

Case Study - Cyclistic Bike share (2019-2020)

Mohd Ali Ansari

14/06/2021

Problem Statement

How do annual members and casual riders use Cyclistic bikes differently?

Objective

This documents contains all the cleaning steps taken in order to clean and transform the data sets and preparing it for next step i.e. analysis Cleaning data set is from the year 2019-2020

About dataset

Cyclistic's historical trip data to analyze and identify trends. Download data from [here](#). (Note: The datasets have a different name because Cyclistic is a fictional company. For the purposes of this case study, the datasets are appropriate and will enable to answer the business questions. The data has been made available by Motivate International Inc. under this license.)

This is public data that can use to explore how different customer types are using Cyclistic bikes. But note that data-privacy issues prohibit from using riders' personally identifiable information. This means that one wouldn't connect pass purchases to credit card numbers to determine if casual riders live in the Cyclistic service area or if they have purchased multiple single passes

```
library(tidyverse)
library(tidyr)
library(dplyr)
library(geosphere)
library(lubridate)
```

Importing the libraries

Loading the dataset The dataset is available in csv format after downloading so we will clean it simultaneously for merging them into one fiscal year Since the companies year starts from April month we will load all the data sets and then after checking for consistency we will merge them to make a complete one year tripdata

```
q2_2019 <- read.csv("Divvy_Trips_2019_Q2.csv")
q3_2019 <- read.csv("Divvy_Trips_2019_Q3.csv")
q4_2019 <- read.csv("Divvy_Trips_2019_Q4.csv")
q1_2020 <- read.csv("Divvy_Trips_2020_Q1.csv")
```

Checking for consistency We have to check for consistency as we have to merge all the datasets into one dataset. So the column names and columns data type should be same for all the datasets

Checking for Column name

```
## [1] "X01...Rental.Details.Rental.ID"
## [2] "X01...Rental.Details.Local.Start.Time"
## [3] "X01...Rental.Details.Local.End.Time"
## [4] "X01...Rental.Details.Bike.ID"
## [5] "X01...Rental.Details.Duration.In.Seconds.Uncapped"
## [6] "X03...Rental.Start.Station.ID"
## [7] "X03...Rental.Start.Station.Name"
## [8] "X02...Rental.End.Station.ID"
## [9] "X02...Rental.End.Station.Name"
## [10] "User.Type"
## [11] "Member.Gender"
## [12] "X05...Member.Details.Member.Birthday.Year"

## [1] "trip_id"          "start_time"       "end_time"
## [4] "bikeid"           "tripduration"     "from_station_id"
## [7] "from_station_name" "to_station_id"    "to_station_name"
## [10] "usertype"         "gender"           "birthyear"

## [1] "trip_id"          "start_time"       "end_time"
## [4] "bikeid"           "tripduration"     "from_station_id"
## [7] "from_station_name" "to_station_id"    "to_station_name"
## [10] "usertype"         "gender"           "birthyear"

## [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"
```

There are total 13 columns in every dataset and the name of some column are not consistent so we will rename the column to same format as q1-2020

Renaming columns We will rename the columns into same format as of 2020 dataset because it is in latest format

```
q4_2019 <- rename(q4_2019
  ,ride_id = trip_id
  ,rideable_type = bikeid
  ,started_at = start_time
  ,ended_at = end_time
  ,start_station_name = from_station_name
  ,start_station_id = from_station_id
  ,end_station_name = to_station_name
  ,end_station_id = to_station_id
  ,member_casual = usertype)
q3_2019 <- rename(q3_2019
  ,ride_id = trip_id
  ,rideable_type = bikeid
  ,started_at = start_time
  ,ended_at = end_time
  ,start_station_name = from_station_name
  ,start_station_id = from_station_id
  ,end_station_name = to_station_name
  ,end_station_id = to_station_id
  ,member_casual = usertype)
```

```
q2_2019 <- rename(q2_2019
  ,ride_id = "X01...Rental.Details.Rental.ID"
  ,rideable_type = "X01...Rental.Details.Bike.ID"
  ,started_at = "X01...Rental.Details.Local.Start.Time"
  ,ended_at = "X01...Rental.Details.Local.End.Time"
  ,start_station_name = "X03...Rental.Start.Station.Name"
  ,start_station_id = "X03...Rental.Start.Station.ID"
  ,end_station_name = "X02...Rental.End.Station.Name"
  ,end_station_id = "X02...Rental.End.Station.ID"
  ,member_casual = "User.Type")
```

