DATS 6401 Visualization of Complex Data

Instructor: Reza Jafari

Final Project Report

Author: Yuan Dang

Y.D 2022-05-03

Table of Contents

Abstraction......3
Introduction......4
Description......5
Data Preprocessing......6
Outlier Detection and Removal......8
Principle Component Analysis(PCA)......10
Normality......12
Data Transformation......13
Heatmap & Pearson Correlation......14
Statistics......15
Visualization......16
Recommendations......26
Reference......27

Abstraction

This final project aims to explore the dataset describing bike trips of a bike-sharing company in New York in May 2018, and try finding the answers to several questions: Who is the largest group of users in May 2018? How was the daily trend of the number of trips and number of users in May 2018? What is the station that the users visit most in May 2018?

Introduction

In order to accomplish the project, since the dataset is very large(1.6M observations), we first need to convert the data file to pickle file for faster processing. Then we can start to manipulate the dataset to identify and handle any missing values, identifiers, date time variables, categorical variables, outliers and non-normality.

After cleaning the dataset, we can start the exploration and visualization and observation.

Description

A bike-sharing service is a shared transport service in which bicycles are made available for shared use to individuals on a short-term basis for a certain price or free. Many bike share systems allow people to borrow a bike from a station and return it at another station belonging to the same system. It is importance for the industry to know where they should provide more bikes and where to reduce the number of bikes to cut the unnecessary cost.

This dataset contains bike trips of a bike-sharing company in New York for one month. The dataset consists of ≈ 1.6M rows and 11 columns. The attributes are:

1. *start_time (numeric): the time when a trip starts (in NYC local time).*

2. stop_time (numeric): the time when a trip is over (in NYC local time).

3. *start_station_id (categorical):* a unique code to identify a station where a trip begins.

4. *start_station_name (categorical):* the name of a station where a trip begins.

5. *end_station_id (categorical):* a unique code to identify a station where a trip is over.

6. *end_station_name (categorical):* the name of a station where a trip is over.

7. *user_type (categorical): the type of bike user.*

8. bike_id (categorical): a unique code to identify a bike user.

9. *gender (categorical):* gender of the user.

10. *age (numeric):* age of the user.

11. *trip_duration (numeric):* the duration of a trip (in minutes), the target variable.

In this case, our dependent variable is *trip_duration,* and other variables are independent variables.

Data Preprocessing

Check Missing Values

var	number of missing values
start_time	0
stop_time	0
start_station_id	0
start_station_name	0
end_station_id	0
end_station_name	0
user_type	0
bike_id	0
gender	0
age	0
trip_duration	0
trip_duration_hour	0
trip_duration_minute	0
trip_duration_second	0

No value is missing in this case.

Handle datetime and categorical variables

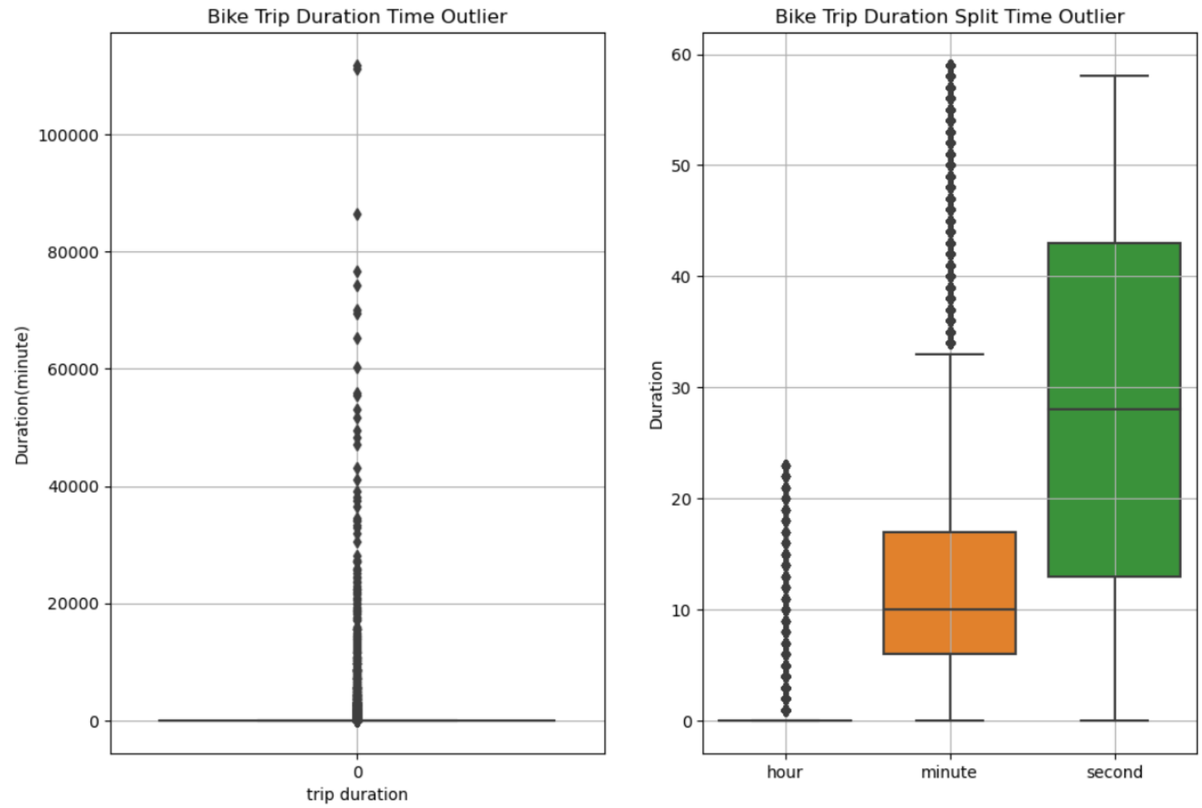
```
Data columns (total 14 columns):
 #   Column                Non-Null Count    Dtype
---  ------                --------------    -----
 0   start_time            1595334 non-null  datetime64[ns]
 1   stop_time             1595334 non-null  datetime64[ns]
 2   start_station_id      1595334 non-null  category
 3   start_station_name    1595334 non-null  category
 4   end_station_id        1595334 non-null  category
 5   end_station_name      1595334 non-null  category
 6   user_type             1595334 non-null  category
 7   bike_id               1595334 non-null  category
 8   gender                1595334 non-null  category
 9   age                   1595334 non-null  int64
 10  trip_duration         1595334 non-null  float64
 11  trip_duration_hour    1595334 non-null  int64
 12  trip_duration_minute  1595334 non-null  int64
 13  trip_duration_second  1595334 non-null  int64
dtypes: category(7), datetime64[ns](2), float64(1), int64(4)
memory usage: 103.9 MB
```

