Google Data Analytics Professional Certificate

Capstone Project Report

Huy Tran
tranbaohuyofficial@outlook.com

Table of Contents

Background Scenario

Cyclistic is a bike – share company in Chicago, with more than 5,800 bikes at 600 docking stations. The company offers service with various types of bike and three pricing models: single – ride pass, full – day pass, and annual subscription. The marketing analytics team believe that in order to maximize the company's success, Cyclistic needs to influence more casual riders to subscribe for annual membership.

I am a junior data analyst that recently joined the marketing analytics team. Lily Moreno – the director of marketing, assigned me the investigation and analyzing of the difference between how casual and annual riders use the bikes. My analysis will be conducted regarding the 6 – phase analytical process: Ask, Prepare, Process, Analyze, Share, Act.

Statement of Business Task (ASK)

The objective of this business task is to answer the question how do annual members and casual riders use Cyclistic bike differently? The big problem we are trying to solve is how to convert more casual riders to annual members. Therefore, knowing the factors that influence the decision of customers to either choose casual or annual subscription will provide insights into the needed changes to make annual membership a more appealing choice for customers.

The scope of this task is to investigate and analyze the habits of renting bikes from customers to find the underlying reasons that influence their choice. The primary stakeholders are the riders, Lily Moreno (marketing director), Cyclistic marketing analytics team, Cyclistic executive team. The secondary stakeholders can be the other employees at Cyclistic.

Description of Data Source (PREPARE)

1, Location

The original data is stored in an AWS S3 bucket named “divvy-tripdata”. This is the location where Cyclistic stores the company’s internal data of the customers’ rental activities.

2, Organization

The original data is organized into tables under the form of csv files. Each row in the table represents a single ride (rental activity) while each column represents a different attribute of that ride. There are currently data from many years from 2013 to 2022. The data of more recent years is divided into 12 tables each year for 12 months while the data of less recent years is divided by quarters. I decided to work with the data from March 2023 with approximately 280,000 rows of data.

3, Sort and Filter

I noticed there were many rows with empty cells and decided to delete those rows to reduce the size of the dataset. Since the dataset consisted of around 280,000 rows, deleting rows with blanks was a feasible and effective option that would not alter the final result. I did this by using COUNTBLANK() function in Excel to count the number of blank cells in a row and delete the rows with that number >= 0.

Then, I applied simple sorting by the type of bicycle (from A – Z) and the start time (from newest – oldest). The dataset still had many flaws and duplicated values that would be dealt with in the Process phase.

4, Quality of Data

I believe my choice of data source satisfies the ROCCC principle. The source is reliable (R), original (O), and Comprehensive (C) since it came from the internal data of the company. The source is from March 2023, which is the most recent month in the database, satisfying the current (C) criteria. Moreover, the data source is the activity trail of all business activities in March; therefore, there is minimal chance of the data being biased.
This public dataset is used under the license from Motivate International Inc.

5, Limitation of Data

There are two limitations with the data. Firstly, the size of the datasets is too large (hundreds thousands of rows for each month). Secondly, due to privacy measure, the purchase and customer details are not accessible. It is impossible to tell whether a casual customer corresponds to many rides id. This forces me to assume that each ride id corresponds to a unique rider.

Documentation of Data Cleaning & Manipulating (PROCESS)

1, Excel

I have done some cleaning to remove rows with empty cells in the Prepare phase. Next, I add two new columns: ride_length and day_of_week as required to the table. The ride_length column shows the duration of the ride. The day_of_week column shows the day of the week corresponds to the start date (1: Sunday, 7: Saturday) with the WEEKDAY() function.

N	O
ride_length	day_of_week
0:10:22	6
0:14:25	6
0:08:34	6
2:26:47	6
0:08:40	6
2:26:58	6

Before exporting the csv file, I use Format Cells to change the datatype of "start_at" and "end_at" columns to the datetime datatype (yyyy-mm-dd hh-mm-ss) of MySQL.

I then export the dataset to a csv file named "202303-divvy-tripdata-prepared.csv" for further cleaning with SQL.

2, SQL

I use MySQL server and MySQL workbench to clean and manipulate the data. To load a large dataset into workbench, I use LOAD DATA INFILE statement for better performance and quicker loading time. The cleaning process consists of several actions:

- Remove duplicate values

  ROW_NUMBER() function is used with PARTITION BY to get the sequential number of each row. Then the row with the sequential number > 1 (duplicated) is deleted.

