Matplotlib

A 2D plotting library in Python.
Used for creating static, animated, and interactive visualizations.
Main module: matplotlib.pyplot (typically imported as plt). Pyplot is also a module within matplotlib

Basic Plotting Workflow

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.plot(x, y)       # Line plot
plt.title("My Graph")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.grid(True)
plt.figure(figsize=(8, 5))     # Set figure size in inches
plt.savefig("plot.png")        # Save to file
plt.show()           # Displays the plot
```

Common Plot Types

Plot Type
Function
Line Plot
plt.plot()
Scatter Plot
plt.scatter()
Bar Chart
plt.bar()
Horizontal Bar
plt.barh()
Histogram
plt.hist()
Pie Chart
plt.pie()
Box Plot
plt.boxplot()

Plot Customisations :

plt.plot() – Line Plot :

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

Parameter

Description
Example
x, y
Coordinates to plot
[1, 2, 3]
color
Line color
red', '#00FF00'
linestyle
Line style: 'solid', '--', ':', '-.'
--'
linewidth
Line width in points
2.5
marker
Marker style: 'o', 's', '*', etc.
o'
markersize
Size of marker
8
label
Label for the legend
Line A'

plt.bar() – Vertical Bar Chart
plt.bar(x, height, color='green', width=0.4, align='center', label='Sales')

Parameter
Description
Example
x
Categories or positions
[1, 2, 3]
height
Heights of the bars
[10, 20, 15]
width
Width of bars
0.5
color
Fill color
skyblue'
edgecolor

Border color of bars
black'
align
Bar alignment ('center' or 'edge')
center'
label
Legend label
Q1 Sales'

plt.scatter() – Scatter Plot
plt.scatter(x, y, color='red', s=100, marker='^', alpha=0.6, label='Group A')

Parameter
Description
Example
x, y
Coordinates of points
[1, 2, 3]
color / c
Color of points
blue'
s
Size of points (area)
100
marker
Shape of marker ('o', '^', 's', etc.)
^'
alpha
Transparency (0 = transparent, 1 = opaque)
0.6
label
Legend label
Category A'

plt.barh() — Horizontal Bar Chart
plt.barh(y, width, color='orange', height=0.4, align='center', label='Sales')
Parameter
Description
Example
y
Categories or positions (y-axis)
['A', 'B', 'C']

width

Length of bars (values)

[10, 15, 7]

height

Thickness of each bar

0.4

color

Fill color of bars

orange'

edgecolor

Border color of bars

black'

align

center' (default) or 'edge'

center'

label

Label for the legend

Q1 Sales'

plt.hist() — Histogram

```
plt.hist(data, bins=5, color='green', edgecolor='black', alpha=0.6)
```

Parameter

Description

Example

data

Numeric data array

[10, 20, 20, 30, 40]

bins

Number of intervals (bars)

5

color

Fill color of bars

green'

edgecolor

Border color of bars

black'

alpha

Transparency (0 to 1)

0.6

density

Normalize to probability (True/False)

TRUE

label

Label for legend
Exam Scores'

plt.pie() — Pie Chart
plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=90)
Parameter
Description
Example
sizes
Values for each wedge
[30, 40, 30]
labels
Category labels
['A', 'B', 'C']
autopct
Display % values on slices
%1.1f%%'
startangle
Start angle of the pie chart
90
colors
List of colors for slices
['red', 'blue', 'green']
explode
Offset slices from the center
[0, 0.1, 0]
shadow
Add shadow effect
TRUE

plt.boxplot() — Box Plot :
plt.boxplot(data, vert=True, patch_artist=True)
Parameter
Description
Example
data
List/array (1D or 2D)
[10, 15, 14, 20, 18]
vert
Orientation: vertical (True) or horizontal
TRUE
patch_artist

Fill the box with color

TRUE

notch

Show notched box for median comparison

TRUE

labels

Custom category labels for boxes

['Group A']

widths

Width of the box

0.6

meanline

Show mean as a line

TRUE

showmeans

Display mean marker

TRUE

Subplots :

With subplots(), you can draw multiple plots in one figure(one beside other, or one above other)

plot 1

X = np.array([0,1,2,3])

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

plt.subplot(1,2,1) —------>> meaning, the fig has 1 row, 2 columns(one chart beside other), and this is the first plot

plt.plot(x,y)

Plot 2

X = np.array([0,1,2,3])

Y = np.array([10,20,30,40])

plt.subplot(1,2,2) —------>> meaning, the fig has 1 row, 2 columns(one chart beside other), and this is the second plot

plt.plot(x,y)

If we want one above the other,

plt.subplot(2,1,1)

plt.subplot(2,1,2)

Other Useful Functions

Function

Description

plt.xlim(), plt.ylim()

Set x/y axis limits

plt.xticks(), plt.yticks()

Customize axis ticks

```
plt.tight_layout()
```

Adjust spacing automatically