

Task-1

Objective: Create a comprehensive documentation guide for **2** of the following Python visualization libraries: **Matplotlib, Seaborn, Plotly, Bokeh, and Pandas**. Your guide should focus on the variety of graphs each library can generate and include practical examples with code snippets.

Python visualization libraries:

Python visualization libraries are the collection of tools that allows to create informative and visually appealing data representations. These libraries are used in various functionalities like data exploration, data analysis and data representation. By using these libraries, we can effectively communicate the complex data insights.

There are various libraries in Python to visualize the complex data, that includes:

1. Matplotlib
2. Seaborn
3. Plotly
4. Bokeh
5. Pandas
6. Altair etc.

Libraries Overview:

1. Seaborn

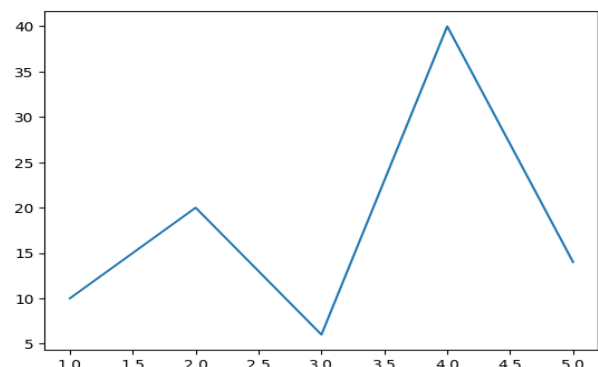
- Seaborn is a Python data visualization library based on [matplotlib](#). It provides a high-level interface for drawing attractive and informative statistical graphics.
- Seaborn helps you to explore and understand your data.
- Use cases of Seaborn includes data visualization in statistical analysis, Machine Learning and data exploration.
- It can be imported into the python program using the “import statement”

Seaborn consists of various types of plotting that are used in the data representation, they include:

a) Line Plot:

- Line plot is also known as line graph, it is a type of chart that is used to display data points connected by a straight line.
- It is used to show the relationship between two variables.

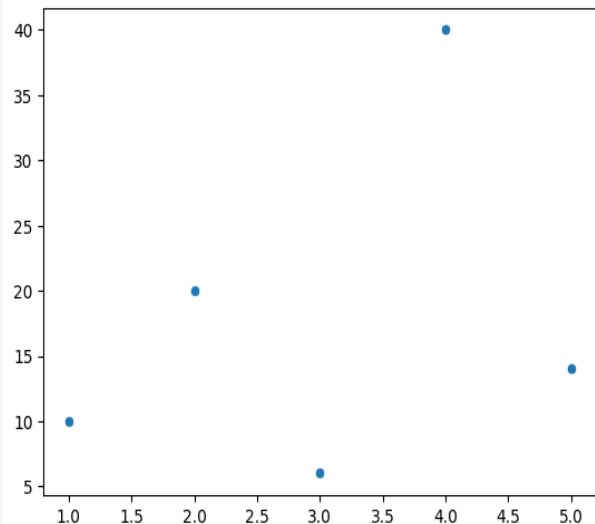
```
#Line plot
import seaborn as sns
import matplotlib.pyplot as plt
x=[1,2,3,4,5]
y=[10,20,6,40,14]
sns.lineplot(x = x,y = y)
plt.show()
```



b) Scatter Plot:

- Scatter plot is tool used in data visualization, used to depict the relationship between two numerical variables.
- Each data point is represented by a dot on a two-dimensional graph, the position of the dot is determined by the values of corresponding variables.

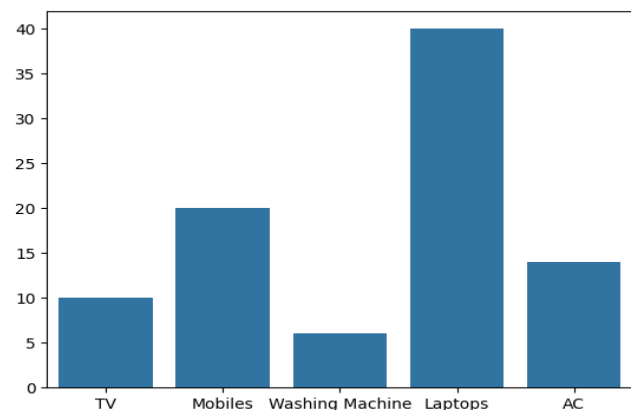
```
#Scatter plot
import seaborn as sns
import matplotlib.pyplot as plt
x=[1,2,3,4,5]
y=[10,20,6,40,14]
sns.scatterplot(x = x,y = y)
plt.show()
```



c) Bar Chart:

