Cyclistic Bike-Share:Case Study

Tade

08/01/2022

Introduction

Google Data Analytics Professional Certificate: Capstone Project

Scenario

In 2016, Cyclistic launched a successful bike-share offering. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system anytime.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes,and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

Cyclistic's finance analysts have concluded that annual members are much more profitable than casual riders. Although the pricing flexibility helps Cyclistic attract more customers, Moreno believes that maximizing the number of annual members will be key to future growth. Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a very good chance to convert casual riders into members. She notes that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs

Goal: Design marketing strategies aimed at converting casual riders into annual members

Primary stakeholders: The director of marketing Lily Moreno and Cyclistic executive team.

Secondary stakeholders: Cyclistic marketing analytics team.

STEP 1: Load packages

tidyverse for data import and wrangling

libridate for date functions

ggplot for visualization

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
library(ggplot2)
library(dplyr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(skimr)
```

Import data

Data collected from the company website which covers a one year period from April 2020 to March 2021. The dataset has been made available under this license [license]agreement(https://ride.divvybikes.com/data-license-agreement) and data source .

In the chunk below, i will use the `read_csv()` function to import data from a .csv in the project folder called "hotel_bookings.csv" and save it as a data frame called `hotel_bookings`.

```
apr_2020 <- read_csv("divvy-trip-data/202004-divvy-tripdata.csv")
```

```
## Rows: 84776 Columns: 13
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
may_2020 <- read_csv("divvy-trip-data/202005-divvy-tripdata.csv")
```

```
## Rows: 200274 Columns: 13
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
jun_2020 <- read_csv("divvy-trip-data/202006-divvy-tripdata.csv")
```

```
## Rows: 343005 Columns: 13
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
```

```
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
jul_2020 <- read_csv("divvy-trip-data/202007-divvy-tripdata.csv")
```

```
## Rows: 551480 Columns: 13

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
aug_2020 <- read_csv("divvy-trip-data/202008-divvy-tripdata.csv")
```

```
## Rows: 622361 Columns: 13

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
sep_2020 <- read_csv("divvy-trip-data/202009-divvy-tripdata.csv")
```

```
## Rows: 532958 Columns: 13

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
oct_2020 <- read_csv("divvy-trip-data/202010-divvy-tripdata.csv")
```

```
## Rows: 388653 Columns: 13

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
nov_2020 <- read_csv("divvy-trip-data/202011-divvy-tripdata.csv")
```

```
## Rows: 259716 Columns: 13
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## 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.
```

```
dec_2020 <- read_csv("divvy-trip-data/202012-divvy-tripdata.csv")
```

```
## Rows: 131573 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.
```

```
jan_2021 <- read_csv("divvy-trip-data/202101-divvy-tripdata.csv")
```

```
## Rows: 96834 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.
```

```
feb_2021 <- read_csv("divvy-trip-data/202102-divvy-tripdata.csv")
```

```
## Rows: 49622 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.
```

```
mar_2021 <- read_csv("divvy-trip-data/202103-divvy-tripdata.csv")
```

```
## Rows: 228496 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.
```

STEP 2: Data wrangling, merging, and cleaning

- Compare file name in each of the dataframes
- Check for inconsistencies and make correction
- Marge files to a single file

```
colnames(apr_2020)
```

```
##  [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"
```

```
colnames(may_2020)
```

```
##  [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"
```

```
colnames(jun_2020)
```

```
##  [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"
```

```
colnames(jul_2020)
```

```
##  [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"
```

```
colnames(aug_2020)
```

```
##  [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"
```

```
colnames(sep_2020)
```

```
##  [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"
```

```
colnames(oct_2020)
```

```
##  [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"
```

```
colnames(nov_2020)
```

```
##  [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"
```

```
colnames(dec_2020)
```

```
##  [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"
```

```
colnames(jan_2021)
```

```
##  [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"
```

```
colnames(feb_2021)
```

```
##  [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"
```

```
colnames(mar_2021)
```

```
##  [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"
```

Column names are consistent across the dataframes ,next step is to check the data types.

Checking for data types

```
str(apr_2020)
```

```
## spec_tbl_df [84,776 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:84776] "A847FADBBC638E45" "5405B80E996FF60D" "5DD24A79A4E006F4" "2A59B
##  $ rideable_type     : chr [1:84776] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:84776], format: "2020-04-26 17:45:14" "2020-04-17 17:08:54" ...
##  $ ended_at          : POSIXct[1:84776], format: "2020-04-26 18:12:03" "2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:84776] "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie S
##  $ start_station_id  : num [1:84776] 86 503 142 216 125 173 35 434 627 377 ...
##  $ end_station_name  : chr [1:84776] "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave & M
##  $ end_station_id    : num [1:84776] 152 499 255 657 323 35 635 382 359 508 ...
##  $ start_lat         : num [1:84776] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:84776] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:84776] 41.9 41.9 41.9 41.9 42 ...
##  $ end_lng           : num [1:84776] -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:84776] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(may_2020)
```

```
## spec_tbl_df [200,274 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:200274] "02668AD35674B983" "7A50CCAF1EDDB28F" "2FFCDFDB91FE9A52" "5899
##  $ rideable_type     : chr [1:200274] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:200274], format: "2020-05-27 10:03:52" "2020-05-25 10:47:11" ...
##  $ ended_at          : POSIXct[1:200274], format: "2020-05-27 10:16:49" "2020-05-25 11:05:40" ...
##  $ start_station_name: chr [1:200274] "Franklin St & Jackson Blvd" "Clark St & Wrightwood Ave" "Kedz
##  $ start_station_id  : num [1:200274] 36 340 260 251 261 206 261 180 331 219 ...
##  $ end_station_name  : chr [1:200274] "Wabash Ave & Grand Ave" "Clark St & Leland Ave" "Kedzie Ave &
##  $ end_station_id    : num [1:200274] 199 326 260 157 206 22 261 180 300 305 ...
##  $ start_lat         : num [1:200274] 41.9 41.9 41.9 42 41.9 ...
##  $ start_lng         : num [1:200274] -87.6 -87.6 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:200274] 41.9 42 41.9 41.9 41.8 ...
##  $ end_lng           : num [1:200274] -87.6 -87.7 -87.7 -87.6 -87.6 ...
##  $ member_casual     : chr [1:200274] "member" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
```

