Visualization Library Documentation

Visualization libraries are used to generate graphs plots by the values of different feature or a single feature of a data. With visualization the patterns and trends in data are easy to find and analyse.

Some major visualization libraries in python are:

- Matplotlib
- Seaborn
- Bokeh
- Plotly

Matplotlib

Matplotlib is a basic level data visualization library built on Numpy Arrays. It is used to generate 2D graphs.

Components of a plot of Matplotlib:

Figure: The figure contains all the elements of the plot. A figure can have multiple axes.

Axes: Axes is the area in the figure where the data is plotted.

Axis: In matplotlib the axis are x-axis and y-axis. These set the limits and scaling of the data.

Artist: All the elements of the figure is an artist like labels, legend, ticks, axis.

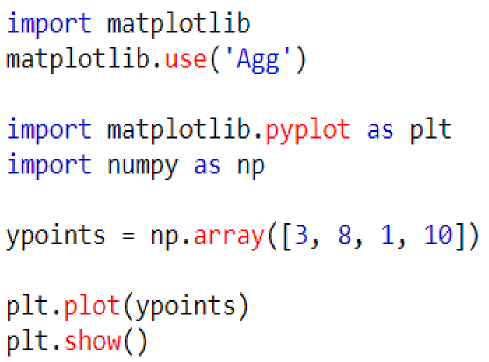
We import *pyplot,* a sub library in matplotlib to define the plots and *numpy for numerical operations.*

Different type of plots in Matplotlib

1. Line Graph
2. Bar Graph
3. Histogram
4. Scatter Plot
5. Pie Chart
6. Box Plot

Line Graph

Line graph is used to plot data of two feature on different axis.

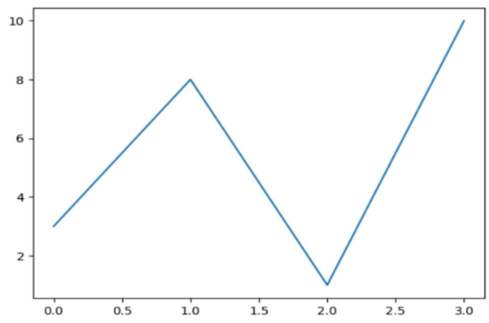


```
import matplotlib
matplotlib.use('Agg')

import matplotlib.pyplot as plt
import numpy as np

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

plt.plot(ypoints)
plt.show()
```

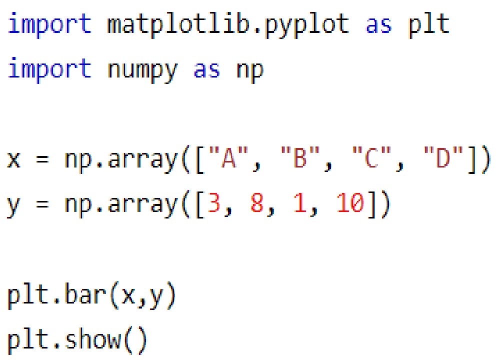


plt - To implement matplotlib.pyplot
array() - To create array
figure() - To allocate size of the plot.
plot() - To provide arguments for axis data,x and y for line plot.
xlabel() - Label for x-axis
ylable() - Label for y-axis
Two separate linear relation of data is represented by using two plot() in the same figure.

Use: Projects the relation between the data.

Bar Graph

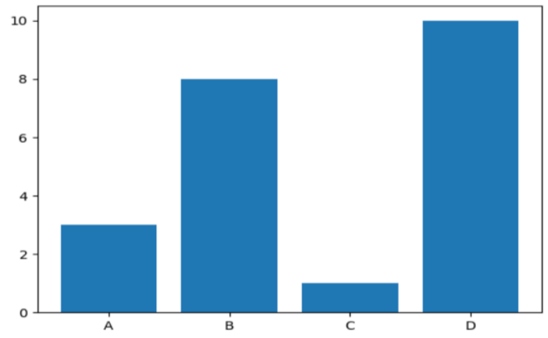
Bar graph is used to represent the strength or value of different data. It can be plotted vertically or horizontally.



