

In [33]: `#task 4/11/25`

MATPLOTLIB

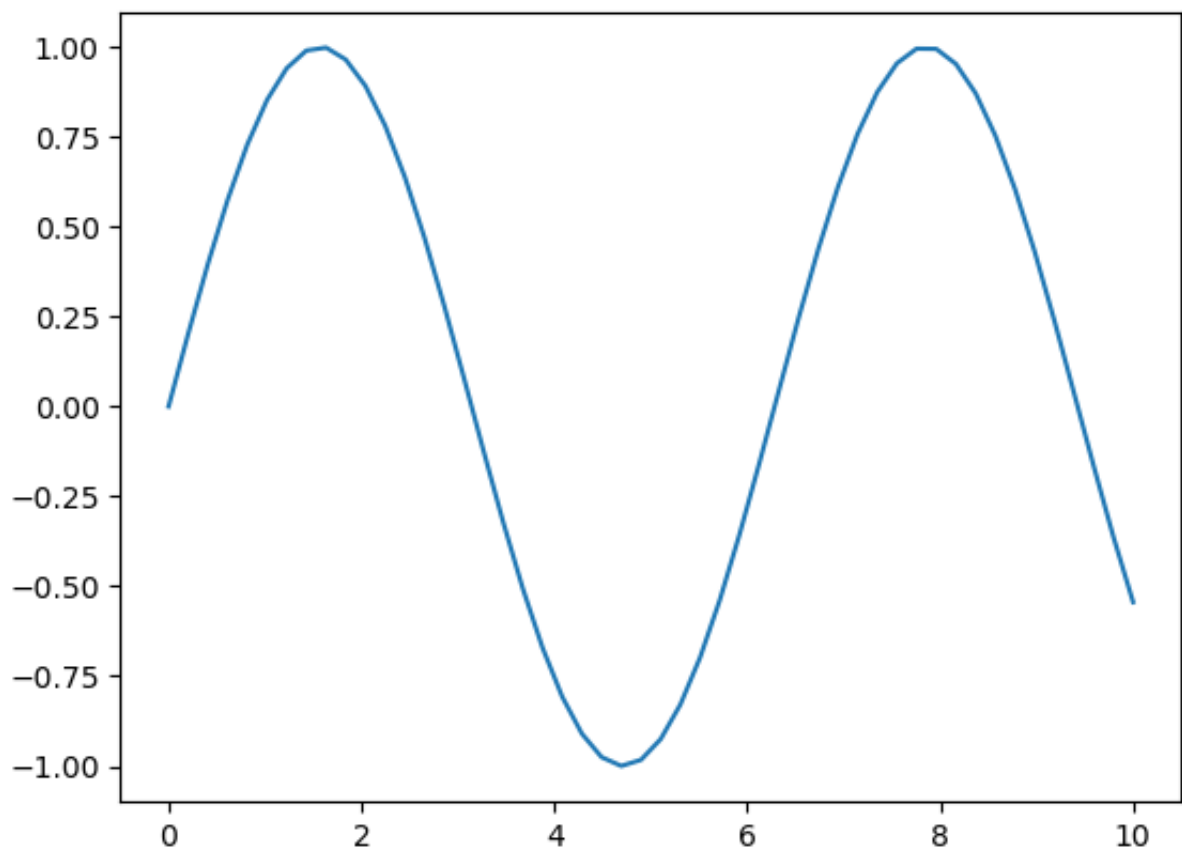
In [34]: `import matplotlib
import matplotlib.pyplot as plt`

In [35]: `import numpy as np
import pandas as pd`

In [36]: `%matplotlib inline
for interactive plot
x1 = np.linspace(0, 10, 50)

create a plot figure
#fig = plt.figure()

