Shri Ramdeobaba College of Engineering and Management, Nagpur

Department of Electronics Engineering

Digital Image Processing (ENT 355-3)

Name: Prajwal Pandurang Shette

Roll No: B1- 12

Experiment No: 02

Aim: Introduction of Python: Matplotlib and Python Modules- NumPy, Pandas, SciPy,

Matplotlib: Installation of Matplotlib

If you have Pytho and PIP already installed on a system, then installation of Matplotlib is very easy.

Install it using this command:

```
C:\Users\Your Name>pip install matplotlib
```

Pyplot

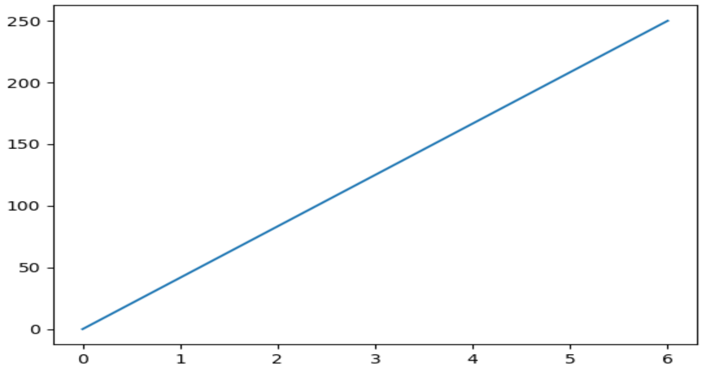
Most of the Matplotlib utilities lies under the pyplot submodule, and are usually imported under the `plt` alias:

```
import matplotlib.pyplot as plt
```

Ex:

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

xpoints = np.array([0, 6])
ypoints = np.array([0, 250])

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

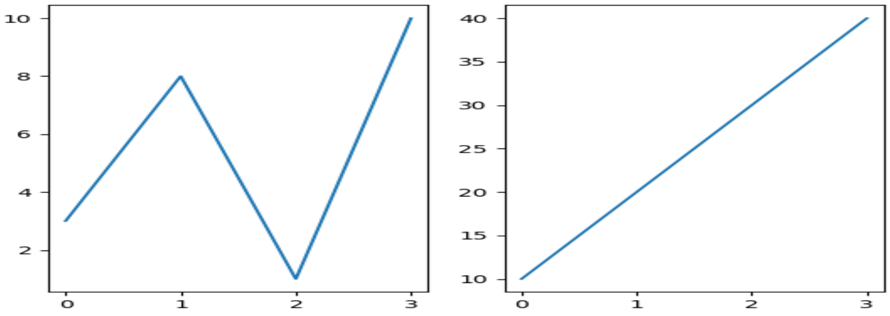
Output:



Ex: Matplotlib subplot

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

#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(1, 2, 1)
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)
plt.plot(x,y)

plt.show()
```

Output:



Ex: Matplotlib Scatter

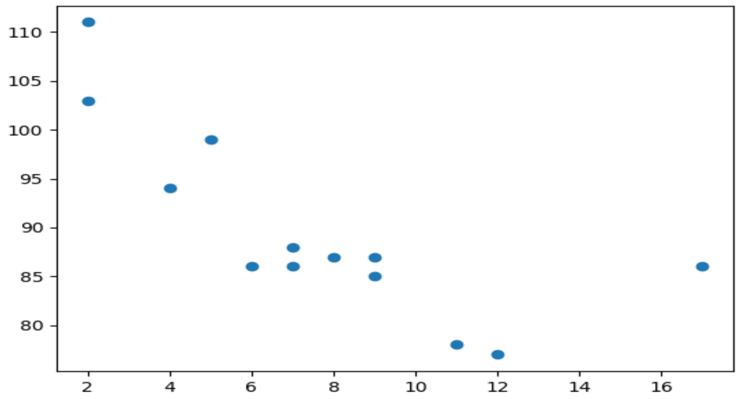
With Pyplot, you can use the `scatter()` function to draw a scatter plot.

The `scatter()` function plots one dot for each observation. It needs two arrays of the same length, one for the values of the x-axis, and one for values on the y-axis:

Input:

```
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()
```

Output:



NumPy: NumPy is a Python library used for working with arrays.It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical Python.

Why Use NumPy?

In Python we have lists that serve the purpose of arrays, but they are slow to process.

NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

The array object in NumPy is called `ndarray`, it provides a lot of supporting functions that make working with `ndarray` very easy.

Arrays are very frequently used in data science, where speed and resources are very important.

Install it using this command:

```
C:\Users\Your Name>pip install numpy
```

Once NumPy is installed, import it in your applications by adding the `import` keyword:

```
import numpy
```

Ex:

```
import numpy

arr = numpy.array([1, 2, 3, 4, 5])

print(arr)
```

Create a NumPy ndarray Object

NumPy is used to work with arrays. The array object in NumPy is called `ndarray`.

We can create a NumPy `ndarray` object by using the `array()` function.

Ex:

```
import numpy as np

arr = np.array([1, 2, 3, 4, 5])

print(arr)

print(type(arr))
```

Pandas: Pandas is a Python library used for working with data sets.

It has functions for analyzing, cleaning, exploring, and manipulating data.

The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.

Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science.

Install it using this command:

```
C:\Users\Your Name>pip install pandas
```

Once Pandas is installed, import it in your applications by adding the `import` keyword:

```
import pandas
```

Ex:

```
import pandas

mydataset = {
  'cars': ["BMW", "Volvo", "Ford"],
  'passings': [3, 7, 2]
}

myvar = pandas.DataFrame(mydataset)

print(myvar)
```

Read CSV Files

A simple way to store big data sets is to use CSV files (comma separated files).

CSV files contains plain text and is a well know format that can be read by everyone including Pandas.

In our examples we will be using a CSV file called 'data.csv'.

Download data.csv. or Open data.csv

Ex:

```
import pandas as pd

df = pd.read_csv('data.csv')

print(df.to_string())
```

Read JSON

Big data sets are often stored, or extracted as JSON.

JSON is plain text, but has the format of an object, and is well known in the world of programming, including Pandas.

In our examples we will be using a JSON file called 'data.json'.

Open data.json.

Ex:

```
import pandas as pd

df = pd.read_json('data.json')

print(df.to_string())
```

SciPy: SciPy is a scientific computation library thatuses NumPy underneath.

SciPy stands for Scientific Python. It provides more utility functions for optimization, stats and signal processing. Like NumPy, SciPy is open source so we can use it freely. SciPy was created by NumPy's creator Travis Olliphant.

Install it using this command:

```
C:\Users\Your Name>pip install scipy
```

Once SciPy is installed, import the SciPy module(s) you want to use in your applications by adding the `from scipy import module` statement:

```
from scipy import constants
```

Ex:
```
from scipy import constants

    print(constants.liter)
```

Ex:
```
from scipy.optimize import root
from math import cos

def eqn(x):
  return x + cos(x)

myroot = root(eqn, 0)

print(myroot.x)
```