Dataset Header

start_time	stop_time	start_station_id	start_station_name	end_station_id	end_station_name	user_type	bike_id	gender	age	trip_duration
2018-05-31 23:59:59	2018-06-01 00:12:57	312	Allen St & Stanton St	460	S 4 St & Wythe Ave	Subscriber	25805	male	32	12.97
2018-05-31 23:59:59	2018-06-01 00:12:26	401	Allen St & Rivington St	360	William St & Pine St	Subscriber	17258	male	24	12.45
2018-05-31 23:59:51	2018-06-01 00:08:09	483	E 12 St & 3 Ave	368	Carmine St & 6 Ave	Subscriber	19692	male	39	8.28
2018-05-31 23:59:48	2018-06-01 00:07:33	3107	Bedford Ave & Nassau Ave	3076	Scholes St & Manhattan Ave	Subscriber	28285	male	28	7.75
2018-05-31 23:59:45	2018-06-01 00:07:48	3341	Central Park West & W 102 St	3400	E 110 St & Madison Ave	Subscriber	21000	female	51	8.05

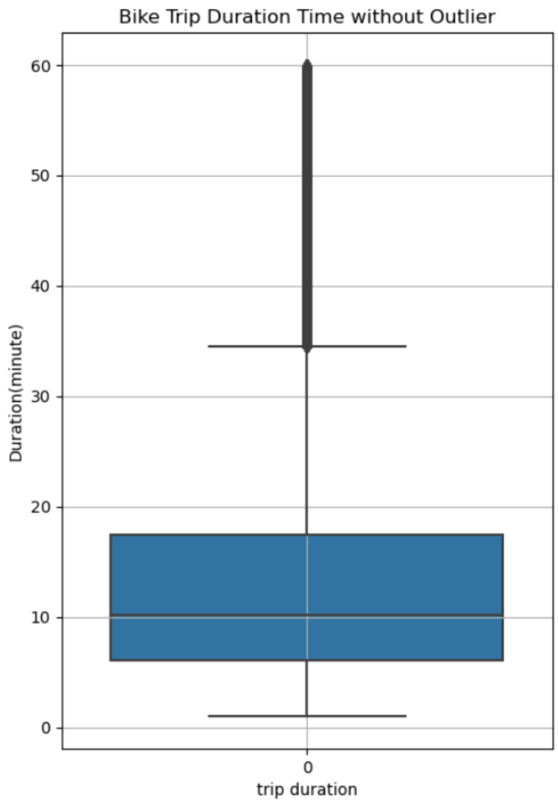
Outlier Detection and Removal

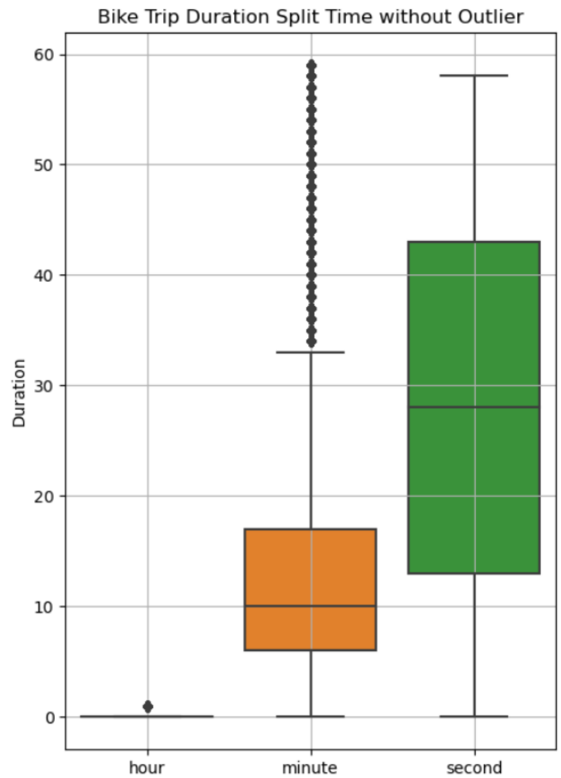
Detection with boxplot:



Notice that pretty much trip duration exceeds 10 hours, which is very unlikely for users to ride a bike for such a long time. It is probably caused by stolen, traffic accident, or inappropriate lock, if users do not appropriately lock the bike, it still counts as 'in-using'. So, let's decide to cut the duration time at 1 hour, in where about 99% of samples stay, and remove everyone more than 1 hour.

After removal:





Although there are still some ‘outliers’ showing in the box plot, we do not want to remove them, since an one hour bike-riding makes sense.

Principle Component Analysis(PCA)

Encoding the categorical variables: user type, gender

Encode these two categorical variables to dummy variables to get numeric columns.

Dummy Variables in Dataset

user_type_Customer	user_type_Subscriber	gender_female	gender_male
0	1	0	1
0	1	0	1
0	1	0	1
0	1	0	1
0	1	1	0

Select features

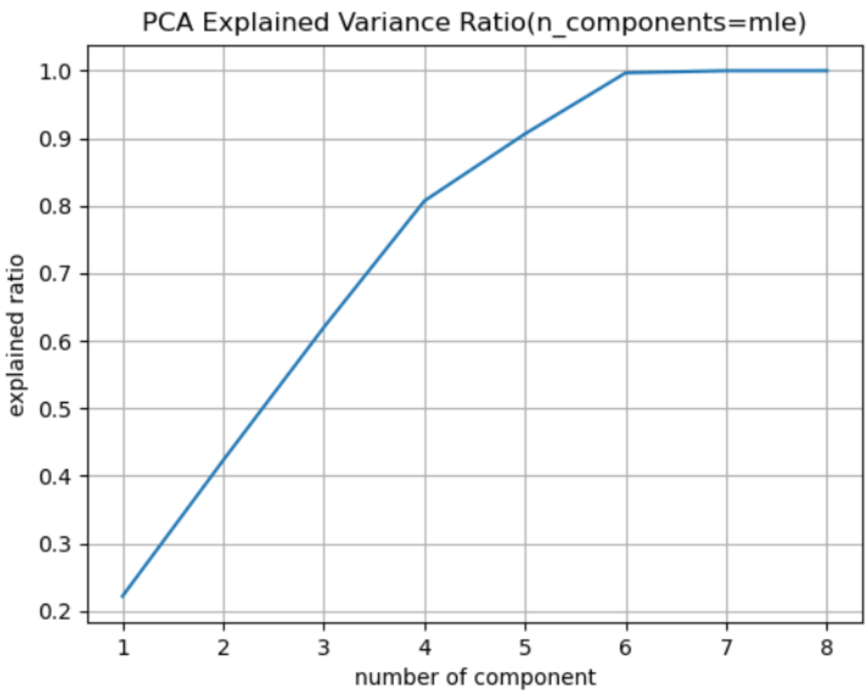
Select these four dummy variables together with original numeric variables: age, trip duration .etc.

