Data Visualization Project

Graph recommendation system

Name ==> Daniyal Muneer

Roll No. ==> 211980031

Data Scientist

Report

Problem statement:

Requirement is to make a system in which user comes to system and upload its .csv file and system shows all columns, system asks user to pick No. of columns on which he wants to perform analysis. System also asks user for analysis type (univariate, bivariate, multivariate) and at last system shows all possible plots.
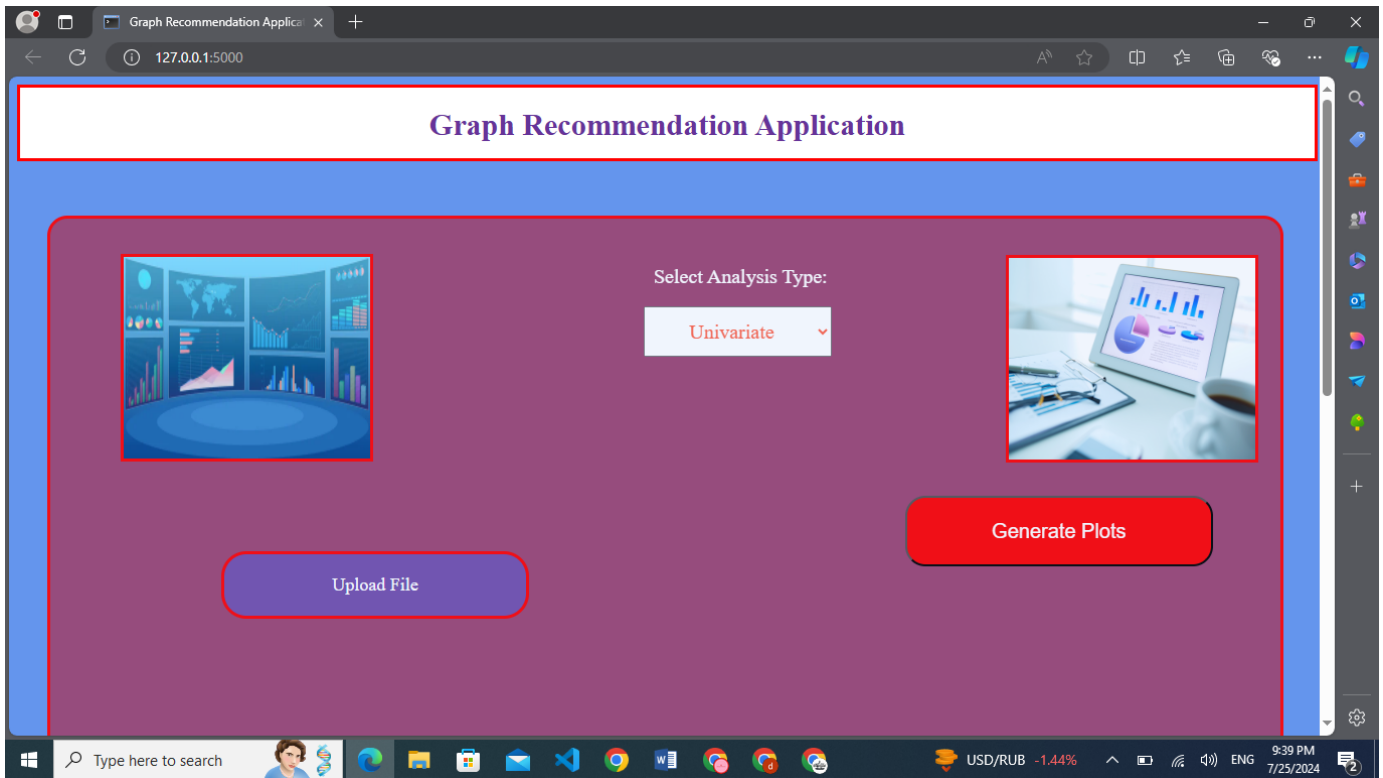
Objective:

- Allow user to upload .csv file
- To select Analysis type
- To pick No. of columns
- To provide relevant recommendation graphs

Architecture:

This is user interface given below.

- I used a button for uploading file.
- I used a dropdown arrow for selecting type of analysis.
- I used a button for generating plots.

Component description:

I used following languages for it.

- I used HTML, CSS, java Script for front-end building.
- I used Python, Flask for back-end building.
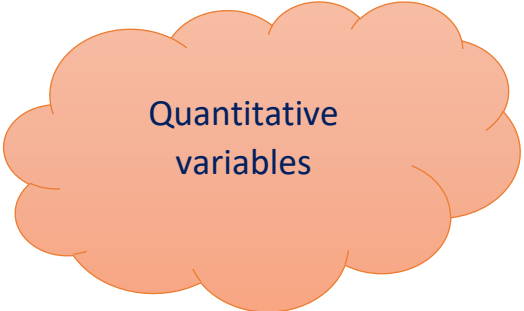
Workflow:

I used switch-cases in this project.

