Feedback Details

Specification Review

Reviewer Note

Hi Udacity Learner, Anas Almudayd

Congratulations on completing the project! 🎉

You have done outstanding work on this project. It was easy for me to navigate through your work as everything was well explained.

ADDITIONAL LINKS 🔗 TO READ IN FREE TIME

7 Fundamental Steps to Complete a Data Analytics Project(opens in a new tab)

A Comprehensive Guide to Data Exploration(opens in a new tab)

Sure you have learned a lot and we encourage you to keep up with this hard work. Have a nice day and good luck forward. 👏

What issues did you face in the project?

How long did it take to complete this project?

Any suggestions or ideas you may have on the project?

Don’t forget to rate my work as a project reviewer! Your detailed feedback is helpful and appreciated - thank you!.

I'll look forward to reading from you. Thanks a lot! :

udacious:

Code Quality

Does the project code follow clean coding practices?

Reviewer Note

All the code runs without any errors. ✅

Comments and docstrings are used to document the code functionality. ✅

Variable names are concise and meaningful. ✅

Functions are used to avoid code repetition. ✅

```
# And now the first plot in bivariate plot is the scatter plot, I will create the relationship between
# member_birth_year and duration_sec, we will see the propotion of who used the bike most based on thier birth year
plt.scatter(df['member_birth_year'], df['duration_sec'], color='red', alpha=0.5)
# I used alpha as 0.5 to make the propotion well focues on the places that contains many users
# the usual below
plt.title('The propotion of most users used the bike based on thier birth year')
plt.xlabel('The generation')
plt.ylabel('Bike using duration (sec)');

# Now here, in y-axis the numbers looks not well for example: 20000, and 40000
# and while they all more than 20K, it better to make them ends with K like this 20K insted of 20000
# I did some search for the best way to apply it, and it turns out that their is a library needs to be import
# Which is ticker from matplotlib, and we need to make function that convert the regular numbers and make it ends with K
# As I did below
import matplotlib.ticker as mtick # Import this library
def thousands_formatter(x, pos): # Here the function that need to make to convert as I searched
    if x == 0:
        return '0' # Here keep the 0 as is without adding K
    else:
        return f'{int(x / 1000)}K' # And here if the number from the thousands will be converted
# And now its time to use the library that imported to apply the function above
plt.gca().yaxis.set_major_formatter(mtick.FuncFormatter(thousands_formatter))
```

Suggestion (Good Practices)

One should always define functions wherever possible to help avoid code repetition.

This is how you can make functions for repeated code:

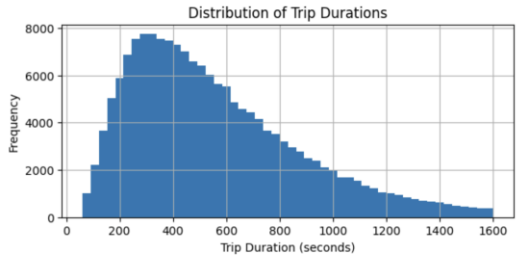
Spot places where you repeat code.

If there are no differences between the two codes, create a function with a descriptive name for that piece of code

Example

You can define a simple function to add axis labels and titles to all the plots.

```
# Now lets plot the histogram
plt.figure(figsize=(7, 3)) # Set the size of the plot
plt.hist(filtered_df['duration_sec'], bins=50)  # Adjust bins as needed
plt.xlabel('Trip Duration (seconds)') # Labeling the x axies
plt.ylabel('Frequency') # Labeling the y axies
plt.title('Distribution of Trip Durations') # Labeling the plot
plt.grid() # The grid gave me a clear view on the exact number or let say the distribution, as I said the grid (to me)
plt.show() # and its time to shoooow
```



Such code repetitions can be easily avoided by defining a simple function to add axis label and title to all the plots.

Look at the below function for reference.

```
#set the default color and method to avoid duplicated code
def x_y_t(xL,yL,title):
    plt.title(title)
    plt.xlabel(xL)
    plt.ylabel(yL)
color = sns.color_palette()[2]
```

ADDITIONAL LINKS

If you want to learn more about python programming you can **follow this link(opens in a new tab)**

15 Python tips and tricks to master Data Science and Machine Learning(opens in a new tab)

The ultimate guide to writing better Python code(opens in a new tab)

Ten Good Coding Practices for Data Scientists(opens in a new tab)

Six steps to more professional data science code(opens in a new tab)

Good coding practices - Describing your code(opens in a new tab)

Code runs without error. Note: a few warnings are okay.

Code is documented with enough relevant comments that a person scanning the code can follow what the code is doing.

Indentation uses 4 spaces (not tabs) and is consistent throughout the project.

Variable names are concise and meaningful.

Functions and loops are used as needed to reduce repetitive code.

Exploratory Data Analysis

Is the data explored systematically using a series of appropriate and varied visualizations?

Reviewer Note

Good job including the different types of plots required for the project.