Normalization

```
# normalization
X = df_dumy[features].values
X = StandardScaler().fit_transform(X)
```

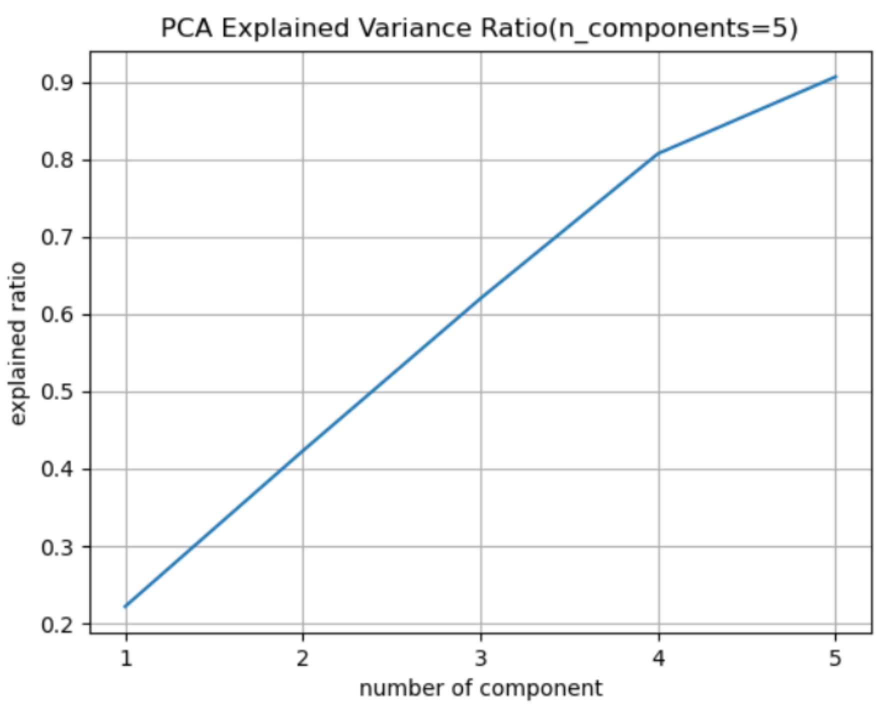
PCA with MLE

```
Original Dimension (1585428, 10)
Transformed Dimension (1585428, 8)
explained variance ratio: [2.21874264e-01 2.01081639e-01 1.96866939e-01 1.87350446e-01
 9.93098641e-02 9.03063492e-02 3.14758201e-03 6.29172918e-05]
```



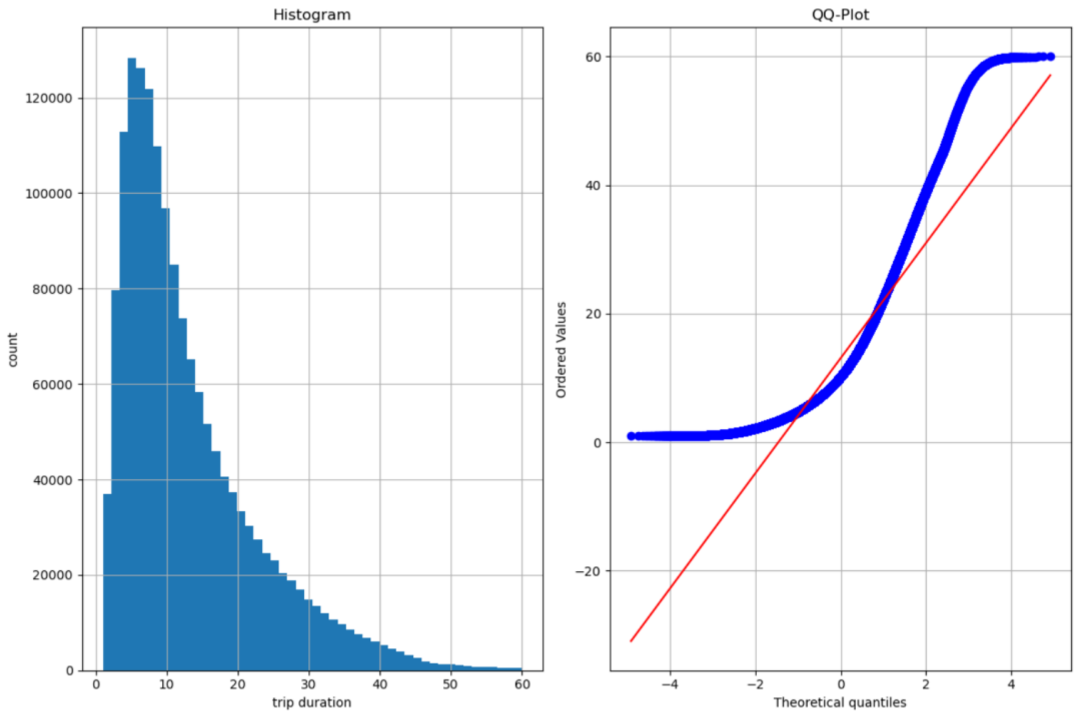
PCA reduced the dimension to 6 variables, but notice the explained variance ratio reach 90% with first five variables.

PCA with n=5



Normality

Histogram & QQ-Plot



The histogram seems to show that data is skewed normal distribution, but from the qq-plot the data seems non-normally distributed.

Normality Test

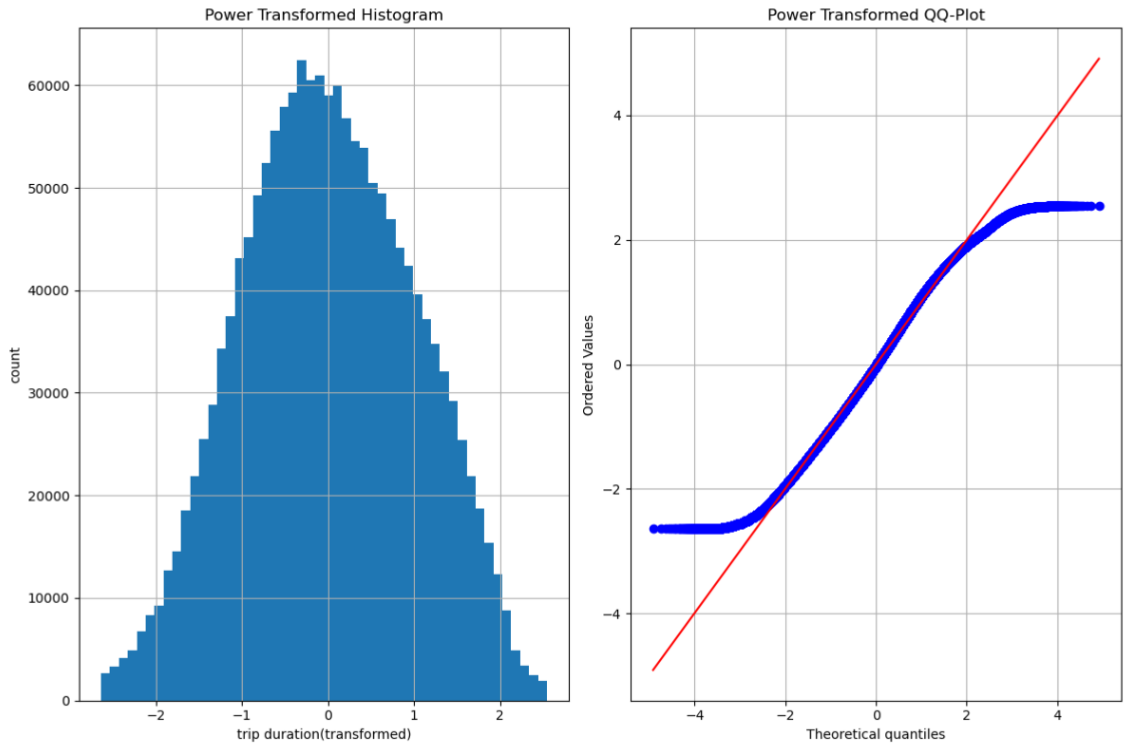
```
=================================================
K-S test: statistics= 0.9630 p-value = 0.0000
K-S test: x dataset looks not Normal
=================================================
```

The K-S test shows the trip duration is not normally distributed with a 99% accuracy.

