

Day1- data visualization-letsupgrade - Laxmi N

July 5, 2021

1 Data visualization

```
[3]: import numpy as np
import pandas as pd
import seaborn as sns
from numpy.random import randn, randint, uniform, sample

import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline
```

```
[2]: print ('Matplotlib version: ', mpl.__version__ ) # >= 2.0.0
```

Matplotlib version: 3.3.2

2 Line Plot

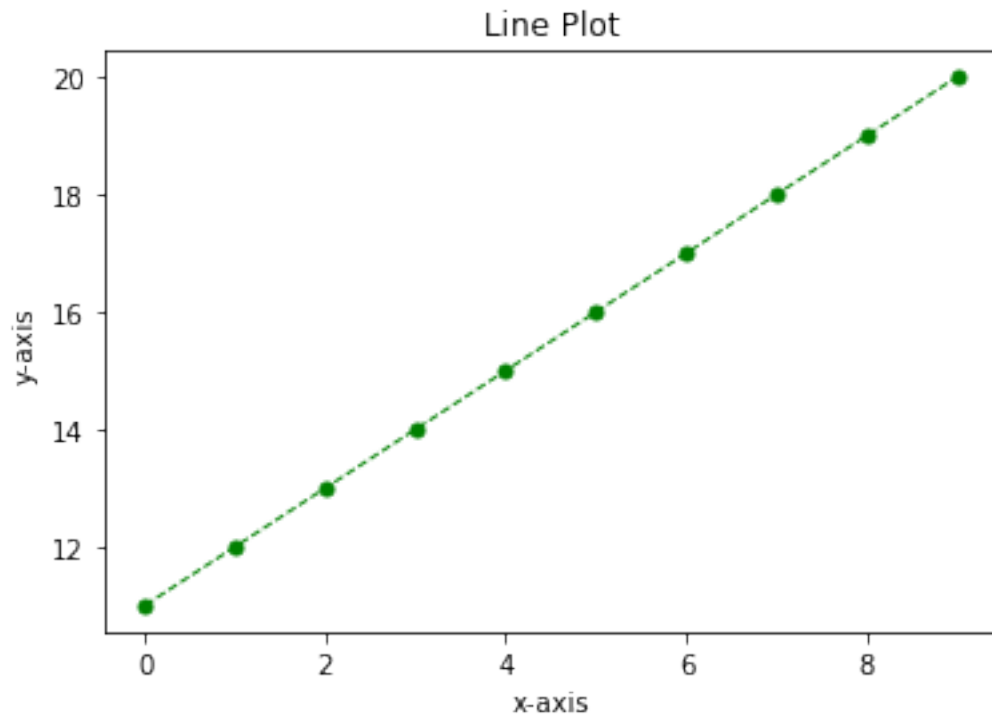
*Displays information as a series of datapoints called markers, connected by straight line segments

```
[4]: ## Simple Examples
```

```
x=np.arange(0,10)
y=np.arange(11,21)
```

```
[6]: plt.plot(x,y,'go--', linewidth=1, markersize=5)
plt.title('Line Plot')
plt.xlabel('x-axis')
plt.ylabel('y-axis')
```

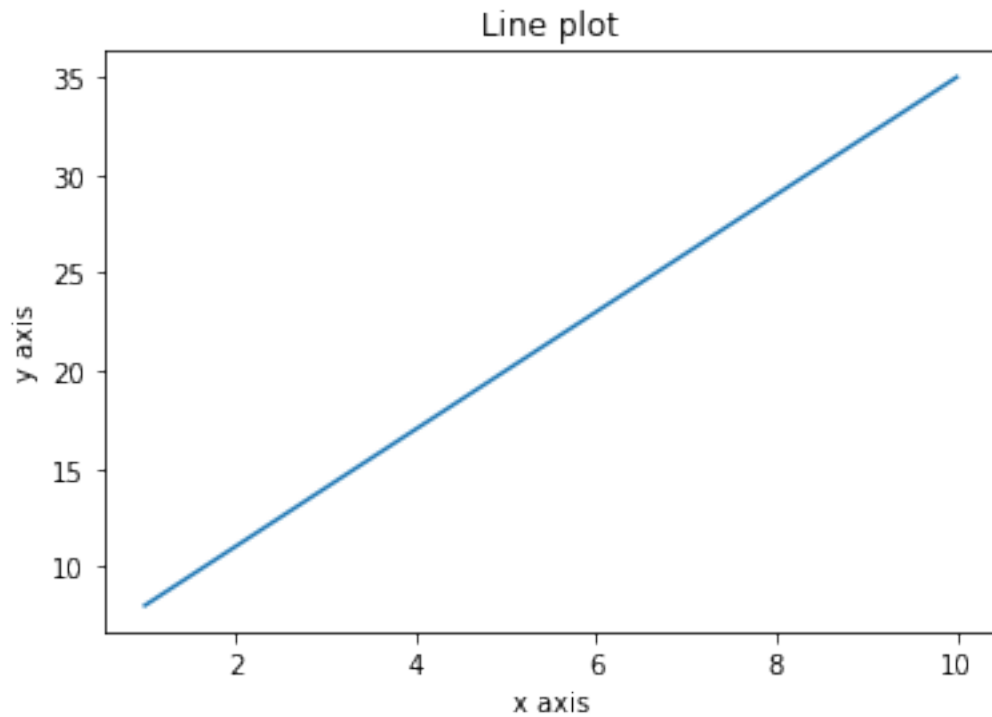
```
[6]: Text(0, 0.5, 'y-axis')
```



```
[7]: x = np.arange(1,11)
      y = 3 * x + 5
```

```
[9]: x = np.arange(1,11)
      y = 3 * x + 5
      plt.title("Line plot")
      plt.xlabel("x axis")
      plt.ylabel("y axis")
      plt.plot(x,y)
```

```
[9]: [<matplotlib.lines.Line2D at 0x1a53c237250>]
```

