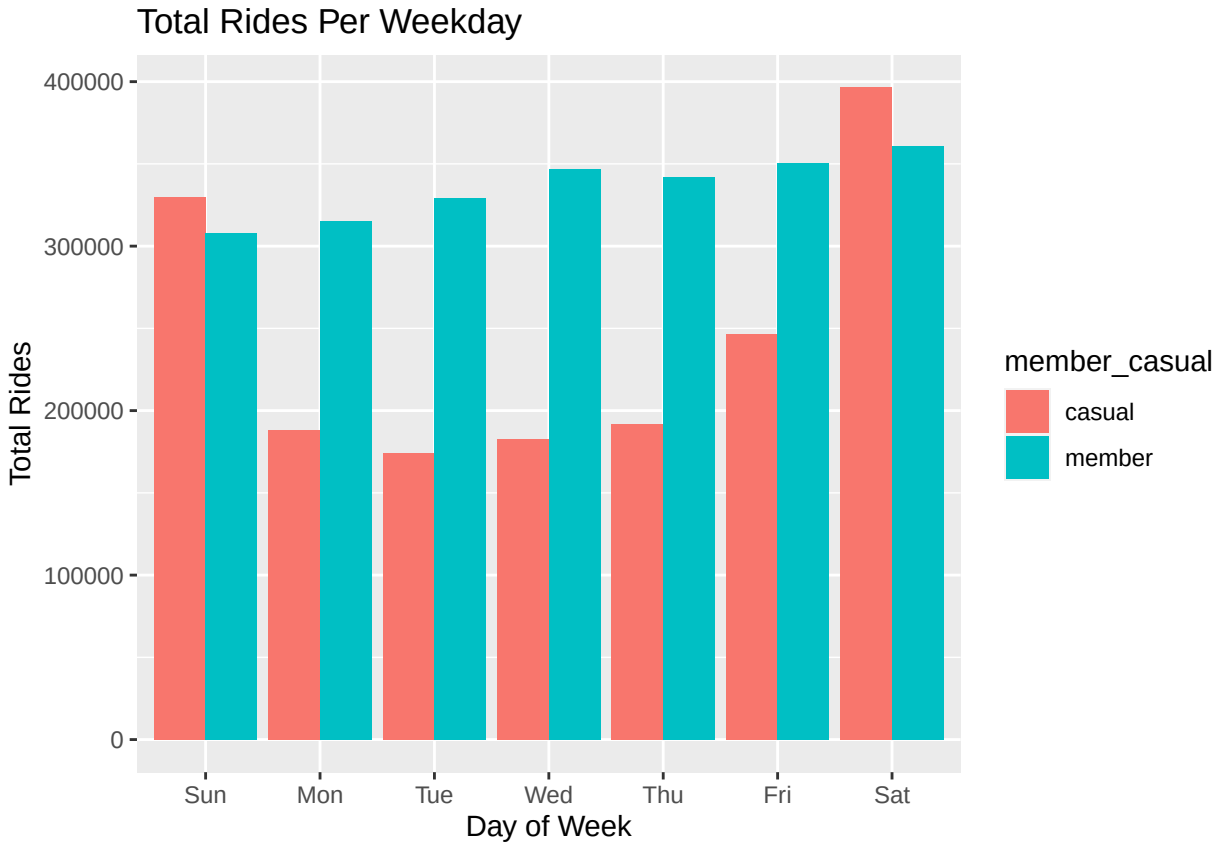
```
Sunday = 1 ... Saturday = 7

    Rider weekday length_secs
1  casual       1   2933.7257
2  member       1   1048.1504
3  casual       2   2533.2514
4  member       2    895.2528
5  casual       3   2280.4173
6  member       3    874.5032
7  casual       4   2299.0170
8  member       4    888.9209
9  casual       5   2394.7949
```