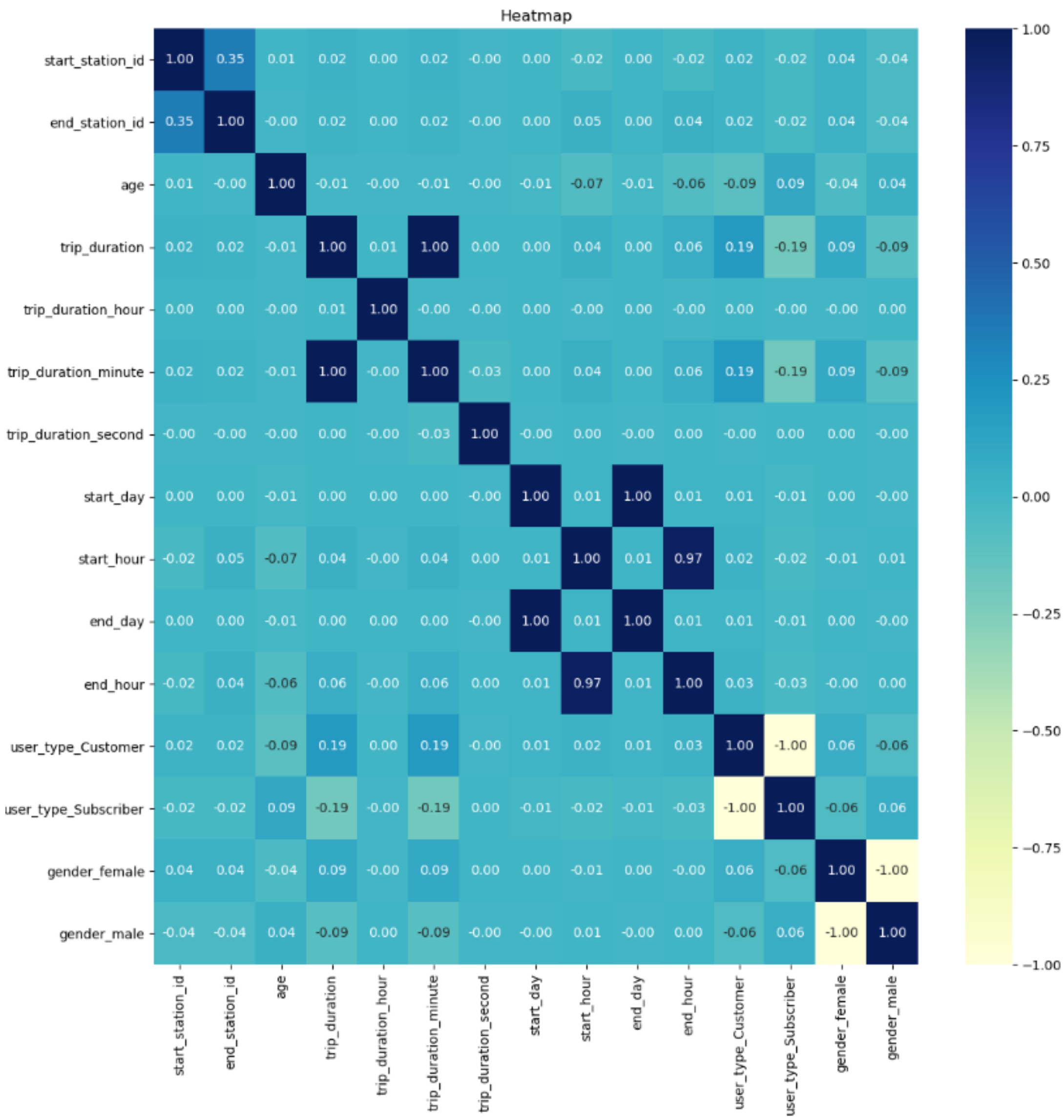
Data Transformation

Power Transformer

use the power transformation to transform the data to normal



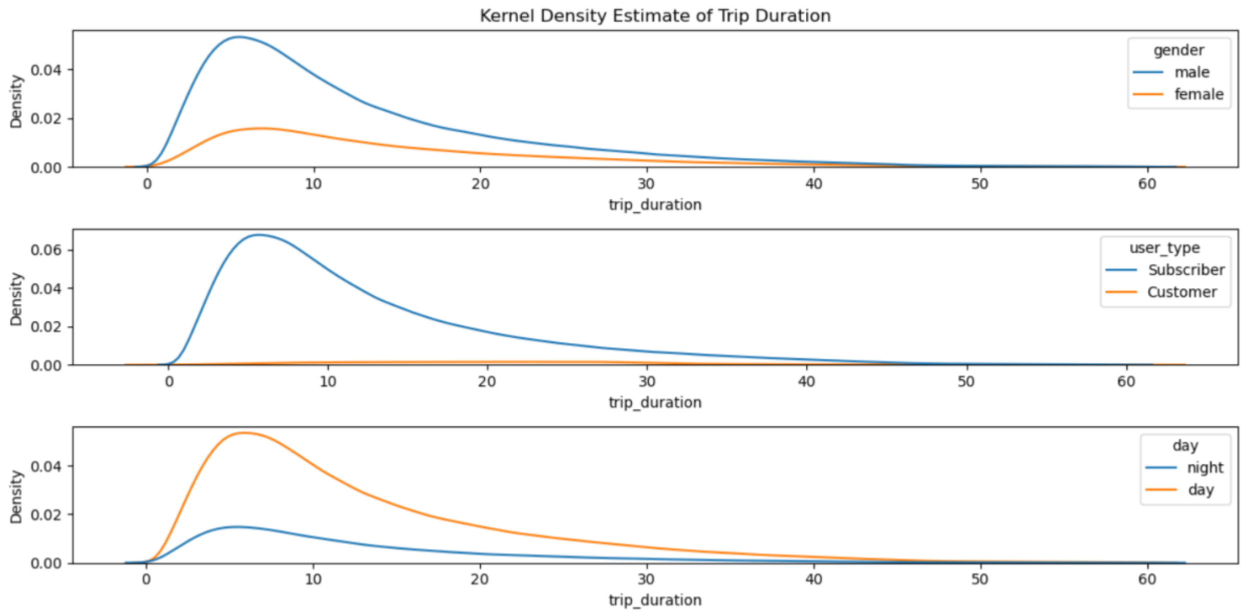
Heatmap & Pearson Correlation



Pearson Correlation Coefficient Matrix

	age	trip_duration	trip_duration_hour	trip_duration_minute	trip_duration_second	user_type_Customer	user_type_Subscriber	gender_female	gender_male
age	1.00000	-0.00504	-0.00036	-0.00503	-0.00026	-0.09380	0.09380	-0.04449	0.04449
trip_duration	-0.00504	1.00000	0.00674	0.99951	0.00249	0.19060	-0.19060	0.08727	-0.08727
trip_duration_hour	-0.00036	0.00674	1.00000	-0.00181	-0.00231	0.00195	-0.00195	-0.00081	0.00081
trip_duration_minute	-0.00503	0.99951	-0.00181	1.00000	-0.02760	0.19055	-0.19055	0.08723	-0.08723
trip_duration_second	-0.00026	0.00249	-0.00231	-0.02760	1.00000	-0.00112	0.00112	0.00071	-0.00071
