Google data Analytics Case study : 1

Sanjeev Kumar Khatri

2023-02-18

Introduction

This notebook provide my ideas for the Google Data Analytics Capstone : Case study 1 . The full document can be accessed from theGoogle Data Analytics Capstone.

for this case study . I will be following the 6 major steps of data analysis.

- ask
- Prepare
- Process
- analyze
- share
- act

Begin with the Ask Phase

for beginning with this lets first try to understand the scenario '

```
you are a junior data analyst working in the marketing analyst team at
Cyclistic, a bike-share company in Chicago. The director of marketing
believes the company's future success depends on maximizing the number of
annual memberships. Therefore, your team wants to understand how casual
riders and annual members use Cyclistic bikes differently. From these
insights, your team will design a new marketing strategy to convert casual
riders into annual members. But first, Cyclistic executives must approve your
recommendations, so they must be backed up with compelling data insights and
professional data visualizations.

Characters and teams

*  Cyclistic: A bike-share program that features more than 5,800 bicycles and
600 docking stations. Cyclistic sets itself apart by also offering reclining
bikes, hand tricycles, and cargo bikes, making bike-share more inclusive to
people with disabilities and riders who can't use a standard two-wheeled
bike. The majority of riders opt for traditional bikes; about 8% of riders
use the assistive options. Cyclistic users are more likely to ride for
leisure, but about 30% use them to commute to work each day.
```

```
* Lily Moreno: The director of marketing and your manager. Moreno is
responsible for the development of campaigns and initiatives to promote the
bike-share program. These may include email, social media, and other
channels.

* Cyclistic marketing analytics team: A team of data analysts who are
responsible for collecting, analyzing, and reporting data that helps guide
Cyclistic marketing strategy. You joined this team six months ago and have
been busy learning about Cyclistic's mission and business goals — as well as
how you, as a junior data analyst, can help Cyclistic achieve them.

* Cyclistic executive team: The notoriously detail-oriented executive team
will decide whether to approve the recommended marketing program.
```

main goal : Design marketing strategies aimed at converting casual riders into annual members.

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

Case study road map

Guiding Questions

1. what is the problem you are trying to solve ?

   here we are trying to convert the casual riders into annual members.

2. How can your insights drive business decisions ?

   This insights is for the marketing team such that it will help them to increase the annual members.

Key Tasks

1. Identify the business task.
2. Consider the key stakeholders.

Deliverable

- A clear statement of the business task

  find the difference between the casual and annual riders?

Prepare

The data is Provided by the company itself and can be access from here.

Case study road map for prepare phase

###Guiding questions ● Where is your data located?

```
data is located in amazon aws.
```

● How is the data organized?

```
data is organized on the bases of month in .zip format upon extraction we can
get the .csv file
```

● Are there issues with bias or credibility in this data? Does your data ROCCC?

```
there is no issues with bias or credibility as the data is from the company
clients its self. The data is ROCCC i.e

reliable, original, comprehensive, current and cited.
```

● How are you addressing licensing, privacy, security, and accessibility?

```
after examining the dataset its found that the licensing is governed by the
company itself and further information is not provided.
```

● How did you verify the data's integrity?

```
as all the file has the consistent column and column has its own data type.
```

● How does it help you answer your question?

```
i might get info about the key insights of the rider
```

● Are there any problems with the data?

```
more information about the station would be helpful(for eg: station more
overcrowded , less crowded)
```

Key Tasks

1. Download data and store it appropriately.
2. Identify how it's organized.
3. Sort and filter the data.
4. Determine the credibility of the data.

Deliverable

- A description of all data sources used
  - CSV files downloaded and saved in one main csv file called(divvy-tripdata_trip_data_2019_q1_to_q4_to_2020_q1)
  - FIles were too large to open in excel and sheets

Process

after merging the csv file we can advance forward it will be easy for now to improve workflow

Case study Roadmap for Process Phase

Guiding Question answers

1. What tools are you choosing and why?

   i am using the R programming langauge as the size of the data is large and merging and combining data will be easier.

2. Have you ensured your data's integrity?

   yes the data is consistent through out the columns

3. What steps have you taken to ensure that your data is clean?

   first of all the duplicate data is removed afterwards we design the columns.

4. How can you verify that your data is clean and ready to analyze?

   code below verifes this.

5. Have you documented your cleaning process so you can review and share those results?

   yes this R document tell all about this.

Key tasks

1. Check the data for errors.
2. Choose your tools.
3. Transform the data so you can work with it effectively.
4. Document the cleaning process.

Deliverable

Documentation of any cleaning or manipulation of data

Installing and loading the required packages as we can think of:

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✔ ggplot2 3.4.1     ✔ purrr   1.0.1
## ✔ tibble  3.1.8     ✔ dplyr   1.1.0
## ✔ tidyr   1.3.0     ✔ stringr 1.5.0
## ✔ readr   2.1.3     ✔ forcats 1.0.0
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(lubridate)

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

for mergint the csv file stackoverflow

reading the csv combine files

```
cycle_merged <- read_csv("/Users/sanju/cyclistic_merged.csv")

## New names:
## Rows: 3489539 Columns: 18
## ── Column specification
## ────────────────────────────────────────────────────────────────────────
Delimiter: "," chr
## (10): ride_id, rideable_type, start_station_name, start_station_id, end...
dbl
## (6): ...1, start_lat, start_lng, end_lat, end_lng, ride_time_m 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.
## • `` -> `...1`
```