- Fix structural errors

  I notice the station_id values of the start station and end station are not consistent (some are TA0000000, some are 000000, some are SL-000).

  I decide to clean the data by three steps:

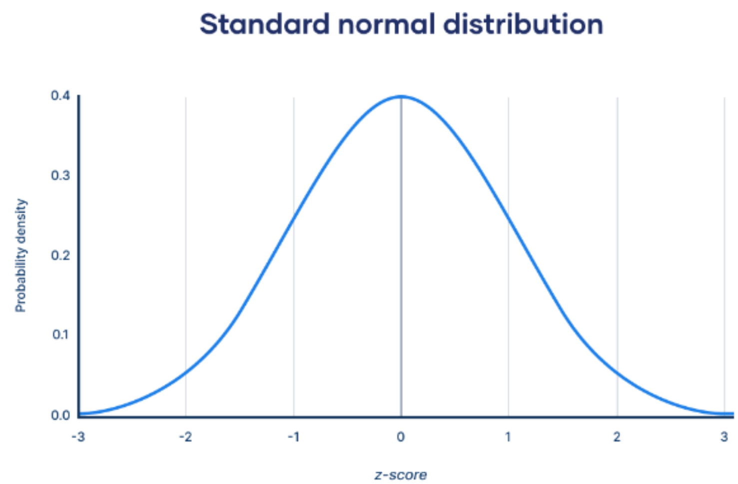
  - remove non-numeric characters
  - delete rows with station_id with length < 5
  - take only the first five digits as the new station_id

  This is approach is simple but can create stations with the same id. However, at the moment, I am not able to perform more complex queries to form a better approach.

- Remove extreme outliers

  There are some extreme outliers in the ride_length attribute (ride that last more than 3 hours). Although they can be dealt more thoroughly with careful consideration, removing them is the simplest option for this large dataset.

  I calculate the zscore (an evaluative metric) by using the AVG() and STD() functions (average and standard deviation). The further away the zscore from 0, the higher possibility of that row being an outlier.



(Photo from Scribbr.com)

Normally, zcore > 3 or < - 3 is enough to differentiate outlier. However, the majority of the dataset (>95%) is rides under 2 hour long. Therefore, I decide to use the condition of zscore > 6, zscore < -6 to remove the extreme outliers. 265 rows are removed.

