

Basics of Programming

L12: Data Visualization

Apr 2020

Dr. Ram P Rustagi
Professor, CSE Dept
KRP, KSGI
rprustagi@ksit.edu.in

Resources

- Visualization with python
 - <https://matplotlib.org>
- Data Science from Scratch, 2nd edition
 - author: Joel Grus
 - publisher: O'Reilly, 2019

Visualization

- Data Visualization
 - A technique to see the data visually i.e.
 - Graphs, curve, scatter plots, images etc.
 - Famous saying
 - A picture is better than thousand words
 - Corollary:
 - A data plot is better than values in the list/tables etc.
- Example:
 - `python hanoi.py 4`

Basic 2D Plot

- A basic 2D plot
 - X-axis (data labels)
 - Y-axis (value, $f(x)$)
 - Graph type
 - Line, bar chart, pie chart, scatter plot (pts) etc
 - Python tool for plotting
 - matplotlib: A library to graph the data
- ```
import matplotlib.pyplot as plt
```
- `-plot()` function to plot the graph

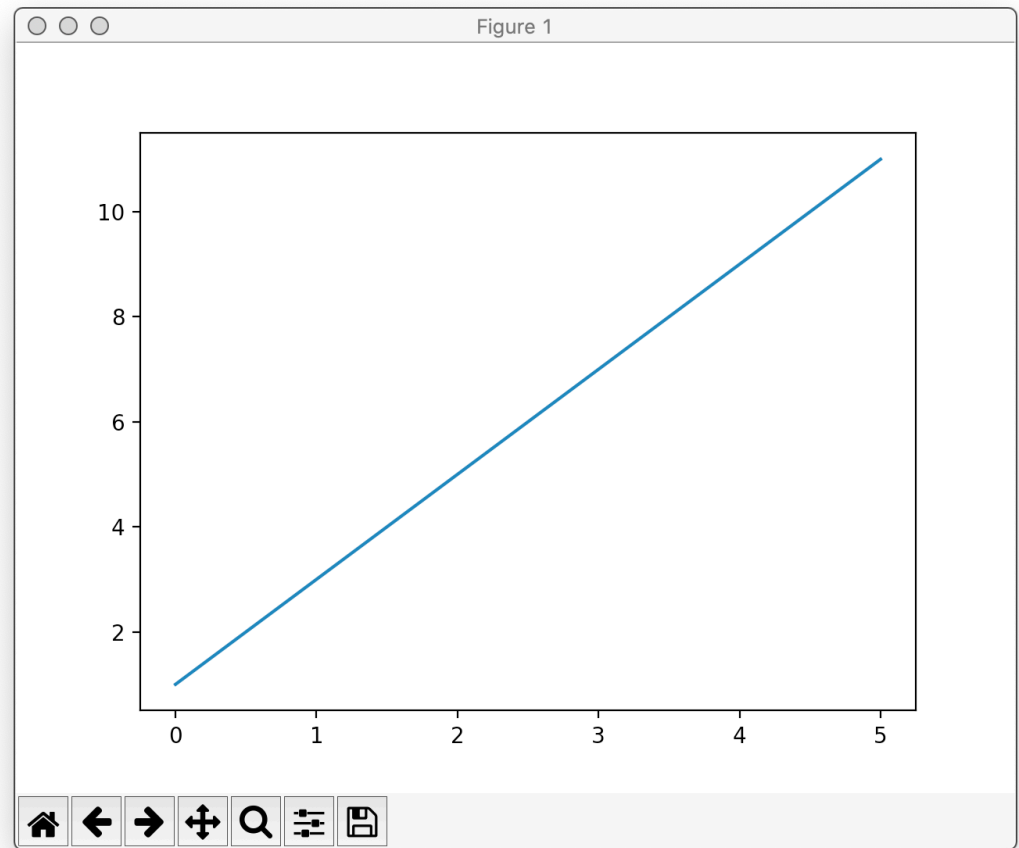
# Simple Examples

- Draw  $y=mx+c$  e.g.  $y=2x+1$

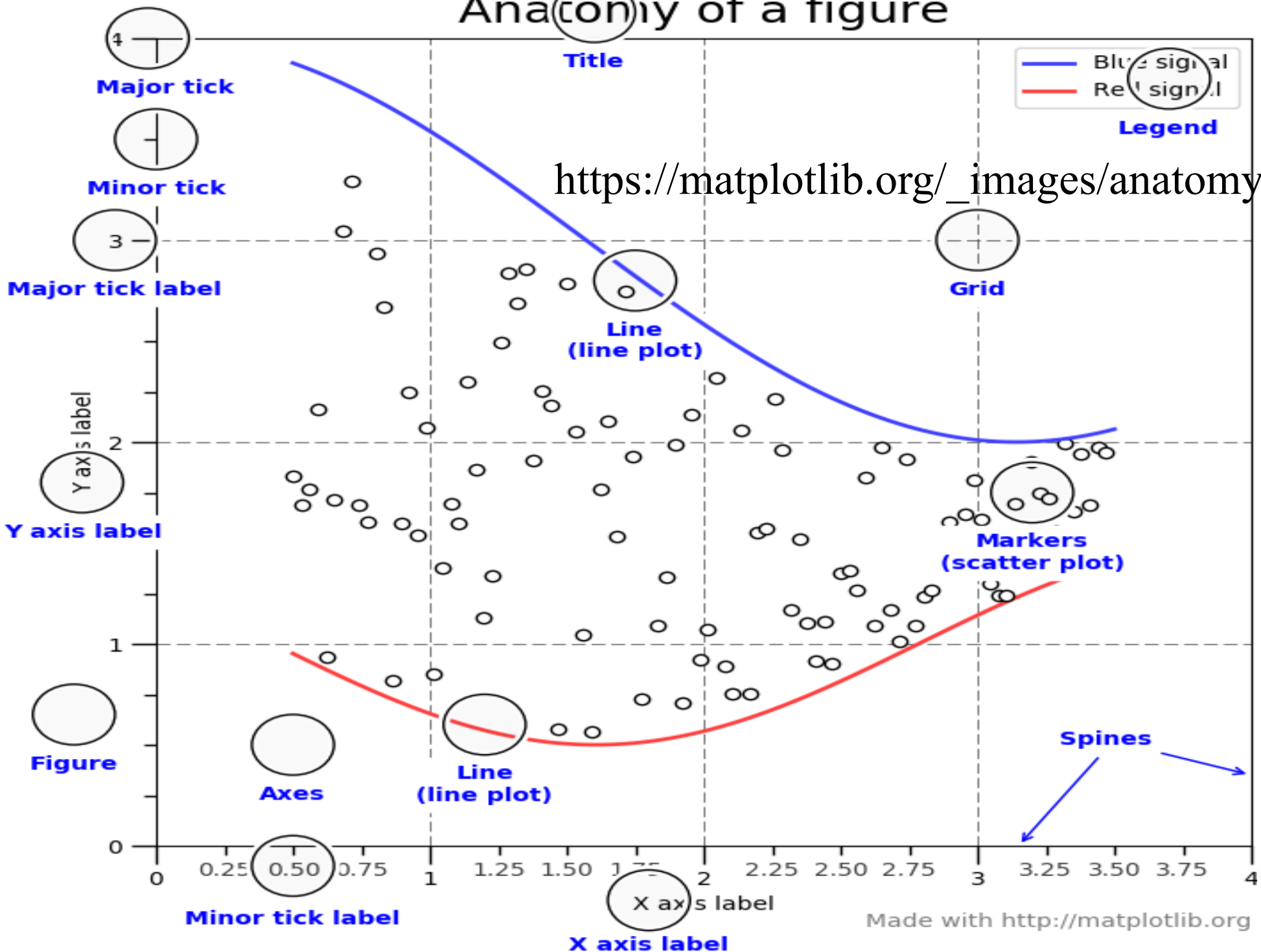
$x = [0, 1, 2, 3, 4, 5]$ ;  $y = [1, 3, 5, 7, 9, 11]$