```
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(jun_2020)
```

```
## spec_tbl_df [343,005 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:343005] "8CD5DE2C2B6C4CFC" "9A191EB2C751D85D" "F37D14B0B5659BCF" "C412
##  $ rideable_type     : chr [1:343005] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:343005], format: "2020-06-13 23:24:48" "2020-06-26 07:26:10" ...
##  $ ended_at          : POSIXct[1:343005], format: "2020-06-13 23:36:55" "2020-06-26 07:31:58" ...
##  $ start_station_name: chr [1:343005] "Wilton Ave & Belmont Ave" "Federal St & Polk St" "Daley Cente
##  $ start_station_id  : num [1:343005] 117 41 81 303 327 327 41 115 338 84 ...
##  $ end_station_name  : chr [1:343005] "Damen Ave & Clybourn Ave" "Daley Center Plaza" "State St & Ha
##  $ end_station_id    : num [1:343005] 163 81 5 294 117 117 81 303 164 53 ...
##  $ start_lat         : num [1:343005] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:343005] -87.7 -87.6 -87.6 -87.6 -87.7 ...
##  $ end_lat           : num [1:343005] 41.9 41.9 41.9 42 41.9 ...
##  $ end_lng           : num [1:343005] -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:343005] "casual" "member" "member" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(jul_2020)
```

```
## spec_tbl_df [551,480 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:551480] "762198876D69004D" "BEC9C9FBA0D4CF1B" "D2FD8EA432C77EC1" "54AE
##  $ rideable_type     : chr [1:551480] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:551480], format: "2020-07-09 15:22:02" "2020-07-24 23:56:30" ...
```

```
##  $ ended_at          : POSIXct[1:551480], format: "2020-07-09 15:25:52" "2020-07-25 00:20:17" ...
##  $ start_station_name: chr [1:551480] "Ritchie Ct & Banks St" "Halsted St & Roscoe St" "Lake Shore D
##  $ start_station_id  : num [1:551480] 180 299 329 181 268 635 113 211 176 31 ...
##  $ end_station_name  : chr [1:551480] "Wells St & Evergreen Ave" "Broadway & Ridge Ave" "Clark St & W
##  $ end_station_id    : num [1:551480] 291 461 156 94 301 289 140 31 191 142 ...
##  $ start_lat         : num [1:551480] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:551480] -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ end_lat           : num [1:551480] 41.9 42 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:551480] -87.6 -87.7 -87.6 -87.6 -87.6 ...
##  $ member_casual     : chr [1:551480] "member" "member" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(aug_2020)
```

```
## spec_tbl_df [622,361 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:622361] "322BD23D287743ED" "2A3AEF1AB9054D8B" "67DC1D133E8B5816" "C79F
##  $ rideable_type     : chr [1:622361] "docked_bike" "electric_bike" "electric_bike" "electric_bike"
##  $ started_at        : POSIXct[1:622361], format: "2020-08-20 18:08:14" "2020-08-27 18:46:04" ...
##  $ ended_at          : POSIXct[1:622361], format: "2020-08-20 18:17:51" "2020-08-27 19:54:51" ...
##  $ start_station_name: chr [1:622361] "Lake Shore Dr & Diversey Pkwy" "Michigan Ave & 14th St" "Colu
##  $ start_station_id  : num [1:622361] 329 168 195 81 658 658 196 67 153 177 ...
##  $ end_station_name  : chr [1:622361] "Clark St & Lincoln Ave" "Michigan Ave & 14th St" "State St & L
##  $ end_station_id    : num [1:622361] 141 168 44 47 658 658 49 229 225 305 ...
##  $ start_lat         : num [1:622361] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:622361] -87.6 -87.6 -87.6 -87.6 -87.7 ...
##  $ end_lat           : num [1:622361] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:622361] -87.6 -87.6 -87.6 -87.6 -87.7 ...
##  $ member_casual     : chr [1:622361] "member" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
```

```
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(sep_2020)
```

```
## spec_tbl_df [532,958 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:532958] "2B22BD5F95FB2629" "A7FB70B4AFC6CAF2" "86057FA01BAC778E" "57F6
##  $ rideable_type     : chr [1:532958] "electric_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:532958], format: "2020-09-17 14:27:11" "2020-09-17 15:07:31" ...
##  $ ended_at          : POSIXct[1:532958], format: "2020-09-17 14:44:24" "2020-09-17 15:07:45" ...
##  $ start_station_name: chr [1:532958] "Michigan Ave & Lake St" "W Oakdale Ave & N Broadway" "W Oakda
##  $ start_station_id  : num [1:532958] 52 NA NA 246 24 94 291 NA NA NA ...
##  $ end_station_name  : chr [1:532958] "Green St & Randolph St" "W Oakdale Ave & N Broadway" "W Oakda
##  $ end_station_id    : num [1:532958] 112 NA NA 249 24 NA 256 NA NA NA ...
##  $ start_lat         : num [1:532958] 41.9 41.9 41.9 42 41.9 ...
##  $ start_lng         : num [1:532958] -87.6 -87.6 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:532958] 41.9 41.9 41.9 42 41.9 ...
##  $ end_lng           : num [1:532958] -87.6 -87.6 -87.6 -87.6 -87.6 ...
##  $ member_casual     : chr [1:532958] "casual" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(oct_2020)
```