3, Result & Limitation

After cleaning with SQL, the number of rows in the dataset reduces from 200448 to 163320. Below is the quick comparison of before and after cleaning.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
1	ride_id	rideable_type	started_at	ended_at	start_station_name	start_station_id	end_station_name	end_station_id	start_lat	start_lng	end_lat	end_lng	member_casual	ide_length	ay_
2	FA97EE7A82714405	classic_bike	3/31/2023 23:59	4/1/2023 0:09	Halsted St & Roscoe	TA1309000025	Lincoln Ave & Diverse	TA1307000064	41.94367	-87.649	41.93223	-87.6586	member	0:10:22	
3	8186F83C4986AA2B	classic_bike	3/31/2023 23:59	4/1/2023 0:13	Lincoln Ave & Divers	TA1307000064	Clybourn Ave & Divisi	TA1307000115	41.93223	-87.6586	41.90461	-87.6406	member	0:14:25	
4	8C1ADFAF3432AD45	classic_bike	3/31/2023 23:58	4/1/2023 0:07	University Ave & 57t	KA1503000071	Shore Dr & 55th St	TA1308000009	41.79148	-87.5999	41.79521	-87.5807	member	0:08:34	
5	B999A5C3BBD9AEA0	classic_bike	3/31/2023 23:58	4/1/2023 2:25	Franklin St & Jacksor	TA1305000025	Franklin St & Jackson	TA1305000025	41.87771	-87.6353	41.87771	-87.6353	casual	2:26:47	
6	1797846702BD8CEA	classic_bike	3/31/2023 23:58	4/1/2023 0:07	Greenview Ave & Di	13294	Stockton Dr & Wright	13276	41.93259	-87.6659	41.93132	-87.6387	member	0:08:40	
7	73389A02CC639AF0	classic_bike	3/31/2023 23:58	4/1/2023 2:25	Franklin St & Jacksor	TA1305000025	Franklin St & Jackson	TA1305000025	41.87771	-87.6353	41.87771	-87.6353	casual	2:26:58	
8	FDC8306D419AE8F0	classic_bike	3/31/2023 23:58	4/1/2023 0:01	Wentworth Ave & 3	15445	Calumet Ave & 33rd S	13217	41.83453	-87.6318	41.8349	-87.6179	member	0:03:38	
9	4EEF8E88AEC5FEB0	classic_bike	3/31/2023 23:57	4/1/2023 2:24	Franklin St & Jacksor	TA1305000025	Franklin St & Jackson	TA1305000025	41.87771	-87.6353	41.87771	-87.6353	casual	2:27:18	
10	056DC876647BCB37	classic_bike	3/31/2023 23:57	4/1/2023 0:06	Kimbark Ave & 53rd	TA1309000037	Harper Ave & 59th St	KA1503000070	41.79957	-87.5947	41.78794	-87.5883	member	0:09:07	
11	6F4451DD01FF5860	classic_bike	3/31/2023 23:54	3/31/2023 23:58	Halsted St & Polk St	TA1307000121	Halsted St & Roosevel	TA1305000017	41.87184	-87.6466	41.86732	-87.6486	member	0:04:11	
12	A83C7A154496EFDB	classic_bike	3/31/2023 23:53	4/1/2023 0:06	McClurg Ct & Ohio S	TA1306000029	Michigan Ave & 8th St	623	41.89259	-87.6173	41.87277	-87.624	casual	0:13:08	
13	454422D26EAEF418	classic_bike	3/31/2023 23:53	3/31/2023 23:59	Milwaukee Ave & Gr	13033	Aberdeen St & Randol	18062	41.89158	-87.6484	41.88411	-87.6543	member	0:06:54	
14	ED47DFEC350E6775	classic_bike	3/31/2023 23:52	3/31/2023 23:55	Damen Ave & Pierce	TA1305000041	Paulina Ave & North A	TA1305000037	41.9094	-87.6777	41.90985	-87.6699	casual	0:02:55	
15	49B3DE91FF5D1621	classic_bike	3/31/2023 23:52	3/31/2023 23:59	Lincoln Ave & Belle	TA1309000026	Leavitt St & Lawrence	TA1309000015	41.956	-87.6802	41.96889	-87.684	member	0:06:25	
16	99932AF08CD922DA	classic_bike	3/31/2023 23:51	4/1/2023 0:24	Sangamon St & Lake	TA1306000015	Pine Grove Ave & Irvi	TA1308000022	41.88578	-87.651	41.95438	-87.648	member	0:32:31	
17	0A5470B8165E6F52	classic_bike	3/31/2023 23:51	4/1/2023 0:05	Clark St & Grace St	TA1307000127	Clark St & Bryn Mawr	KA1504000151	41.95078	-87.6592	41.98359	-87.6692	member	0:14:09	
18	754076E627644A17	classic_bike	3/31/2023 23:51	4/1/2023 0:02	Halsted St & Fulton	23003	LaSalle St & Illinois St	13430	41.89	-87.65	41.89076	-87.6317	casual	0:11:03	
19	8C5E39A9DCEBBF7A	classic_bike	3/31/2023 23:51	4/1/2023 0:02	Halsted St & Fulton	23003	LaSalle St & Illinois St	13430	41.89	-87.65	41.89076	-87.6317	member	0:11:14	
20	F25C544745A59C67	classic_bike	3/31/2023 23:50	3/31/2023 23:55	Morgan St & Polk St	TA1307000130	Halsted St & Maxwell	TA1309000001	41.87174	-87.651	41.86488	-87.6471	member	0:04:52	
21	46EA756BD97C2999	classic_bike	3/31/2023 23:50	4/1/2023 0:13	California Ave & Cor	17660	Central Park Ave & Ell	15644	41.90036	-87.6967	41.93534	-87.7169	casual	0:23:29	