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

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

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



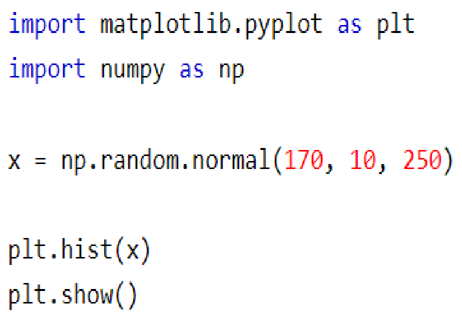
plt - To implement matplotlib.pyplot
array() - To create array
figure() - To allocate size of the plot.
bar() - To provide arguments for axis data,x and y for bar graph.
xlabel() - Label for x-axis
ylabel() - Label for y-axis

y = x*2 gives values of list of x mutiplied by 2 seperately.

Use: Compare different types of data with respect to their strength, count, etc,.

Histogram

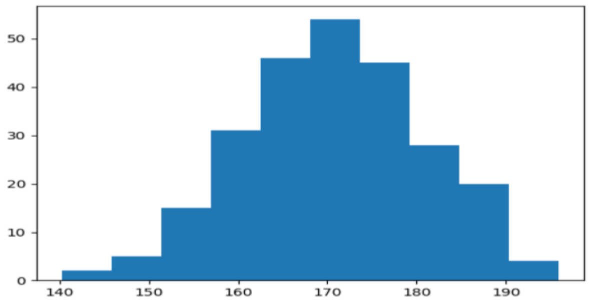
Histogram is used to show the frequency distribution of different data values throughout a range of values. It consists of bin values to aggregate a group of data.



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

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

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

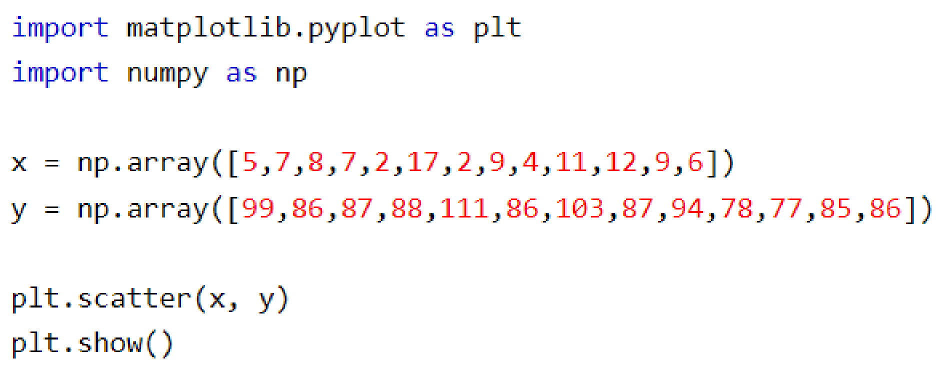


plt - To implement matplotlib.pyplot
np - To implement numpy
random.normal() - To generate list of 500 random numbers.
figure() - To allocate size of the plot.
hist() - To provide arguments for axis data,x and y for histogram as value and frequency of the values.
xlabel() - Label for x-axis
ylabel() - Label for y-axis

Use: Compare different types of data with respect to their strength, count, etc,.