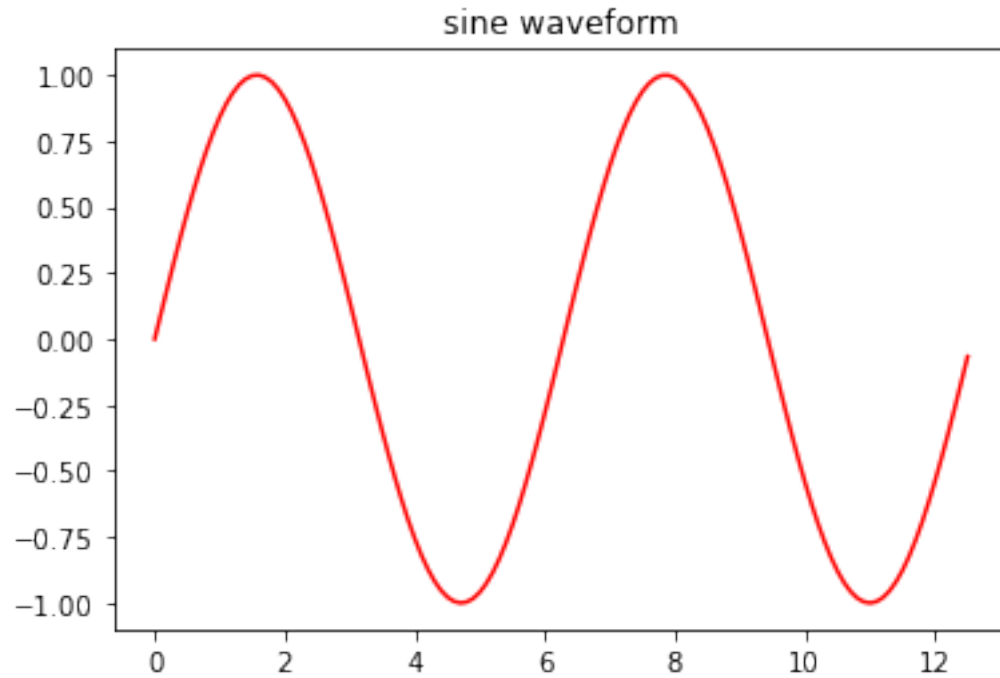


```
[10]: np.pi
```

```
[10]: 3.141592653589793
```