I. Quantitative vs. Quantitative
II. Categorical vs. Categorical
III. Quantitative vs. Categorical
IV. Categorical vs. Quantitative

Data Visualization functions:

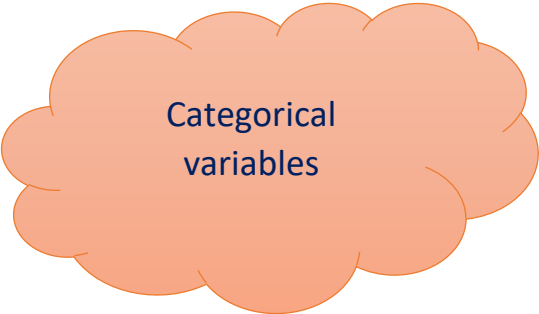I used a lot of graphs in this project.

Univariate Analysis:

1. Histogram
2. Empirical cumulative distribution
3. Area chart
4. Lollipop chart
5. Dot chart

Quantitative variables

6. Count plot
7. Bar chart
8. Bubble chart
9. Pie chart
10. Donut chart
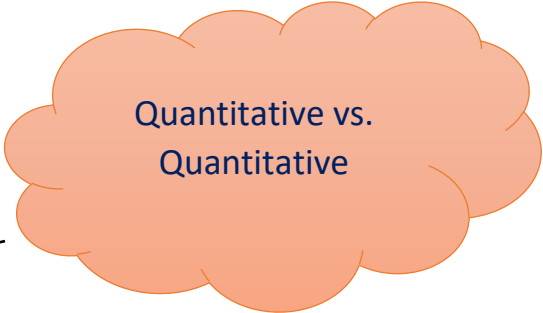
Categorical variables

Bivariate Analysis:
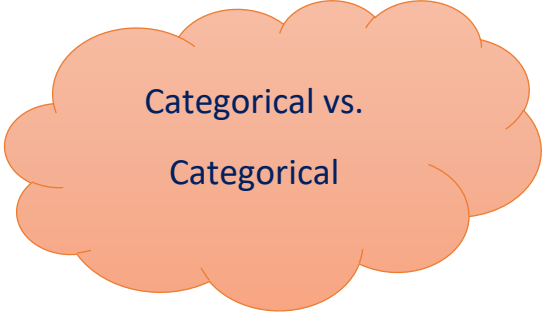
11. Scatter plot
12. Line plot

Quantitative vs. Quantitative

13. Stacked bar chart
14. Grouped bar chart

15. Segmented bar chart

Categorical vs. Categorical

----

16. Density plot
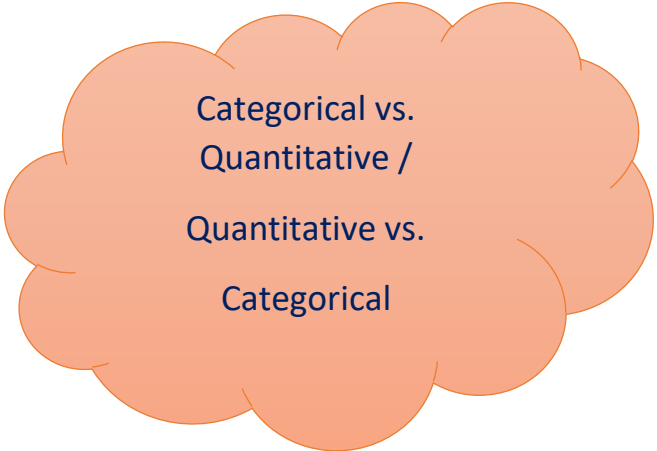
17. Strip plot

18. Box plot
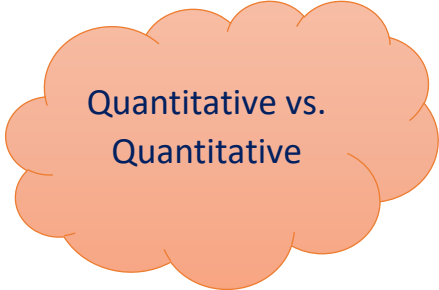
19. Boxen plot

20. Violin plot

21. Swarm plot

Categorical vs. Quantitative /

Quantitative vs.

