Visualization Library Documentation

1. Library Overview

Matplotlib

Matplotlib is a widely used Python library for creating static, interactive, and animated visualizations. It provides extensive customization options and supports a variety of plot types. Matplotlib is particularly known for its ability to create detailed, publication-quality plots.

Key Features

- **Comprehensive support for 2D plotting**
- **Highly customizable plots**
- **Supports multiple plot types including line plots, scatter plots, bar charts, and histograms**

Typical Use Cases

- **Exploratory data analysis**
- **Presentation of data insights**
- **Scientific and engineering visualizations**

Seaborn

Seaborn is a statistical data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

Key Features

- **Built-in themes for aesthetic plots**
- **Integration with pandas data structures**
- **Simplified syntax for complex visualizations**

Typical Use Cases

- **Statistical data analysis and visualization**
- **Exploratory data visualization with integrated statistical graphics**

2. Graph Types

Matplotlib

Line Plot

Description: A line plot connects data points with a continuous line.

Use Case: Used to visualize trends over time.

Example Code:

```
import matplotlib.pyplot as plt

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

y = [2, 4, 6, 8, 10]

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

plt.title('Line Plot')

plt.xlabel('X-axis')

plt.ylabel('Y-axis')

plt.show()
```

Bar Chart

Description: A bar chart represents categorical data with rectangular bars.

Use Case: Used to compare different categories.

Example Code:

```
categories = ['A', 'B', 'C', 'D']

values = [10, 15, 7, 12]

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

plt.title('Bar Chart')

plt.xlabel('Categories')

plt.ylabel('Values')

plt.show()
```

Histogram

Description: A histogram displays the distribution of a dataset.

Use Case: Used to visualize frequency distributions.

Example Code:

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

plt.hist(data, bins=4, color='green', edgecolor='black')

plt.title('Histogram')

plt.xlabel('Value')

plt.ylabel('Frequency')

plt.show()
```

Seaborn

Scatter Plot

Description: A scatter plot displays points for pairs of values.

Use Case: Used to explore relationships between variables.

Example Code:

```
import seaborn as sns

import matplotlib.pyplot as plt

data = {'X': [1, 2, 3, 4, 5], 'Y': [2, 4, 6, 8, 10]}

sns.scatterplot(x='X', y='Y', data=data)

plt.title('Scatter Plot')

plt.show()
```

Heatmap

Description: A heatmap visualizes data in matrix form using color gradients.

Use Case: Used for correlation matrices and displaying relationships.

Example Code:

```
import seaborn as sns

import numpy as np
```

```
data = np.random.rand(5, 5)

sns.heatmap(data, annot=True, cmap='coolwarm')

plt.title('Heatmap')

plt.show()
```

Box Plot

Description: A box plot summarizes data distributions with quartiles and outliers.

Use Case: Used to detect outliers and understand distribution shapes.

Example Code:

```
sns.boxplot(data=[7, 8, 8, 10, 12, 15, 15, 18, 22])

plt.title('Box Plot')

plt.show()
```

3. Comparison

Feature	Matplotlib	Seaborn
Ease of Use	Requires more code for complex plots	Simplifies creation of complex statistical plots
Customization	Highly customizable with detailed control	Limited customization compared to Matplotlib
Interactivity	Basic static plots	Integrates well with pandas and statistical libraries
Performance	Handles large datasets with extensive configuration	Suitable for medium datasets with statistical focus

4. Resources

- [Matplotlib Documentation]
- [Seaborn Documentation]