Checking data types of columns

```
## Rows: 1,108,163
## Columns: 12
## $ ride_id <int> 22178529, 22178530, ~
## $ started_at <chr> "2019-04-01 00:02:22~
## $ ended_at <chr> "2019-04-01 00:09:48~
## $ rideable_type <int> 6251, 6226, 5649, 41~
## $ X01...Rental.Details.Duration.In.Seconds.Uncapped <chr> "446.0", "1,048.0", ~
## $ start_station_id <int> 81, 317, 283, 26, 20~
## $ start_station_name <chr> "Daley Center Plaza"~
## $ end_station_id <int> 56, 59, 174, 133, 12~
## $ end_station_name <chr> "Desplaines St & Kin~
## $ member_casual <chr> "Subscriber", "Subsc~
## $ Member.Gender <chr> "Male", "Female", "M~
## $ X05...Member.Details.Member.Birthday.Year <int> 1975, 1984, 1990, 19~

## Rows: 1,640,718
## Columns: 12
## $ ride_id <int> 23479388, 23479389, 23479390, 23479391, 23479392, 2~
## $ started_at <chr> "2019-07-01 00:00:27", "2019-07-01 00:01:16", "2019~
## $ ended_at <chr> "2019-07-01 00:20:41", "2019-07-01 00:18:44", "2019~
## $ rideable_type <int> 3591, 5353, 6180, 5540, 6014, 4941, 3770, 5442, 295~
## $ tripduration <chr> "1,214.0", "1,048.0", "1,554.0", "1,503.0", "1,213.~
## $ start_station_id <int> 117, 381, 313, 313, 168, 300, 168, 313, 43, 43, 511~
## $ start_station_name <chr> "Wilton Ave & Belmont Ave", "Western Ave & Monroe S~
## $ end_station_id <int> 497, 203, 144, 144, 62, 232, 62, 144, 195, 195, 84,~
## $ end_station_name <chr> "Kimball Ave & Belmont Ave", "Western Ave & 21st St~
## $ member_casual <chr> "Subscriber", "Customer", "Customer", "Customer", "~
## $ gender <chr> "Male", "", "", "", "", "Male", "", "", "", "", "",~
## $ birthyear <int> 1992, NA, NA, NA, NA, 1990, NA, NA, NA, NA, NA, NA,~

## Rows: 704,054
## Columns: 12
## $ ride_id <int> 25223640, 25223641, 25223642, 25223643, 25223644, 2~
## $ started_at <chr> "2019-10-01 00:01:39", "2019-10-01 00:02:16", "2019~
## $ ended_at <chr> "2019-10-01 00:17:20", "2019-10-01 00:06:34", "2019~
## $ rideable_type <int> 2215, 6328, 3003, 3275, 5294, 1891, 1061, 1274, 601~
## $ tripduration <chr> "940.0", "258.0", "850.0", "2,350.0", "1,867.0", "3~
## $ start_station_id <int> 20, 19, 84, 313, 210, 156, 84, 156, 156, 336, 77, 1~
## $ start_station_name <chr> "Sheffield Ave & Kingsbury St", "Throop (Loomis) St~
## $ end_station_id <int> 309, 241, 199, 290, 382, 226, 142, 463, 463, 336, 5~
## $ end_station_name <chr> "Leavitt St & Armitage Ave", "Morgan St & Polk St",~
## $ member_casual <chr> "Subscriber", "Subscriber", "Subscriber", "Subscrib~
```

```
## $ gender          <chr> "Male", "Male", "Female", "Male", "Male", "Female", ~
## $ birthyear       <int> 1987, 1998, 1991, 1990, 1987, 1994, 1991, 1995, 199~

## Rows: 426,887
## Columns: 13
## $ ride_id         <chr> "EACB19130B0CDA4A", "8FED874C809DC021", "789F3C21E4~
## $ rideable_type   <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at      <chr> "2020-01-21 20:06:59", "2020-01-30 14:22:39", "2020~
## $ ended_at        <chr> "2020-01-21 20:14:30", "2020-01-30 14:26:22", "2020~
## $ start_station_name <chr> "Western Ave & Leland Ave", "Clark St & Montrose Av~
## $ start_station_id <int> 239, 234, 296, 51, 66, 212, 96, 96, 212, 38, 117, 1~
## $ end_station_name <chr> "Clark St & Leland Ave", "Southport Ave & Irving Pa~
## $ end_station_id   <int> 326, 318, 117, 24, 212, 96, 212, 212, 96, 100, 632,~
## $ start_lat        <dbl> 41.9665, 41.9616, 41.9401, 41.8846, 41.8856, 41.889~
## $ start_lng        <dbl> -87.6884, -87.6660, -87.6455, -87.6319, -87.6418, --
## $ end_lat          <dbl> 41.9671, 41.9542, 41.9402, 41.8918, 41.8899, 41.884~
## $ end_lng          <dbl> -87.6674, -87.6644, -87.6530, -87.6206, -87.6343, --
## $ member_casual    <chr> "member", "member", "member", "member", "member", "~
```

