Library Overview

Matplotlib

Matplotlib is a foundational Python library for creating static, animated, and interactive plots. It is widely used due to its extensive customization options and broad functionality, making it a favourite for generating publication-quality graphics.

- **Key Features**: Flexible customization, large library support, suited for static plots.
- **Common Use Cases**: Scientific visualization, academic and professional graphics, in-depth data analysis.

Seaborn

Seaborn is built on top of Matplotlib, designed specifically for statistical data visualization. It provides an easy interface for creating visually appealing plots and includes additional features that streamline statistical visualization.

- **Key Features**: Attractive aesthetics, supports statistical plots, ideal for visualizing data distributions.
- **Common Use Cases**: Exploratory data analysis, statistical graphics, visual summaries of data distributions.

Graph Types and Examples:

- Matplotlib

1. Line Plot

Description: Displays data trends over a continuous interval.
Use Case: Tracking changes in stock prices, temperature fluctuations over time.

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
```

```
y = [2, 3, 5, 7, 11]

plt.plot(x, y, marker='o')

plt.title("Line Plot")

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

plt.show()
```

2. Scatter Plot

Description: Plots individual data points, useful for visualizing relationships between two variables.
Use Case: Comparing height vs. weight, product sales vs. advertising spend.

```
plt.scatter([1, 2, 3, 4], [10, 20, 25, 30], color='r')

plt.title("Scatter Plot")

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

plt.show()
```

3. Bar Chart

Description: Visualizes data as rectangular bars, showing comparisons across categories.
Use Case: Comparing sales figures for different products or regions.

```
categories = ["Product A", "Product B", "Product C"]

values = [10, 20, 15]

plt.bar(categories, values, color='skyblue')

plt.title("Bar Chart")

plt.xlabel("Categories")

plt.ylabel("Values")

plt.show()
```

4. Histogram

Description: Displays frequency distribution of a dataset, often used to visualize data spread.
Use Case: Distribution of exam scores, age group analysis.

```
data = [1, 2, 2, 3, 3, 3, 4, 4, 5]

plt.hist(data, bins=5, color='purple', edgecolor='black')
plt.title("Histogram")
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.show()
```

- Seaborn

1. Distribution Plot

Description: Represents the distribution of a single variable, useful for analyzing data spread.
Use Case: Income distribution, frequency of events.

```
import seaborn as sns
import numpy as np
data = np.random.normal(size=100)
sns.histplot(data, kde=True)
plt.title("Distribution Plot")
plt.show()
```

2. Pair Plot

Description: Creates a matrix of scatter plots to examine relationships between variables in a dataset.
Use Case: Comparing features in the Iris dataset to observe species characteristics.

```
data = sns.load_dataset("iris")
sns.pairplot(data, hue="species")
plt.show()
```

3. Box Plot

Description: Displays data distribution across categories, highlighting quartiles and outliers.
Use Case: Comparing salary distributions across job roles or regions.

```
sns.boxplot(x="species", y="sepal_length", data=data)
plt.title("Box Plot")
plt.show()
```

4. Heatmap

Description: Visualizes matrix data with colors representing values, effective for showing correlations.
Use Case: Correlation matrices, regional election results.

```
data = np.random.rand(10, 10)

sns.heatmap(data, annot=True, cmap="YlGnBu")

plt.title("Heatmap")

plt.show()
```

Comparison

Feature	Matplotlib	Seaborn
Ease of Use	Medium (setup required)	High (more intuitive interface)
Customization	Extensive (detailed control)	Moderate (with good defaults)
Interactivity	Limited (mostly static)	Limited (better with Plotly)
Performance	Good (large datasets)	Suitable for moderate datasets
Specialization	General-purpose plotting	Statistical visualization

Additional Resources

Resources which I referred to:

Matplotlib: https://matplotlib.org/stable/users/explain/quick_star t.html#quick-start

Seaborn:https://seaborn.pydata.org/tutorial/introduction.html