looking deep into the data

```
head(cycle_merged)

## # A tibble: 6 × 18
##    ...1 ride_id  ridea…¹ started_at          ended_at            start…²
start…³
##   <dbl> <chr>    <chr>   <dttm>              <dttm>              <chr>
<chr>
## 1     1 A847FAD… docked… 2020-04-26 17:45:14 2020-04-26 18:12:03
Eckhar… 86
## 2     2 5405B80… docked… 2020-04-17 17:08:54 2020-04-17 17:17:03 Drake
… 503
## 3     3 5DD24A7… docked… 2020-04-01 17:54:13 2020-04-01 18:08:36
McClur… 142
## 4     4 2A59BBD… docked… 2020-04-07 12:50:19 2020-04-07 13:02:31
Califo… 216
## 5     5 27AD306… docked… 2020-04-18 10:22:59 2020-04-18 11:15:54 Rush
S… 125
## 6     6 356216E… docked… 2020-04-30 17:55:47 2020-04-30 18:01:11 Mies
v… 173
## # … with 11 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
```

```
## #   member_casual <chr>, ride_time_m <dbl>, year_month <chr>, weekday
<chr>,
## #   start_hour <chr>, and abbreviated variable names ¹rideable_type,
## #   ²start_station_name, ³start_station_id

str(cycle_merged)

## spc_tbl_ [3,489,539 × 18] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ...1              : num [1:3489539] 1 2 3 4 5 6 7 8 9 10 ...
##  $ ride_id           : chr [1:3489539] "A847FADBBC638E45"
"5405B80E996FF60D" "5DD24A79A4E006F4" "2A59BBDF5CDBA725" ...
##  $ rideable_type     : chr [1:3489539] "docked_bike" "docked_bike"
"docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:3489539], format: "2020-04-26 17:45:14"
"2020-04-17 17:08:54" ...
##  $ ended_at          : POSIXct[1:3489539], format: "2020-04-26 18:12:03"
"2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:3489539] "Eckhart Park" "Drake Ave &
Fullerton Ave" "McClurg Ct & Erie St" "California Ave & Division St" ...
##  $ start_station_id  : chr [1:3489539] "86" "503" "142" "216" ...
##  $ end_station_name  : chr [1:3489539] "Lincoln Ave & Diversey Pkwy"
"Kosciuszko Park" "Indiana Ave & Roosevelt Rd" "Wood St & Augusta Blvd" ...
##  $ end_station_id    : chr [1:3489539] "152" "499" "255" "657" ...
##  $ start_lat         : num [1:3489539] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:3489539] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:3489539] 41.9 41.9 41.9 41.9 42 ...
##  $ end_lng           : num [1:3489539] -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:3489539] "member" "member" "member" "member"
...
##  $ ride_time_m       : num [1:3489539] 26.82 8.15 14.38 12.2 52.92 ...
##  $ year_month        : chr [1:3489539] "2020 - 04 (Apr)" "2020 - 04 (Apr)"
"2020 - 04 (Apr)" "2020 - 04 (Apr)" ...
##  $ weekday           : chr [1:3489539] "7 - Sun" "5 - Fri" "3 - Wed" "2 -
Tue" ...
##  $ start_hour        : chr [1:3489539] "18" "17" "18" "13" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ...1 = col_double(),
##   ..   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(),
##   ..   ride_time_m = col_double(),
##   ..   year_month = col_character(),
##   ..   weekday = col_character(),
##   ..   start_hour = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>

colnames(cycle_merged)

##  [1] "...1"               "ride_id"            "rideable_type"
##  [4] "started_at"         "ended_at"           "start_station_name"
##  [7] "start_station_id"   "end_station_name"   "end_station_id"
## [10] "start_lat"          "start_lng"          "end_lat"
## [13] "end_lng"            "member_casual"      "ride_time_m"
## [16] "year_month"         "weekday"            "start_hour"

length(colnames(cycle_merged))

## [1] 18
```

Data cleaning

Removing the duplicates if any based on ride_id for more info and here

```
cycle_no_dup <- cycle_merged %>%
  distinct(.keep_all = TRUE)
head(cycle_no_dup)

## # A tibble: 6 × 18
##    ...1 ride_id  ridea…¹ started_at          ended_at            start…²
start…³
##   <dbl> <chr>    <chr>   <dttm>              <dttm>              <chr>
<chr>
## 1     1 A847FAD… docked… 2020-04-26 17:45:14 2020-04-26 18:12:03
Eckhar… 86
## 2     2 5405B80… docked… 2020-04-17 17:08:54 2020-04-17 17:17:03 Drake
… 503
## 3     3 5DD24A7… docked… 2020-04-01 17:54:13 2020-04-01 18:08:36
McClur… 142
## 4     4 2A59BBD… docked… 2020-04-07 12:50:19 2020-04-07 13:02:31
Califo… 216
## 5     5 27AD306… docked… 2020-04-18 10:22:59 2020-04-18 11:15:54 Rush
S… 125
## 6     6 356216E… docked… 2020-04-30 17:55:47 2020-04-30 18:01:11 Mies
v… 173
## # … with 11 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, ride_time_m <dbl>, year_month <chr>, weekday
<chr>,
## #   start_hour <chr>, and abbreviated variable names ¹rideable_type,
## #   ²start_station_name, ³start_station_id
```

Parsing date time columns

The POSIXct class stores date/time values as the number of seconds since January 1, 1970, while the POSIXlt class stores them as a list with elements for second, minute, hour, day, month, and year, among others. Unless you need the list nature of the POSIXlt class, the POSIXct class is the usual choice for storing dates in R.

```
cycle_no_dup$started_at <- as.POSIXct(cycle_no_dup$started_at, "%Y-%m-%d
%H:%M:%S")
cycle_no_dup$ended_at <- as.POSIXct(cycle_no_dup$ended_at, "%Y-%m-%d
%H:%M:%S")
```

creating new column ride_length (mutate will add new col)

```
library(hms)

##
## Attaching package: 'hms'

## The following object is masked from 'package:lubridate':
##
##     hms

cycle_no_dup <- cycle_no_dup %>%
  mutate(ride_length = hms(as.numeric(ended_at - started_at)))
head(cycle_no_dup)

## # A tibble: 6 × 19
##    ...1 ride_id  ridea…¹ started_at          ended_at            start…²
start…³
##   <dbl> <chr>    <chr>   <dttm>              <dttm>              <chr>
<chr>
## 1     1 A847FAD… docked… 2020-04-26 17:45:14 2020-04-26 18:12:03
Eckhar… 86
## 2     2 5405B80… docked… 2020-04-17 17:08:54 2020-04-17 17:17:03 Drake
… 503
## 3     3 5DD24A7… docked… 2020-04-01 17:54:13 2020-04-01 18:08:36
McClur… 142
## 4     4 2A59BBD… docked… 2020-04-07 12:50:19 2020-04-07 13:02:31
Califo… 216
## 5     5 27AD306… docked… 2020-04-18 10:22:59 2020-04-18 11:15:54 Rush
S… 125
## 6     6 356216E… docked… 2020-04-30 17:55:47 2020-04-30 18:01:11 Mies
v… 173
## # … with 12 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, ride_time_m <dbl>, year_month <chr>, weekday
<chr>,
## #   start_hour <chr>, ride_length <time>, and abbreviated variable names
## #   ¹rideable_type, ²start_station_name, ³start_station_id
```

Explanation: we use the hms package to convert the length in HH:MM:SS format. first we subtract the ended and started time then convert it to the numeric value as hms only accept the numeric value.

```
cycle_no_dup <- cycle_no_dup %>%
    mutate(day_of_week = paste(as.integer(strftime(cycle_no_dup$started_at,
"%u")) %% 7 + 1 , "-", strftime(cycle_no_dup$started_at, "%a")))
```

