Data 607 - Assignment 1

Richie R.

Overview

We will be looking into a dataset I personally like, the tripdata dataset published by citibike of their trips. Specifically, we will look at the data corresponding with December 2023 and look at the most popular start and end location for electric bikes (e-bikes). This data is not natively available online as a delimited file, so we will need to follow the below instructions to be able to access this dataset as a dataframe:

This dataset can be found at the url below: https://s3.amazonaws.com/tripdata/index.html

We will be using the '202312-citibike-tripdata.csv.zip' dataset.

1. Download the zipped file to the working directory
2. Unzip the downloaded dataset
3. Read the unzipped dataset as a dataframe
4. Create a filtered_df dataset which is a dataset to only include e-bike rides
5. Group filtered_df by the starting location and count the number of distinct Ride IDs and sort the data by count descending
6. Repeat step 4, but use the ending location rather than the starting location

1. Downloading the dataset to the working directory

```r
file_name       <- "202312-citibike-tripdata.csv"

zip_file_name   <- paste(file_name, ".zip", sep="")

download_url    <- paste("https://s3.amazonaws.com/tripdata/", zip_file_name, sep="")

download.file(download_url, dest=zip_file_name, mode="wb")
```

2. Unzipping the downloaded dataset into the same directory

```r
unzipped_folder = "data_citibike_rides"
unzip(zip_file_name, exdir=unzipped_folder)
```

3. Read in the data as an R dataframe

We'll do so by creating the csv path with the variables established above and then display the first few rows using the head() function.

```r
csv_path = paste(unzipped_folder, "/", file_name, sep="")

citi_df = read.csv(csv_path)

head(citi_df)
```

```
##            ride_id rideable_type          started_at            ended_at
## 1 FB18F431791D6F97  classic_bike 2023-12-07 12:40:22 2023-12-07 12:47:09
## 2 73DF56B794079C50  classic_bike 2023-12-29 13:47:27 2023-12-29 13:54:02
## 3 E3BA5AF851CC1CF0  classic_bike 2023-12-14 19:57:46 2023-12-14 20:15:12
## 4 8F2CBCCB503B0398 electric_bike 2023-12-20 16:55:15 2023-12-20 17:04:03
## 5 A28FFC9585DE8CC5  classic_bike 2023-12-30 14:43:15 2023-12-30 14:56:33
## 6 3AA77BAAC5F3D561  classic_bike 2023-12-21 16:48:10 2023-12-21 16:52:34
##        start_station_name start_station_id             end_station_name
## 1   Allen St & Stanton St          5484.09           Carmine St & 6 Ave
## 2   Carlton Ave & Dean St          4199.12             Union St & 4 Ave
## 3 W 84 St & Amsterdam Ave          7409.04 W 48 St &  Rockefeller Plaza
## 4      E 85 St & York Ave          7146.04  Central Park West & W 85 St
## 5 W 84 St & Amsterdam Ave          7409.04              E 58 St & 3 Ave
## 6       Bergen St & 4 Ave          4322.06           3 Ave & Carroll St
##   end_station_id start_lat start_lng  end_lat   end_lng member_casual
## 1        5763.03  40.72182 -73.98917 40.73039 -74.00215        member
## 2        4175.15  40.68097 -73.97101 40.67727 -73.98282        member
## 3        6626.11  40.78625 -73.97545 40.75777 -73.97929        member
## 4        7354.01  40.77537 -73.94803 40.78476 -73.96986        member
## 5        6762.02  40.78624 -73.97548 40.76096 -73.96724        member
## 6        4143.04  40.68263 -73.98002 40.67703 -73.98650        member
```

4. Create an R dataframe with the subset of the rows which are e-bikes

To do so, we will begin by filtering the dataframe to entries where ridable_type = "electric_bike"

Once that's done, we will only keep the following columns:

- start_station_name
- end_station_name

```r
filtered_df <- subset(
    citi_df
    , rideable_type == "electric_bike"
    , select = c(start_station_name, end_station_name)
)
```

5. With filtered_df, we will look at the most common starting station

To do so, we'll summarize the data grouping by start_station_name and counting the number of times each station shows up.

```r
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
```

```
filtered_df %>%
  group_by(start_station_name) %>%
  summarize(Count=n()) %>%
  arrange(desc(Count))
```

```
## # A tibble: 2,039 x 2
##    start_station_name         Count
##    <chr>                      <int>
##  1 7 Ave & Central Park South   296
##  2 2 Ave & E 29 St              269
##  3 W 21 St & 6 Ave              263
##  4 Broadway & E 14 St           259
##  5 E 33 St & 1 Ave              258
##  6 Broadway & W 58 St           234
##  7 E 17 St & Broadway           234
##  8 11 Ave & W 41 St             228
##  9 W 31 St & 7 Ave              220
## 10 6 Ave & W 33 St              214
## # i 2,029 more rows
```

From the block above, we can see that '7 Ave & Central Park South' was the most popular starting location in December.

6. With fitlered_df, we will look at the most common ending location

```
filtered_df %>%
  group_by(end_station_name) %>%
  summarize(Count=n()) %>%
  arrange(desc(Count))
```

```
## # A tibble: 2,066 x 2
##    end_station_name           Count
##    <chr>                      <int>
##  1 7 Ave & Central Park South   305
##  2 W 21 St & 6 Ave              271
##  3 E 17 St & Broadway           260
##  4 E 33 St & 1 Ave              255
##  5 Broadway & E 14 St           253
##  6 2 Ave & E 29 St              248
##  7 Broadway & W 58 St           240
##  8 W 31 St & 7 Ave              231
##  9 Central Park S & 6 Ave       227
## 10 6 Ave & W 33 St              226
## # i 2,056 more rows
```

Interestingly enough, the same station (‘7 Ave & Central Park South’) was also the most popular ending location!

Conclusion

It appears that ‘7 Ave & Central Park South’ is the most popular start and end station for rides in December 2023.