```
## spec_tbl_df [388,653 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:388653] "ACB6B40CF5B9044C" "DF450C72FD109C01" "B6396B54A15AC0DF" "44A4
##  $ rideable_type     : chr [1:388653] "electric_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:388653], format: "2020-10-31 19:39:43" "2020-10-31 23:50:08" ...
##  $ ended_at          : POSIXct[1:388653], format: "2020-10-31 19:57:12" "2020-11-01 00:04:16" ...
##  $ start_station_name: chr [1:388653] "Lakeview Ave & Fullerton Pkwy" "Southport Ave & Waveland Ave"
##  $ start_station_id  : num [1:388653] 313 227 102 165 190 359 313 125 NA 174 ...
##  $ end_station_name  : chr [1:388653] "Rush St & Hubbard St" "Kedzie Ave & Milwaukee Ave" "University
##  $ end_station_id    : num [1:388653] 125 260 423 256 185 53 125 313 199 635 ...
##  $ start_lat         : num [1:388653] 41.9 41.9 41.8 42 41.9 ...
##  $ start_lng         : num [1:388653] -87.6 -87.7 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num [1:388653] 41.9 41.9 41.8 42 41.9 ...
```

```
##  $ end_lng           : num [1:388653] -87.6 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:388653] "casual" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(nov_2020)
```

```
## spec_tbl_df [259,716 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:259716] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E533
##  $ rideable_type     : chr [1:259716] "electric_bike" "electric_bike" "electric_bike" "electric_bike
##  $ started_at        : POSIXct[1:259716], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" ...
##  $ ended_at          : POSIXct[1:259716], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" ...
##  $ start_station_name: chr [1:259716] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake Shor
##  $ start_station_id  : num [1:259716] 110 672 76 659 2 72 76 NA 58 394 ...
##  $ end_station_name  : chr [1:259716] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federal S
##  $ end_station_id    : num [1:259716] 211 29 41 185 2 76 72 NA 288 273 ...
##  $ start_lat         : num [1:259716] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:259716] -87.6 -87.6 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:259716] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:259716] -87.6 -87.7 -87.6 -87.7 -87.6 ...
##  $ member_casual     : chr [1:259716] "casual" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(dec_2020)
```

```
## spec_tbl_df [131,573 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:131573] "70B6A9A437D4C30D" "158A465D4E74C54A" "5262016E0F1F2F9A" "BE11
##  $ rideable_type     : chr [1:131573] "classic_bike" "electric_bike" "electric_bike" "electric_bike"
##  $ started_at        : POSIXct[1:131573], format: "2020-12-27 12:44:29" "2020-12-18 17:37:15" ...
##  $ ended_at          : POSIXct[1:131573], format: "2020-12-27 12:55:06" "2020-12-18 17:44:19" ...
##  $ start_station_name: chr [1:131573] "Aberdeen St & Jackson Blvd" NA NA NA ...
##  $ start_station_id  : chr [1:131573] "13157" NA NA NA ...
##  $ end_station_name  : chr [1:131573] "Desplaines St & Kinzie St" NA NA NA ...
##  $ end_station_id    : chr [1:131573] "TA1306000003" NA NA NA ...
##  $ start_lat         : num [1:131573] 41.9 41.9 41.9 41.9 41.8 ...
##  $ start_lng         : num [1:131573] -87.7 -87.7 -87.7 -87.7 -87.6 ...
##  $ end_lat           : num [1:131573] 41.9 41.9 41.9 41.9 41.8 ...
##  $ end_lng           : num [1:131573] -87.6 -87.7 -87.7 -87.7 -87.6 ...
##  $ member_casual     : chr [1:131573] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_character(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_character(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(jan_2021)
```

```
## spec_tbl_df [96,834 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:96834] "E19E6F1B8D4C42ED" "DC88F20C2C55F27F" "EC45C94683FE3F27" "4FA453
##  $ rideable_type     : chr [1:96834] "electric_bike" "electric_bike" "electric_bike" "electric_bike"
##  $ started_at        : POSIXct[1:96834], format: "2021-01-23 16:14:19" "2021-01-27 18:43:08" ...
##  $ ended_at          : POSIXct[1:96834], format: "2021-01-23 16:24:44" "2021-01-27 18:47:12" ...
##  $ start_station_name: chr [1:96834] "California Ave & Cortez St" "California Ave & Cortez St" "Cali
##  $ start_station_id  : chr [1:96834] "17660" "17660" "17660" "17660" ...
##  $ end_station_name  : chr [1:96834] NA NA NA NA ...
##  $ end_station_id    : chr [1:96834] NA NA NA NA ...
##  $ start_lat         : num [1:96834] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:96834] -87.7 -87.7 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num [1:96834] 41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num [1:96834] -87.7 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr [1:96834] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
```

```
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_character(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_character(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(feb_2021)
```