explanation: The `paste()` function is used to concatenate two strings together with a "-" separator, and the two strings are created using the `strftime()`function to extract the day of the week from the "ended_at" column in the dataset.`%u` represents the day of the week as a decimal number (1-7, with 1 representing Monday), and `%a`represents the abbreviated weekday name (e.g., "Sun" for Sunday). 7 + 1 is added cause the in data set it sun = 7 and mon = 1

```
head(cycle_no_dup)
```

```
## # A tibble: 6 × 20
##    ...1 ride_id  ridea…¹ started_at          ended_at            start…²
start…³
##   <dbl> <chr>    <chr>   <dttm>              <dttm>              <chr>
<chr>
## 1     1 A847FAD… docked… 2020-04-26 17:45:14 2020-04-26 18:12:03
Eckhar… 86
## 2     2 5405B80… docked… 2020-04-17 17:08:54 2020-04-17 17:17:03 Drake
… 503
## 3     3 5DD24A7… docked… 2020-04-01 17:54:13 2020-04-01 18:08:36
McClur… 142
## 4     4 2A59BBD… docked… 2020-04-07 12:50:19 2020-04-07 13:02:31
Califo… 216
## 5     5 27AD306… docked… 2020-04-18 10:22:59 2020-04-18 11:15:54 Rush
S… 125
## 6     6 356216E… docked… 2020-04-30 17:55:47 2020-04-30 18:01:11 Mies
v… 173
## # … with 13 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, ride_time_m <dbl>, year_month <chr>, weekday
<chr>,
## #   start_hour <chr>, ride_length <time>, day_of_week <chr>, and
abbreviated
## #   variable names ¹rideable_type, ²start_station_name, ³start_station_id
```

saving the result in new csv file

```
cycle_no_dup %>%
  write.csv("clean_cycle.csv")
```

Analyze

Case Study Road Map Analyze

Key tasks

1. Aggregate your data so it's useful and accessible.
2. Organize and format your data.
3. Perform calculations.
4. Identify trends and relationships.

Deliverable

A summary of your analysis

The data exploration will consist of building a profile for annual members and how they differ from casual riders.

```
cycle <- cycle_no_dup
head(cycle)
```

```
## # A tibble: 6 × 20
##    ...1 ride_id  ridea…¹ started_at          ended_at            start…²
start…³
##   <dbl> <chr>    <chr>   <dttm>              <dttm>              <chr>
<chr>
## 1     1 A847FAD… docked… 2020-04-26 17:45:14 2020-04-26 18:12:03
Eckhar… 86
## 2     2 5405B80… docked… 2020-04-17 17:08:54 2020-04-17 17:17:03 Drake
… 503
## 3     3 5DD24A7… docked… 2020-04-01 17:54:13 2020-04-01 18:08:36
McClur… 142
## 4     4 2A59BBD… docked… 2020-04-07 12:50:19 2020-04-07 13:02:31
Califo… 216
## 5     5 27AD306… docked… 2020-04-18 10:22:59 2020-04-18 11:15:54 Rush
S… 125
## 6     6 356216E… docked… 2020-04-30 17:55:47 2020-04-30 18:01:11 Mies
v… 173
## # … with 13 more variables: end_station_name <chr>, end_station_id <chr>,
## #   start_lat <dbl>, start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, ride_time_m <dbl>, year_month <chr>, weekday
<chr>,
## #   start_hour <chr>, ride_length <time>, day_of_week <chr>, and
abbreviated
## #   variable names ¹rideable_type, ²start_station_name, ³start_station_id
```

let's get the summary of the dataset

```
summary(cycle)
```

```
##       ...1           ride_id          rideable_type
##  Min.   :      1   Length:3489539     Length:3489539
```

```
##  1st Qu.: 872386   Class :character   Class :character
##  Median :1744770   Mode  :character   Mode  :character
##  Mean   :1744796
##  3rd Qu.:2617154
##  Max.   :3489748
##
##    started_at                       ended_at
##  Min.   :2020-04-01 00:00:30.00   Min.   :2020-04-01 00:10:45.00
##  1st Qu.:2020-07-14 19:36:28.00   1st Qu.:2020-07-14 20:11:10.50
##  Median :2020-08-29 14:47:30.00   Median :2020-08-29 15:18:24.00
##  Mean   :2020-09-10 01:13:26.91   Mean   :2020-09-10 01:39:55.75
##  3rd Qu.:2020-10-20 18:07:35.50   3rd Qu.:2020-10-20 18:21:47.00
##  Max.   :2021-03-31 23:59:08.00   Max.   :2021-04-06 11:00:11.00
##
##  start_station_name start_station_id   end_station_name   end_station_id
##  Length:3489539     Length:3489539     Length:3489539     Length:3489539
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    start_lat       start_lng         end_lat         end_lng
##  Min.   :41.64   Min.   :-87.87   Min.   :41.54   Min.   :-88.07
##  1st Qu.:41.88   1st Qu.:-87.66   1st Qu.:41.88   1st Qu.:-87.66
##  Median :41.90   Median :-87.64   Median :41.90   Median :-87.64
##  Mean   :41.90   Mean   :-87.64   Mean   :41.90   Mean   :-87.64
##  3rd Qu.:41.93   3rd Qu.:-87.63   3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.08   Max.   :-87.52   Max.   :42.16   Max.   :-87.44
##                                   NA's   :4737    NA's   :4737
##  member_casual       ride_time_m         year_month          weekday
##  Length:3489539     Min.   :-29049.97   Length:3489539     Length:3489539
##  Class :character   1st Qu.:     7.88   Class :character   Class
:character
##  Mode  :character   Median :    14.52   Mode  :character   Mode
:character
##                     Mean   :    26.48
##                     3rd Qu.:    26.63
##                     Max.   : 58720.03
##
##   start_hour        ride_length       day_of_week
##  Length:3489539     Length:3489539    Length:3489539
##  Class :character   Class1:hms        Class :character
##  Mode  :character   Class2:difftime   Mode  :character
##                     Mode  :numeric
##
##
##
```

one thing we notice that the min value of ride_time_m is negative and max is 58720.03