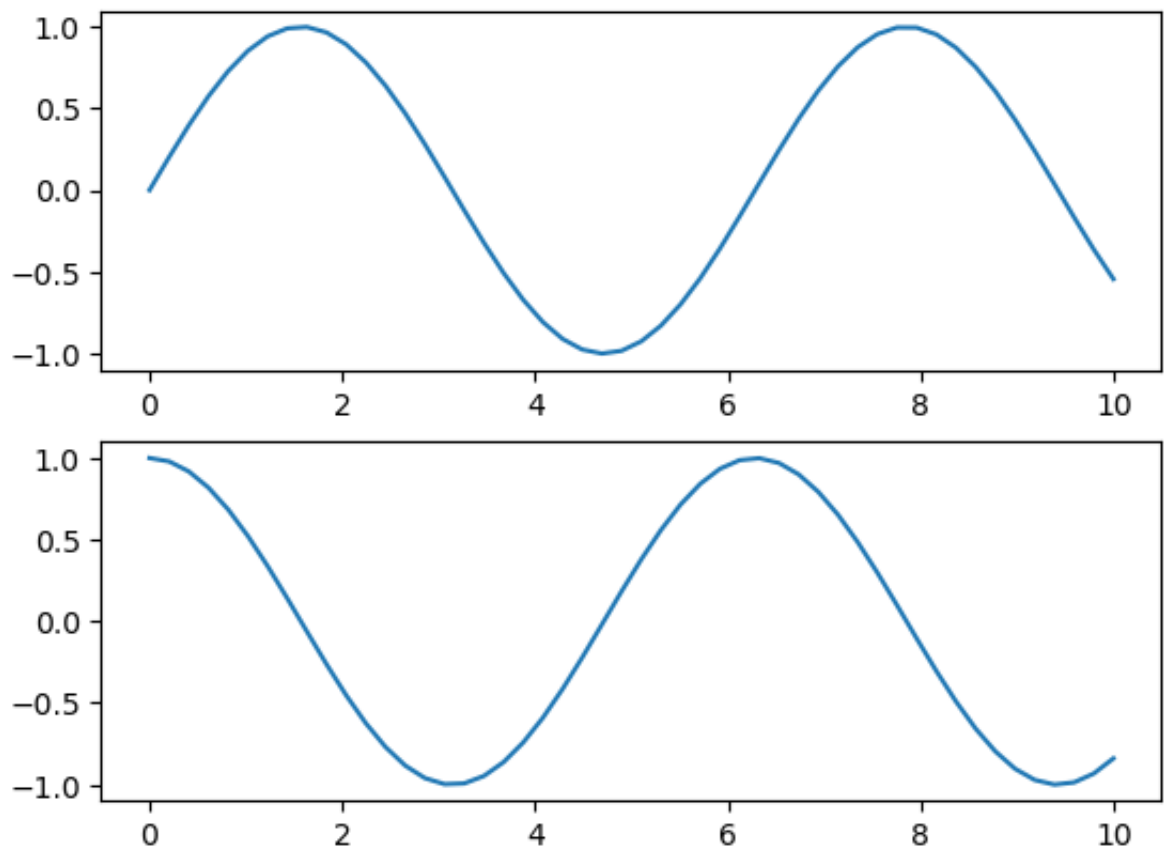
plt.plot(x1, np.sin(x1), '-r')
#plt.plot(x1, np.cos(x1), '--')
#plt.plot(x1, np.tan(x1), '--')
plt.show()`



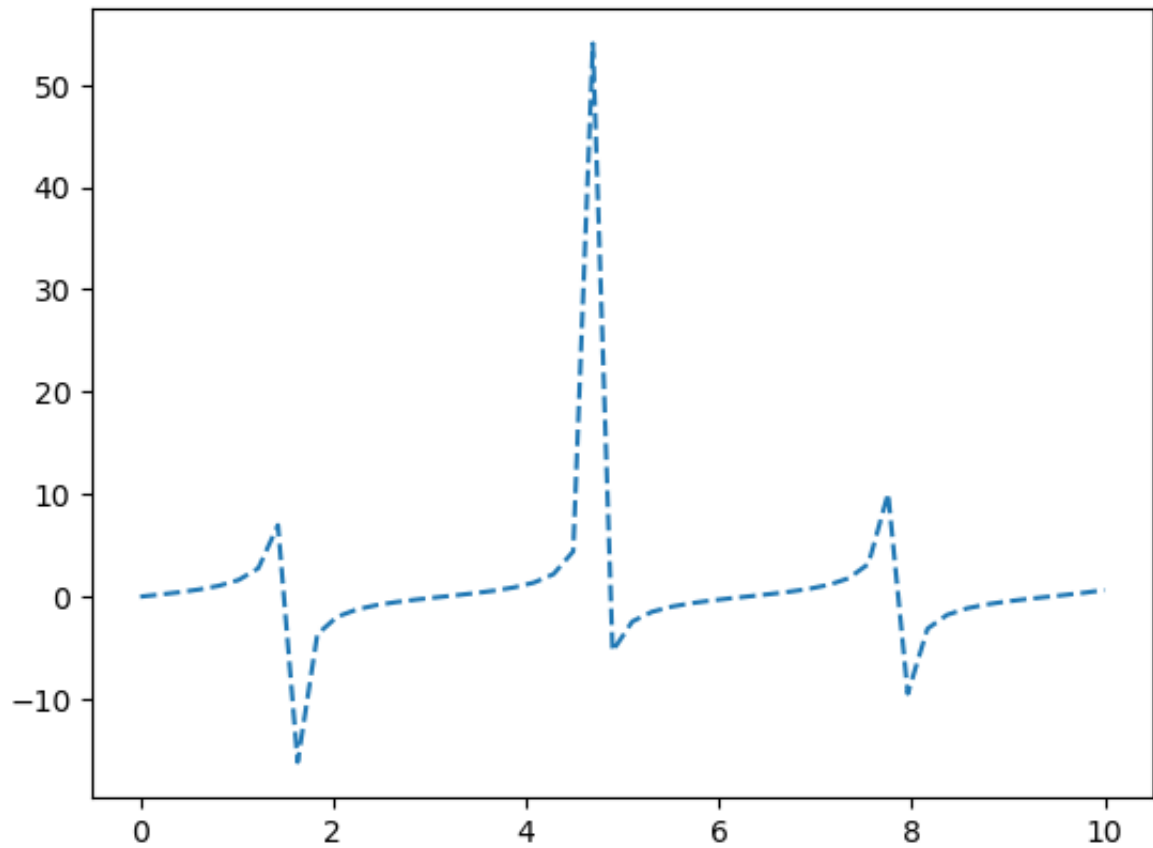
In [44]: `# create the first of two panels and set current axis
plt.subplot(2, 1, 1) # (rows, columns, panel number)
plt.plot(x1, np.sin(x1))`

```
# create the second of two panels and set current axis
plt.subplot(2, 1, 2) # (rows, columns, panel number)
plt.plot(x1, np.cos(x1));
```