From `q2_2019` on wards the `ride_id` is in integer form but it should be in character form

```
q4_2019 <- mutate(q4_2019, ride_id = as.character(ride_id)
,rideable_type = as.character(rideable_type))
q3_2019 <- mutate(q3_2019, ride_id = as.character(ride_id)
,rideable_type = as.character(rideable_type))
q2_2019 <- mutate(q2_2019, ride_id = as.character(ride_id)
,rideable_type = as.character(rideable_type))
```

Changing data type *Checking one dataset if the conversion happens or not*

```
glimpse(q3_2019)
```

```
## Rows: 1,640,718
## Columns: 12
## $ ride_id         <chr> "23479388", "23479389", "23479390", "23479391", "23~
## $ started_at      <chr> "2019-07-01 00:00:27", "2019-07-01 00:01:16", "2019~
## $ ended_at        <chr> "2019-07-01 00:20:41", "2019-07-01 00:18:44", "2019~
## $ rideable_type   <chr> "3591", "5353", "6180", "5540", "6014", "4941", "37~
## $ tripduration    <chr> "1,214.0", "1,048.0", "1,554.0", "1,503.0", "1,213.~
## $ start_station_id <int> 117, 381, 313, 313, 168, 300, 168, 313, 43, 43, 511~
## $ start_station_name <chr> "Wilton Ave & Belmont Ave", "Western Ave & Monroe S~
## $ end_station_id   <int> 497, 203, 144, 144, 62, 232, 62, 144, 195, 195, 84,~
## $ end_station_name <chr> "Kimball Ave & Belmont Ave", "Western Ave & 21st St~
## $ member_casual    <chr> "Subscriber", "Customer", "Customer", "Customer", "~
## $ gender          <chr> "Male", "", "", "", "", "Male", "", "", "", "", "", ~
## $ birthyear        <int> 1992, NA, NA, NA, NA, 1990, NA, NA, NA, NA, NA, NA,~
```

Merging the datasets Now all the datasets is in the same order so we can now combine them into one single dataset to do the further cleaning and transform

```
all_trips <- bind_rows(q2_2019, q3_2019, q4_2019, q1_2020)
```