- The simple program for this line

```
import matplotlib.pyplot as plt
xval=[i for i in range(6)]
yval=[2*i+1 for i in range(6)]
fig = plt.figure()
plt.plot(xval, yval)
plt.show()
```



# Anatomy of a figure

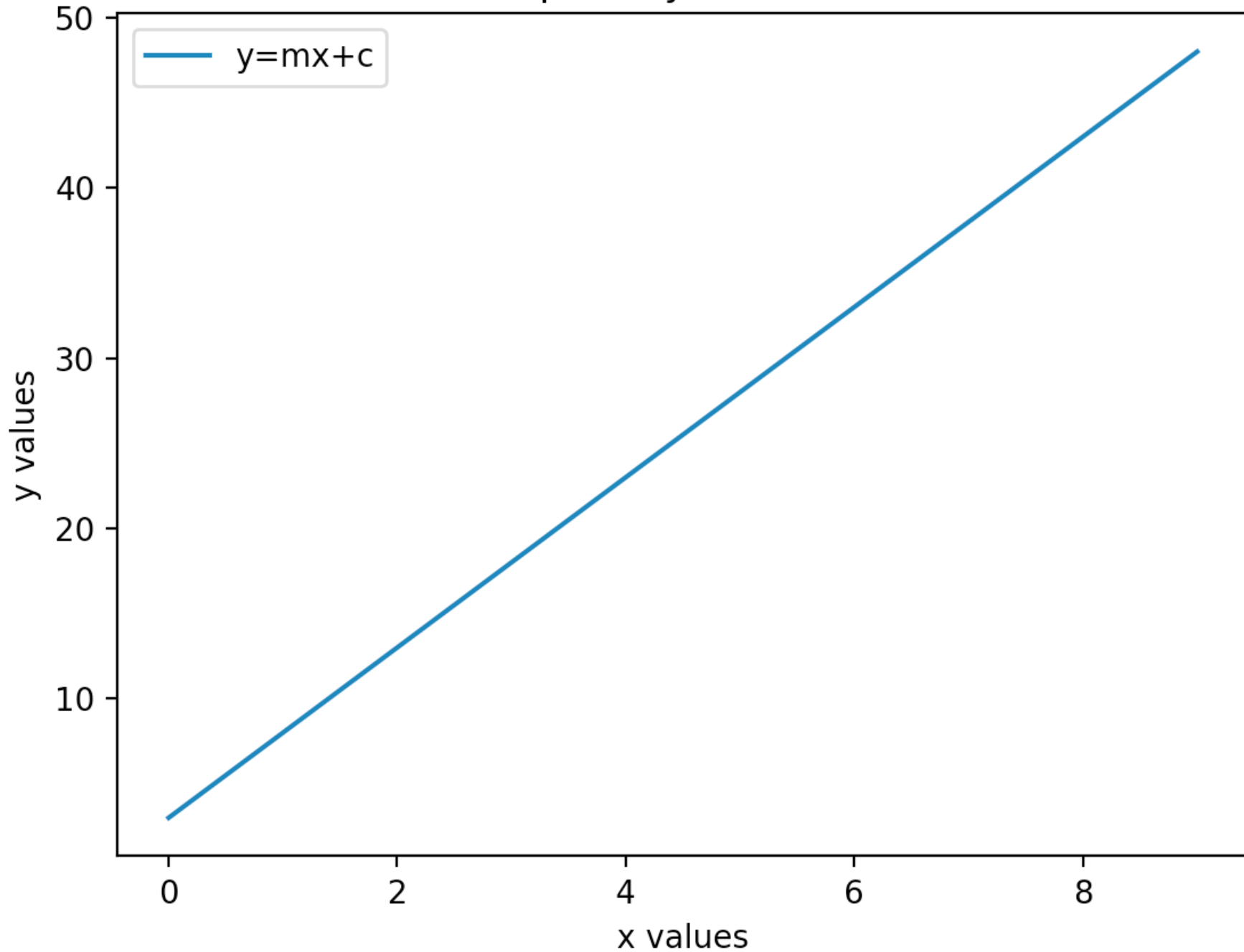


[https://matplotlib.org/\\_images/anatomy.png](https://matplotlib.org/_images/anatomy.png)

