Matplotlib

Matplotlib

Matplotlib is easy to use and an amazing visualizing library in Python. It is built on NumPy arrays and designed to work with the broader SciPy stack and consists of several plots like line, bar, scatter, histogram, etc.

Installing Matplotlib

```
!pip install matplotlib
```

Importing Matplotlib

```
import matplotlib
```

Checking Matplotlib Version

```
matplotlib.__version__
```

Pyplot

Most of the Matplotlib utilities lies under the pyplot submodule, and are usually imported under the plt alias
Each Pyplot function makes some change to a figure. For example, a function creates a figure, a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

Importing Pyplot

Import matplotlb.pyplot as plt

plt.plot()

The plot() function is used to draw points (markers) in a diagram.
By default, the plot() function draws a line from point to point.
The function takes parameters for specifying points in the diagram.

Useful plot functions

function	Description
plt.title("title")	Set a title for the plot
plt.xlabel("label")	Add label to X-axis
plt.ylabel("label")	Add label to Y-axis
plt.legend("location")	Specify location for legend (default: 'best')
plt.show()	Display all open figures

Line plot

A line chart or line plot or line graph or curve chart is a type of chart which displays information as a series of data points called 'markers' connected by straight line segments.

Line Plots are widely used in time series to identify the pattern of the data.

Syntax

plt.plot(data, color , linestyle , linewidth, marker)

data : The data for line plot

color : color of line (Name of color or hexadecimal color format)

Please refer the following link to get list of markers : https://matplotlib.org/stable/tutorials/colors/colors.html

linestyle Style	Symbol
'solid' (default)	'-'
'dotted'	':'
'dashed'	'--'
'dashdot'	'-.'

linewidth : Width of the line

Marker : Style to represent point on line

Please refer the following link to get list of markers : https://matplotlib.org/stable/api/markers_api.html

Scatter plot

Scatter plots are the graphs that present the relationship between two variables in a data-set. It represents data points on a two-dimensional plane or on a Cartesian system.

It is used in data analysis for checking correlation between two variables.

Syntax

```
plt.scatter(x, y, c, s, marker)
```

x : Array of X axis
y : Array of Y axis
c : Color of points
s : Size of points
marker : Style to represent point on line

Axes Class

Axes object is the region of the image with the data space. A given figure can contain many Axes, but a given Axes object can only be in one Figure.
Axes object is added to figure by calling the add_axes() method.

Syntax

```
ax=fig.add_axes([0,0,1,1])
```

Important Axes Function

function	Description
ax.legend()	adds a legend to the plot figure.
ax.plot()	plots values of one array versus another as lines or markers.
ax.set_title()	Set title to the plot
ax.set_xlabel()	Add label to X-axis
ax.set_ylabel()	Add label to Y-axis

Bar plot using ax

A bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

A bar graph shows comparisons among discrete categories.

Syntax

ax.bar(x, height, width, bottom, align)

x : Data values

height : the height(s) of the bars.

width : the width(s) of the bars default 0.8

bottom : the y coordinate(s) of the bars default None.

align : {‘center’, ‘edge’}, optional, default ‘center’

Seaborn

Seaborn

Seaborn is a library mostly used for statistical plotting in Python. It is built on top of Matplotlib and provides beautiful default styles and color palettes to make statistical plots more attractive.

Installing seaborn

```
!pip install seaborn
```

Importing seaborn

```
import seaborn as sns
```

Checking Matplotlib Version

```
sns.__version__
```

Dist plot & Line plot

Distplot stands for distribution plot, it takes as input an array and plots a curve corresponding to the distribution of points in the array.

Syntax

```
sns.distplot(arr)
```

arr: Data values

Line plot function draw's a line plot with the possibility of several semantic groupings.

Syntax

```
sns.lineplot(x,y,data,style)
```

x, y: Input data variables; must be numeric.

data: input data frame

style: Grouping variable that will produce lines with different dashes and/or markers. Can have a numeric dtype but will always be treated as categorical.

Count plot & Bar plot

Count plot is used to Show the counts of observations in each categorical bin using bars.

Syntax

```
sns.countplot(x, data)
```

x: This parameter take names of variables in data or vector data, optional, Inputs for plotting long-form data.
data : input data frame

Bar plot is used to draw bar chart.

Syntax

```
sns.barplot(x,y ,data)
```

x, y: Input data variables;
data : input data frame

Scatter plot & Box plot

Syntax

```
sns.scatterplot(x,y, data)
```

x, y: Input data variables;
data : input data frame

A box plot helps to maintain the distribution of quantitative data in such a way that it facilitates the comparisons between variables or across levels of a categorical variable.
The main body of the box plot showing the quartiles and the median's confidence intervals if enabled. The medians have horizontal lines at the median of each box and while whiskers have the vertical lines extending to the most extreme, non-outlier data points and caps are the horizontal lines at the ends of the whiskers.

Syntax

```
sns.boxplot(x,y data)
```

x, y: Input data variables;
data : input dataframe

Pair plot & Heat Map

Pair plot gives multiple pairwise bivariate distributions in a dataset.

Syntax

```
sns.pairplot(data)
```

data : input data frame

Heatmap is defined as a graphical representation of data using colors to visualize the value of the matrix. In this, to represent more common values or higher activities brighter colors basically reddish colors are used and to represent less common or activity values, darker colors are preferred.

Syntax

```
sns.heatmap(data)
```

data : input dataframe


Thank you!