Scatter Plot

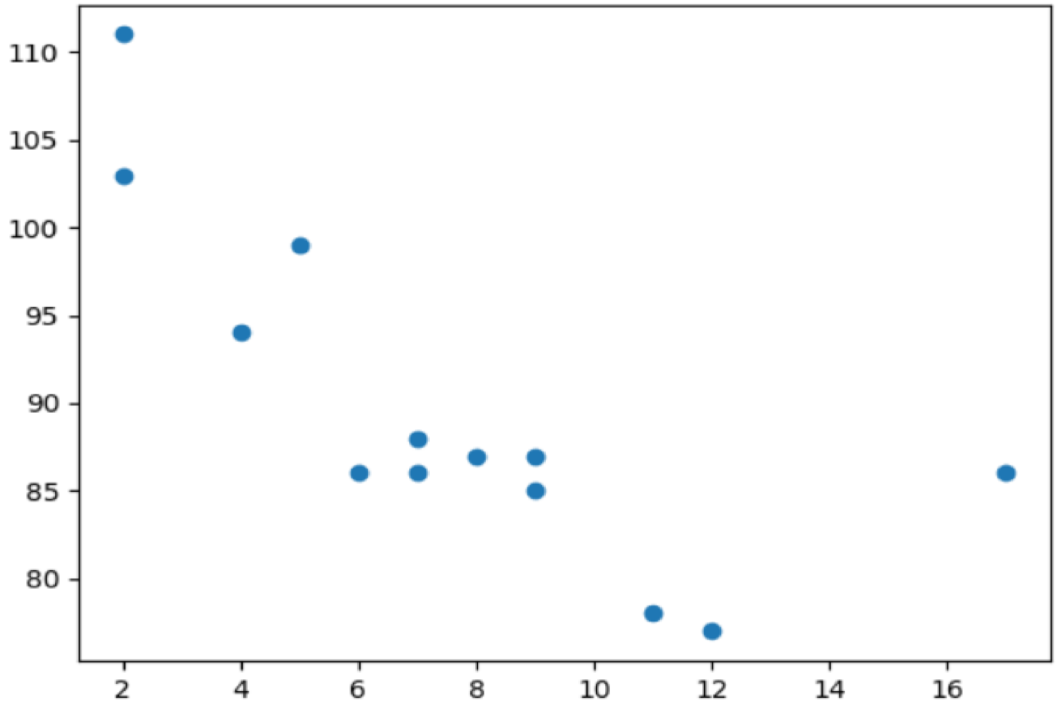
Scatter plot uses the data to show the relationship between them by plotting markers in data coordinates.



```
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()
```



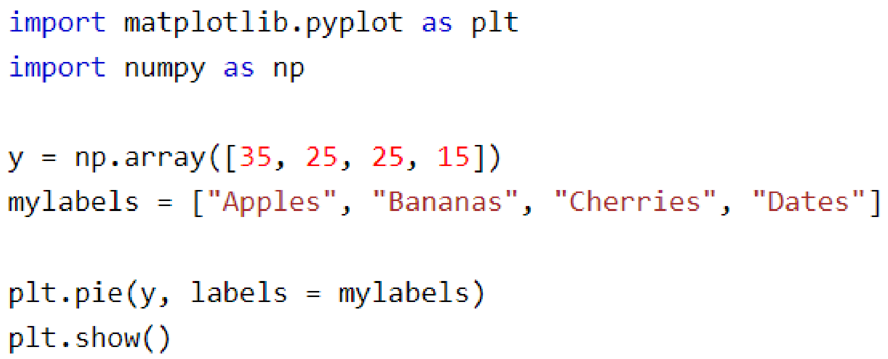
plt - To implement matplotlib.pyplot
np - To implement numpy
array() - To convert list into numpy array.
scatter() - To provide arguments for axis data,x and y for bar graph including color of bubbles, edgecolor of bubbles , and linewidth of the edge of bubbles.

Bubble=x*100 to multiply the values of list x with 100.

Use: Identifying outliers easily.