# Linear Eqn: $y=mx+c$

```
Get m,c and cnt as cmd arguments
xval=[i for i in range(cnt)]
yval=[m*i + c for i in range(cnt)]
plt.plot(xval, yval,label="y=mx+c")
plt.xlabel("x values")
plt.ylabel("y values")
plt.title("Graph for y=" + str(m) +
"x+" + str(c))
plt.legend()
plt.show()
```

Graph for  $y=5.0x+3.0$





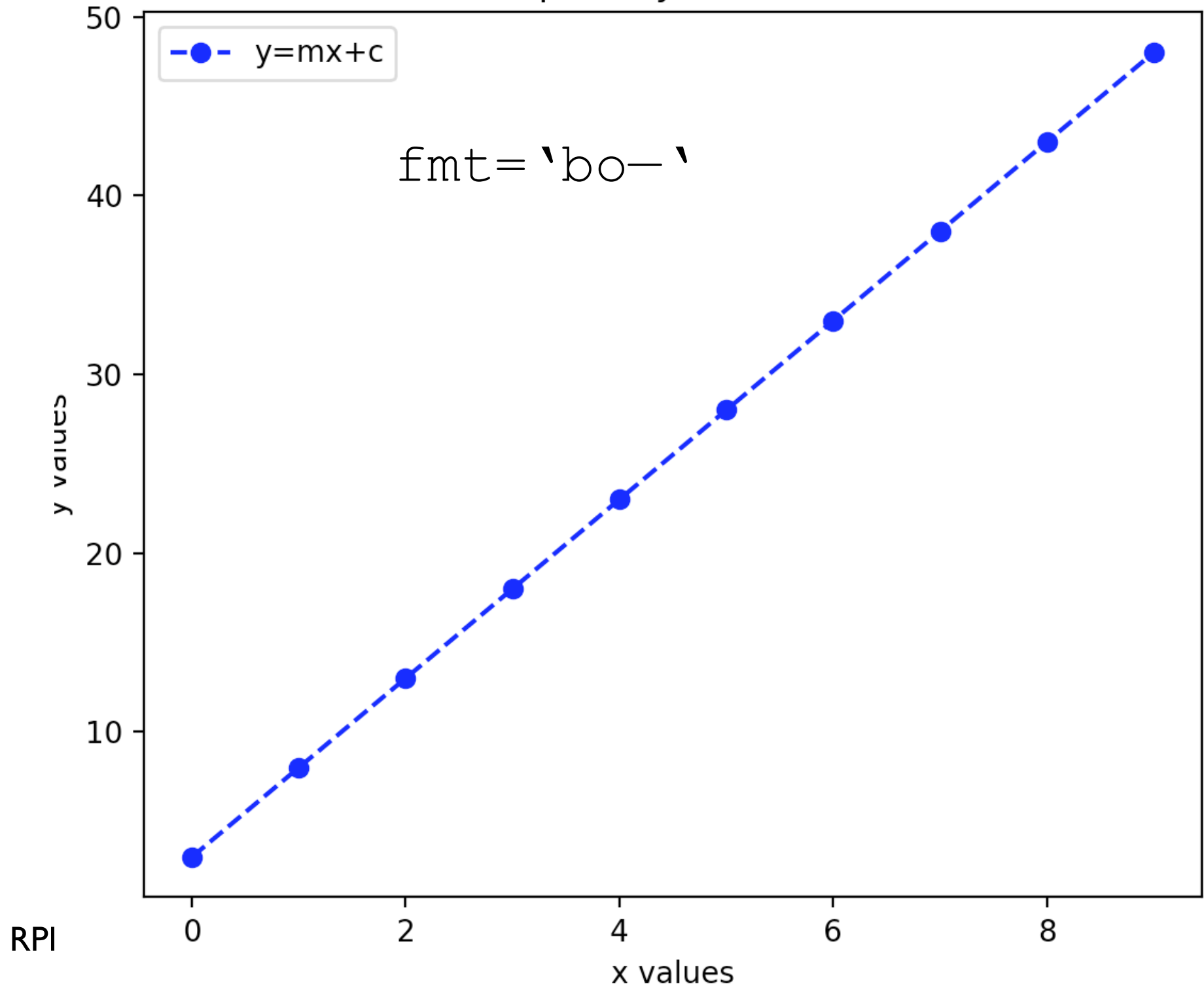
# Formatting Style of Plot

- documentation:
  - [https://matplotlib.org/api/as\\_gen/matplotlib.pyplot.plot.html#matplotlib.pyplot.plot](https://matplotlib.org/api/as_gen/matplotlib.pyplot.plot.html#matplotlib.pyplot.plot)
- `plt.axis(low_x, upper_x, lower_y, upper_y)`
  - Defines range of X and Y axis
- `plt.plot([xlist], ylist, [fmt])`
  - format option : [color][marker][line]
  - b- : solid blue line
  - bo: blue circles
  - ro: red circle
  - r+: red plusses

# Formatting Style of Plot

- colors
  - b: blue
  - g: green
  - r: red
  - c: cyan
  - m: magenta
  - y: yellow
  - k: black
  - w: white
- line styles
  - solid line
  - dashed line
  - . dash dot line
  - : dotted line
- markers
  - . point marker
  - , pixel marker
  - o circle marker
  - v triangle down
  - ^ triangle up
  - < triangle left
  - > triangle right
  - p pentagon marker
  - x x marker
  - D diamond
  - d thin diamond
  - \* star marker
  - + plus marker

Graph for  $y=5.0x+3.0$



# Multiline plots

- `plot(x,y1,fmt1, x,y2,fmt2, x,y3,fmt3)`

- **Example**

```
import matplotlib.pyplot as plt
x=[i for i in range(10)]
y1=[i for i in range(10)]
y2=[i*i for i in range(10)]
y3=[i*i*i for i in range(10)]
plt.plot(x,y1,"b-", x,y2,"r--.", x,y3,"gs")
plt.show()
```