1	ride_id	rideable_type	started_at	ended_at	start_station_name	start_station_id	end_station_name	end_station_id	start_lat	start_lng	end_lat	end_lng	member_casual	ride_length
2	FA97EE7A82714405	classic_bike	3/31/2023 23:59	4/1/2023 0:09	Halsted St & Roscoe St	13090	Lincoln Ave & Dive	13070	41.944	-87.649	41.932	-87.659	member	0:10:22
3	8186F83C4986AA2B	classic_bike	3/31/2023 23:59	4/1/2023 0:13	Lincoln Ave & Diversey	13070	Clybourn Ave & Div	13070	41.932	-87.659	41.905	-87.641	member	0:14:25
4	8C1ADFAF3432AD45	classic_bike	3/31/2023 23:58	4/1/2023 0:07	University Ave & 57th S	15030	Shore Dr & 55th St	13080	41.791	-87.600	41.795	-87.581	member	0:08:34
5	B999A5C3BBD9AEA0	classic_bike	3/31/2023 23:58	4/1/2023 2:25	Franklin St & Jackson Bl	13050	Franklin St & Jacks	13050	41.878	-87.635	41.878	-87.635	casual	2:26:47
6	1797846702BD8CEA	classic_bike	3/31/2023 23:58	4/1/2023 0:07	Greenview Ave & Divers	13294	Stockton Dr & Wrig	13276	41.933	-87.666	41.931	-87.639	member	0:08:40
7	73389A02CC639AF0	classic_bike	3/31/2023 23:58	4/1/2023 2:25	Franklin St & Jackson Bl	13050	Franklin St & Jacks	13050	41.878	-87.635	41.878	-87.635	casual	2:26:58
8	FDC8306D419AE8F0	classic_bike	3/31/2023 23:58	4/1/2023 0:01	Wentworth Ave & 33rd	15445	Calumet Ave & 33r	13217	41.835	-87.632	41.835	-87.618	member	0:03:38
9	4EEF8E88AEC5FEB0	classic_bike	3/31/2023 23:57	4/1/2023 2:24	Franklin St & Jackson Bl	13050	Franklin St & Jacks	13050	41.878	-87.635	41.878	-87.635	casual	2:27:18
10	056DC876647BCB37	classic_bike	3/31/2023 23:57	4/1/2023 0:06	Kimbark Ave & 53rd St	13090	Harper Ave & 59th	15030	41.800	-87.595	41.788	-87.588	member	0:09:07
11	6F4451DD01FF5860	classic_bike	3/31/2023 23:54	3/31/2023 23:58	Halsted St & Polk St	13070	Halsted St & Roose	13050	41.872	-87.647	41.867	-87.649	member	0:04:11
12	454422D26EAEF418	classic_bike	3/31/2023 23:53	3/31/2023 23:59	Milwaukee Ave & Gran	13033	Aberdeen St & Ran	18062	41.892	-87.648	41.884	-87.654	member	0:06:54
13	ED47DFEC350E6775	classic_bike	3/31/2023 23:52	3/31/2023 23:55	Damen Ave & Pierce Av	13050	Paulina Ave & Nort	13050	41.909	-87.678	41.910	-87.670	casual	0:02:55
14	49B3DE91FF5D1621	classic_bike	3/31/2023 23:52	3/31/2023 23:59	Lincoln Ave & Belle Plai	13090	Leavitt St & Lawre	13090	41.956	-87.680	41.969	-87.684	member	0:06:25
15	99932AF08CD922DA	classic_bike	3/31/2023 23:51	4/1/2023 0:24	Sangamon St & Lake St	13060	Pine Grove Ave & I	13080	41.886	-87.651	41.954	-87.648	member	0:32:31
16	0A5470B8165E6F52	classic_bike	3/31/2023 23:51	4/1/2023 0:05	Clark St & Grace St	13070	Clark St & Bryn Ma	15040	41.951	-87.659	41.984	-87.669	member	0:14:09
17	754076E627644A17	classic_bike	3/31/2023 23:51	4/1/2023 0:02	Halsted St & Fulton St	23003	LaSalle St & Illinois	13430	41.890	-87.650	41.891	-87.632	casual	0:11:03

Before
After

I use Format Cells to limit the number of character after decimal point to 3 to reduce the size of longitude, latitude columns.
My cleaning process is still very simple and has limitation. The process of reformatting the station_id can result in duplicate id. The outliers can be treated more thoroughly. More cleaning processes can be applied. My future projects will be done better. The data is exported as a CSV file named “202303-divvy-tripdata-processed.csv”.

Summary of Analysis (ANALYZE)

I use R for the Analyze phase since this is a powerful tool to work with a large dataset and I would like to practice my skills with R. The process is fairly straightforward since there are clear instructions. I imported the csv file from the Process phase into a new dataframe named "huy_data" and then conduct some descriptive analysis.