Pie Chart

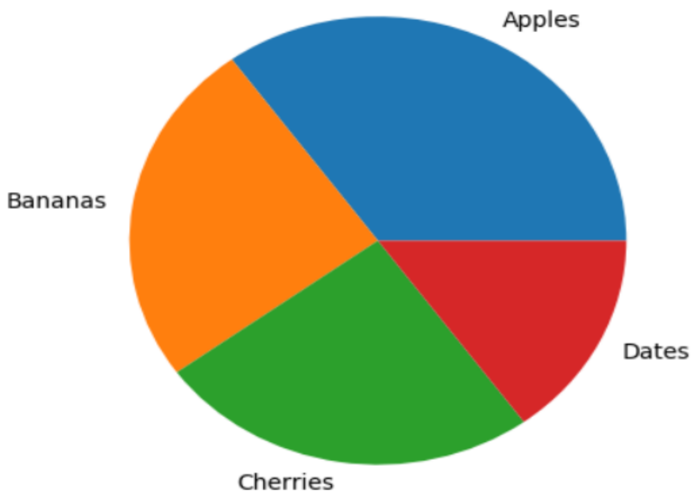
Pie chart is a circular plot used to show the composition of types of data in categorical data. The whole circle represents 100% of the data.



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

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(y, labels = mylabels)
plt.show()
```



plt - To implement matplotlib.pyplot

pie() - To provide arguments for axis data for pie chart including labels color wedges, edgecolor of wedges, and explode to highlight categories seperately.

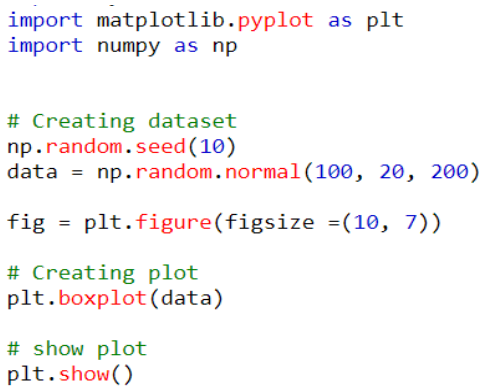
Fruit - Contains list of fruit name for label.
Data - List of value for each fruit.
Color - List of colors to be given to different wedge.

Exp - Contains list of values to define the stregth of explode

Wp - Contains wedge property for line width and edge color

Box Plot

Box plot is used to represent the summary of used data with a line divided box and extended line markers.



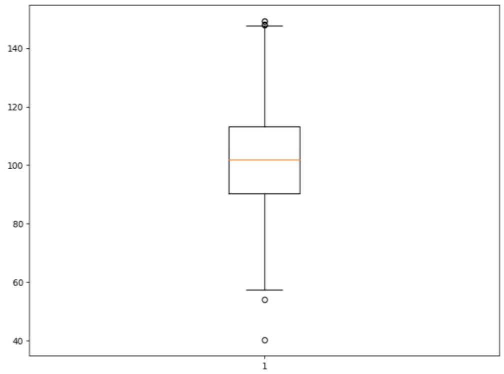
```
import matplotlib.pyplot as plt
import numpy as np


# Creating dataset
np.random.seed(10)
data = np.random.normal(100, 20, 200)

fig = plt.figure(figsize =(10, 7))