casual versus members

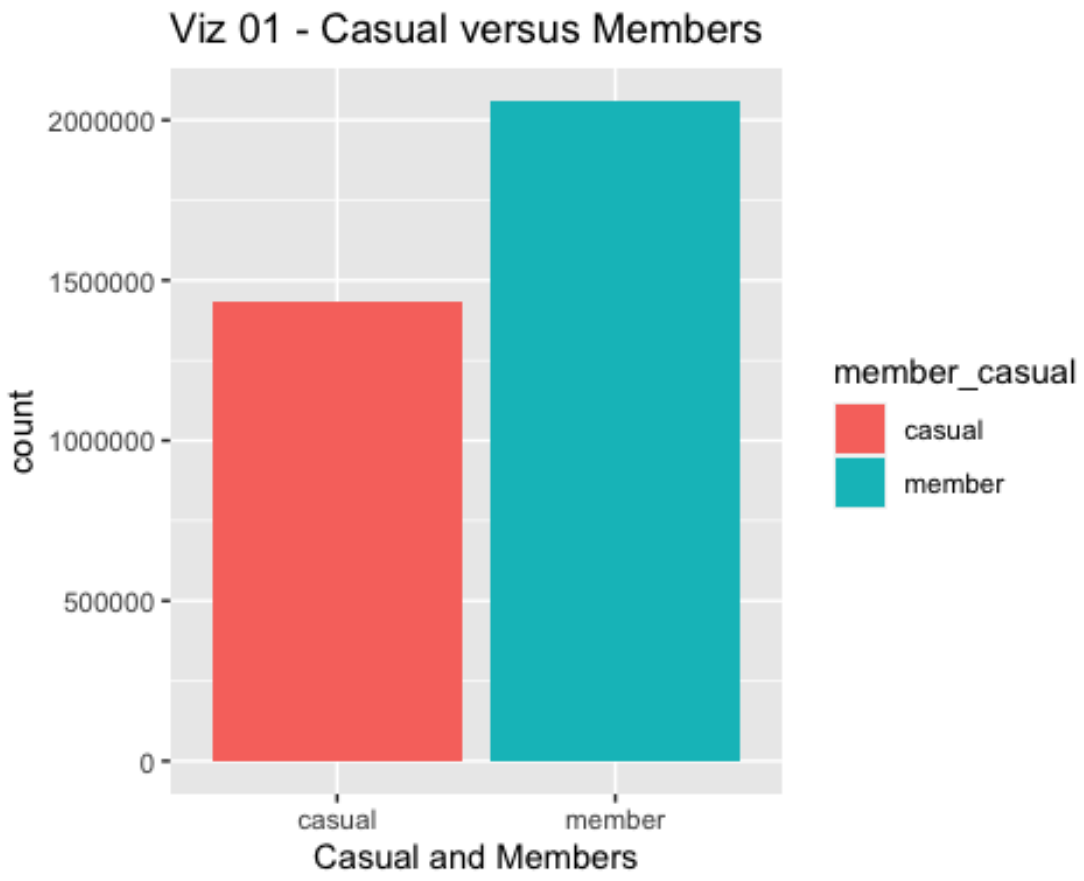
```
cycle %>%
  group_by(member_casual) %>%
  summarise(count = length(ride_id),
            '%' = (length(ride_id)/ nrow(cycle))* 100
            )
```

```
## # A tibble: 2 × 3
##   member_casual   count   `%`
##   <chr>           <int> <dbl>
## 1 casual        1430351  41.0
## 2 member        2059188  59.0
```

explanation : at first we group by the column member_casual afterwards we use the summarize function to summarize the casual and member.

lets try to visualize this

```
library(ggplot2)
ggplot(data = cycle) +
  geom_bar(mapping = aes(x = member_casual, fill = member_casual)) +
  labs(x = "Casual and Members", title = "Viz 01 - Casual versus Members")
```



we can see that the member has bigger proporction of the data.

lets check data distribution per month

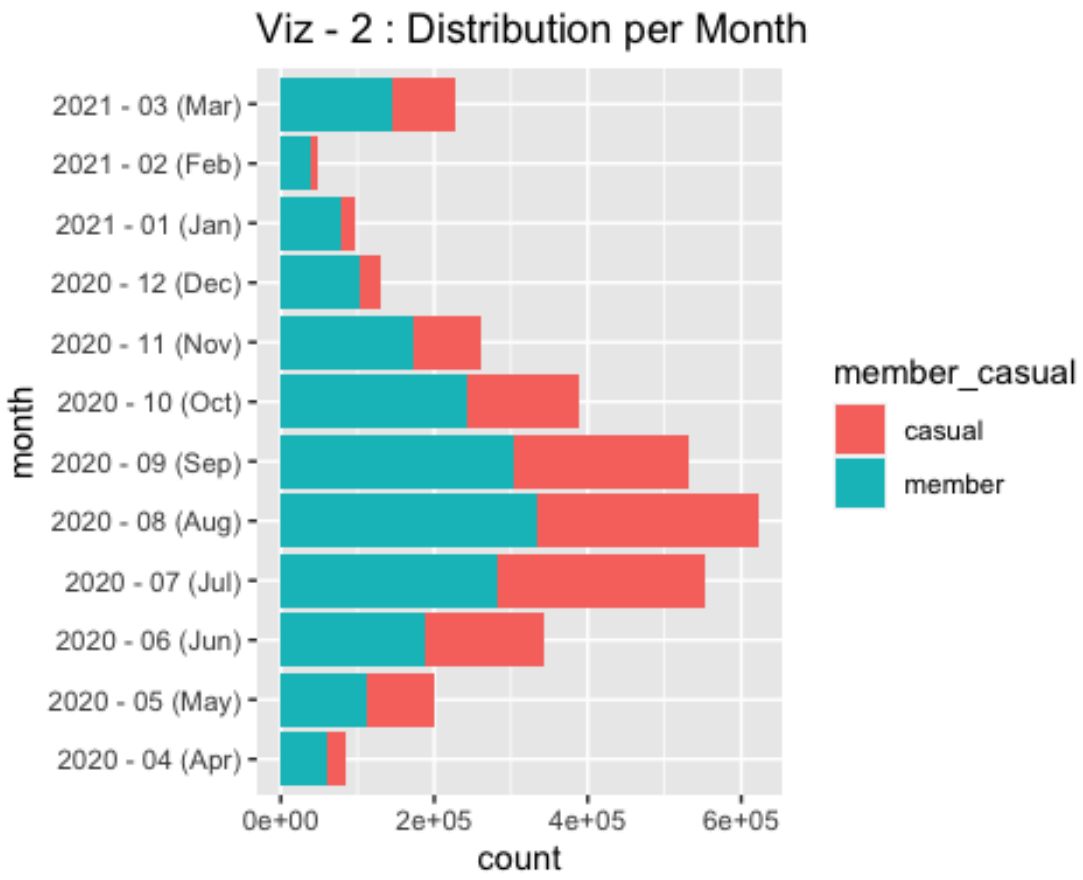
```
cycle %>%
  group_by(year_month) %>%
  summarize(count = length(ride_id),
            '%' = (length(ride_id) / nrow(cycle)) * 100,
            'member_m'= (sum(member_casual == "member") / length(ride_id) ) *
100,
            'casual_m'=sum(member_casual == "casual") / length(ride_id) *
100,
            'member - casual' = member_m - casual_m
            )
```

```
## # A tibble: 12 × 6
##    year_month       count   `%` member_m casual_m `member - casual`
##    <chr>            <int> <dbl>    <dbl>    <dbl>              <dbl>
##  1 2020 - 04 (Apr)  84776  2.43     72.1     27.9               44.3
##  2 2020 - 05 (May) 200274  5.74     56.6     43.4               13.2
##  3 2020 - 06 (Jun) 343005  9.83     54.9     45.1                9.79
##  4 2020 - 07 (Jul) 551480 15.8      51.2     48.8                2.34
##  5 2020 - 08 (Aug) 622361 17.8      53.5     46.5                6.92
##  6 2020 - 09 (Sep) 532958 15.3      56.7     43.3               13.4
##  7 2020 - 10 (Oct) 388653 11.1      62.7     37.3               25.4
```

```
##  8 2020 - 11 (Nov) 259716  7.44     66.1     33.9             32.2
##  9 2020 - 12 (Dec) 131364  3.76     77.1     22.9             54.2
## 10 2021 - 01 (Jan)  96834  2.77     81.3     18.7             62.6
## 11 2021 - 02 (Feb)  49622  1.42     79.6     20.4             59.2
## 12 2021 - 03 (Mar) 228496  6.55     63.2     36.8             26.4
```

lets visualize this

```
ggplot(data = cycle) +
  geom_bar(mapping = aes(x = year_month , fill = member_casual)) +
  labs(x = "month" , title = "Viz - 2 : Distribution per Month") +
  coord_flip()
```



explanation from the chart :

from chart we find that there are more member rider than that of the casual one and at the month of august there is the highest data points nearly of ~18%.