```
## spec_tbl_df [49,622 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:49622] "89E7AA6C29227EFF" "0FEFDE2603568365" "E6159D746B2DBB91" "B32D3
##  $ rideable_type     : chr [1:49622] "classic_bike" "classic_bike" "electric_bike" "classic_bike" ..
##  $ started_at        : POSIXct[1:49622], format: "2021-02-12 16:14:56" "2021-02-14 17:52:38" ...
##  $ ended_at          : POSIXct[1:49622], format: "2021-02-12 16:21:43" "2021-02-14 18:12:09" ...
##  $ start_station_name: chr [1:49622] "Glenwood Ave & Touhy Ave" "Glenwood Ave & Touhy Ave" "Clark St
##  $ start_station_id  : chr [1:49622] "525" "525" "KA1503000012" "637" ...
##  $ end_station_name  : chr [1:49622] "Sheridan Rd & Columbia Ave" "Bosworth Ave & Howard St" "State S
##  $ end_station_id    : chr [1:49622] "660" "16806" "TA1305000029" "TA1305000034" ...
##  $ start_lat         : num [1:49622] 42 42 41.9 41.9 41.8 ...
##  $ start_lng         : num [1:49622] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:49622] 42 42 41.9 41.9 41.8 ...
##  $ end_lng           : num [1:49622] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ member_casual     : chr [1:49622] "member" "casual" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_character(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_character(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(mar_2021)
```

```
## spec_tbl_df [228,496 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:228496] "CFA86D4455AA1030" "30D9DC61227D1AF3" "846D87A15682A284" "994D
##  $ rideable_type     : chr [1:228496] "classic_bike" "classic_bike" "classic_bike" "classic_bike" ..
##  $ started_at        : POSIXct[1:228496], format: "2021-03-16 08:32:30" "2021-03-28 01:26:28" ...
##  $ ended_at          : POSIXct[1:228496], format: "2021-03-16 08:36:34" "2021-03-28 01:36:55" ...
```

```
##  $ start_station_name: chr [1:228496] "Humboldt Blvd & Armitage Ave" "Humboldt Blvd & Armitage Ave" "
##  $ start_station_id  : chr [1:228496] "15651" "15651" "15443" "TA1308000021" ...
##  $ end_station_name  : chr [1:228496] "Stave St & Armitage Ave" "Central Park Ave & Bloomingdale Ave
##  $ end_station_id    : chr [1:228496] "13266" "18017" "TA1308000043" "13323" ...
##  $ start_lat         : num [1:228496] 41.9 41.9 41.8 42 42 ...
##  $ start_lng         : num [1:228496] -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num [1:228496] 41.9 41.9 41.8 42 42.1 ...
##  $ end_lng           : num [1:228496] -87.7 -87.7 -87.6 -87.6 -87.7 ...
##  $ member_casual     : chr [1:228496] "casual" "casual" "casual" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_character(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_character(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Correct inconsistencies with data types

Convert start_station_id and end_station_id to character so that they can stack correctly.

Start_station_id and end_station_id Columns apr_2020 - nov_2020 dataframes are doubles.

```
apr_2020 <-  mutate(apr_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
may_2020 <-  mutate(may_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
jun_2020 <-  mutate(jun_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
jul_2020 <-  mutate(jul_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
aug_2020 <-  mutate(aug_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
sep_2020 <-  mutate(sep_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
oct_2020 <-  mutate(oct_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
nov_2020 <-  mutate(nov_2020, start_station_id = as.character(start_station_id)
                    ,end_station_id = as.character(end_station_id))
```

Stack individual quarter's data frames into one big data frame

```
all_trips <- bind_rows(apr_2020, may_2020, jun_2020, jul_2020, aug_2020, sep_2020, oct_2020, nov_2020
                       , dec_2020, jan_2021, feb_2021, mar_2021)
```

```
head(all_trips)
```

```
## # A tibble: 6 x 13
##   ride_id rideable_type started_at          ended_at            start_station_n~
##   <chr>   <chr>         <dttm>              <dttm>              <chr>
## 1 A847FA~ docked_bike   2020-04-26 17:45:14 2020-04-26 18:12:03 Eckhart Park
## 2 5405B8~ docked_bike   2020-04-17 17:08:54 2020-04-17 17:17:03 Drake Ave & Ful~
## 3 5DD24A~ docked_bike   2020-04-01 17:54:13 2020-04-01 18:08:36 McClurg Ct & Er~
## 4 2A59BB~ docked_bike   2020-04-07 12:50:19 2020-04-07 13:02:31 California Ave ~
## 5 27AD30~ docked_bike   2020-04-18 10:22:59 2020-04-18 11:15:54 Rush St & Hubba~
## 6 356216~ docked_bike   2020-04-30 17:55:47 2020-04-30 18:01:11 Mies van der Ro~
## # ... with 8 more variables: start_station_id <chr>, end_station_name <chr>,
## #   end_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, member_casual <chr>
```

Remove irrelevant data from the data frame

I will remove ride_id, station_id and end_station_id. In the code chunk below I use the select() and c() functions to remove irrelevant data columns from the dataframe.

```
all_trips <- all_trips %>%
  select(-c(ride_id, start_station_id, end_station_id))
```

Clean up data

Checking the new table that i created

```
skim_without_charts(all_trips)
```

Table 1: Data summary

Name	all_trips
Number of rows	3489748
Number of columns	10
Column type frequency:	
character	4
numeric	4
POSIXct	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
rideable_type	0	1.00	11	13	0	3	0
start_station_name	122175	0.96	10	53	0	708	0
end_station_name	143242	0.96	10	53	0	706	0
member_casual	0	1.00	6	6	0	2	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
start_lat	0	1	41.90	0.04	41.64	41.88	41.90	41.93	42.08
start_lng	0	1	-87.64	0.03	-87.87	-87.66	-87.64	-87.63	-87.52
end_lat	4738	1	41.90	0.04	41.54	41.88	41.90	41.93	42.16
end_lng	4738	1	-87.65	0.03	-88.07	-87.66	-87.64	-87.63	-87.44