- A bar chart is a fundamental and powerful tool in data visualization that displays categorical data using rectangular bars.
- Each bar corresponds to a specific category, and its length or height is proportional to the value associated with that category.
- Bar charts are used to: compare values across categories, Track changes over time etc.

```
#bar chart
import seaborn as sns
import matplotlib.pyplot as plt
items=["TV","Mobile","Washing Machine","Laptop","AC"]
availability=[10,20,6,40,14]
sns.barplot(x = items,y = availability)
plt.show()
```

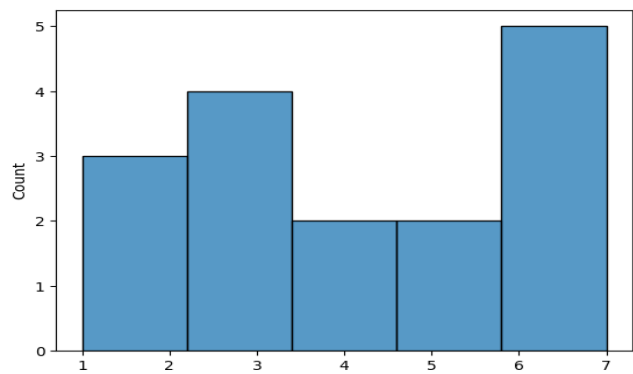


d) Histogram:

- Histograms are the powerful tool in data visualization, used to represent the distribution of a numerical dataset.
- It consists of bars that represent the frequency (or count) of data points within specified ranges, known as bins.

#Histogram

```
import seaborn as sns
import matplotlib.pyplot as plt
data=[1,2,2,3,3,3,3,4,4,5,6,6,6,7]
sns.histplot(data)
plt.show()
```

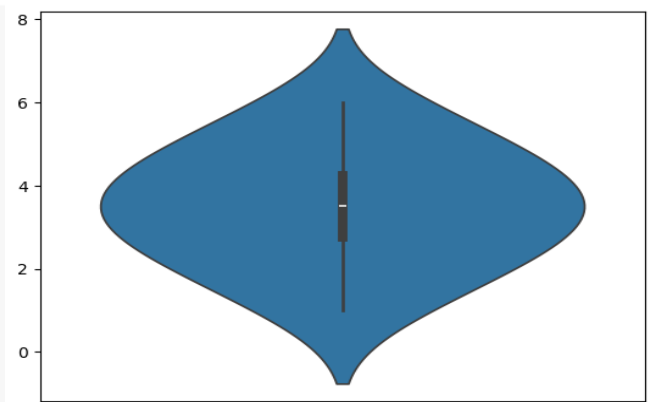


e) Violin Plot:

- A violin plot is a powerful data visualization tool that combines aspects of a box plot and a kernel density plot.
- They provide a more detailed representation of the distribution of a numerical dataset, especially when the data is skewed or has multiple peaks.

#Violin plot

```
import seaborn as sns
import matplotlib.pyplot as plt
data=[1,2,2,3,3,3,4,4,4,5,5,6]
sns.violinplot(data)
plt.show()
```



2) Plotly

- The [plotly Python library](#) is an interactive, [open-source](#) plotting library that supports over 40 unique chart types covering a wide range of statistical, financial, geographic, scientific, and 3-dimensional use-cases.
- Plotly is a powerful and flexible library that empowers you to create visually appealing and interactive data visualizations in Python.
- Plotly can be used in a variety of use cases, including:
 - Data Analysis and Visualization
 - Exploratory data analysis (EDA)
 - Hypothesis testing
 - Data storytelling

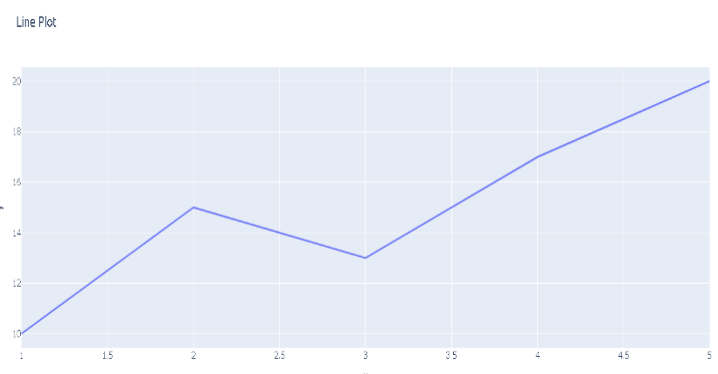
Plotly consists of various types of plotting that are used in the data representation, they include:

a) Line Plot:

- Line plot is also known as line graph, it is a type of chart that is used to display data points connected by a straight line.