lets check data distribution in week

```
cycle %>%
  group_by(weekday) %>%
  summarise(count = length(ride_id),
            "%" = (length(ride_id) / nrow(cycle)) * 100,
            "members_w"= (sum(member_casual == "member") / length(ride_id)) *
```

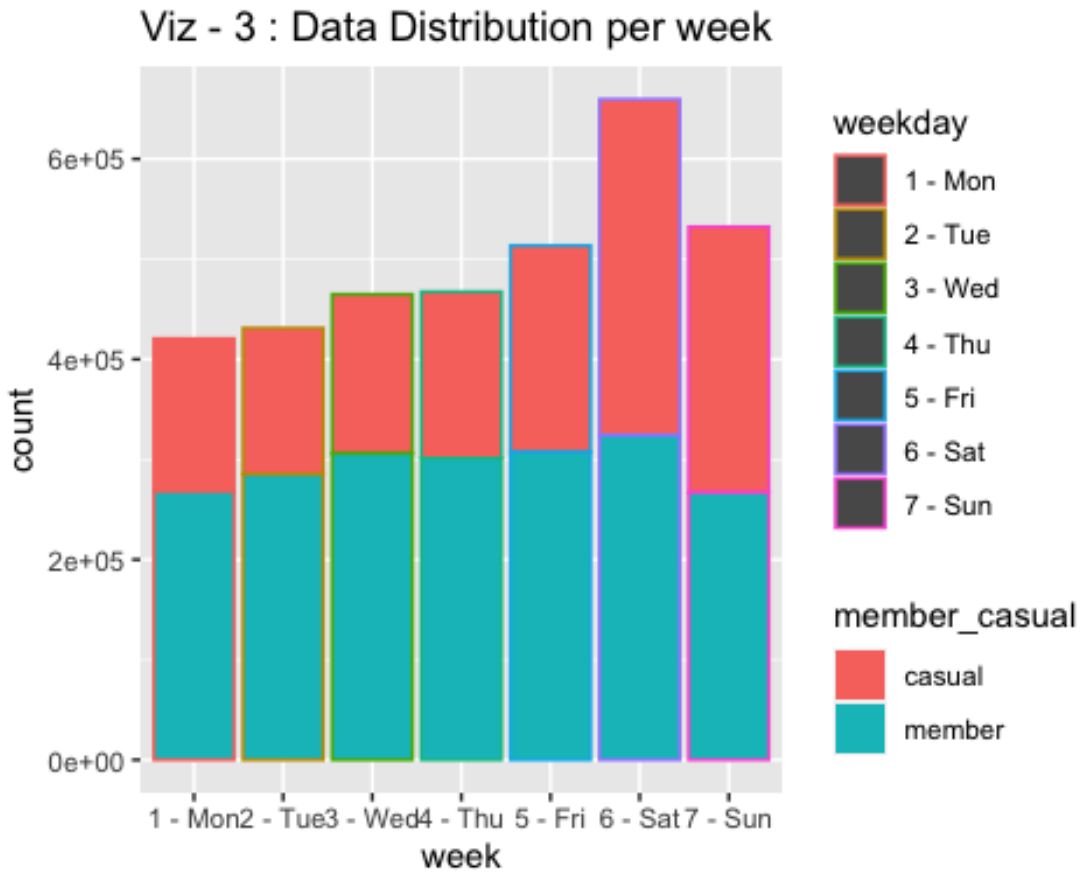
```
100,
            "casual_w" = (sum(member_casual == "casual") / length(ride_id)) *
100,
            "members - casual" = members_w - casual_w

            )
```

```
## # A tibble: 7 × 6
##   weekday  count   `%` members_w casual_w `members - casual`
##   <chr>    <int> <dbl>     <dbl>    <dbl>              <dbl>
## 1 1 - Mon 420613  12.1      63.7     36.3              27.5
## 2 2 - Tue 431131  12.4      66.2     33.8              32.3
## 3 3 - Wed 464879  13.3      65.9     34.1              31.7
## 4 4 - Thu 467450  13.4      64.4     35.6              28.9
## 5 5 - Fri 513585  14.7      59.8     40.2              19.6
## 6 6 - Sat 659637  18.9      49.2     50.8              -1.67
## 7 7 - Sun 532244  15.3      50.2     49.8               0.321
```

lets visualize this too

```
ggplot(data = cycle) +
  geom_bar(mapping = aes(x =weekday , fill = member_casual , color =
weekday)) +
  labs(x = "week" , title ="Viz - 3 : Data Distribution per week")
```



explanation :

we see that sat has the highest data points nearly 19%. we also see mostly data is occupied by the member except for the saturday.

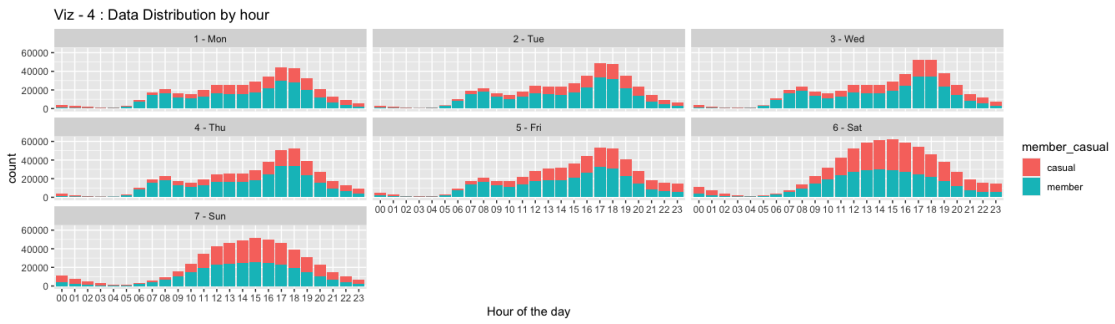
lets find the distribution based on the hour

```
cycle %>%
  group_by(start_hour) %>%
  summarise(count = length(ride_id),
            "%" = (length(ride_id) / nrow(cycle)) * 100,
            "member_h"= (sum(member_casual == "member") / length(ride_id) ) *
100,
            "casual_h"= (sum(member_casual == "casual") / length(ride_id)) *
100,
            "member - casual" = member_h - casual_h


            )
```

```
## # A tibble: 24 × 6
##    start_hour  count   `%` member_h casual_h `member - casual`
##    <chr>       <int> <dbl>    <dbl>    <dbl>             <dbl>
##  1 00          41924 1.20      33.4     66.6             -33.2
##  2 01          26372 0.756     29.5     70.5             -41.1
##  3 02          15386 0.441     27.5     72.5             -45.1
##  4 03           9038 0.259     27.6     72.4             -44.7
##  5 04           7391 0.212     41.3     58.7             -17.5
##  6 05          17987 0.515     75.0     25.0              50.0
##  7 06          56915 1.63      81.5     18.5              63.0
##  8 07         106045 3.04      81.6     18.4              63.2
##  9 08         133253 3.82      78.4     21.6              56.8
## 10 09         123699 3.54      72.0     28.0              44.0
## # … with 14 more rows
```