# Trigonometric Curves

- $\sin\theta$  curve from 0 to  $2\pi$

```
import sys
import math
import matplotlib.pyplot as plt

cnt=100
xmax = 2*math.pi
x=[xmax*i/cnt for i in range(cnt)]
y0=[0 for i in range(cnt)] # base line
y1=[math.sin(xmax*i/cnt) for i in
range(cnt)]

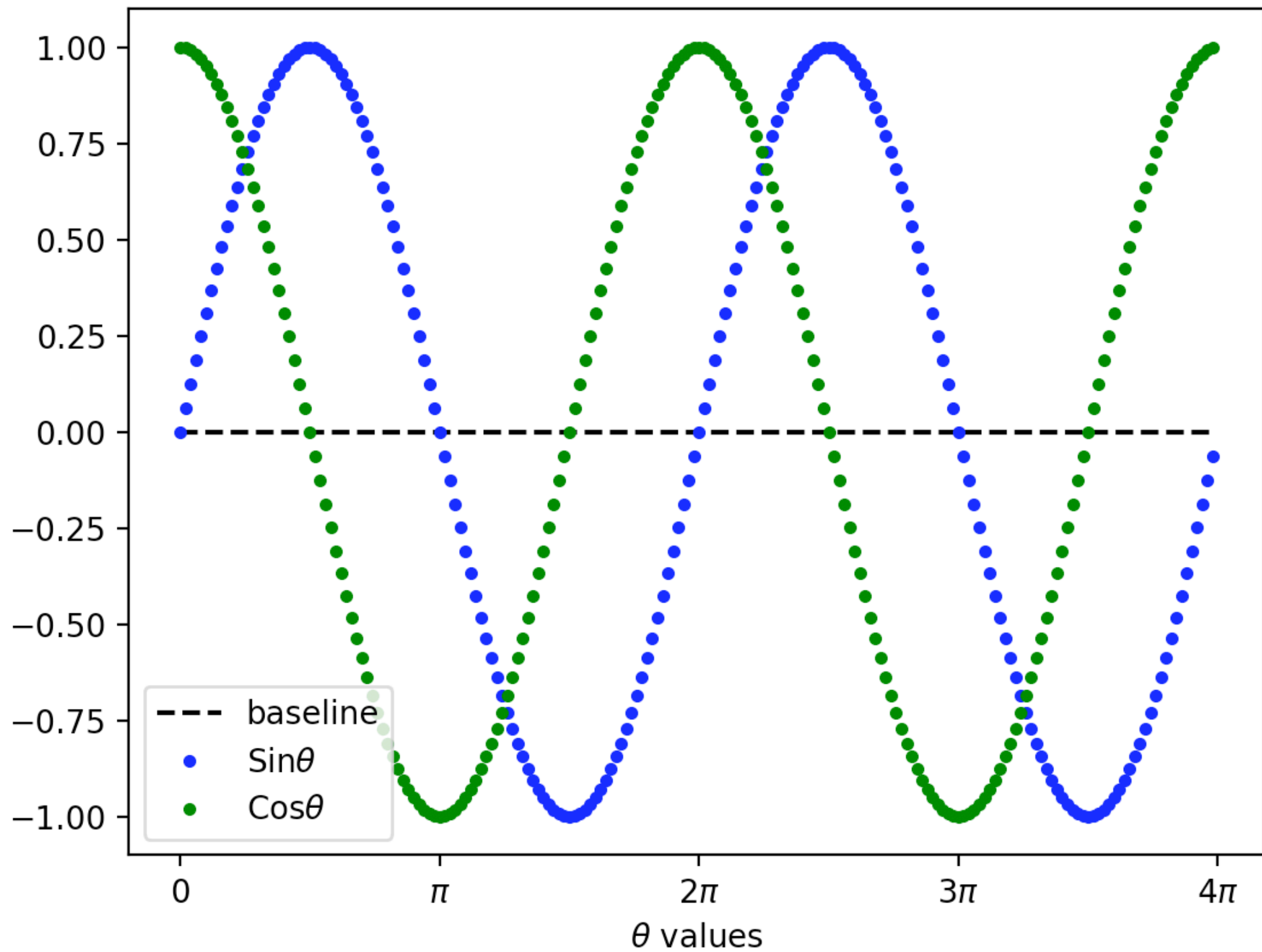
plt.plot(x, y0, 'k--', x, y1, 'b.')
plt.show()
```

# Trigonometric Curves

- $\sin\theta$  and  $\cos\theta$  curve from 0 to  $4\pi$

```
xmax = 4*math.pi
x=[xmax*i/cnt for i in range(100)]
y0=[0 for i in range(100)]
y1=[math.sin(xmax*i/100) for i in range(100)]
y2=[math.cos(xmax*i/100) for i in range(100)]

plt.plot(x,y0,'k--', label="baseline")
plt.plot(x,y1,'b.', label="Sin" +r'θ')
plt.plot(x,y2,'g.', label="Cos" +r'θ')
plt.xlabel(r"θ values")
plt.ylabel("Sin and cos curves")
plt.xticks([i*math.pi for i in range(5)],
 [r'0',r'π',r'2π',r'3π',r'4π'])
plt.legend()
plt.show()
```



# Exercises

- Ex 01: Plot a curve of your percentage marks from class 8 to class 12
- Ex 02: Draw a graph for  $\sin^2\theta + \cos^2\theta$  and show that value is always 1.
- Ex 03: Draw a graph for last 7 days, number of whatsapp messages sent and received.



# Questions

