

Visualization Library Documentation

Matplotlib:

- **Introduction:**
Matplotlib is the foundation of Python visualization. It mimics MATLAB-style plotting and provides the most control over figure elements. It is a low-level library, meaning users must explicitly define details like axis labels, colors, and markers, which offers maximum flexibility at the cost of verbosity.
- **Unique Features:**
 - Can create virtually any 2D visualization (and limited 3D plots via `mpl_toolkits`).
 - Integrates with NumPy arrays and Pandas DataFrames.
 - Allows exporting figures in multiple formats (PNG, PDF, SVG).
 - Supports animations and interactive visualizations in Jupyter notebooks.
- **Typical Use Cases:**
 - Academic research papers needing precise control.
 - Data scientists wanting publication-ready plots.
 - Developers needing custom visualizations integrated into applications.

A. Matplotlib Graphs

1. Line Plot

- **Description:**
A line plot connects individual data points with straight lines. It is widely used to show trends over time (time-series data) or changes in a variable.
- **Use Case:**
Tracking stock prices, temperature over days, or sales over months.

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]

plt.plot(x, y, marker='o', color='blue', linestyle='--')

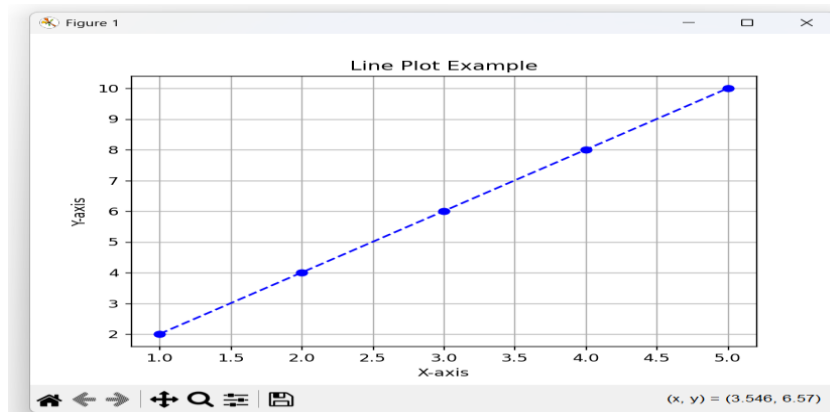
plt.title("Line Plot Example")

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

plt.grid(True)

plt.show()
```



2. Scatter Plot

- Description:**
 A scatter plot displays individual data points in a 2D space. It helps visualize relationships, clusters, or outliers.
- Use Case:**
 Checking correlation between income vs. expenditure, or height vs. weight.

```
import matplotlib.pyplot as plt
```

```
x = [5, 7, 8, 7, 6, 9, 5]
```

```
y = [99, 86, 87, 88, 100, 86, 103]
```

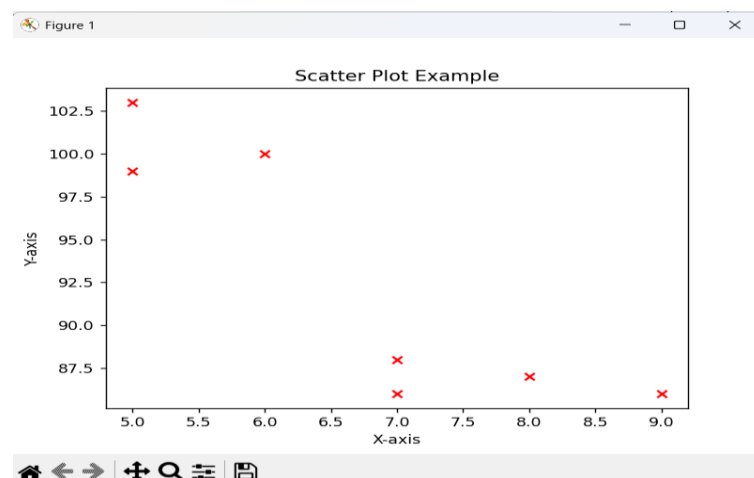
```
plt.scatter(x, y, color='red', marker='x')
```

```
plt.title("Scatter Plot Example")
```

```
plt.xlabel("X-axis")
```

```
plt.ylabel("Y-axis")
```

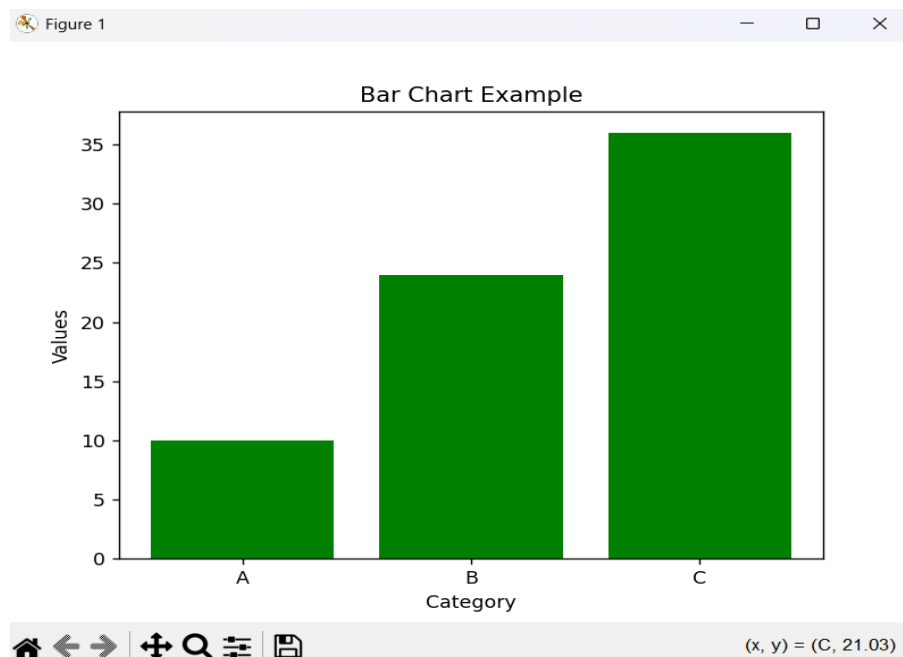
```
plt.show()
```



3. Bar Chart

- **Description:**
A bar chart uses rectangular bars to represent categorical data. Bar lengths represent values.
- **Use Case:**
Comparing sales across product categories, or population across countries.

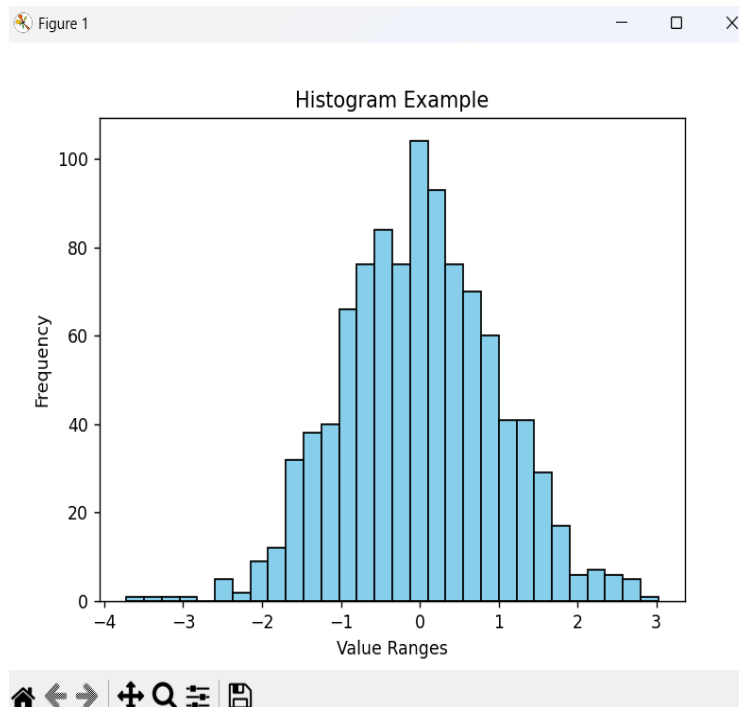
```
import matplotlib.pyplot as plt  
categories = ['A', 'B', 'C']  
values = [10, 24, 36]  
plt.bar(categories, values, color='green')  
plt.title("Bar Chart Example")  
plt.xlabel("Category")  
plt.ylabel("Values")  
plt.show()
```



4. Histogram

- **Description:**
A histogram shows the distribution of data by dividing it into bins. Useful for identifying skewness, normality, and spread of data.
- **Use Case:**
Understanding exam score distributions, income levels, or sensor measurements.

```
import matplotlib.pyplot as plt  
import numpy as np  
data = np.random.randn(1000)  
plt.hist(data, bins=30, color='skyblue', edgecolor='black')  
plt.title("Histogram Example")  
plt.xlabel("Value Ranges")  
plt.ylabel("Frequency")  
plt.show()
```



5. Pie Chart

- **Description:**
A pie chart represents percentage share of different categories.
- **Use Case:**
Market share of companies, survey responses, or budget allocations.

```
import matplotlib.pyplot as plt

import numpy as np

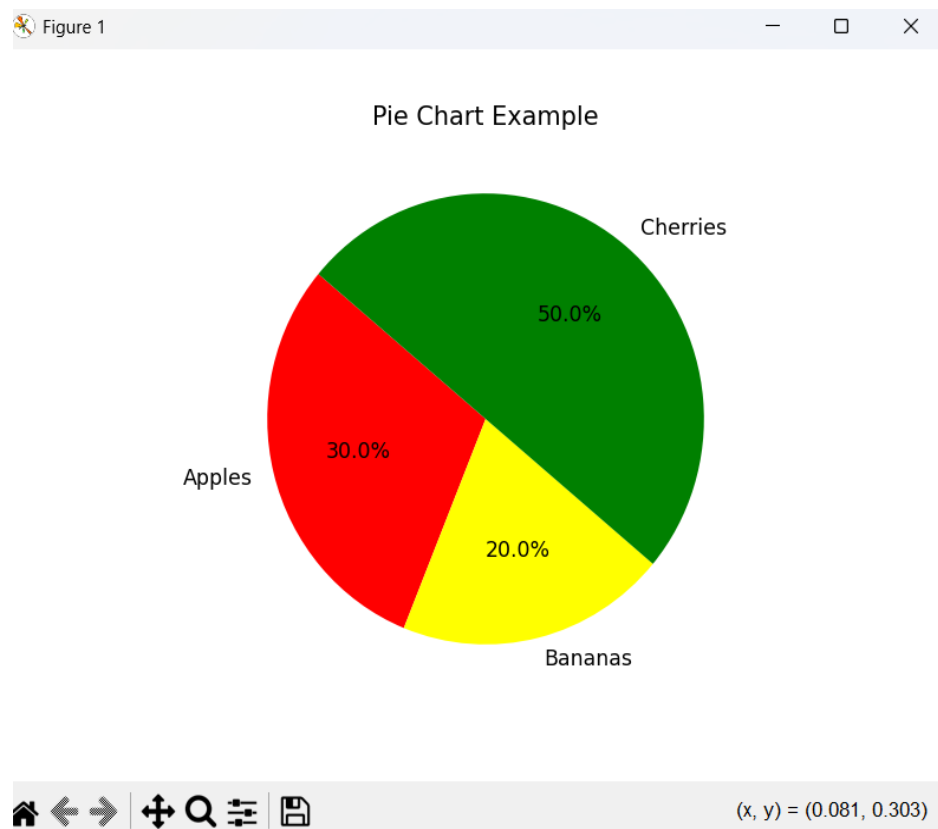
sizes = [30, 20, 50]

labels = ['Apples', 'Bananas', 'Cherries']

plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=140,
        colors=['red', 'yellow', 'green'])

plt.title("Pie Chart Example")

plt.show()
```



B. Seaborn Graphs

1. Line Plot:

- **Description:**
Seaborn line plots extend Matplotlib by adding statistical estimations and confidence intervals automatically.
- **Use Case:**
Visualizing trends while accounting for statistical variation.

```
import seaborn as sns

import matplotlib.pyplot as plt

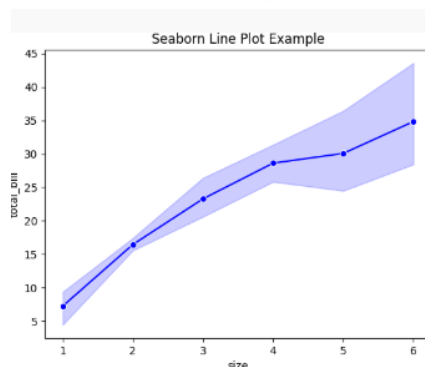
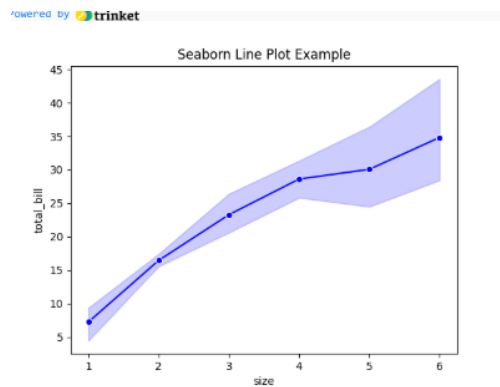
tips = sns.load_dataset("tips")

sns.lineplot(x="size", y="total_bill", data=tips, marker="o", color="blue")

plt.title("Seaborn Line Plot Example")

plt.savefig("plot.png") # <-- This saves the plot as a PNG file

plt.show()
```



[plot.png](#)

[trinket_plot.png](#)

2. Scatter Plot

- **Description:**
Seaborn's scatter plots support grouping using the hue, style, and size arguments.
- **Use Case:**
Comparing spending habits by gender or class in datasets.

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

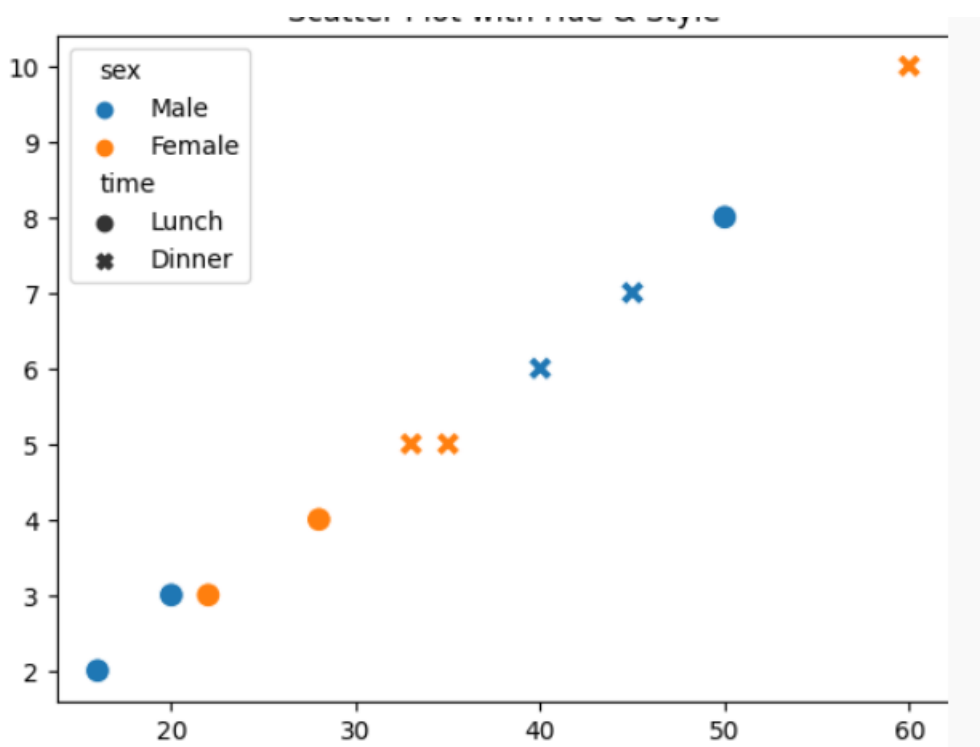
```
import pandas as pd
```

```
tips = pd.DataFrame({  
    "total_bill": [16, 35, 40, 20, 60, 28, 45, 33, 50, 22],  
    "tip":        [2, 5, 6, 3, 10, 4, 7, 5, 8, 3],  
    "sex":        ["Male", "Female", "Male", "Male", "Female", "Female", "Male",  
                  "Female", "Male", "Female"],  
    "time":       ["Lunch", "Dinner", "Dinner", "Lunch", "Dinner", "Lunch",  
                  "Dinner", "Dinner", "Lunch", "Lunch"]  
})
```

```
sns.scatterplot(x="total_bill", y="tip", hue="sex", style="time", data=tips, s=100)
```

```
plt.title("Scatter Plot with Hue & Style")
```

```
plt.show()
```



3. Bar Plot

- **Description:**
Unlike Matplotlib, Seaborn's bar plots include confidence intervals to show uncertainty.
- **Use Case:**
Comparing average sales per day with variation included.