Variable type: POSIXct

skim_variable	n_missing	complete_rate	min	max	median	n_unique
started_at	0	1	2020-04-01 00:00:30	2021-03-31 23:59:08	2020-08-29 14:50:36	3040228
ended_at	0	1	2020-04-01 00:10:45	2021-04-06 11:00:11	2020-08-29 15:21:13	3027775

```
head(all_trips)
```

```
## # A tibble: 6 x 10
##   rideable_type started_at          ended_at            start_station_name
##   <chr>         <dttm>              <dttm>              <chr>
## 1 docked_bike   2020-04-26 17:45:14 2020-04-26 18:12:03 Eckhart Park
## 2 docked_bike   2020-04-17 17:08:54 2020-04-17 17:17:03 Drake Ave & Fullerton A~
## 3 docked_bike   2020-04-01 17:54:13 2020-04-01 18:08:36 McClurg Ct & Erie St
## 4 docked_bike   2020-04-07 12:50:19 2020-04-07 13:02:31 California Ave & Divisi~
## 5 docked_bike   2020-04-18 10:22:59 2020-04-18 11:15:54 Rush St & Hubbard St
## 6 docked_bike   2020-04-30 17:55:47 2020-04-30 18:01:11 Mies van der Rohe Way &~
## # ... with 6 more variables: end_station_name <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, member_casual <chr>
```

```
tail(all_trips)
```

```
## # A tibble: 6 x 10
##   rideable_type started_at          ended_at            start_station_name
##   <chr>         <dttm>              <dttm>              <chr>
## 1 electric_bike 2021-03-14 01:59:38 2021-03-14 03:13:09 Larrabee St & Armitage ~
## 2 docked_bike   2021-03-20 14:58:56 2021-03-20 17:22:47 Michigan Ave & Oak St
## 3 classic_bike  2021-03-02 11:35:10 2021-03-02 11:43:37 Kingsbury St & Kinzie St
## 4 classic_bike  2021-03-09 11:07:36 2021-03-09 11:49:11 Michigan Ave & Oak St
## 5 classic_bike  2021-03-01 18:11:57 2021-03-01 18:18:37 Kingsbury St & Kinzie St
## 6 electric_bike 2021-03-26 17:58:14 2021-03-26 18:06:43 <NA>
## # ... with 6 more variables: end_station_name <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, member_casual <chr>
```

Adding Data

I want to add new columns for 'date', 'month', 'day', 'year, 'day_of_week', and 'hour'. These new columns will allow me to aggregate the data for different time periods of each ride

The default format is yyyy-mm-dd

```
all_trips$date <- as.Date(all_trips$started_at)
all_trips$month <- format(as.Date(all_trips$date), "%m")
all_trips$day <- format(as.Date(all_trips$date), "%d")
```

```
all_trips$year <- format(as.Date(all_trips$date), "%Y")
all_trips$day_of_week <- format(as.Date(all_trips$date), "%A")
```

Adding a new variable called "ride_length" to all_trips

```
all_trips$ride_length <- difftime(all_trips$ended_at, all_trips$started_at)
```

inspect the structure of the columns

```
str(all_trips)
```

```
## tibble [3,489,748 x 16] (S3: tbl_df/tbl/data.frame)
##  $ rideable_type     : chr [1:3489748] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:3489748], format: "2020-04-26 17:45:14" "2020-04-17 17:08:54" ...
##  $ ended_at          : POSIXct[1:3489748], format: "2020-04-26 18:12:03" "2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:3489748] "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie
##  $ end_station_name  : chr [1:3489748] "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave &
##  $ start_lat         : num [1:3489748] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:3489748] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:3489748] 41.9 41.9 41.9 41.9 42 ...
##  $ end_lng           : num [1:3489748] -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:3489748] "member" "member" "member" "member" ...
##  $ date              : Date[1:3489748], format: "2020-04-26" "2020-04-17" ...
##  $ month             : chr [1:3489748] "04" "04" "04" "04" ...
##  $ day               : chr [1:3489748] "26" "17" "01" "07" ...
##  $ year              : chr [1:3489748] "2020" "2020" "2020" "2020" ...
##  $ day_of_week       : chr [1:3489748] "Sunday" "Friday" "Wednesday" "Tuesday" ...
##  $ ride_length       : 'difftime' num [1:3489748] 1609 489 863 732 ...
##   ..- attr(*, "units")= chr "secs"
```

Convert "ride_length" from Factor to numeric so we can run calculations on the data

```
is.factor(all_trips$ride_length)
```

```
## [1] FALSE
```

```
all_trips$ride_length <- as.numeric(as.character(all_trips$ride_length))
is.numeric(all_trips$ride_length)
```

```
## [1] TRUE
```

Remove data error

The dataframe includes a few hundred entries when bikes were taken out of docks and checked for quality by Divvy or ride_length was negative

I will create a new version of the dataframe (v2) since data is being removed

```
all_trips_v2 <- all_trips[!(all_trips$ride_length<0),]
```

Remove NA values

```
all_trips_v2 <- na.omit(all_trips_v2)
```

STEP 3: Analyze

Descriptive analysis on ride_length (all figures in seconds)

```
summary(all_trips_v2$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     485     885    1683    1615 3523202
```

Compare members and casual users

straight average (total ride length / rides)

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = mean)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                2726.5873
## 2                     member                 955.9582
```

midpoint number in the ascending array of ride lengths

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = median)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                      1298
## 2                     member                       696
```

Longest ride

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = max)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                   3341033
## 2                     member                   3523202
```

Shortest ride

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = min)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                         0
## 2                     member                         0
```

The average ride time by each day for members vs casual users

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual + all_trips_v2$day_of_week, FUN = mean)
```

```
##    all_trips_v2$member_casual all_trips_v2$day_of_week all_trips_v2$ride_length
```