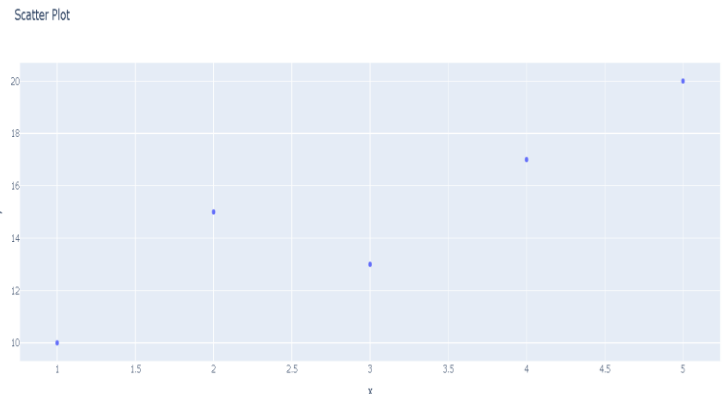
#Line plot

```
import plotly.express as px
data = {
    'x': [1, 2, 3, 4, 5],
    'y': [10, 15, 13, 17, 20]
}
fig = px.line(data, x='x', y='y', title='Line Plot')
fig.show()
```



- Scatter plot is tool used in data visualization, used to depict the relationship between two numerical variables.
- Each data point is represented by a dot on a two-dimensional graph, the position of the dot is determined by the values of corresponding variables.

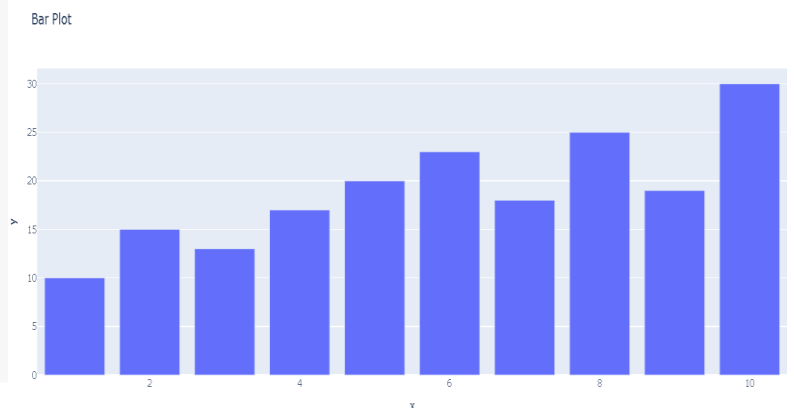
```
#Scatter plot
import plotly.express as px
data = {
    'x': [1, 2, 3, 4, 5],
    'y': [10, 15, 13, 17, 20]
}
fig = px.scatter(data, x='x', y='y', title='Scatter Plot')
fig.show()
```



c) Bar Chart:

- A bar chart is a fundamental and powerful tool in data visualization that displays categorical data using rectangular bars.
- Each bar corresponds to a specific category, and its length or height is proportional to the value associated with that category.

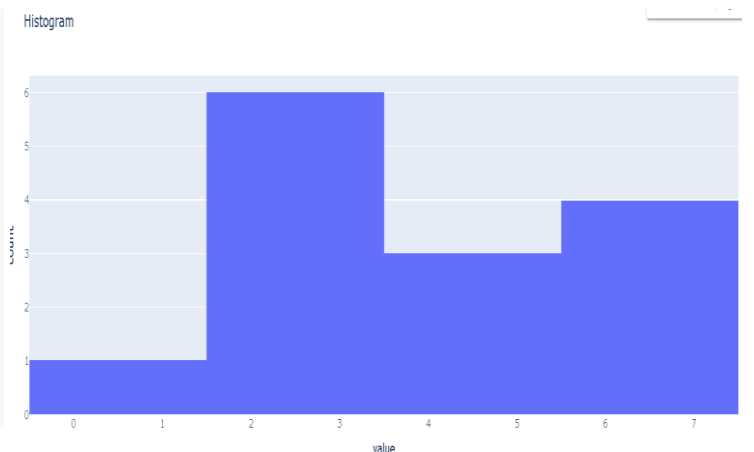
```
#Bar plot
import plotly.express as px
data = {
    'x': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'y': [10, 15, 13, 17, 20, 23, 18, 25, 19, 30]
}
fig = px.bar(data, x='x', y='y', title='Bar Plot')
fig.show()
```



d) Histogram:

- Histograms are the powerful tool in data visualization, used to represent the distribution of a numerical dataset.
- It consists of bars that represent the frequency (or count) of data points within specified ranges, known as bins.

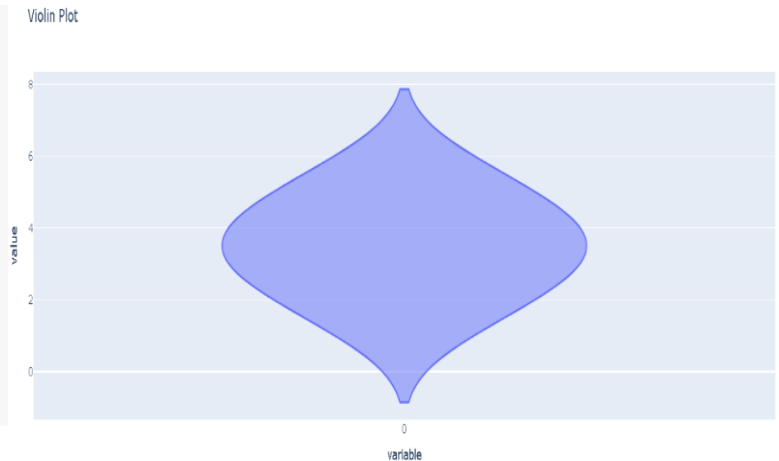
```
#Histogram
import plotly.express as px
data = [1, 2, 2, 3, 3, 3, 3, 4, 4, 5, 6, 6, 6, 7]
fig = px.histogram(data, title='Histogram')
fig.show()
```



e) Violin Plot:

- Violin plot is a powerful data visualization tool that combines aspects of a box plot and a kernel density plot.
- They provide a more detailed representation of the distribution of a numerical dataset, especially when the data is skewed or has multiple peaks.

```
#Violin plot
import plotly.express as px
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 5, 5, 6]
fig = px.violin(data, title='Violin Plot')
fig.show()
```



Overview of Seaborn library:

Advantages:

- Seaborn comes with a variety of themes and color palettes that make it easy to create visually appealing plots.
- Seaborn has advanced visualization functions like violin plots, heatmaps, and cluster maps.
- Seaborn integrates with Pandas DataFrames, making it easy to visualize data directly from a dataset.
- Seaborn's high-level API abstracts away many of the low-level details and boilerplate code that you need to write with Matplotlib.
- Seaborn automatically handles the aesthetics, labels, legends, and axes of your plots.
- Seaborn's concise syntax streamlines complex plotting tasks.

Disadvantages:

- Seaborn's ability to automate the creation of multiple figures can lead to memory usage issues.
- Seaborn has a different API from Matplotlib.
- Seaborn's legend is outside of the plot by default, so it cannot be part of a larger matplotlib figure.

Overview of Plotly library:

Advantages:

- Plotly is more interactive than other tools like Matplotlib or Seaborn.
- Plotly can be used with many languages and tools, including Python, R, MATLAB, Perl, Julia, and Arduino.
- Plotly can easily share interactive plots online with others.
- Plotly has a simple syntax for creating interactive plots.
- Plotly is compatible with ggplots in R and Python.

Disadvantages:

- The documentation for Plotly can be out of date.
- Some customization options require complex and technical solutions.
- Plotly may experience performance issues with large datasets or complex visualizations.
- Cloud Some advanced features are only available through Plotly's online cloud service.
- The development site URL can't be shared with anyone except app co-owners and administrators.

Conclusion:

In summary, Seaborn and Plotly serve various purposes in Python visualization. Seaborn is perfect for creating visually appealing statistical graphics, while Plotly is used in generating interactive plots that enhance data exploration and storytelling. Choosing between them depends on whether you prioritize aesthetics or interactivity in your visualizations.