# Creating plot
plt.boxplot(data)

# show plot
plt.show()
```



plt - To implement matplotlib.pyplot

np - To implement numpy

Random.normal(x,y,z) - To generate numpy array.

boxplot() - To provide data fro the box plot.

d1,d2,d3 - Three lists are generated using random.normal() function.

Data - List that stores three series of list d1,d2,d3.

Orange line in centre shows mean.

The box represents the IQR Quartile1 , Quartile 2 and Quartile 3.

The marked lines represent the range of the the data.

The dots outside represent the outlier data which are not fit in the dataset.

Use: Find mean, median, IQR, Outliers.

Seaborn

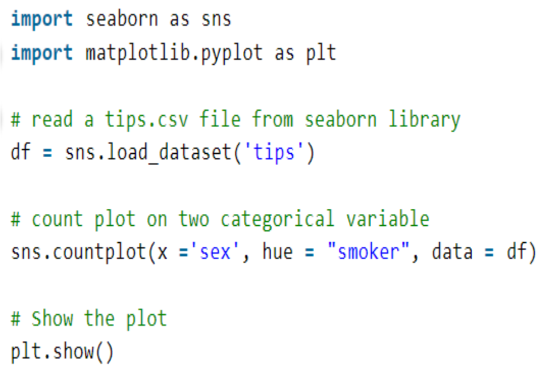
Seaborn is an advanced python visualization library built on top of matplotlib and pandas. It is used to provide enhanced visual representation and easy to implement multiple types of plots in same figure.

Different type of plots in Seaborn

1. Count Plot
2. Box Plot
3. Strip Plot
4. Pair Plot
5. Heat Map
6. Cat Plot
7. KDE plot

Count Plot

Count plot is used to represent the strength of categorical data like bar graph.

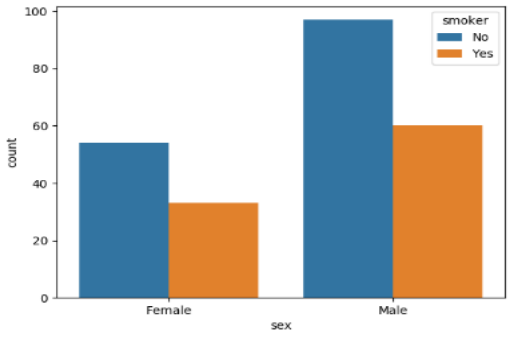


```
import seaborn as sns
import matplotlib.pyplot as plt

# read a tips.csv file from seaborn library
df = sns.load_dataset('tips')

# count plot on two categorical variable
sns.countplot(x ='sex', hue = "smoker", data = df)

# Show the plot
plt.show()
```



pd - To implement pandas
sns - To implement seaborn
load_dataset() - To get dataset from external file.
countplot() - Takes parameters as x value and data to plot.

df- Contains the dataset.
x - Given the data values of gender.

Advantage: Does not require to specify count property or to give color for data.

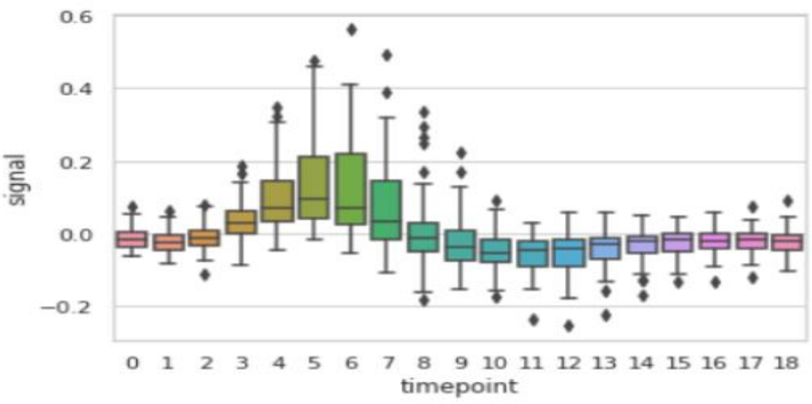
Box Plot

Box plot is used to represent the summary of used data. The mean, median, and IQR is represented with a line divided box and extended line markers.

```
import seaborn


seaborn.set(style='whitegrid')
fmri = seaborn.load_dataset("fmri")

