ASSIGNMENT

Part 1 : Exploring Data Visualization in Python with Matplotlib

Objective: In this assignment, you will learn how to use Matplotlib, a popular data visualization library in Python. You will practice creating different types of visualizations such as line plots, scatter plots, histograms, bar charts, and heatmaps using Matplotlib. You will use the "Iris" dataset from the UCI Machine Learning Repository.

Tasks:

1. Import the necessary libraries: Start by importing the required libraries, including Matplotlib, NumPy, and pandas.

2. Load the dataset: Load the Iris dataset(load_iris) from sklearn.datasets convert it into a pandas DataFrame. The dataset contains 150 rows and 5 columns (4 features and 1 target variable).

3. Line Plot: Create a line plot to visualize the trend in the data over time. Since this is not a time-series dataset, create a line plot for each feature against the target variable (species). Customize the plot by adding a title, labels for the x and y-axis, and a legend.

4. Scatter Plot: Create a scatter plot to visualize the relationship between two variables. Create scatter plots for each pair of features, and color the points by species. Customize the plot by adding a title, labels for the x and y-axis, and a legend.

5. Histogram: Create a histogram to visualize the distribution of a single variable. Create histograms for each feature, and color the bars by species. Customize the plot by adding a title, labels for the x and y-axis, and setting the number of bins.

6. Bar Chart: Create a bar chart to visualize the frequency or count of a categorical variable. Create a bar chart for the target variable (species), showing the count of each species. Customize the plot by adding a title, labels for the x and y-axis, and changing the color scheme.

7. Heatmap: Create a heatmap to visualize the correlation between multiple variables. Create a heatmap showing the correlation matrix between the features. Customize the plot by adding a title and changing the color scheme.

8. Bonus: Choose a different dataset and create a unique visualization using Matplotlib. Be creative!

Submission:

Submit a Jupyter Notebook file (.ipynb) containing your code, visualizations, and a brief explanation of your thought process for each task. Make sure to include any necessary comments in your code for clarity.

Part 2 : Handling Null Values in Python with Pandas

Objective: In this assignment, you will learn how to handle null values in a dataset using Python with the Pandas library. You will practice different techniques such as identifying null values, dropping null values, filling null values, and using interpolation methods. You will use the "California Housing Prices" dataset from Scikit-learn.

Tasks:

1. Import the necessary libraries: Start by importing the required libraries, including Pandas, NumPy, and Scikit-learn.

2. Load the dataset: Load the California Housing Prices(fetch_california_housing) dataset from sklearn.datasets convert that into a pandas DataFrame using Scikit-learn. The dataset contains 20,640 rows and 8 columns.

3. Identify Null Values: Check the dataset for any missing values and count the number of missing values in each column.

4. Dropping Null Values: Remove any rows that have missing values.

5. Filling Null Values: Replace the missing values in the "total_bedrooms" column with the median value of the column.

6. Interpolation: Fill the missing values in the "total_bedrooms" column using interpolation. Visualize the interpolated values using a line plot.

7. Bonus: Choose a different dataset and handle null values using different techniques. Be creative!

Submission:

Submit a Jupyter Notebook file (.ipynb) containing your code, a brief explanation of your thought process for each task, and any necessary comments for clarity. Make sure to test your code and provide examples of the output for each task.