Inspect the dataset Since now the dataset of all the month is transform into one dataset we will inspect it and then clean and process it to make ready for analysis

```
#checking column names
colnames(all_trips)
```

```
## [1] "ride_id"
## [2] "started_at"
## [3] "ended_at"
## [4] "rideable_type"
## [5] "X01...Rental.Details.Duration.In.Seconds.Uncapped"
## [6] "start_station_id"
## [7] "start_station_name"
## [8] "end_station_id"
## [9] "end_station_name"
## [10] "member_casual"
## [11] "Member.Gender"
## [12] "X05...Member.Details.Member.Birthday.Year"
## [13] "tripduration"
## [14] "gender"
## [15] "birthyear"
## [16] "start_lat"
## [17] "start_lng"
## [18] "end_lat"
## [19] "end_lng"
```

```
#checking data types
glimpse(all_trips)
```

```
## Rows: 3,879,822
## Columns: 19
## $ ride_id                <chr> "22178529", "2217853~
## $ started_at             <chr> "2019-04-01 00:02:22~
## $ ended_at               <chr> "2019-04-01 00:09:48~
## $ rideable_type          <chr> "6251", "6226", "564~
## $ X01...Rental.Details.Duration.In.Seconds.Uncapped <chr> "446.0", "1,048.0", ~
## $ start_station_id       <int> 81, 317, 283, 26, 20~
## $ start_station_name     <chr> "Daley Center Plaza"~
## $ end_station_id         <int> 56, 59, 174, 133, 12~
## $ end_station_name       <chr> "Desplaines St & Kin~
## $ member_casual          <chr> "Subscriber", "Subsc~
## $ Member.Gender          <chr> "Male", "Female", "M~
## $ X05...Member.Details.Member.Birthday.Year         <int> 1975, 1984, 1990, 19~
## $ tripduration           <chr> NA, NA, NA, NA, NA, ~
## $ gender                 <chr> NA, NA, NA, NA, NA, ~
## $ birthyear              <int> NA, NA, NA, NA, NA, ~
## $ start_lat              <dbl> NA, NA, NA, NA, NA, ~
## $ start_lng              <dbl> NA, NA, NA, NA, NA, ~
## $ end_lat                <dbl> NA, NA, NA, NA, NA, ~
## $ end_lng                <dbl> NA, NA, NA, NA, NA, ~
```

Remove unnecessary columns Removing unnecessary columns for consistency

```
all_trips <- all_trips %>%
  select(-c(start_lat, start_lng, end_lat, end_lng, birthyear, gender, "X01...Rental.Details.Duration.In.Seconds.Uncapped"))
```

Converting data types Convert started_at and ended_at to date and time

```
all_trips$started_at<-ymd_hms(all_trips$started_at)
all_trips$ended_at <- ymd_hms(all_trips$ended_at)
```

Removing inconsistency There are four unique values in member_casual subscriber, member, customer, casual but 2020 onwards these member has been changed into two unique values member, casual

```
all_trips <- all_trips %>%
  mutate(member_casual = recode(member_casual
                                , "Subscriber" = "member"
                                , "Customer" = "casual"))
```

Ride length (new column) ride_length is the distance between started time and ended time

```
all_trips$ride_length <- difftime(all_trips$ended_at,all_trips$started_at,units = "mins")
head(all_trips$ride_length)
```

```
## Time differences in mins
## [1]  7.433333 17.466667  4.200000  5.950000 16.783333  4.283333
```

Also we will convert the ride_length into numeric for further calculations

```
all_trips$ride_length <- round(as.numeric(as.character(all_trips$ride_length)),2)
```

Round trip (new column) We will produce a new column named round_trip = "Yes" where start_station_name is equal to end_station_name

```
all_trips <- all_trips %>%
  mutate(round_trip=case_when(
    start_station_name==end_station_name ~ "Yes",
    start_station_name!=end_station_name ~ "No"
  ))
head(all_trips$round_trip)
```

```
## [1] "No" "No" "No" "No" "No" "No"
```