user_type_Customer	-0.09380	0.19060	0.00195	0.19055	-0.00112	1.00000	-1.00000	0.05606	-0.05606
user_type_Subscriber	0.09380	-0.19060	-0.00195	-0.19055	0.00112	-1.00000	1.00000	-0.05606	0.05606
gender_female	-0.04449	0.08727	-0.00081	0.08723	0.00071	0.05606	-0.05606	1.00000	-1.00000
gender_male	0.04449	-0.08727	0.00081	-0.08723	-0.00071	-0.05606	0.05606	-1.00000	1.00000

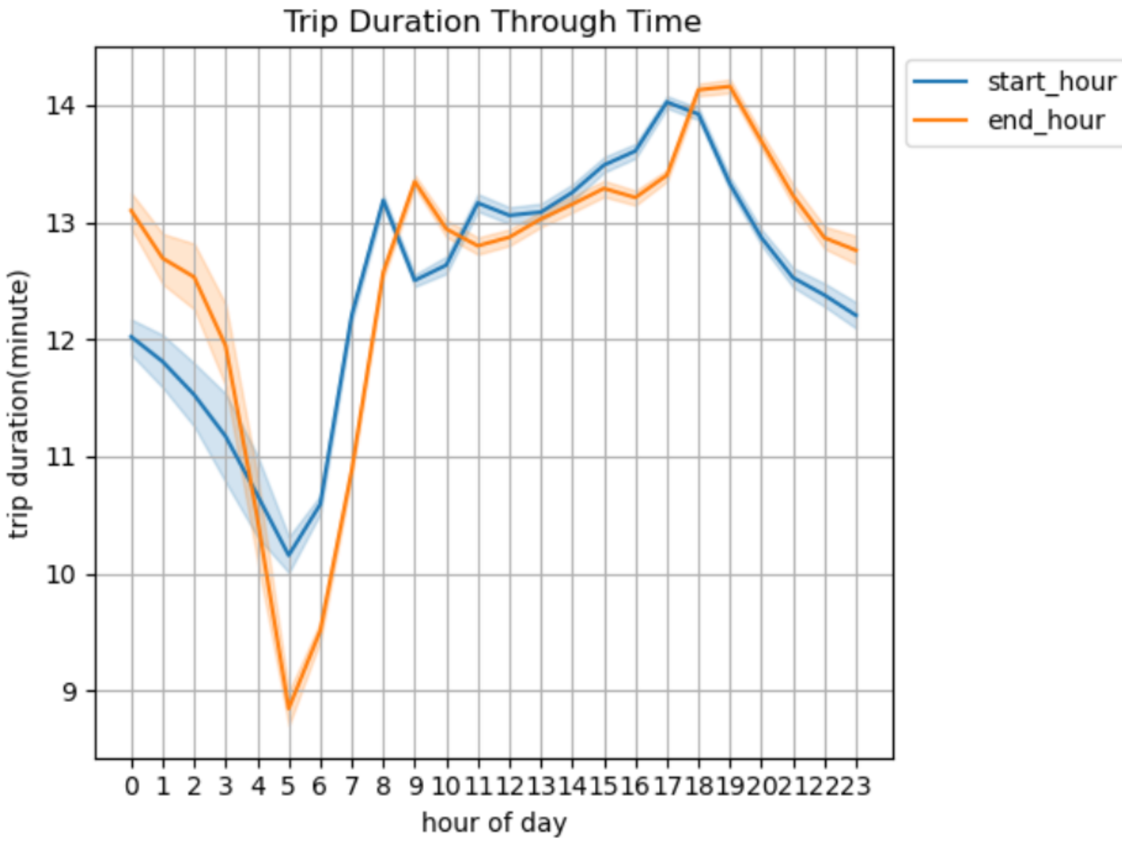
Statistics



The kernel density shows that there are more male users than female users, and among those users, pretty many of then subscribed Citibike. Moreover, people tend to use Citibike more often during the day time.

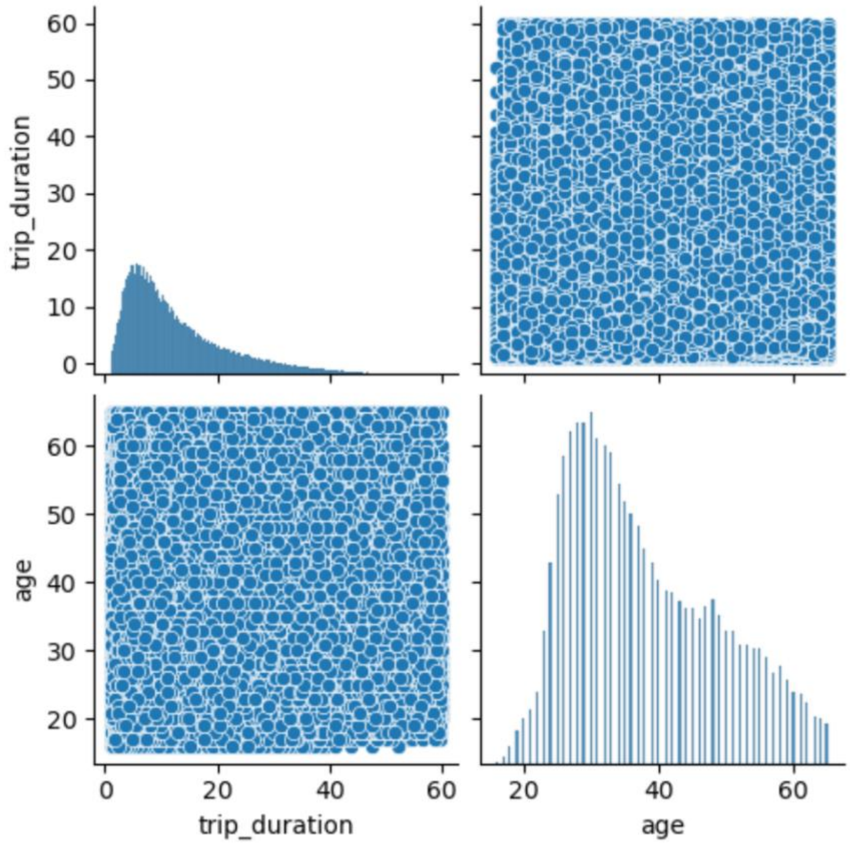
Visualization

line plot



The line plot shows the estimated time of bike usage given trip start time or the end time. The shade shows a 95% confidence interval of estimation.