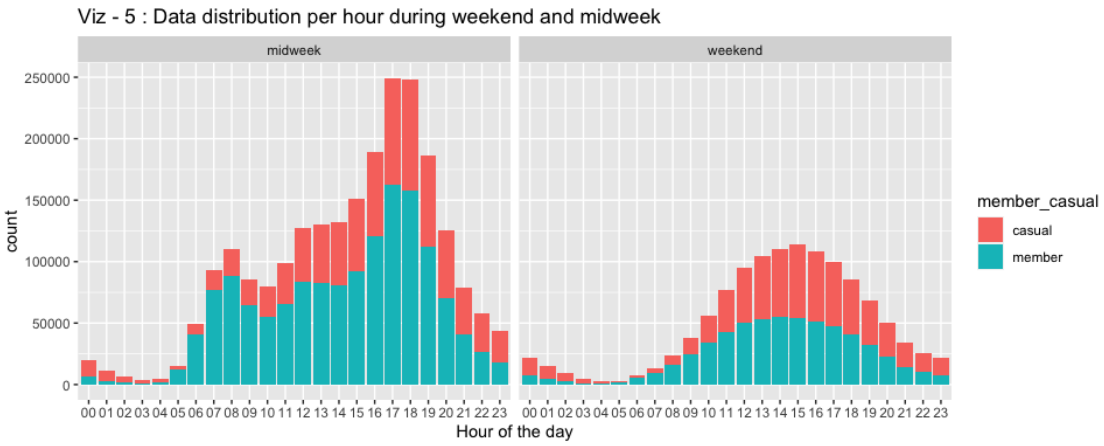
lets visualize it



there is huge difference in mid week and weekend.

```
cycle %>%
  mutate(type_of_weekday = ifelse(weekday == "7 - Sun" | weekday == "6 -
```

```
Sat", 'weekend', 'midweek' )) %>%
  ggplot(aes(x = start_hour ,fill = member_casual)) +
  geom_bar()+
  labs(x = "Hour of the day" , title = "Viz - 5 : Data distribution per hour
during weekend and midweek") +
  facet_wrap(~type_of_weekday)
```



explanation:

1. we can visualize and conclude that during the weekend days the data points are in smooth flow than that of the midweek.
2. there is big increase of data in between 6 AM to 8 AM and the data decrease drastically.
3. During weekend mid dat we see gradually increase in data.

its fundamental to question who are the riders, who use the bike mostly. during the midweek as in [2] we can conclude that it might be the office hour so their is increase in data in morning and same goes for the evening time.

let's check someother data in our datasets

```
summary(cycle$ride_time_m)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -29049.97      7.88     14.52     26.48     26.63  58720.03
```

looks like there are some negative values in this data frames.the negative values may have occurred due to the misplacement or the some station might have return the bad date.

```
ventiles <- quantile(cycle$ride_time_m , seq(0,1, by = 0.05))
ventiles
```

```
##            0%            5%           10%           15%           20%
## -29049.966667      3.100000      4.516667      5.683333      6.783333
##           25%           30%           35%           40%           45%
##      7.883333      9.033333     10.250000     11.533333     12.950000
##           50%           55%           60%           65%           70%
##     14.516667     16.283333     18.300000     20.650000     23.393333
```

```
##           75%           80%           85%           90%           95% 
##     26.633333     30.583333     36.400000     46.100000     73.050000 
##          100% 
##  58720.033333
```

we can see that the difference between 0% and 100% is nearly 87769 and the difference between 95% and 5% is 69 so we will use the data from 5% to 95% i.e we remove the outliers in the 5% of the data and use the subset of the data.

```
cycle_no_outliers <- cycle %>%
  filter(ride_time_m > as.numeric(ventiles['5%'])) %>%
  filter(ride_time_m < as.numeric(ventiles['95%']))

print(paste("removed" , nrow(cycle) - nrow(cycle_no_outliers), "rows as
outliers"))
```

```
## [1] "removed 350103 rows as outliers"
```

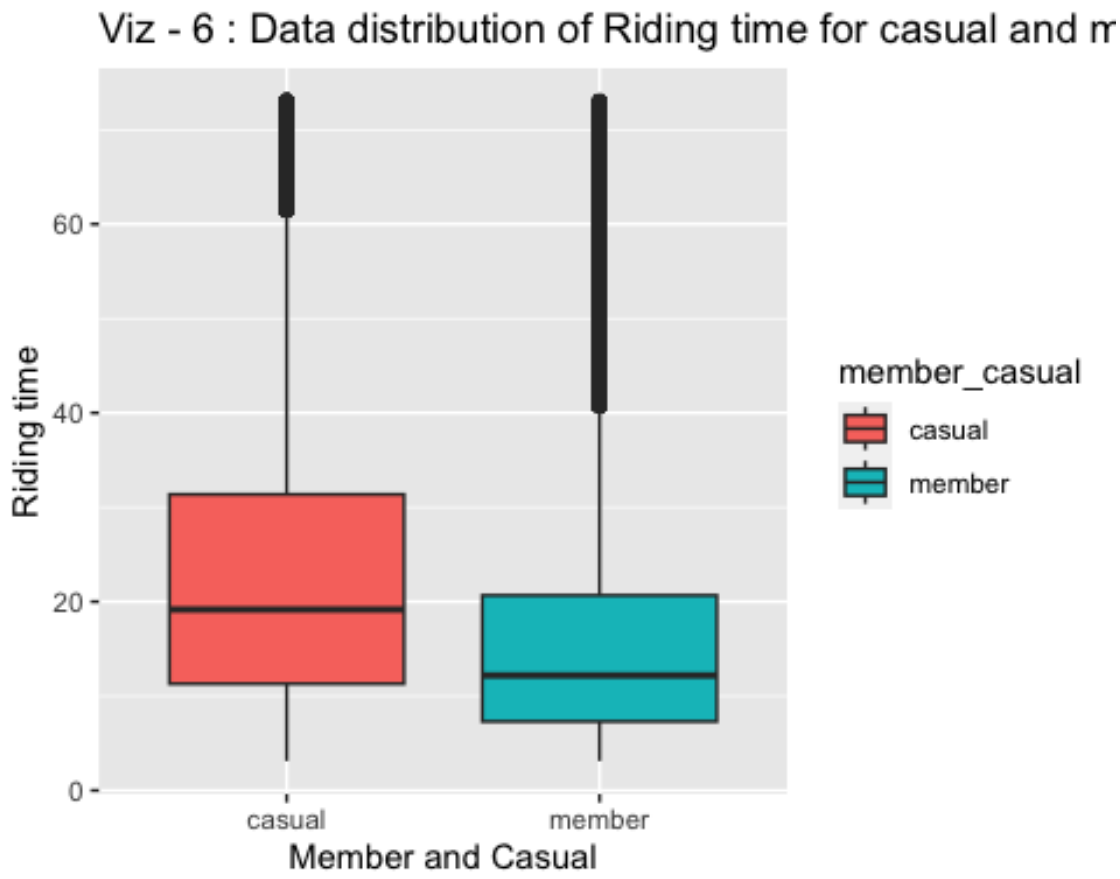
lets find the Q1,median,Q3

```
cycle_no_outliers %>%
  group_by(member_casual) %>%
  summarise(mean = mean(ride_time_m),
            'Q1' = as.numeric(quantile(ride_time_m, 0.25)),
            'median' = median(ride_time_m),
            'Q3' = as.numeric(quantile(ride_time_m , 0.75)),
            'IR'= Q3 - Q1
            )
```

```
## # A tibble: 2 × 6
##   member_casual  mean    Q1 median    Q3    IR
##   <chr>         <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 casual         23.7  11.3   19.2  31.4  20.1
## 2 member         15.5   7.3   12.2  20.7  13.4
```