```
## 1                      casual                  Friday             2594.6604
## 2                      member                  Friday              933.9941
## 3                      casual                  Monday             2728.8925
## 4                      member                  Monday              905.5943
## 5                      casual                Saturday             2837.8823
## 6                      member                Saturday             1060.9659
## 7                      casual                  Sunday             3067.5597
## 8                      member                  Sunday             1088.5590
## 9                      casual                Thursday             2614.9029
## 10                     member                Thursday              901.7930
## 11                     casual                 Tuesday             2455.1566
## 12                     member                 Tuesday              898.2407
## 13                     casual               Wednesday             2452.5521
## 14                     member               Wednesday              902.8226
```

The days of the week are out of order. Let's fix that

```
all_trips_v2$day_of_week <- ordered(all_trips_v2$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "
```

Let's run the average ride time by each day for members vs casual users

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual + all_trips_v2$day_of_week, FUN = mean)
```

```
##    all_trips_v2$member_casual all_trips_v2$day_of_week all_trips_v2$ride_length
## 1                      casual                  Sunday             3067.5597
## 2                      member                  Sunday             1088.5590
## 3                      casual                  Monday             2728.8925
## 4                      member                  Monday              905.5943
## 5                      casual                 Tuesday             2455.1566
## 6                      member                 Tuesday              898.2407
## 7                      casual               Wednesday             2452.5521
## 8                      member               Wednesday              902.8226
## 9                      casual                Thursday             2614.9029
## 10                     member                Thursday              901.7930
## 11                     casual                  Friday             2594.6604
## 12                     member                  Friday              933.9941
## 13                     casual                Saturday             2837.8823
## 14                     member                Saturday             1060.9659
```

Analyze ridership data by type and weekday

```
all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%  #creates weekday field using wday()
  group_by(member_casual, weekday) %>%  #groups by usertype and weekday
  summarise(number_of_rides = n()                              #calculates the number of rides and average
  ,average_duration = mean(ride_length)) %>%        # calculates the average duration
  arrange(weekday, member_casual)
```

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

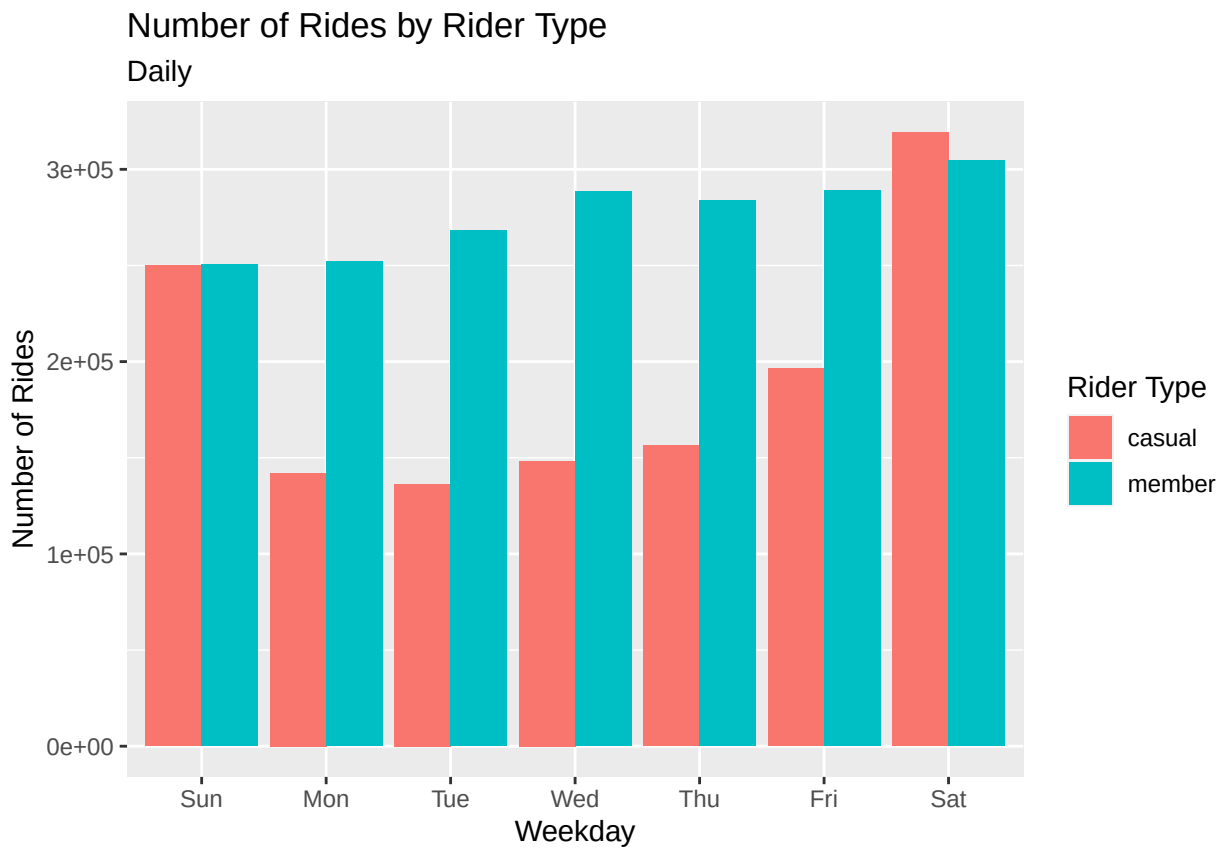
```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
```

```
##  1 casual        Sun              249837           3068.
##  2 member        Sun              250547           1089.
##  3 casual        Mon              142071           2729.
##  4 member        Mon              251938            906.
##  5 casual        Tue              136258           2455.
##  6 member        Tue              268267            898.
##  7 casual        Wed              148401           2453.
##  8 member        Wed              288443            903.
##  9 casual        Thu              156253           2615.
## 10 member        Thu              283783            902.
## 11 casual        Fri              196542           2595.
## 12 member        Fri              288961            934.
## 13 casual        Sat              319124           2838.
## 14 member        Sat              304684           1061.
```