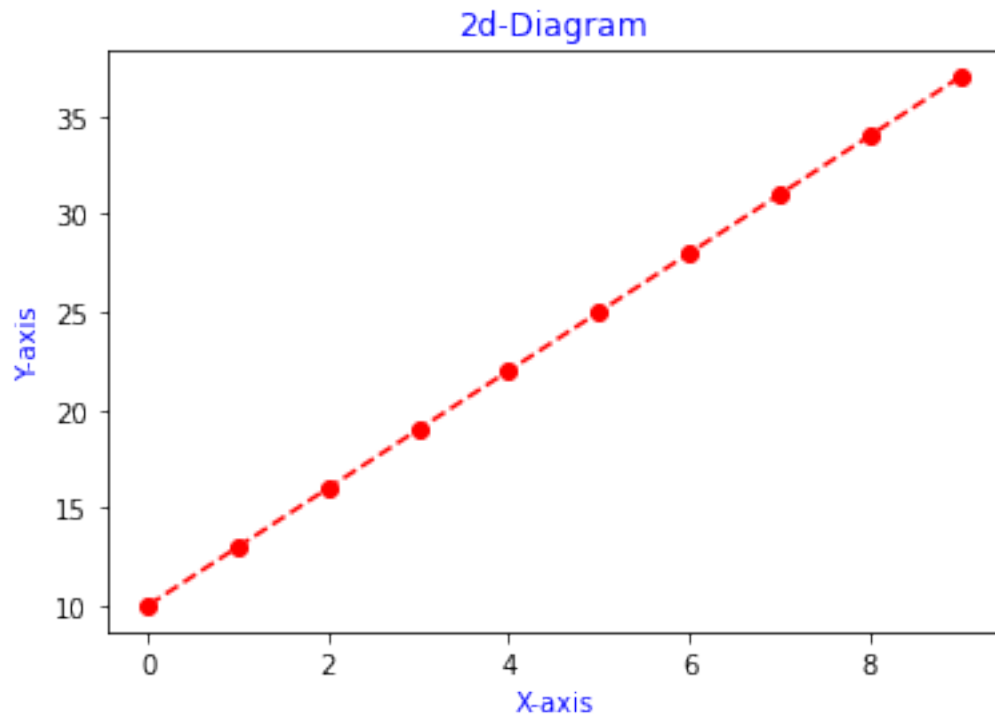
```
[11]: # Compute the x and y coordinates for points on a sine curve
x = np.arange(0, 4 * np.pi, 0.1)
y = np.sin(x)
plt.title("sine waveform")

# Plot the points using matplotlib
plt.plot(x, y, color='r')
plt.show()
```



```
[14]: x=np.arange(0,10)
      y= 3*x + 10
      ## plt plot
      plt.plot(x,y, 'ro--')
      #plt.plot(x,y,'r*',linestyle='dashed',linewidth=2, markersize=12)
      plt.xlabel('X-axis', color='blue')
      plt.ylabel('Y-axis', color='blue')
      plt.title('2d-Diagram', color='blue')
```

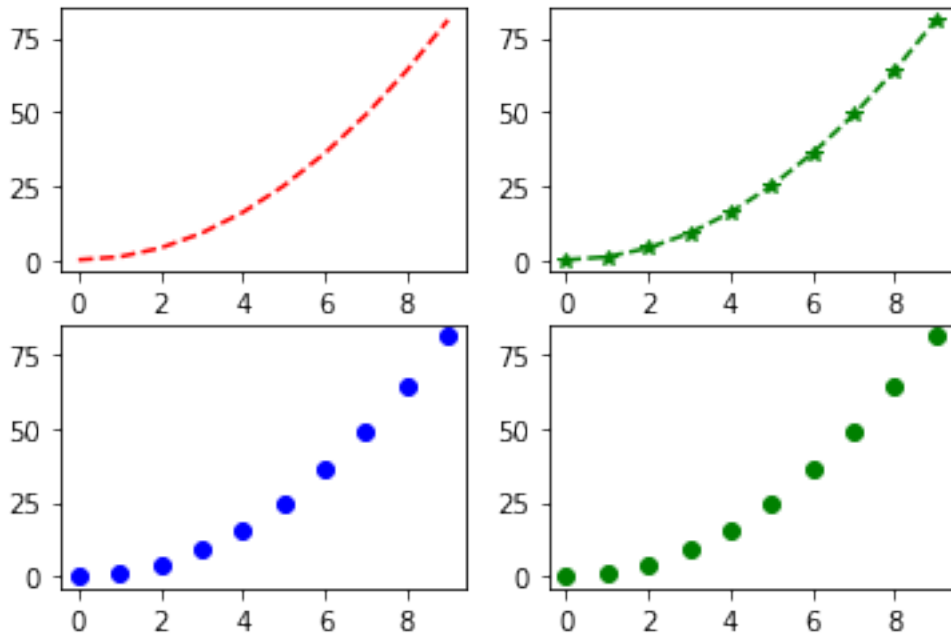
```
[14]: Text(0.5, 1.0, '2d-Diagram')
```



[17]: *## Creating Subplots*

```
plt.subplot(2,2,1)
plt.plot(x,y, 'r--')
plt.subplot(2,2,2)
plt.plot(x,y, 'g*--')
plt.subplot(2,2,3)
plt.plot(x,y, 'bo')
plt.subplot(2,2,4)
plt.plot(x,y, 'go')
```

[17]: [<matplotlib.lines.Line2D at 0x1a53c56a3a0>]



[18]: *# Plot a line plot for*

```
x=np.arange(0,10)
y=x*x
```

[20]: *##Practice exercise: plot a line plot between a and b:*

```
a=np.arange(40,50)
b=np.arange(50,60)
```

[21]: *# Plot a line plot showing the sales trend in company 1 and 2:*

```
days = [1,2,3,4,5,6,7] #days of d week
sales_1 = [160,150,140,145,175,165,180] #sales of company1
sales_2 = [70,90,160,150,140,145,175] #sales of company2
```

[22]: *# Create a 3 by 3 subplots:*

```
## multiple plots

x = [1,2,3,4]
y1 = [4,3,2,1]
y2 = [10,20,30,40]
y3 = [40,30,20,10]
y4 = [1,2,1,2]
y5 = [40,70,90,70]
```

[]:

[]:

[]: