

DOCUMENTATION

1. Matplotlib:

Matplotlib is a popular data visualization library in Python. It's often used for creating static, interactive, and animated visualizations in Python. The name "Matplotlib" combines "MATLAB" and "plotting," reflecting its origin as a tool for producing plots similar to those made in MATLAB, a popular numerical computing environment. Matplotlib allows you to generate plots, histograms, bar charts, scatter plots, etc., with just a few lines of code.

Commonly used module:

- `matplotlib.pyplot`: This module provides a MATLAB-like interface for creating plots, making it easy for users familiar with MATLAB to get started with Matplotlib.

Key Features

1. **Variety of Plots:** Supports many types of plots including:
 - Line plots
 - Bar charts
 - Histograms
 - Scatter plots
2. **Variety of Plots:** Supports many types of plots including:
 - Line plots
 - Bar charts
 - Histograms
 - Scatter plots

1.Line plot:

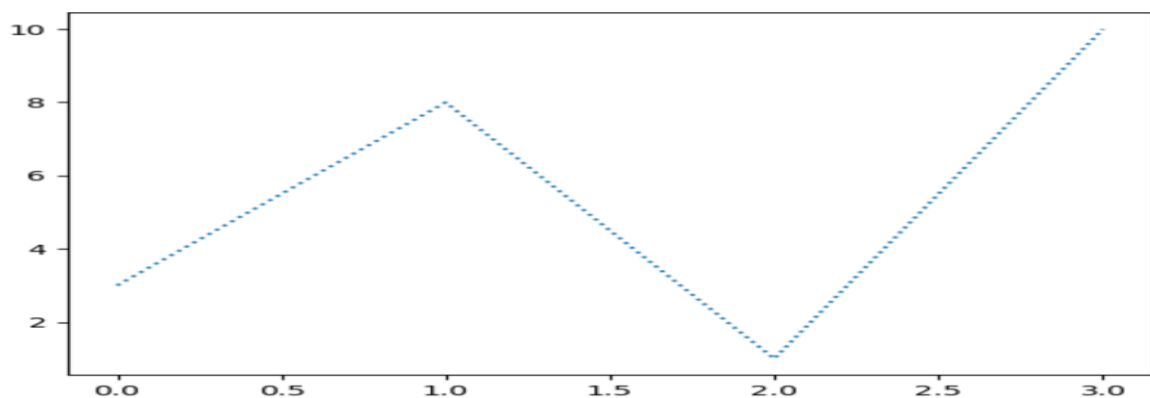
A line plot is a type of graph that displays information as a series of data points called 'markers' connected by straight line segments. It's commonly used to visualize data trends over time or other continuous variables.

```
import matplotlib.pyplot as plt
import numpy as np

ypoints = np.array([3, 8, 1, 10])

plt.plot(ypoints, linestyle = 'dotted')
plt.show()
```

OUTPUT:



Use Cases

Line plots are useful for:

- Showing trends over time (e.g., stock prices, temperature changes).
- Comparing different datasets on the same graph.

2.Histogram

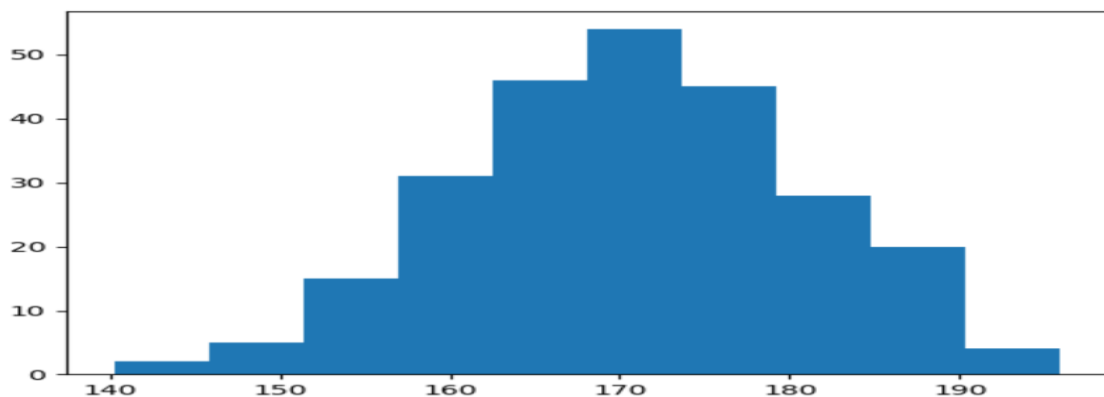
Histogram is a graphical representation of the distribution of numerical data, showing the frequency of data points within specified ranges (bins). It's useful for understanding the underlying frequency distribution of your data.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.normal(170, 10, 250)

plt.hist(x)
plt.show()
```

OUTPUT:



Use Cases

Histograms are useful for:

- Visualizing the distribution of data (normal, skewed, bimodal, etc.).
- Identifying outliers.
- Understanding the spread and central tendency of the data.

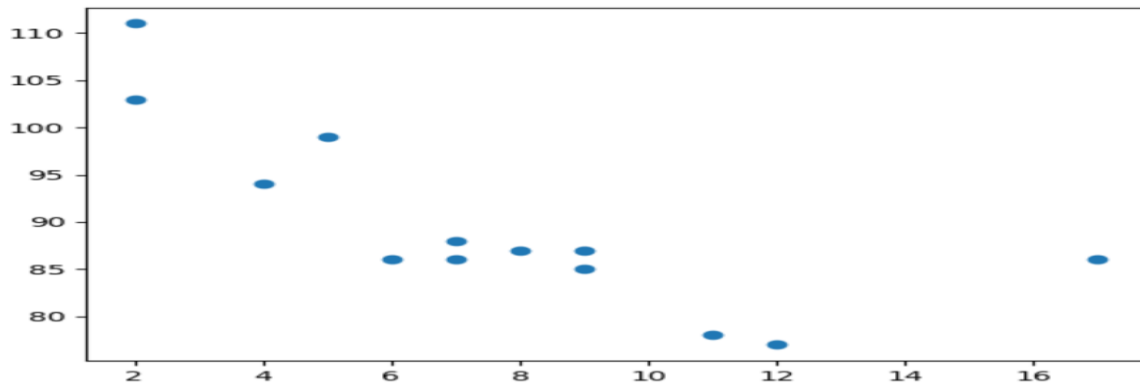
3.Scatter plot

A scatter plot is a type of data visualization that uses dots to represent the values obtained for two different variables—one plotted along the x-axis and the other along the y-axis. It's particularly useful for showing the relationship or correlation between these two variables.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])

plt.scatter(x, y)
plt.show()
```



Use Cases:

Scatter plots are great for:

- Identifying relationships between variables (correlation).
- Visualizing the distribution of data points.

2. Plotly:

Plotly library in Python is an open-source library that can be used for data visualization and understanding data simply and easily. Plotly is a powerful graphing library that enables users to create interactive visualizations. Plotly supports a wide range of charts and graphs, including line plots, bar charts, scatter plots, and 3D surface plots, among others. So you all must be wondering why Plotly is over other visualization tools or libraries.

It supports Plotly is particularly popular for its ability to produce interactive web-based visualizations that can be easily shared and embedded.

Key Features:

1. **Interactivity:** Users can hover over data points to see additional information, zoom in and out, and pan across the plot.
2. **Diverse Chart Types:** Supports various chart types, including statistical, financial, geographical, and scientific visualizations.

Common uses:

1. Data Visualization

- **Interactive Graphs:** Create interactive charts that allow users to explore data dynamically (hovering, zooming, etc.).
- **Wide Range of Chart Types:** Support for line charts, scatter plots, bar charts, histograms, box plots, 3D plots, and more.

2. Data Exploration

- **Exploratory Data Analysis (EDA):** Quickly visualize data to identify trends, outliers, and patterns, facilitating better understanding and insights.

1.BAR CHART

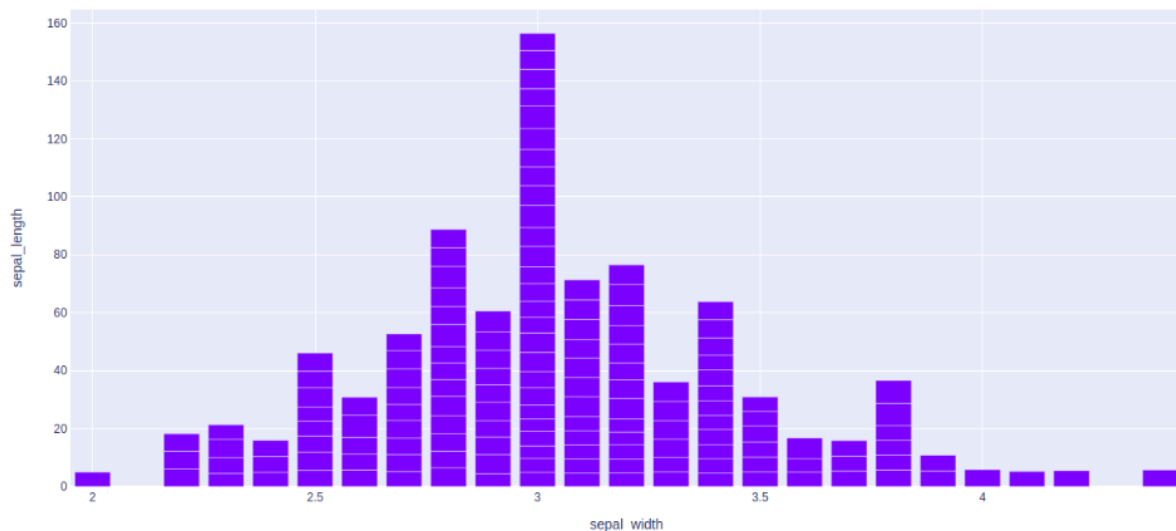
- In Python, a **bar chart** created with Plotly is a type of visualization that represents categorical data with rectangular bars. The length of each bar corresponds to the value it represents, making it easy to compare different categories or groups. Bar charts are widely used in data analysis, reporting, and presentations due to their simplicity and effectiveness in conveying categorical comparisons.

```
[2]: import plotly.express as px

df = px.data.iris()

fig = px.bar(df, x="sepal_width", y="sepal_length")

fig.show()
```



Use Cases:

- **Comparison:** Easily compare different categories, such as sales figures across different products.
- **Distribution:** Show the distribution of data points across categories.
- **Trends Over Time:** Display changes in categories over time when combined with time series data.

2.Histograms

It contains a rectangular area to display the statistical information which is proportional to the frequency of a variable and its width in successive numerical intervals. A graphical representation that manages a group of data points into different specified ranges. It has a special feature that shows no gaps between the bars and similar to a vertical bar graph.

```
import plotly.express as px

df = px.data.iris()

fig = px.histogram(df, x="sepal_length", y="petal_width")

fig.show()
```

OUTPUT:

