**Shadowfox Data Science Task 1**

**Matplotlib**

Matplotlib is one of the most widely used Python libraries for creating static, interactive, and animated visualizations. It serves as a foundation for other libraries like Seaborn, offering a high degree of control over plot aesthetics. It is particularly useful for generating publication-quality figures in a variety of formats.

* **Typical Use Cases**: Line plots, scatter plots, histograms, and subplots for data analysis and research.
* **Unique Features**: Extensive customization options, ability to export figures to multiple formats (PNG, PDF), and fine-grained control over every element of a plot.

**Seaborn**

Seaborn is a statistical data visualization library built on top of Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn simplifies complex visualization processes and automatically manages many of the styling and plotting intricacies, making it ideal for exploratory data analysis.

* **Typical Use Cases**: Statistical visualization, data distributions, heatmaps, and complex multi-plot grids.
* **Unique Features**: Built-in themes, colour palettes, and integration with Pandas for seamless plotting.

**2. Graph Types**

**Matplotlib Graphs**

1. **Line Plot**

* **Description**: Displays data as a series of points connected by straight lines. Useful for tracking changes over time.

1. **Scatter Plot**

* **Description**: Visualizes the relationship between two numerical variables. Each point represents an observation.

1. **Histogram**

* **Description**: Represents the distribution of a dataset by grouping data into bins.

**Seaborn Graphs**

1. **Bar Plot**
   * **Description**: Displays categorical data with rectangular bars. Each bar’s height is proportional to the count or value it represents.
2. **Heatmap**

* **Description**: Shows a matrix of data values with color-coding to indicate different ranges of values. Useful for displaying correlations.

1. **Pair Plot**

* **Description**: Creates a matrix of scatter plots for all numerical variables in a dataset, along with histograms for each variable.

**3. Comparison**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Matplotlib** | **Seaborn** |
| **Ease of Use** | High learning curve due to extensive customization options. | User-friendly, especially for creating complex statistical plots quickly. |
| **Customization** | Offers detailed control over every plot element, but can require more code. | Less customizable but has sensible defaults for style and layout. |
| **Interactivity** | Basic interactivity, better suited for static plots. | Limited interactivity; mainly for static plots. |
| **Performance** | Good performance for medium datasets but may slow down with very large datasets. | Built on Matplotlib, so performance is similar. Handles large datasets well for statistical plots. |
| **Strengths** | Highly customizable, supports a wide range of plot types. | Easy to create visually appealing plots with minimal effort, strong integration with Pandas. |
| **Weaknesses** | Requires more code for common tasks, and the syntax can be verbose. | Less flexible for detailed customization compared to Matplotlib. |



Embedded the assignment in the above pdf.