- Examine mean, median, max, min of the ride_length attribute.

```
> mean(huy_data$ride_length)
Time difference of 649.2728 secs
> median(huy_data$ride_length)
00:07:32
> max(huy_data$ride_length)
Time difference of 10798 secs
> min(huy_data$ride_length)
Time difference of 0 secs
```

These are important metrics to describe the average duration of rides.

- Compare mean, median of ride_length between casual riders and annual members.

```
> aggregate(huy_data$ride_length ~ huy_data$member_casual, FUN = mean)
  huy_data$member_casual huy_data$ride_length
1                 casual        879.3337 secs
2                 member        581.0334 secs
> aggregate(huy_data$ride_length ~ huy_data$member_casual, FUN = median)
  huy_data$member_casual huy_data$ride_length
1                 casual                  550
2                 member                  426
```

This tells that annual members tend to have shorter rides. This can be explained by the fact that many annual members use bikes to commute on fixed routes to work, to school, which cannot be too long since they are doing it daily.

- Compare the mean of ride_length between casual riders and annual members by weekday.

```
> aggregate(huy_data$ride_length ~ huy_data$member_casual + huy_data$day_of_week, FUN = mean)
   huy_data$member_casual huy_data$day_of_week huy_data$ride_length
1                  casual                    1       1068.2885 secs
2                  member                    1        632.0353 secs
3                  casual                    2        845.9633 secs
4                  member                    2        561.3674 secs
5                  casual                    3        879.4678 secs
6                  member                    3        587.4279 secs
7                  casual                    4        831.8446 secs
8                  member                    4        585.3482 secs
9                  casual                    5        800.2825 secs
10                 member                    5        563.1532 secs
11                 casual                    6        870.1672 secs
12                 member                    6        554.2706 secs
13                 casual                    7        906.9335 secs
14                 member                    7        621.2026 secs
```

We can see that Sunday (1) is when people tend to ride bikes the longest.

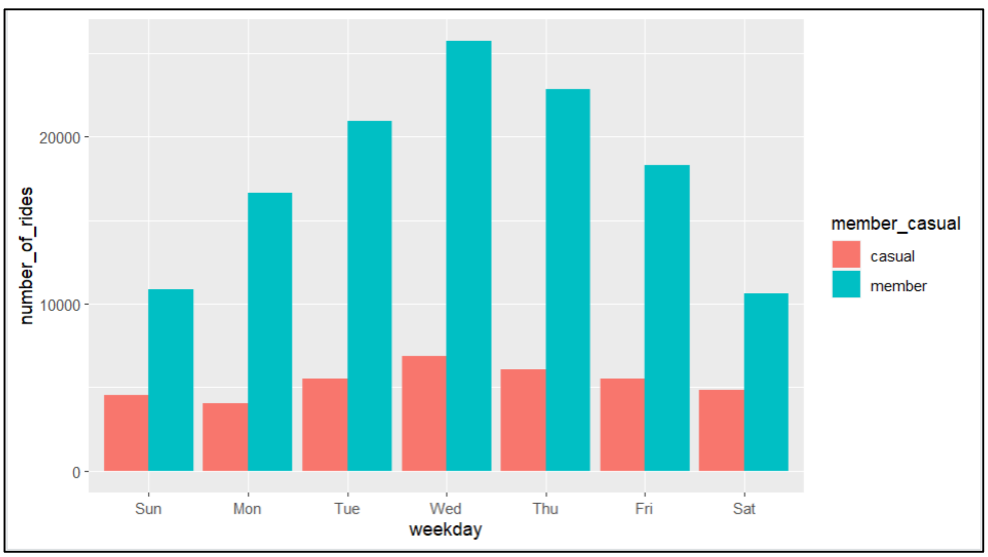
- Compare the number of rides and average_duration between casual riders and members

```
> huy_data %>%
+     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 duration
+               ,average_duration = mean(ride_length)) %>%                         # calculates the average duration
+     arrange(member_casual, weekday)
`summarise()` has grouped output by 'member_casual'. You can override using the `.groups` argument.
# A tibble: 14 × 4
# Groups:   member_casual [2]
   member_casual weekday number_of_rides average_duration
   <chr>         <ord>             <int> <drtn>
 1 casual        Sun                4538 1068.2885 secs
 2 casual        Mon                4010  845.9633 secs
 3 casual        Tue                5496  879.4678 secs
 4 casual        Wed                6880  831.8446 secs
 5 casual        Thu                6075  800.2825 secs
 6 casual        Fri                5533  870.1672 secs
 7 casual        Sat                4829  906.9335 secs
 8 member        Sun               10874  632.0353 secs
 9 member        Mon               16630  561.3674 secs
10 member        Tue               20938  587.4279 secs
11 member        Wed               25763  585.3482 secs
12 member        Thu               22851  563.1532 secs
13 member        Fri               18302  554.2706 secs
14 member        Sat               10600  621.2026 secs
```

We can see that there is a higher number of rides during week days, especially with members, almost doubling the number in the weekends. This is because members use the bikes to perform daily commute tasks (work, school, shopping, etc).