Let's visualize the number of rides by rider type

```
all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Number of Rides by Rider Type", subtitle = "Daily", fill = "Rider Type") +
  xlab("Weekday") + ylab("Number of Rides")
```

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

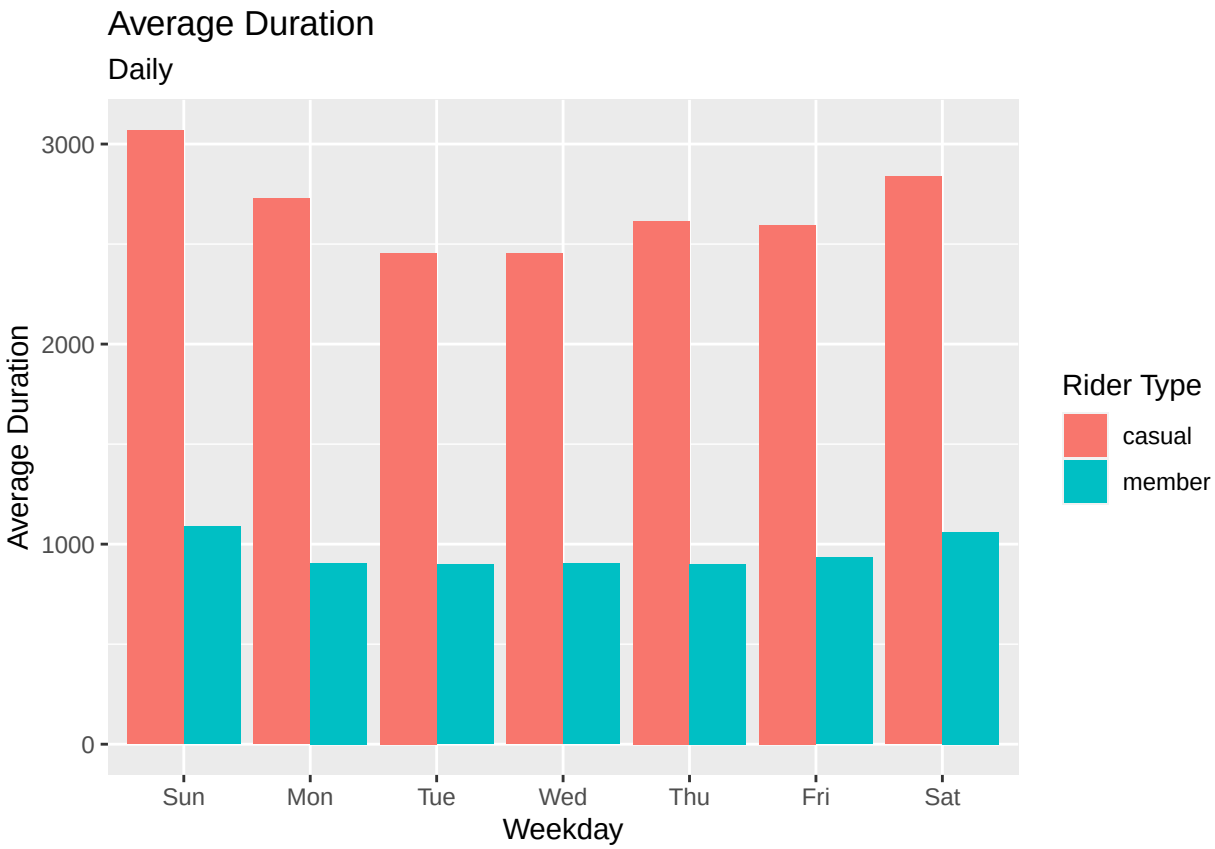


From the bar graph that the member riders have higher number of rides on weekdays and casual riders are higher on Saturdays. Saturdays have the highest number of riders for both casual and member.

Let's create a visualization for average duration

```
all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)  %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Average Duration", subtitle = "Daily", fill = "Rider Type") +
  xlab("Weekday") + ylab("Average Duration")
```

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



Casual riders have higher average duration than member riders. During weekends we see a higher average duration than weekdays for both rider types. For member riders, there's a small change of duration between weekdays and weekends but a higher change for casual riders.

Analyze ridership data by type and month

```
all_trips_v2 %>%
  mutate(monthly = month(started_at, label = TRUE)) %>%  #creates monthly field using month()
  group_by(member_casual, monthly, year) %>%  #groups by usertype and monthly
  summarise(number_of_rides = n()                              #calculates the number of rides and average
  ,average_duration = mean(ride_length)) %>%        # calculates the average duration
  arrange(year, monthly, member_casual)                                 # sorts
```

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

```
## # A tibble: 24 x 5
## # Groups:   member_casual, monthly [24]
##    member_casual monthly year  number_of_rides average_duration
##    <chr>         <ord>   <chr>           <int>            <dbl>
##  1 casual        Apr     2020            23570            4349.
##  2 member        Apr     2020            61056            1282.
##  3 casual        May     2020            86699            3036.
##  4 member        May     2020           113083            1175.
##  5 casual        Jun     2020           154342            3074.
##  6 member        Jun     2020           187727            1112.
##  7 casual        Jul     2020           268126            3557.
```

```
##  8 member        Jul     2020          280556           1054.
##  9 casual        Aug     2020          282050           2654.
## 10 member        Aug     2020          323843            994.
## # ... with 14 more rows
```