Day (new column) calculating the day using the started_date column

```
all_trips$day <- day(all_trips$started_at)
head(all_trips$day)
```

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

```
all_trips$day_of_week <- weekdays(all_trips$started_at)
head(all_trips$day_of_week)
```

Day of the week (new column)

```
## [1] "Monday" "Monday" "Monday" "Monday" "Monday" "Monday"
```

Month of the year (new column) calculating month using the started_date column

```
all_trips$month <- months.Date(all_trips$started_at)
head(all_trips$month)
```

```
## [1] "April" "April" "April" "April" "April" "April"
```

Year (new column) finally year column for summarizing the data by year

```
all_trips$year <- year(all_trips$started_at)
head(all_trips$year)
```

```
## [1] 2019 2019 2019 2019 2019 2019
```

Deleting/Filtering bad data The start_station_name = "HQ QR" is not relevant because it is the maintenance station for the bike so we have to remove it

Also the negative ride_length is not good for analysis as the ended_time is less than the started time which is simply a bad data

```
all_trips <- all_trips %>%
  filter(!(all_trips$ride_length < 0 | all_trips$start_station_name == "HQ QR" ))
```

Saving the transform data Finally saved the transform data for analysis

```
write.csv(all_trips, row.names = F, "Bike_sharing_clean/2019-20_tripdatas.csv")
```

Aggregating the file After cleaning, merging and saving all the file its time to aggregate them because the file size is too large to work with them so it is a must to aggregate them into most suitable form

We will use ride_length for aggregating the data since it is a numerical column and it is most important for our analysis

Loading the cleaned data

```
trip <- read.csv("Bike_sharing_clean/2019-20_tripdatas.csv")
head(trip)
```

```
##   ride_id      started_at      ended_at rideable_type
## 1 22178529 2019-04-01 00:02:22 2019-04-01 00:09:48      6251
## 2 22178530 2019-04-01 00:03:02 2019-04-01 00:20:30      6226
## 3 22178531 2019-04-01 00:11:07 2019-04-01 00:15:19      5649
## 4 22178532 2019-04-01 00:13:01 2019-04-01 00:18:58      4151
## 5 22178533 2019-04-01 00:19:26 2019-04-01 00:36:13      3270
## 6 22178534 2019-04-01 00:19:39 2019-04-01 00:23:56      3123
##   start_station_id      start_station_name end_station_id
## 1                81      Daley Center Plaza           56
## 2                317      Wood St & Taylor St           59
## 3                283 LaSalle St & Jackson Blvd          174
## 4                 26 McClurg Ct & Illinois St           133
## 5                202      Halsted St & 18th St           129
## 6                420      Ellis Ave & 55th St           426
##   end_station_name member_casual ride_length round_trip day
## 1 Desplaines St & Kinzie St      member          7.43      No   1
## 2 Wabash Ave & Roosevelt Rd      member         17.47      No   1
## 3      Canal St & Madison St      member          4.20      No   1
## 4 Kingsbury St & Kinzie St      member          5.95      No   1
## 5 Blue Island Ave & 18th St      member         16.78      No   1
## 6      Ellis Ave & 60th St      member          4.28      No   1
##   day_of_week month year
## 1      Monday April 2019
## 2      Monday April 2019
## 3      Monday April 2019
## 4      Monday April 2019
```

```
## 5      Monday April 2019
## 6      Monday April 2019
```