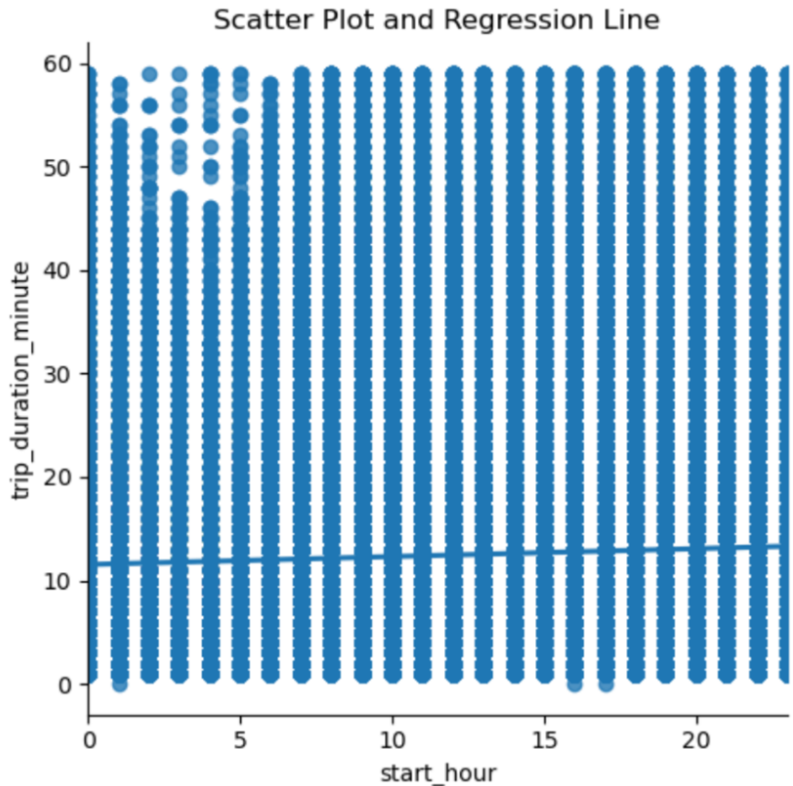
pair plot



We use two numeric variables in the dataset to make a pairplot.

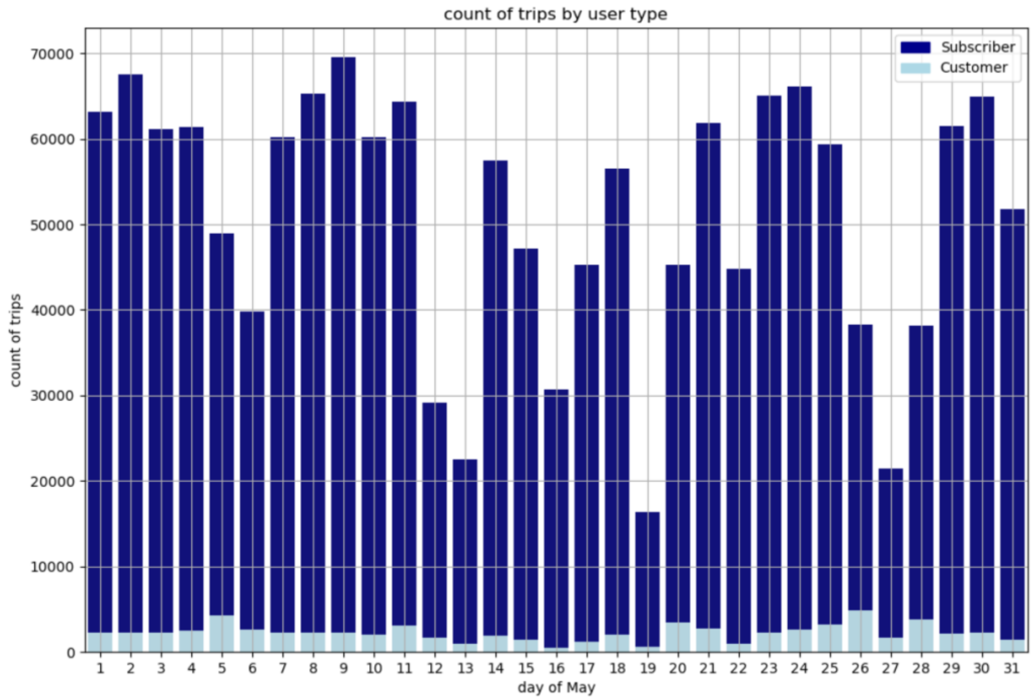
The scatter plots in this pair plot do not give much information since there are too many observations, and the relationship between age and trip duration is not clear(they are not linear related).

Scatter plot and regression line



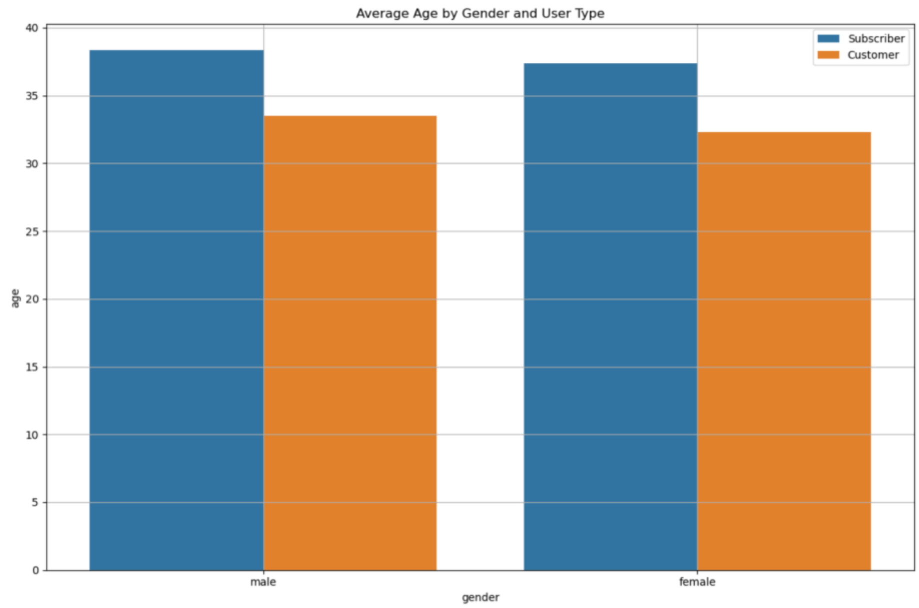
The start hour and trip duration do not seem to fit a linear regression model here.

bar plot(stack)