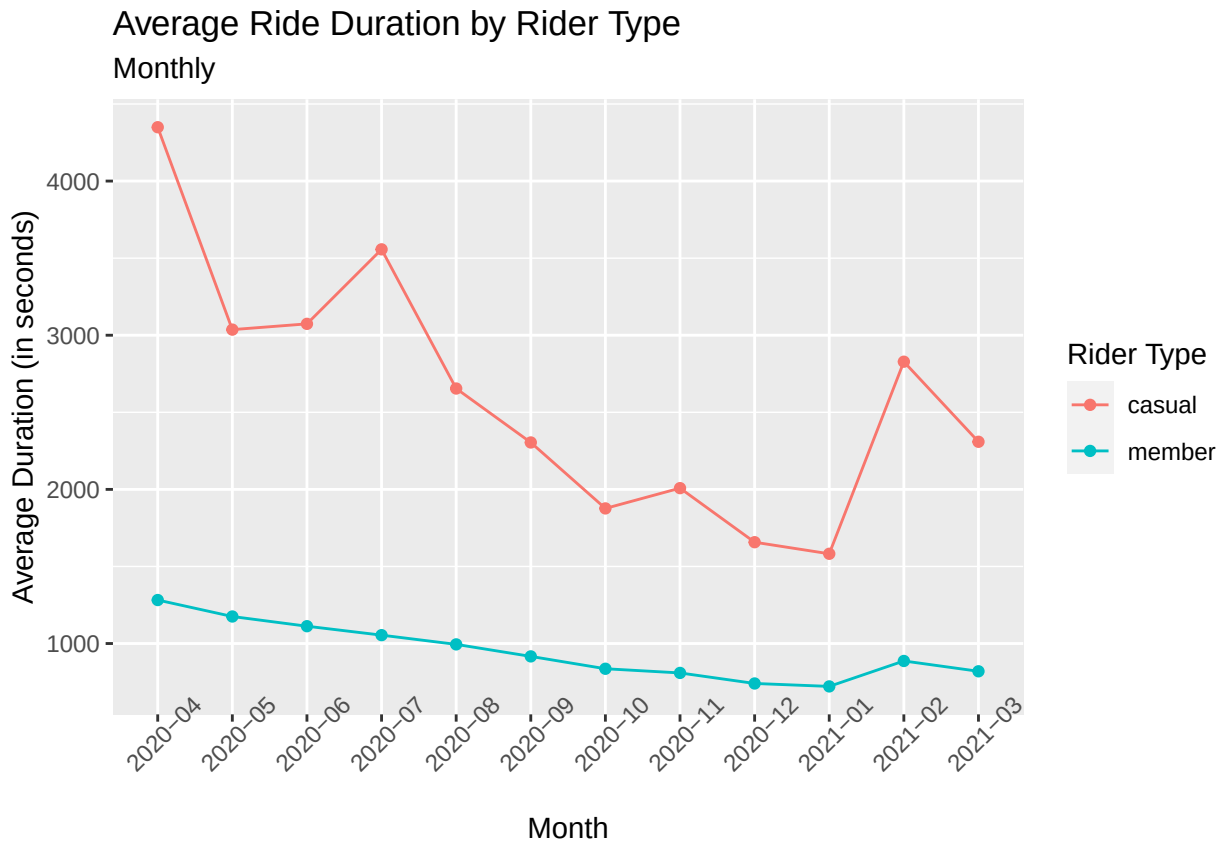
Let's create a visualization for the number of rides by rider type monthly

```
all_trips_v2 %>%
  mutate(monthly = format(as.Date(started_at), "%Y-%m")) %>%
  group_by(member_casual, monthly, year) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length), .groups = "keep") %>%
  arrange(year, monthly, member_casual)  %>%
  ggplot(aes(x = monthly, y = number_of_rides, group = member_casual)) +
  geom_line(aes(color=member_casual)) +
  geom_point(aes(color=member_casual)) +
  theme(axis.text.x = element_text(angle = 45)) +
  labs(title = "Rider type and the number of rides", subtitle = "Monthly", color = "Rider Type") +
  xlab("Month") + ylab("Number of Rides")
```

* Month of August 2020 had the hightest number of ride for both casual and member * Month of February 2021 had the lowest number of ride for both casual and member * All through the year, member riders has highest number of rides.

Let's create a visualization for average duration monthly

```
all_trips_v2 %>%
  mutate(monthly = format(as.Date(started_at), "%Y-%m")) %>%
  group_by(member_casual, monthly, year) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length), .groups = "keep") %>%
  arrange(year, monthly, member_casual)  %>%
  ggplot(aes(x = monthly, y = average_duration, group = member_casual)) +
  geom_line(aes(color=member_casual)) +
  geom_point(aes(color=member_casual)) +
  theme(axis.text.x = element_text(angle = 45)) +
  labs(title = "Average Ride Duration by Rider Type", subtitle = "Monthly", color = "Rider Type") +
  xlab("Month") + ylab("Average Duration (in seconds)")
```



- Member riders, the highest average duration was in April 2020 and lowest was in January 2021
- Casual riders, the highest average duration was in April 2020 and lowest was in January 2021

What we found during this analysis:

- The casual riders rent the bikes more during the weekends while member riders are renting consistently throughout the week.
- The casual riders' average duration is almost triple the time of the member riders.
- Both casual and member riders ride the bikes more during summer and less during winter
- Both casual and member riders ride the bike longer during summer and shorter during winter

STEP 4 Recommendations

- Make a “weekend campaign” membership sign-up emphasizing how much they can save if they convert into member riders.

*Make an ad that focus on saving as member for longer ride duration to get casual sign-up as member riders.

*Make ads and campaigns during summer because there are higher numbers of casual riders at that time.