Checking the data type of data

```
glimpse(trip)
```

```
## Rows: 3,876,042
## Columns: 15
## $ ride_id          <chr> "22178529", "22178530", "22178531", "22178532", "22~
## $ started_at       <chr> "2019-04-01 00:02:22", "2019-04-01 00:03:02", "2019~
## $ ended_at         <chr> "2019-04-01 00:09:48", "2019-04-01 00:20:30", "2019~
## $ rideable_type     <chr> "6251", "6226", "5649", "4151", "3270", "3123", "64~
## $ start_station_id <int> 81, 317, 283, 26, 202, 420, 503, 260, 211, 211, 304~
## $ start_station_name <chr> "Daley Center Plaza", "Wood St & Taylor St", "LaSal~
## $ end_station_id   <int> 56, 59, 174, 133, 129, 426, 500, 499, 211, 211, 232~
## $ end_station_name <chr> "Desplaines St & Kinzie St", "Wabash Ave & Roosevel~
## $ member_casual    <chr> "member", "member", "member", "member", "member", "~
## $ ride_length      <dbl> 7.43, 17.47, 4.20, 5.95, 16.78, 4.28, 9.13, 6.38, 3~
## $ round_trip       <chr> "No", "No", "No", "No", "No", "No", "No", "No", "Ye~
## $ day              <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ day_of_week      <chr> "Monday", "Monday", "Monday", "Monday", "Monday", "~
## $ month            <chr> "April", "April", "April", "April", "April", "April~
## $ year             <int> 2019, 2019, 2019, 2019, 2019, 2019, 2019, 2019, 201~
```

Changing the data_type We will convert `year` data type since we will consider it as the categorical data when we will merge all the dataset when aggregating

```
trip$year <- as.character(trip$year)
```

Checking the Statistics Since we will aggregate the data based on `ride_length`, its important to check its statistics to decide the aggregate parameter but we have already aggregate the 2016-17 data based on median due to skewed column, we will consider this parameter for all the aggregation for consistency

```
summary(trip$ride_length)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
##      0.02     6.87     11.87     24.65    21.48 156450.40
```

We can easily see that the `ride_length` is right-skewed since there is a BIG difference between *Q3* and *Max* value. So we will use median instead of mean for aggregation which is more ideal in this case of skewed column

Aggregating the data

```
trip_data<-aggregate(trip$ride_length~trip$member_casual+trip$round_trip+trip$day_of_week+trip$month+trip$year,
data=trip, FUN=median)
head(trip_data)
```

```
##      trip$member_casual trip$round_trip trip$day_of_week trip$month trip$year
## 1          casual          No          Friday          April          2019
## 2          member          No          Friday          April          2019
## 3          casual          Yes          Friday          April          2019
## 4          member          Yes          Friday          April          2019
## 5          casual          No          Monday          April          2019
## 6          member          No          Monday          April          2019
##      trip$ride_length
## 1          24.100
## 2           8.970
## 3          37.815
```



```
## 4          14.750
## 5          26.350
## 6           9.680
```

Saving the aggregate

Finally, last step is to save the data so we can use this data to merge all other aggregates data

We will merge the data with the old data we saved while aggregating

```
trip_old <- read.csv("Bike_sharing_clean/tripdata_aggregate.csv")
head(trip_old)
```

```
##   trip.member_casual trip.round_trip trip.day_of_week trip.month trip.year
## 1          casual          No          Friday      April      2016
## 2          member          No          Friday      April      2016
## 3          casual          Yes          Friday      April      2016
## 4          member          Yes          Friday      April      2016
## 5          casual          No          Monday      April      2016
## 6          member          No          Monday      April      2016
##   trip.ride_length
## 1              19
## 2               9
## 3              22
## 4              10
## 5              22
## 6              10
```

Transforming the old data to merge perfectly We have to make column name and type consistent before merging

```
trip_old$trip.year <- as.character(trip_old$trip.year)
trip_old <- rename(trip_old,
  "trip$member_casual"=trip.member_casual,
  "trip$round_trip"=trip.round_trip,
  "trip$day_of_week"=trip.day_of_week,
  "trip$month"=trip.month,
  "trip$year"=trip.year,
  "trip$ride_length"=trip.ride_length
)
```

```
trip_merged <- bind_rows(trip_old, trip_data)
write.csv(trip_merged, row.names = F, "Bike_sharing_clean/tripdata_aggregate.csv")
```