seaborn.boxplot(x="timepoint",
                y="signal",
                data=fmri)
```



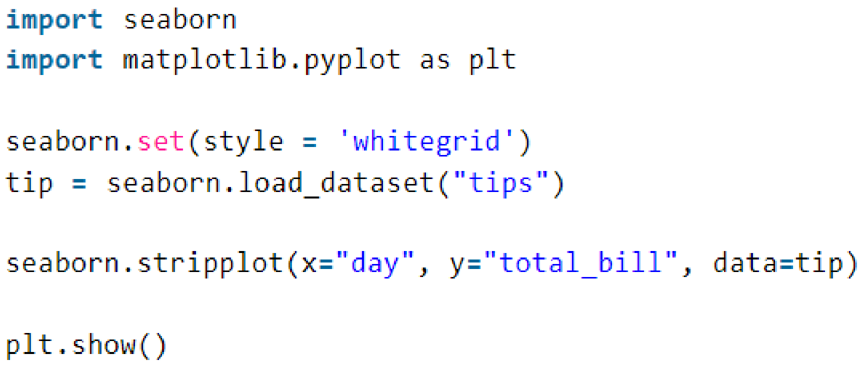
pd - To implement pandas
sns - To implement seaborn
boxplot() - Takes parameters as x, y, and data to plot.

dt - Contains the dataset.

x - Given the data values of timepoint.
y - Given the data values of signal.

Advantage: Automatic color representation.

Strip Plot
Strip plot is similar to scatter plot but used to represent values of categorical data.

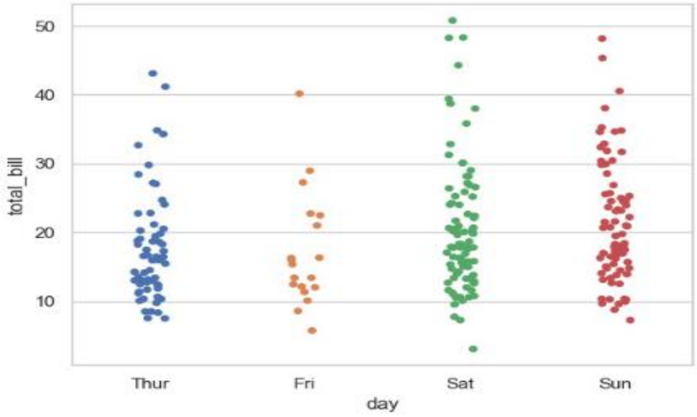


```
import seaborn
import matplotlib.pyplot as plt

seaborn.set(style = 'whitegrid')
tip = seaborn.load_dataset("tips")

seaborn.stripplot(x="day", y="total_bill", data=tip)

plt.show()
```



pd - To implement pandas

sns - To implement seaborn
stripplot() - Takes parameters as x, y, and data to plot.

dt - Contains the dataset.

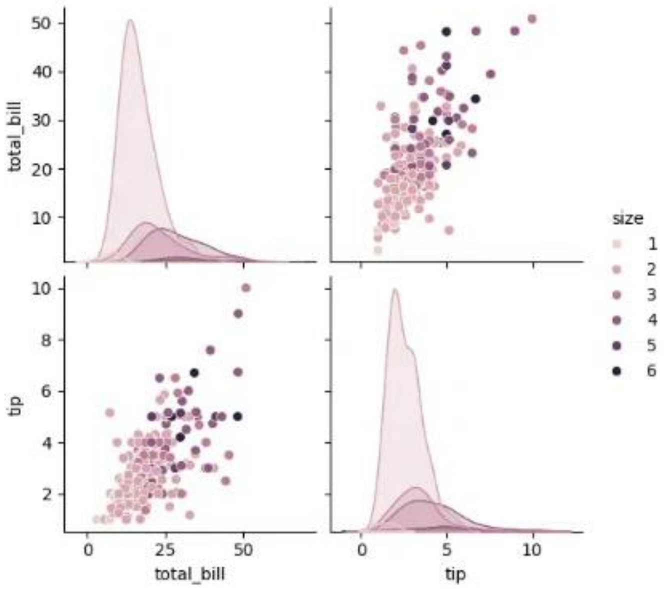
x - Given the data values of day.
y - Given the data values of total bill.
hue - To show difference in genre through differenrt colors.

Advantage: Represent multivariate data easily.

Pair Plot

Pair plot is used to represent the plotting for each pair of feature in single figure.