```
import matplotlib.pyplot as plt

import seaborn as sns

tips = sns.load_dataset("tips")

sns.lineplot(x="size", y="total_bill", data=tips, marker="o", color="blue")

plt.title("Seaborn Line Plot Example")

plt.show()

sns.scatterplot(x="total_bill", y="tip", hue="sex", style="time", data=tips)

plt.title("Seaborn Scatter Plot Example")

plt.show()

sns.barplot(x="day", y="total_bill", data=tips, ci="sd", palette="Set2")

plt.title("Seaborn Bar Plot Example")

plt.show()

sns.histplot(tips["total_bill"], bins=30, kde=True, color="purple")

plt.title("Seaborn Histogram + KDE Example")

plt.show()

sns.boxplot(x="day", y="total_bill", data=tips, palette="coolwarm")

plt.title("Seaborn Box Plot Example")

plt.show()

sns.violinplot(x="day", y="total_bill", data=tips, palette="Set3")

plt.title("Seaborn Violin Plot Example")

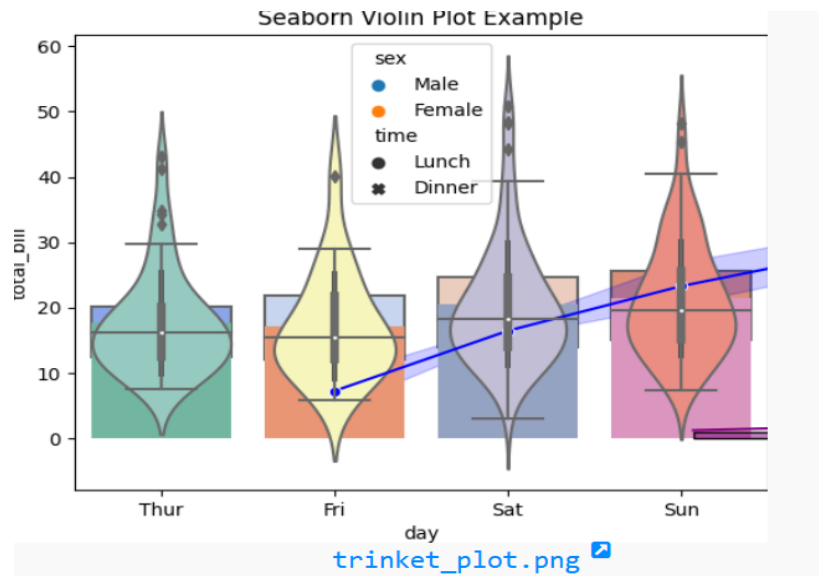
plt.show()

corr = tips.corr(numeric_only=True)

sns.heatmap(corr, annot=True, cmap="Blues")

plt.title("Seaborn Heatmap Example")

plt.show()
```

4. Histogram (Distribution Plot)

- Description:**
 Seaborn's histplot combines histograms with optional KDE (Kernel Density Estimation) to visualize probability density.
- Use Case:**
 Analyzing income distribution, scores, or waiting times.

```
import seaborn as sns
```

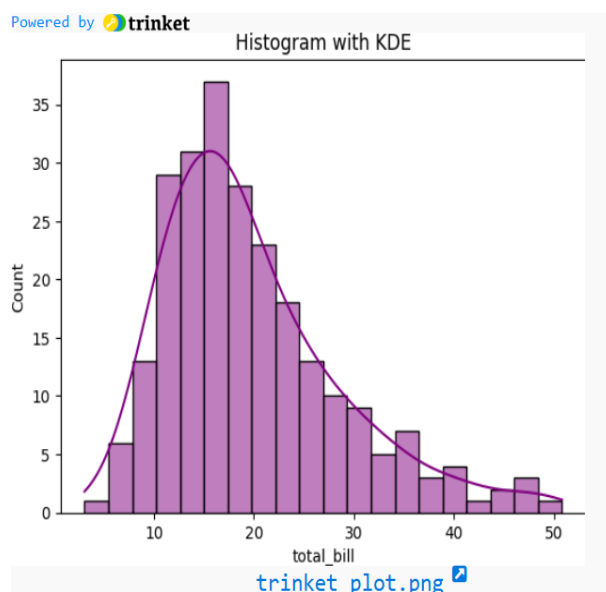
```
import matplotlib.pyplot as plt
```

```
tips = sns.load_dataset("tips")
```

```
sns.histplot(tips["total_bill"], kde=True, bins=20, color="purple")
```

```
plt.title("Histogram with KDE")
```

```
plt.show()
```



5. Box Plot

- **Description:**
Box plots show the spread, quartiles, and outliers in data.
- **Use Case:**
Comparing salaries across departments, or tips across days.

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
tips = sns.load_dataset("tips")
```

```
sns.boxplot(x="day", y="total_bill", data=tips, palette="pastel")
```

```
plt.title("Box Plot Example")
```

```
plt.show()
```

