Cyclistic Case Study

Vincent Liu

2023-08-31

Cyclistic Case Study

Data Cleaning in RStudio

I used RStudio to clean and append the data together. I will use Cyclistic's historical bike trip data from August 2022 to July 2023, which is publicly available here. The Cyclistic Data is available on a month to month basis, with each month's information stored in its own CSV file.

The data includes Ride ID, user type, bike type, and start & end details (times, positions, station names).

I will first load the packages required for the data cleaning process.

```
library(tidyverse) #for wrangling data
```

```
## Warning: package 'tidyverse' was built under R version 4.2.3

## Warning: package 'ggplot2' was built under R version 4.2.3

## Warning: package 'tibble' was built under R version 4.2.3

## Warning: package 'tidyr' was built under R version 4.2.3

## Warning: package 'readr' was built under R version 4.2.3

## Warning: package 'purrr' was built under R version 4.2.3

## Warning: package 'dplyr' was built under R version 4.2.3

## Warning: package 'stringr' was built under R version 4.2.3

## Warning: package 'forcats' was built under R version 4.2.3

## Warning: package 'lubridate' was built under R version 4.2.3

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
```

```
## 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(lubridate) #for wrangling data attributes
library(readr) #for uploading data
```

Next, I will upload the CSV files I have downloaded from each month into a dataframe in R.

```
m7_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202307-divvy-tripdata.csv")
```

```
## Rows: 767650 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m6_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202306-divvy-tripdata.csv")
```

```
## Rows: 719618 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m5_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202305-divvy-tripdata.csv")
```

```
## Rows: 604827 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m4_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202304-divvy-tripdata.csv")
```

```
## Rows: 426590 Columns: 13
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m3_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202303-divvy-tripdata.csv")
```

```
## Rows: 258678 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m2_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202302-divvy-tripdata.csv")
```

```
## Rows: 190445 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m1_2023 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202301-divvy-tripdata.csv")
```

```
## Rows: 190301 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m12_2022 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202212-divvy-tripdata.csv")
```

```
## Rows: 181806 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
```

```
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m11_2022 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202211-divvy-tripdata.csv")
```

```
## Rows: 337735 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m10_2022 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202210-divvy-tripdata.csv")
```

```
## Rows: 558685 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m9_2022 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202209-divvy-publictripdata.csv")
```

```
## Rows: 701339 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

```
m8_2022 <- read_csv("C:/Users/vliu1/Desktop/Google Case Study (Divvy)/202208-divvy-tripdata.csv")
```

```
## Rows: 785932 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, end_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## 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.
```

I will drop the columns from each dataframe that will not be necessary for our analysis, such as start and end station ID as Ill as ride IDs (*start_station_id*, *end_station_id*, *ride_id*).

```
m7_2023 <- m7_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m6_2023 <- m6_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m5_2023 <- m5_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m4_2023 <- m4_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m3_2023 <- m3_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m2_2023 <- m2_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m1_2023 <- m1_2023 %>% select (-c(start_station_id, end_station_id, ride_id))
m12_2022 <- m12_2022 %>% select (-c(start_station_id, end_station_id, ride_id))
m11_2022 <- m11_2022 %>% select (-c(start_station_id, end_station_id, ride_id))
m10_2022 <- m10_2022 %>% select (-c(start_station_id, end_station_id, ride_id))
m9_2022 <- m9_2022 %>% select (-c(start_station_id, end_station_id, ride_id))
m8_2022 <- m8_2022 %>% select (-c(start_station_id, end_station_id, ride_id))
```

Now I will combine each dataframe into one.

```
total_trips <- bind_rows (m7_2023, m6_2023, m5_2023, m4_2023, m3_2023, m2_2023, m1_2023, m12_2022,m11_2
```

I will check if the data is correct. The *member_casual* column should only have casual or member output, and the *rideable_type* column should only have classic bike, docked bike, or electric bike.

```
table(total_trips$member_casual) #should only be casual or member
```

```
##
##  casual  member
## 2169555 3554051
```

```
table(total_trips$rideable_type) #should only result in classic bike, docked bike, or electric bike
```

```
##
##  classic_bike   docked_bike electric_bike
##       2484277        128904       3110425
```

Next, I will format the dates into months, year, and day of the week.

```
total_trips$date <- as.Date(total_trips$started_at)
total_trips$month <- format(as.Date(total_trips$date), "%m")
total_trips$year <- format(as.Date(total_trips$date), "%Y")
total_trips$day_of_week <- format(as.Date(total_trips$date), "%A")
```

Finally, the dataframe is ready for analysis. I will convert the dataframe into a .csv file on my local directory.

```
write.csv(total_trips, "overall-trips.csv", row.names = FALSE)
```