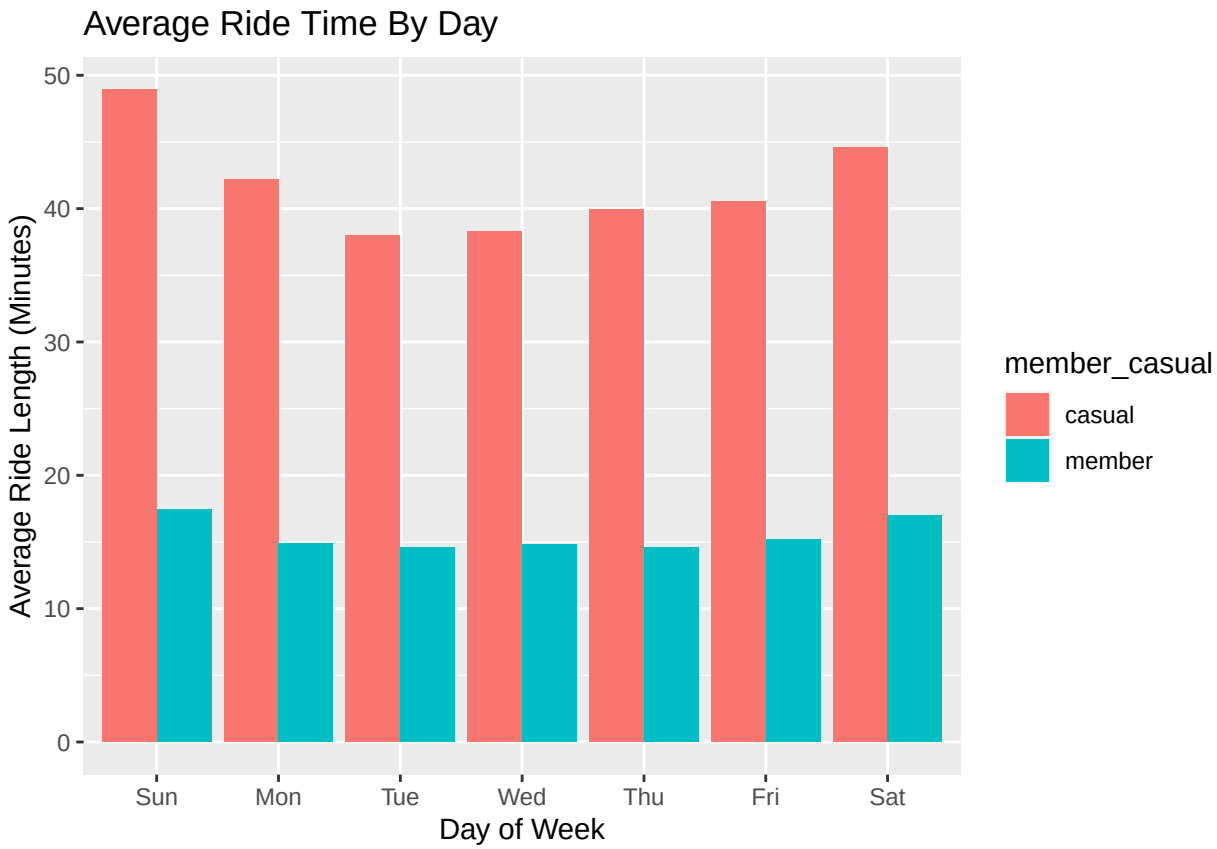
```
10 member       5    876.9482
11 casual       6   2430.4863
12 member       6    912.9767
13 casual       7   2674.2422
14 member       7   1019.6252
```

**Analyze ridership by type and weekday**

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              329920            2934.
##  2 casual        Mon              188393            2533.
##  3 casual        Tue              174391            2280.
##  4 casual        Wed              182540            2299.
##  5 casual        Thu              191733            2395.
##  6 casual        Fri              246620            2430.
##  7 casual        Sat              396397            2674.
##  8 member        Sun              307785            1048.
##  9 member        Mon              315457             895.
## 10 member        Tue              329375             875.
## 11 member        Wed              347149             889.
## 12 member        Thu              341939             877.
## 13 member        Fri              350544             913.
## 14 member        Sat              360531            1020.
```

**Visualize ridership by type**




Average Ride Time By Day
Average Ride Length (Minutes)
50
40
30
20
10
0
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Day of Week
member_casual
casual
member