A Histogram plot is Included in the Univariate section. ✅

A Scatterplot is Included in the Bivariate section. ✅

A Boxplot is Included in the Bivariate section. ✅

A FacetGridd plot is included in the Multivariate section. ✅

Also, at least 7plots were included in the exploration part

Your aesthetic and labeling choices have made the plots readily interpretable.

Additional Links 🔗

[The Python Graph Gallery(opens in a new tab)]

[How to use Python Seaborn for Exploratory Data Analysis(opens in a new tab)]

[Univariate, Bivariate, and Multivariate(opens in a new tab)]

[Fundamentals of Data Visualization(opens in a new tab)]

[How to avoid overplotting with python(opens in a new tab)]

Student has provided seven (7) exploratory data visualizations distributed over univariate, bivariate, and multivariate plots to explore many relationships in the data set.

For each exploration category, complete the required chart(s) and choose one additional visualization type.

Univariate Exploration

Histogram - required

Bar Chart

Count Plot

Bivariate Exploration

Scatterplots - required

Box Plots - required

Clustered Bar Chart

Heatmap

Multivariate Exploration

Facet Plot - required

Plot Matrix

Scatterplot with multiple encodings

Plots should include relevant annotations, such as; lines denoting the average value of a distribution or the values in each cell of a cluster plot. Appropriate scales, axis labels, axis limits, and encodings should be used.

Are questions and observations documented in the report?

Reviewer Note

great job placing the question in between your exploration. it helps in going through the flow of the exploration. I was able to follow along, and understand what was going on in your head while you were trying to answer those questions!

You have added observations after the plots which helped.

Good job framing questions at the end of the section and answering them based on findings

As we see it turns out the hiest people used the following path to start and finish their trip! starting from Berry St at 4th St, and submit their bike in San Francisco Ferry Building (Harry Bridges Plaza)

Talk about some of the relationships you observed in this part of the investigation. How did the feature of interest vary with other features in the dataset?

> In this part the duration in seconds helps a lot to make the relationship among the dufferents plot such as the scatter and box plots, I can say that its the main one while the others helps to make the relationships

Did you observe any interesting relationships between the other features?

> Yes, Specifically in the heat map, the relationships between the start & end station gave me an idea about the most path that really works and active!!

Good job using
Question-Visualization-Observations framework

ADDITIONAL LINKS THAT YOU CAN READ IN YOUR FREE TIME. 🔗

[A Data Analyst's Guide to Asking the Right Questions(opens in a new tab)]

[How to frame the right questions to be answered using data(opens in a new tab)]

Questions and observations are placed regularly throughout the report.

For each exploration category (univariate, bivariate, multivariate) state the assumptions and questions the charts should answer.

After each plot or set of related plots, describe what you found.

Explanatory Data Analysis

Does the explanatory data analysis tell a story?

Reviewer Note

Student presents a clear overarching theme that was used to guide their data analysis. ✅

Student provides some context for the dataset analyzed. ✅

Student presents their findings and key insights. ✅

Dataset Overview and Executive Summary

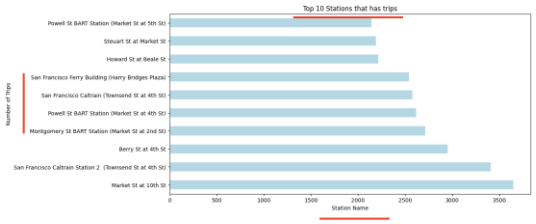
The dataset, 2019-fordgobike-tripdata.csv, is downloaded from Udacity and licensed by Ford GoBike. This dataset includes 183,412 trips with 16 features such as duration_sec (time of take the bike in seconds), start_time and end, and type of user and its gender, there are also id for the bike, stations, and finally wethear the bike shared durnig the trip or not .

summary of data wrangling process that performed:

- The null velue have been removed
- The start_time & end_time have been converted into datetime, they were 'object' as we see above
- The start_time & end_time have been splited into date_start, time_start, date_end, and time_end to make the structre better
- The start_time & end_time columns have been revomed after perform the split step
- The columns start_station_longitude and end_station_longitude values converted into + values
- The outliers have been dealt with

Visualizations from their exploratory analysis are used to support findings. ✅

Visualizations should be polished with clear axis, labels, and annotations. ✅



All the plots in the explanatory part are well polished that is they have axis label, title and legends wherever required

---

ADDITIONAL LINKS THAT YOU CAN READ IN YOUR FREE TIME.

[Explanatory & Exploratory Data Analytics: What You Need to Know(opens in a new tab)]

[EXPLORATORY VS EXPLANATORY DATA ANALYSIS(opens in a new tab)]

The explanatory analysis follows a logical flow and is organized in the following manner:

Student presents a clear overarching question or theme that was used to guide their data analysis.

Student provides some context for the dataset analyzed.

Student presents their findings and key insights.

Visualizations from their exploratory analysis are used to support findings.

Visualizations should be polished with clear axis, labels, and annotations.