- Visualize the number of rides between casual riders and members

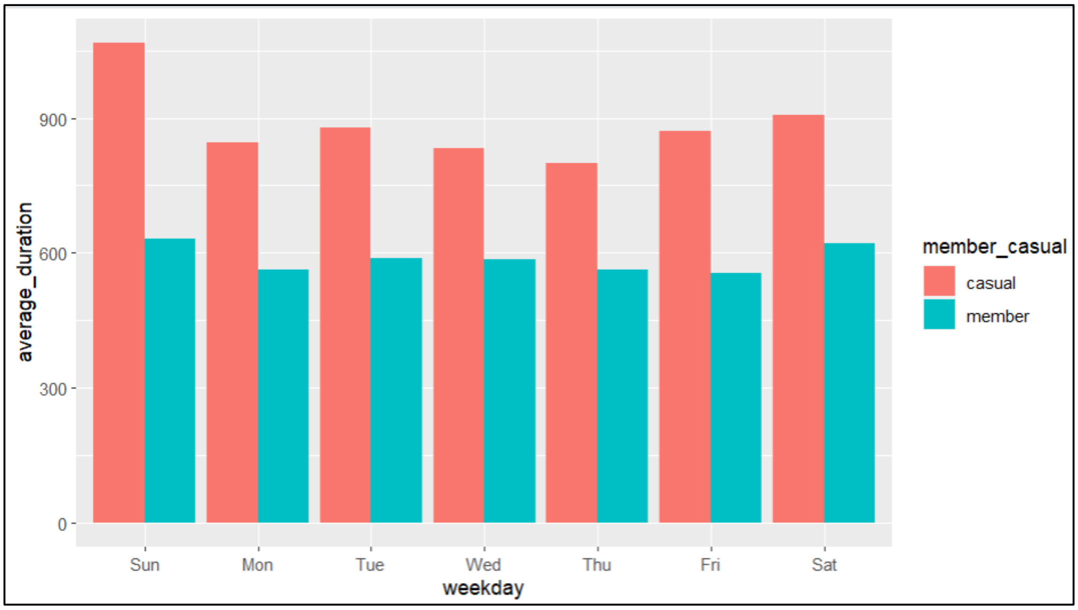
```
> huy_data %>%
+     mutate(weekday = wday(started_at, label = TRUE)) %>%
+     group_by(member_casual, weekday) %>%
+     summarise(number_of_rides = n()
+               ,average_duration = mean(ride_length)) %>%
+     arrange(member_casual, weekday)  %>%
+     ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
+     geom_col(position = "dodge")
`summarise()` has grouped output by 'member_casual'. You can override using the `.groups`
argument.
>
```



The graph further supports my earlier insight on the number of rides.