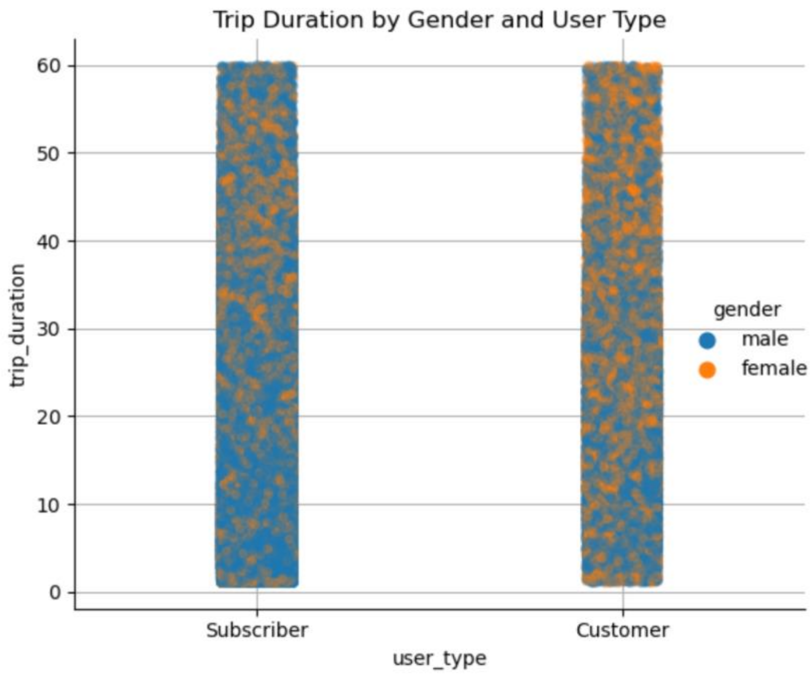
This stacked bar plot shows the number of trips everyday and the proportion of subscriber and non-subscriber users. The result shows there are much more subscriber users, and during 12th, 13th and 19th of May, there are less users.

barplot(group)



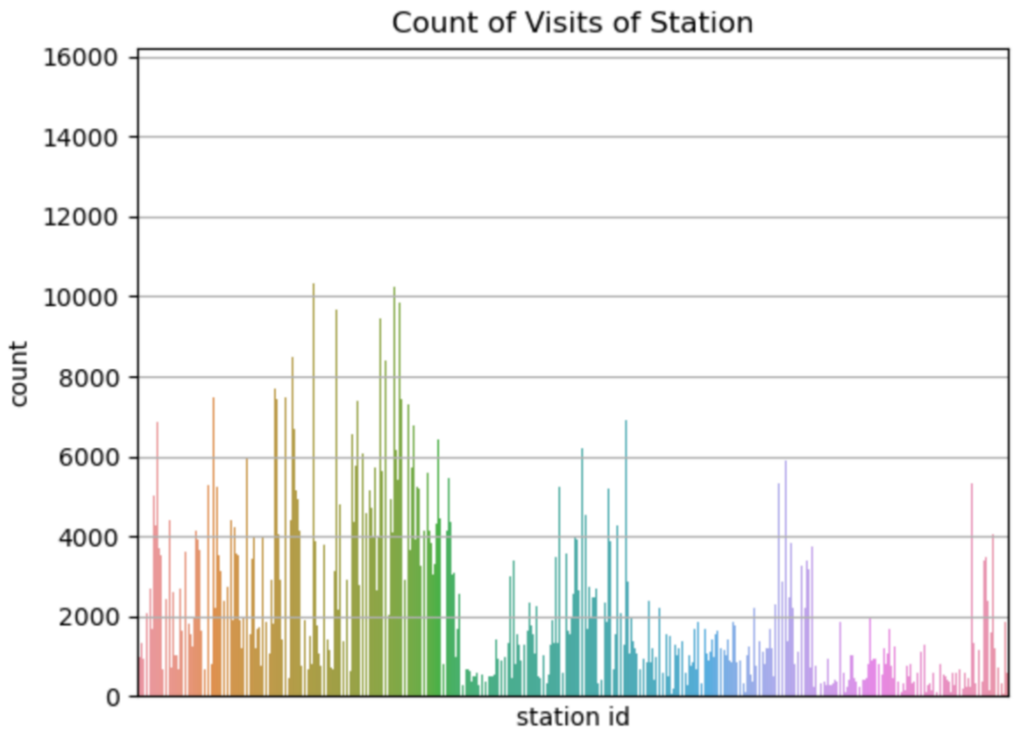
The user-type grouped bar plot shows again that there are more subscribers. Additionally, there are slightly more male users than female users.

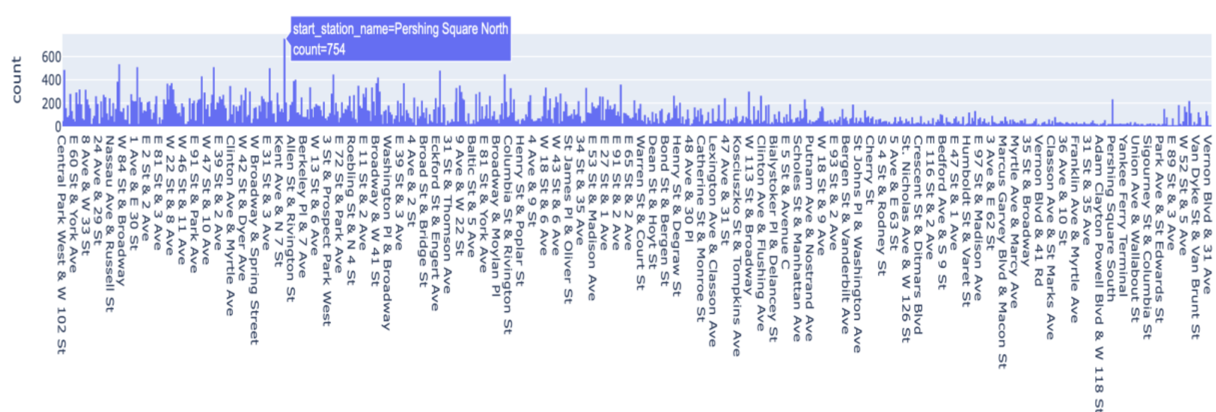
catplot



Since the sample size is very large, the catplot does not give a lot of information here.