Categorical

Multivariate Analysis:

22. Heat Map

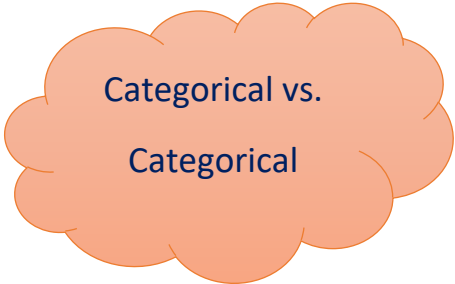
Quantitative vs. Quantitative

23. Tree Map
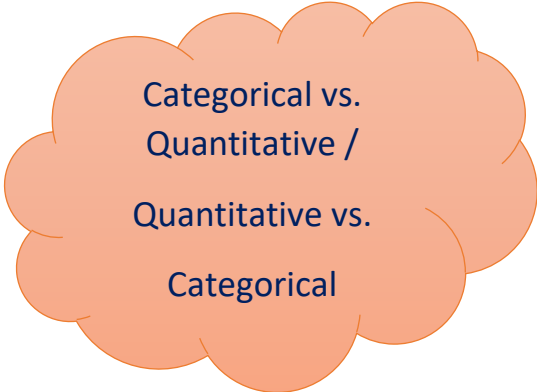
24. Mosaic plot

Categorical vs.

Categorical

25. Multiple plot

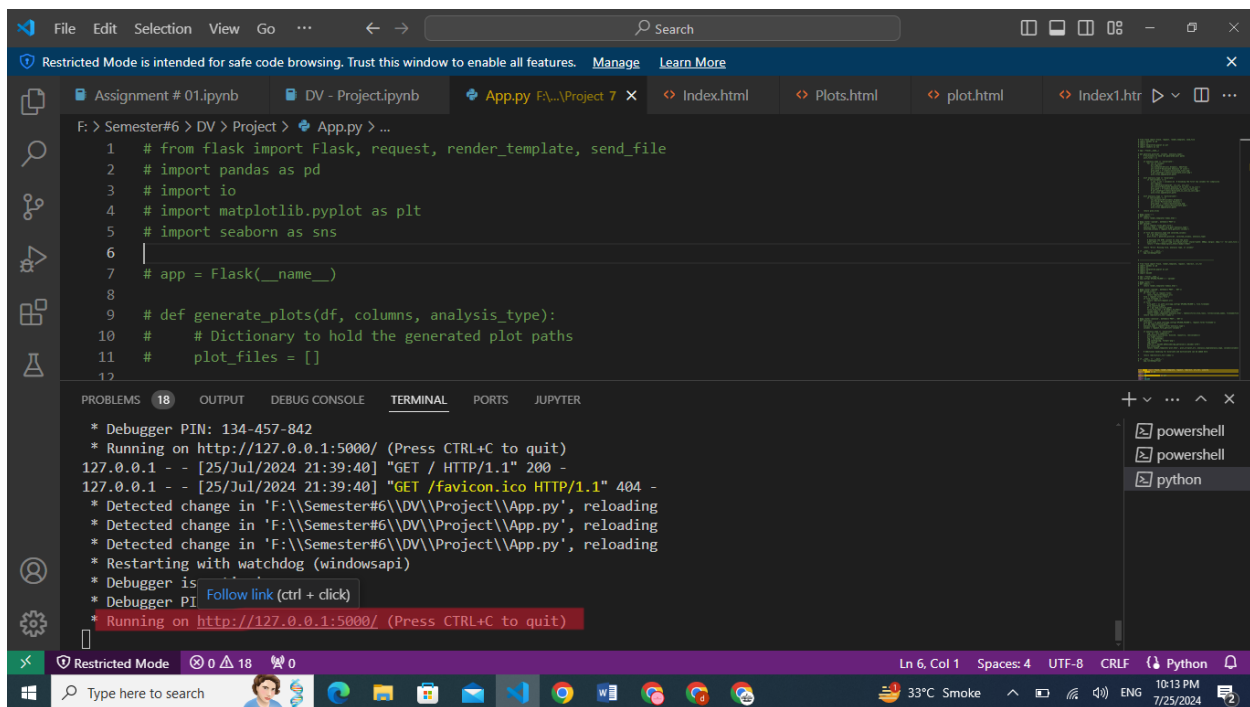
Categorical vs. Quantitative /

Quantitative vs.

Categorical

How to run Code?

I.   First of all, user will come to flask file.
II. He runs the code at going terminal, there will be appeared an IP and click the link.
III. A front-end Website will be opened.
IV. User will upload file, select analysis type and pick No. of columns for visualization.

Flask structure:

Uploads: This folder has all .csv files.

Templates: This folder has all front-end files i.e .html

Static: This folder has all styling files for front-end i.e .css and all images that are present in front-end website.

```php
data-analysis-visualization/
|
├── app/
|   ├── App.py
|   ├── static/
|   |   ├── images/              # Directory for images (.png, .jpg)
|   |   └── css/
|   |       └── style.css
|   └── templates/
|       ├── index.html
|       └── plot.html
|
├── uploads/                     # Directory to store .csv files
```

Conclusion:

This project is very useful for such a user who have no understanding about data analysis and data visualization. Its interface is very simple and user-friendly. It supports all various types of analysis, making it valuable tool for data analyst. It is very efficient for handling large datasets.

*****************THANKS***************