lets visualize it

```
ggplot(data = cycle_no_outliers) +
  geom_boxplot(mapping = aes(x = member_casual ,  y=ride_time_m , fill =
member_casual )) +
  labs(x = "Member and Casual" , y = "Riding time" , title = "Viz - 6 : Data
distribution of Riding time for casual and member")
```

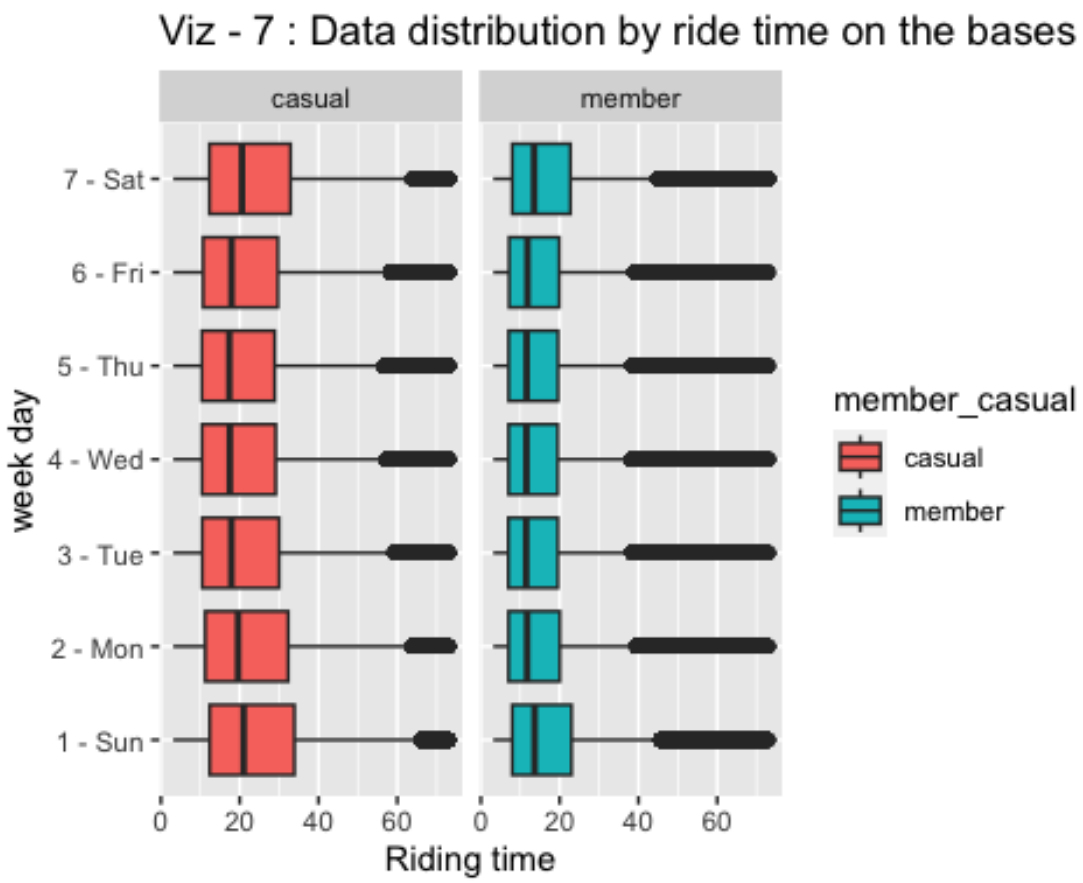


Explanation:

it is seen that the riding time for the casual is more than that of the member. mean is also more in casual than member.

lets try to plot same but for the week

```
ggplot(data = cycle_no_outliers) +
  geom_boxplot(mapping = aes(x = day_of_week, y = ride_time_m, fill =
member_casual)) +
  labs(x = "week day" , y = "Riding time" , title = "Viz - 7 : Data
distribution by ride time on the bases of week") +
  facet_wrap(~member_casual) +
  coord_flip()
```