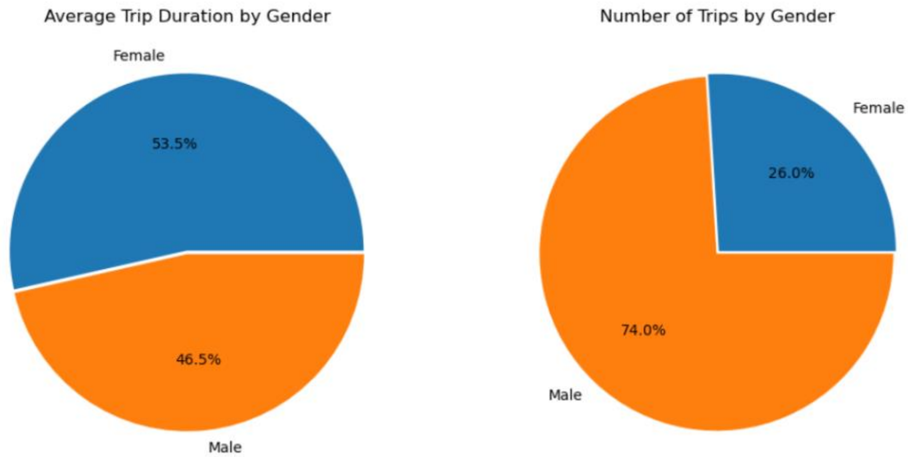
count plot





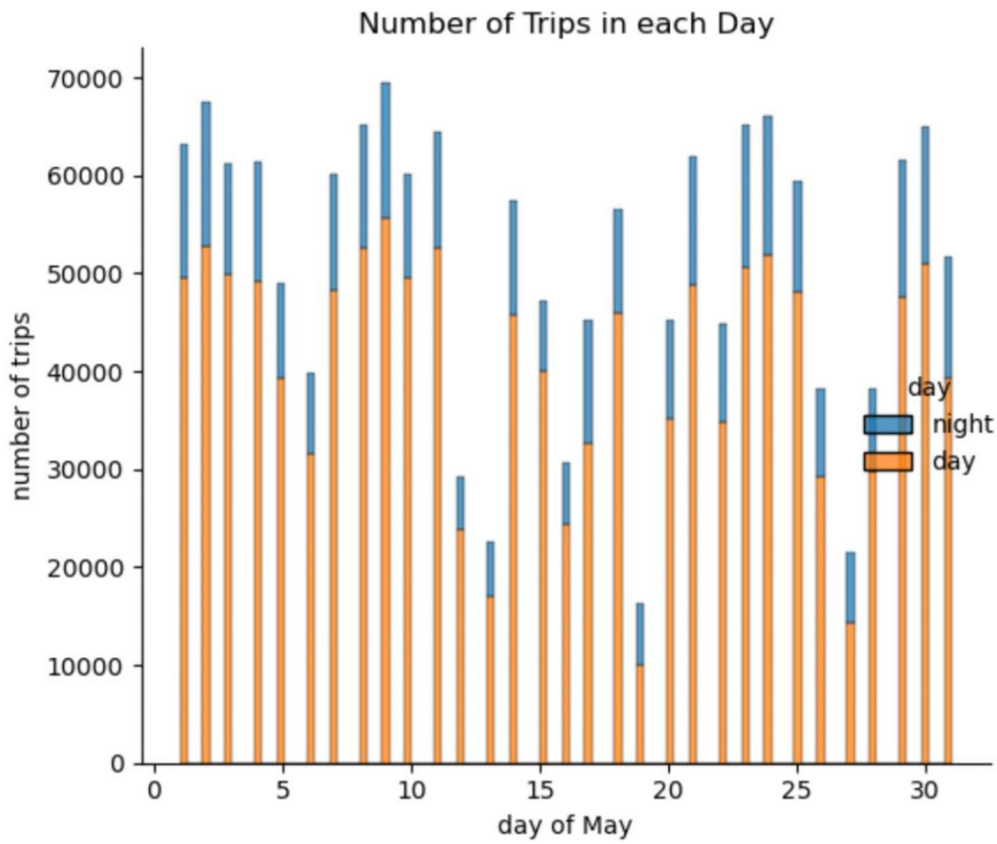
The count plot counts number of usages at each station, and looks like station “Pershing Square North” is most often visited.

pie chart



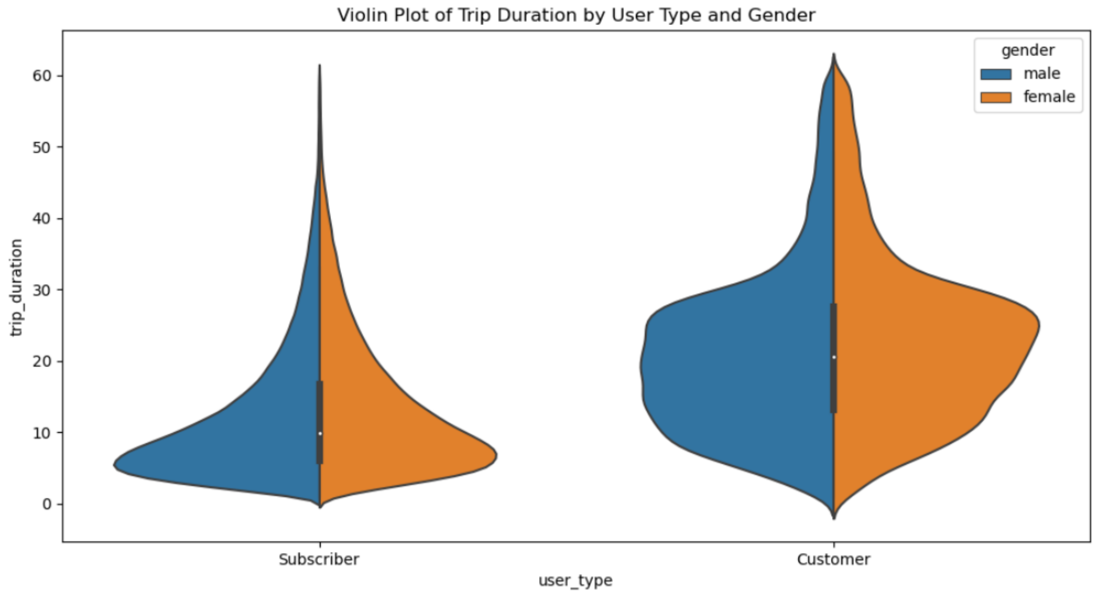
These two pie chart shows that although males ride Citibike much more often, they generally do not ride as long as females do.

displot



The dtribution plot shows how number of trips are distributed over month, and proportion of trips during the day the trips during the night(6pm – 12am). Most trips are during the day.

violin plot



The violin plot gives the kde as well. For subscribers, they mostly take a short ride, around 6 minutes, for customer, they tend to ride longer, around 10-20 minutes.

Recommendations

As a conclusion, and to answer the previous question in abstraction section, we know most bike trips happens during the day time. Men ride more often than women, but women ride longer than men. The most often visited station is “Pershing Square North”, most users are subscriber, and they usually take short ride, but they also ride more frequently. The developer of Citibike can consider to put more bikes at the most frequently visited station, such as “Pershing Square North”.

In the dash app, users can choose the variables and what kind of plot they want. So they can manipulate with the database themselves, and there are guide on the app to tell what the buttons or dropdown menus do.

Reference

-Dash Footer Layout

https://community.plotly.com/t/holy-grail-layout-with-dash-bootstrap-components/40818/2

-Bar Plot Example

https://python-graph-gallery.com/stacked-and-percent-stacked-barplot

- pmlm_utilities_shallow.ipynb by Yuxiao Huang,

https://github.com/yuxiaohuang/teaching/tree/master/gwu/machine_learning_I/spring_2022/

code/utilities/p2_shallow_learning

- case_study.ipynb by Yuxiao Huang,

https://github.com/yuxiaohuang/teaching/tree/master/gwu/machine_learning_I/spring_2022/

code/p2_shallow_learning/p2_c2_supervised_learning/p2_c2_s4_shallow_neural_networks/ca

se_study