```
import seaborn
import matplotlib.pyplot as plt
df = seaborn.load_dataset('tips')
seaborn.pairplot(df,hue ='size')
plt.show()
```



pairplot seabon

pd - To implement pandas

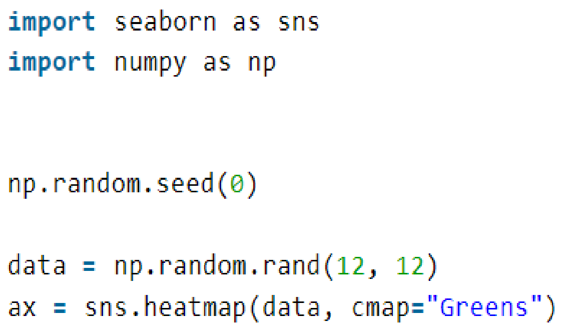
sns - To implement seaborn
pairplot() - Takes parameter data to plot.

df - Contains the dataset.
hue - To show difference in genre through differenrt colors.

Advantage: Provide subplots for pattern analysis with each feature.

Heat Map

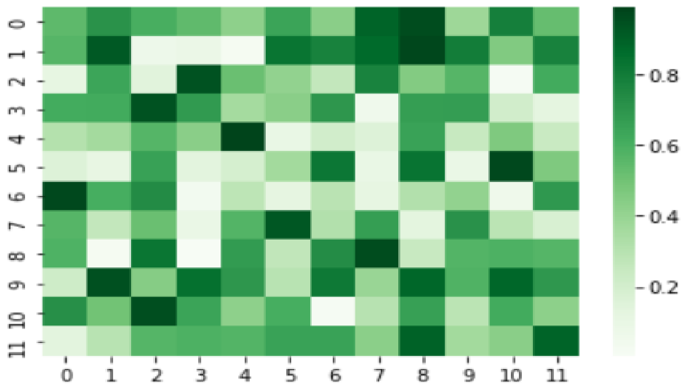
Heatmap is used to represent to strength of relation between the data features.



```
import seaborn as sns
import numpy as np


np.random.seed(0)

data = np.random.rand(12, 12)
ax = sns.heatmap(data, cmap="Greens")
```



Heatmap with a sequential colormap

np - To implement numpy
sns - To implement seaborn

random.seed() - To generate a matrix of size (10,10), with values ranging from 1 to 50.

heatmap() - Takes parameters as data, vmin and vmax to plot.

dt - Contains the dataset.

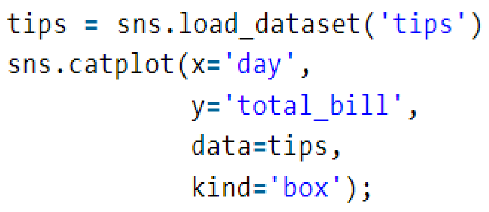
vmax - Minimum value in the plot color range.

vmin - Maximum value in the plot color range.

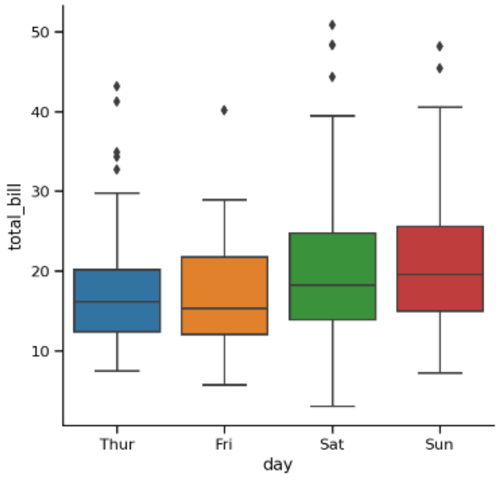
Advantage: Color coded representation of strength of relation.

Cat Plot

Cat plot is special plot in seaborn, with cat plot any other categorical plot can be defined with the help of "kind" parameter given with name of the type of plot.