- Visualize the average_duration between casual riders and members

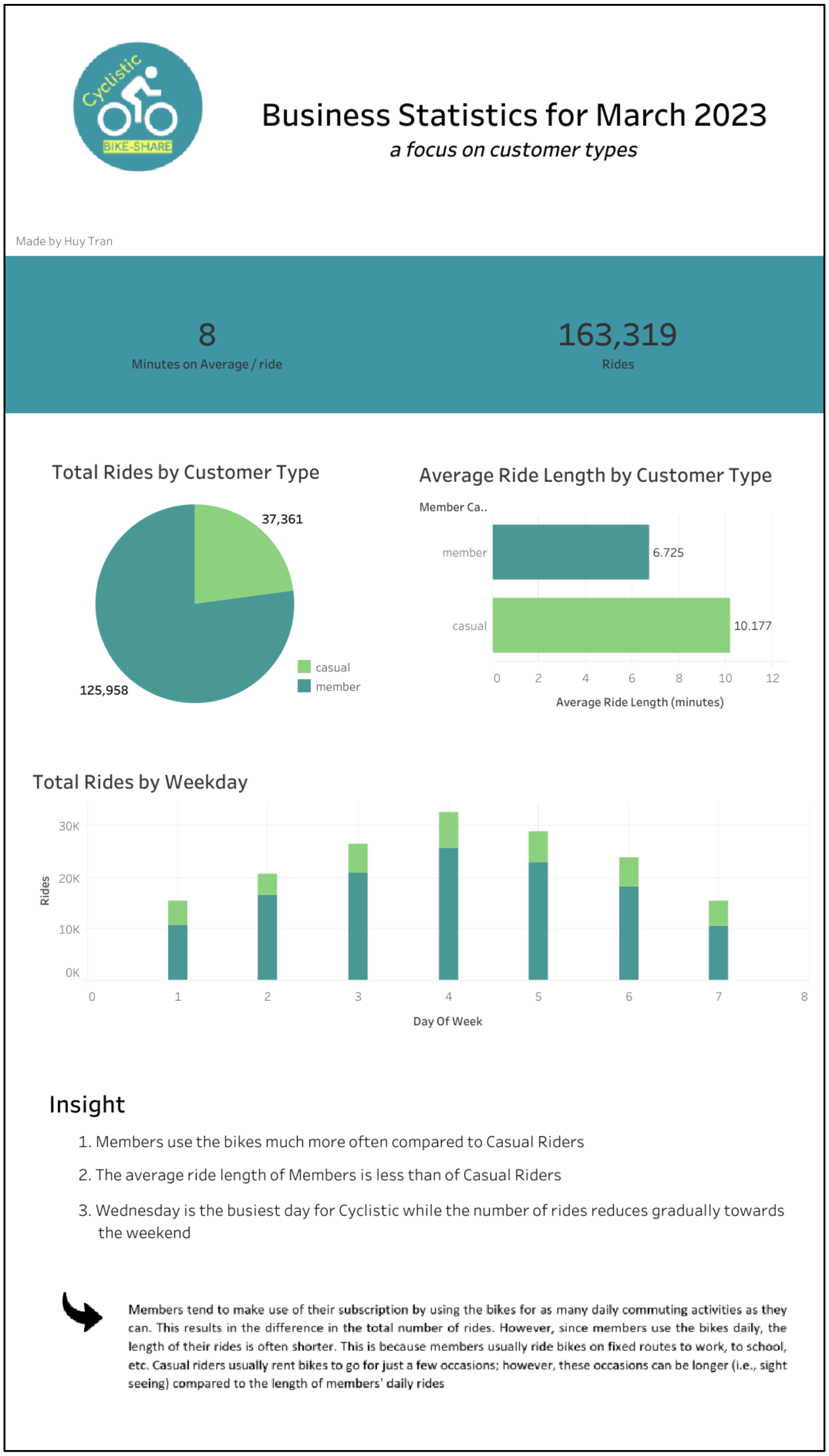
```
> huy_data %>%
+     mutate(weekday = wday(started_at, label = TRUE)) %>%
+     group_by(member_casual, weekday) %>%
+     summarise(number_of_rides = n()
+               ,average_duration = mean(ride_length)) %>%
+     arrange(member_casual, weekday)  %>%
+     ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
+     geom_col(position = "dodge")
`summarise()` has grouped output by 'member_casual'. You can override using the
`.groups` argument.
Don't know how to automatically pick scale for object of type <difftime>. Defaulting
to continuous.
>
```



The graph further supports my earlier insight on the average duration.

Visualization & Key Findings (SHARE)

I use Tableau to create a dashboard to illustrate my findings to the executive team. However, visualization can also be done by spreadsheets, Excel, R, or PowerBI.



Recommendation (ACT)

Based on my analysis, here is the top 3 recommendations to encourage more casual customers to subscribe for annual subscription:

- Set up marketing campaign to illustrate the benefit of using bikes for daily commuting.
- Give out more promotion upon signing up for the annual subscription.
- Create a community for annual members on social media to promote the daily use of bikes and arrange socializing event between members.

Limitation & Assumption

This project is an introductory level project with many limitations during some phases. The most important and challenging phase was the cleaning of the data (Prepare and Process); however, this can be done better by more detailed approaches. The methods used were not ideal and optimized due to the complexity of the dataset.

I finished this project in a short time frame to recollect the new knowledge I had gained from the certification. My future projects will certainly be improved. Thank you.