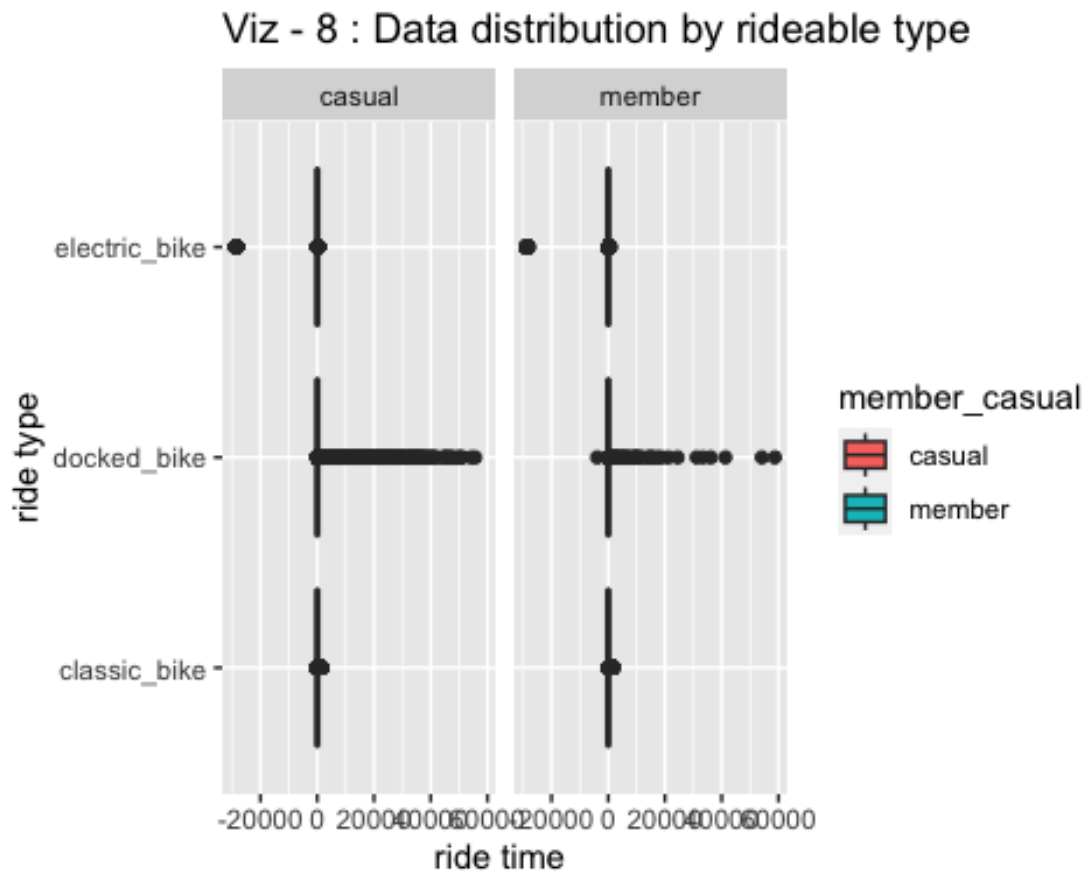


Explanation

we can see that the riding time remains unchanged during the mid-week for the member but changed during the weekend. casual has bell curve like structure looks like there is peak on sunday and saturday.

lets check the rideable_type

```
ggplot(data = cycle_no_dup) +
  geom_boxplot(mapping = aes(x= rideable_type, y = ride_time_m , fill
=member_casual )) +
  labs(x = "ride type" , y ="ride time " , title= "Viz - 8 : Data
distribution by rideable type") +
  facet_wrap(~member_casual) +
  coord_flip()
```



Explanation: Electric bike has less riding time than other two types . similarly docked bike has more riding time.

Guiding Questions:

1.How should you organize your data to perform analysis on it?

```
first of all all the month data is combined into single csv file.
```

2. Has your data been properly formatted?

   yes all the data is completely formatted we can confirm it by using the `str()` and `summary()`function

3. What surprises did you discover in the data?

   initially when i analyze the data based on month it seems there are more member than casual but when i analyze based on weekdays it is found that there are more casual then member.

4. What trends or relationships did you find in the data?

   - there are more member than the casual.

- during august there is high data points i.e more riders
- during weeked there is increaese in the rides.
- members use the bike on schedule during the weekdays generally from 6 to 8 at the morning and 4 to 6 in the evening
- member has the less riding time
- both casual and member like to ride the docked bike most and electric bike less

5. How will these insights help answer your business questions?

   as our main motive is to convert the casual to members this will help to build profile for the members.

Share

The share phase is usually done by building a presentation. But for this case study, the best representation of the analysis and conclusions is it's own notebook.

lets find what we have found and try to conclude our conclusion:

firstly what we know about the data:

```
 1. there are more member than that of the casual one.
 2. from viz - 2 we can say that during last semester of the 2020 there are
more data points especially during august it has more data points.
 3. during weekend there is increase in the number of the rides.
 4. during midweek morning time and evening time has more ride counts.
 5. Casual has more riding time than that of the member.
```

Now lets see how member differs from the casual one.

```
* members has the highest number of data besides the saturday. during this
day casual has more data points we can see this in viz - 3.
* we have more members during the morning time and more casual during the mid
day during the weekdays.
* There's a big increase of data points in the midweek between 6am to 8am for
members. Then it fell a bit. Another big increase is from 5pm to 6pm.
* there's a big flow of casual during the weekend especially during the 11 AM
to 3PM and start to slightly decrease proprtionaly afterwards.
* from viz - 6 we can conclude that though there are more members but the
casual has more riding time.
* bike preference can be concluded by the viz - 8
```

our findings shows that the members have a fixed used for bike than the casual ones as they use the bike for exercise or for going to work . this findings is from the use of the bike from the viz - 5. This can also be proven as we state that we have more members in between 6am to 8am and at 5pm to 6pm. Also, members may have set routes when using the bikes, as proven by riding time for members keeps unchanged during the midweek.

members has also more reference on the classic bike as they can exercise while going to work or they make them fit.

Case study road map share

Guiding questions

● Were you able to answer the question of how annual members and casual riders use Cyclistic bikes differently?

```
yes data points from different visualization says it all.
```

● What story does your data tell?

```
The main story the data tells is that members have set schedules, as seen on
viz - 05 on key timestamps. Those timestamps point out that members use the
bikes for routine activities, like going to work. Viz like 07 also point out
that they have less riding time, because they have a set route to take.
```

● How do your findings relate to your original question?

```
our main questions was "How do annual members and casual riders use Cyclistic
bikes differently?".
```

● Who is your audience? What is the best way to communicate with them?

```
the main audience is the market analysis team and our analysis head Moreno.
The best way of communicating them is through the slide presentaion.
```

● Can data visualization help you share your findings?

```
yes all the viz chart created so far help share our findings.
```

● Is your presentation accessible to your audience?

```
yes the plots are understandable and vibrant colors are used.
```

###Key tasks

1. Determine the best way to share your findings.
2. Create effective data visualizations.
3. Present your findings.
4. Ensure your work is accessible.

Deliverable

Supporting visualizations and key findings

Act

the act phase will be implemented by the marketing team. the main takeway will be the three recommendations based on this analysis.

Guiding questions

1. What is your final conclusion based on your analysis?

   as a final conclusion i can conclude on based on my analysis that both casual and member has different perception and habit while riding the bikes.

2. How could your team and business apply your insights?

   The insights could be implemented when preparing a marketing campaign for turning casual into members. The marketing can have a focus on workers as a non effect on enivronment way to get to work.

3. What next steps would you or your stakeholders take based on your findings?

   Further analysis could be done to improve the findings, besides that, the marketing team can take the main information to build a marketing campaign.

4. Is there additional data you could use to expand on your findings?

   - more info about the members could be helpful
   - data related to climate impact would be helpful on determining the effect on riders

Deliverable

Your top three recommendations based on your analysis

1. first of all the market campaign must focus on how the fuel based transport is increasing global warming , climate change and how it will impact on our future generation and how use of cycle will help on reducing it. as a whole how we can maintain the green planet concept.
2. social media platform should be used as a part of the marketing strategy and discount should be provided to the annual members like free ride during the weekend.
3. as analysis shows all the bikers are more engaged during the weekend may be as the part of the exercise during such period ads campaign can be conducted .

Conclusion

this case study from the Google data analytics certificate helps and teach me a lot about the different phases of the data analysis and help me analyzing and visualizing the data using the R.