```
tips = sns.load_dataset('tips')
sns.catplot(x='day',
            y='total_bill',
            data=tips,
            kind='box');
```



pd - To implement pandas

sns - To implement seaborn

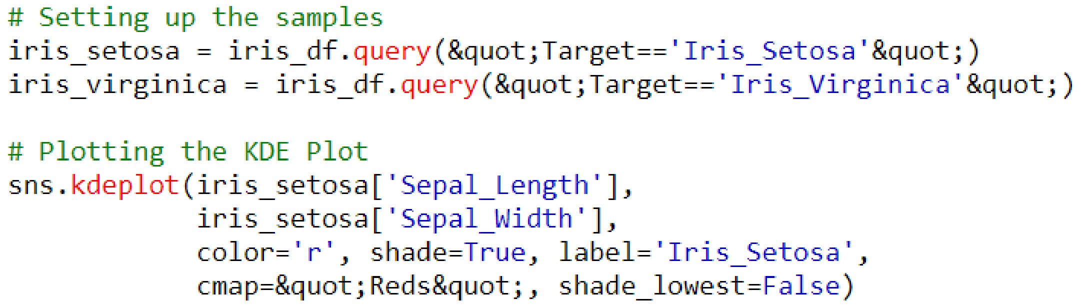
read_csv() - To get dataset from external file.

catplot() - Takes parameters x, y, data to plot with hue and kind of plot.

dt - Contains the dataset.
x - Given the values of gender.
y - Given the values of age.
kind - Given as “box” to plot box plot.
hue - show difference in genre through different colors.
Advantage: Easy to represent different types of plots.

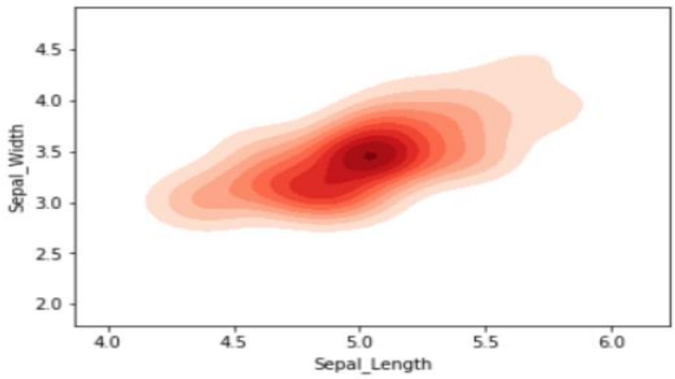
KDE Plot

Kernel Density Estimate (KDE) Plot is a powerful tool for estimating the probability density function of continuous or non-parametric data. KDE plot is implemented through the kdeplot function in Seaborn.



```
# Setting up the samples
iris_setosa = iris_df.query(&quot;Target=='Iris_Setosa'&quot;)
iris_virginica = iris_df.query(&quot;Target=='Iris_Virginica'&quot;)

# Plotting the KDE Plot
sns.kdeplot(iris_setosa['Sepal_Length'],
            iris_setosa['Sepal_Width'],
            color='r', shade=True, label='Iris_Setosa',
            cmap=&quot;Reds&quot;, shade_lowest=False)
```



pd - To implement pandas
sns - To implement seaborn
kdeplot()-Takes parameters x, y, data to plot with hue and kind of plot.

x - Given the values of width.
y - Given the values of length.
df- Dataframe for the external data

Advantage: Easy to represent different types of plots.

Matplotlib v/s Seaborn

Matplotlib	**Seaborn**
Graphs with basic themes.	Graphs with advanced themes.
Uses long and complex syntax.	Short and simple syntax.
Compatible with Numpy and Pandas.	Only compatible with pandas.
Needed to unzip dataset.	Automatically unzip dataset.