Assignment: 5

Problem Statement:

Visualize the data using R/Python by plotting the graphs for assignment no. 1 and 2. Consider a suitable data set.

a) Use Scatter plot, bar plot, Box plot and Histogram

OR

b) Perform the data visualization operations using Tableau for the given dataset.

Software Library Package:

Python with pandas , matplotlib and seaborn.

1. Theory:

Seaborn is a Python data visualization library that simplifies the creation of attractive statistical graphics. Built on Matplotlib, it offers a high-level interface for generating various plots like scatter plots, bar plots, and heatmaps. Seaborn's default themes enhance aesthetics, and its support for visualizing categorical and numerical data makes it ideal for exploratory data analysis. With intuitive customization options, Seaborn integrates seamlessly with Pandas DataFrames, making it popular among researchers, data scientists, and analysts for creating visually appealing and insightful plots.

1.1 Methodology:

The program utilizes Python's libraries, primarily matplotlib and seaborn, for data visualization. It employs specific functions from these libraries to create different types of plots.

1.2 Advantages and Applications:

-Functionality: Utilizing functions from matplotlib and seaborn libraries provides a wide range of capabilities for creating visually appealing and informative plots.

- Efficiency: Using pre-built functions simplifies the process of creating plots, reducing the need for manual coding and enhancing efficiency.

- Customization: These libraries offer extensive customization options through function parameters, allowing users to tailor plots according to their requirements.

1.3 Limitations:

- Learning Curve: Understanding the various functions and their parameters within matplotlib and seaborn may require time and effort, especially for beginners.

- Dependency: The program relies on external libraries, which could lead to compatibility issues or version dependencies.

- Performance: Depending on the complexity of the plots and dataset size, the performance of the functions may vary, potentially impacting execution time.

2. Working/Algorithm:

The program implements several functions from matplotlib and seaborn libraries to create different types of plots:

1. `plt.scatter`: Generates a scatter plot to visualize the relationship between speeding and total crashes.

2. `plt.bar`: Creates a bar plot to compare speeding and not distracted attributes.

3. `plt.hist`: Produces a histogram to display the frequency distribution of the speeding attribute.

4. `sns.regplot`: Constructs a regression plot showing the regression line between alcohol and speeding attributes.

5. `plt.plot`: Draws a point plot to plot the total crashes against data points.

These functions take input data and parameters to generate the desired plots, with each function serving a specific visualization purpose.

3. Conclusion:

By leveraging functions from matplotlib and seaborn libraries, the program effectively creates diverse visualizations for the provided dataset. These functions offer flexibility, efficiency, and customization options, making them valuable tools for data visualization tasks. However, users should be mindful of potential limitations such as the learning curve and performance considerations associated with using these libraries. Overall, the program demonstrates the utility of leveraging pre-built functions for efficient and insightful data visualization in Python.