In [45]: `plt.show()`



In [48]: `#plt.plot(x1, np.cos(x1), '--')`
`plt.plot(x1, np.tan(x1), '--')`
`plt.show()`



```
In [49]: print(plt.gcf ( ) )
```

Figure(640x480)

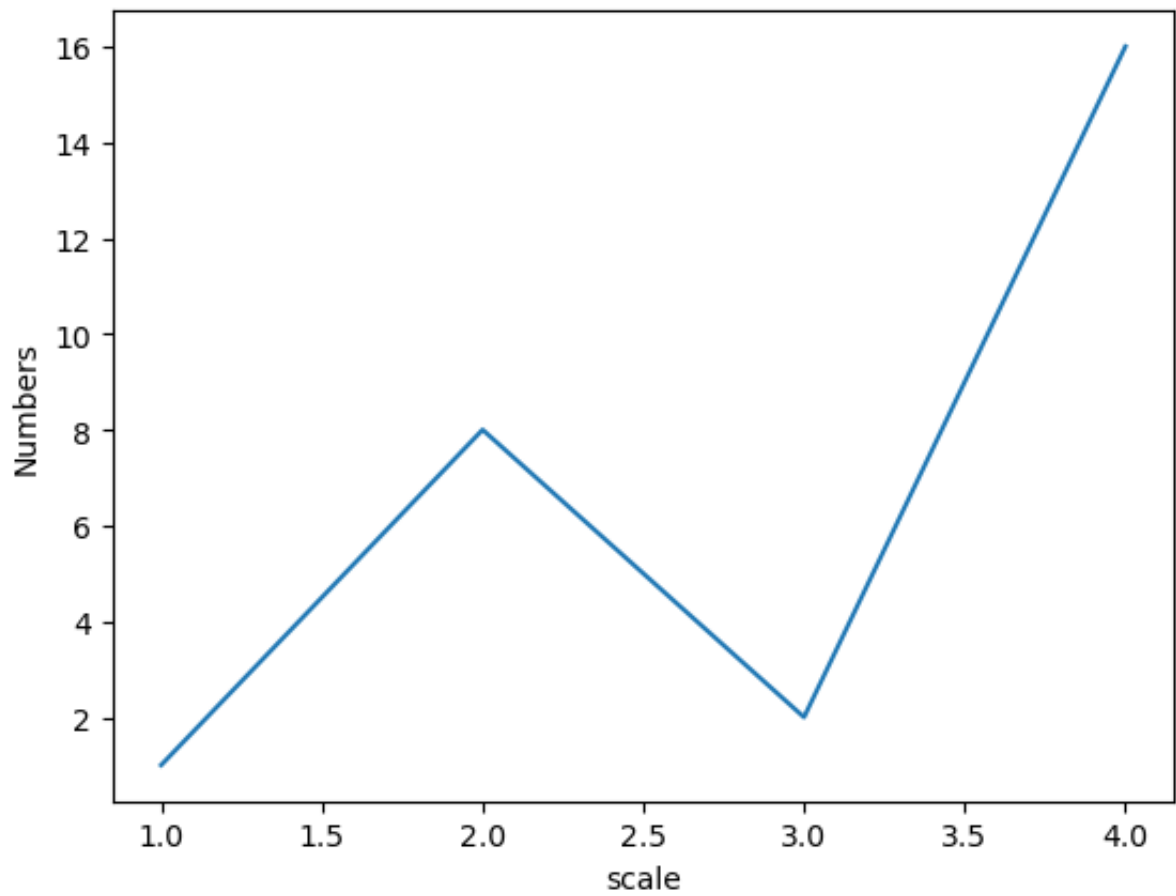
```
In [51]: print(plt.gca ( ) )
```

Axes(0.125,0.11;0.775x0.77)

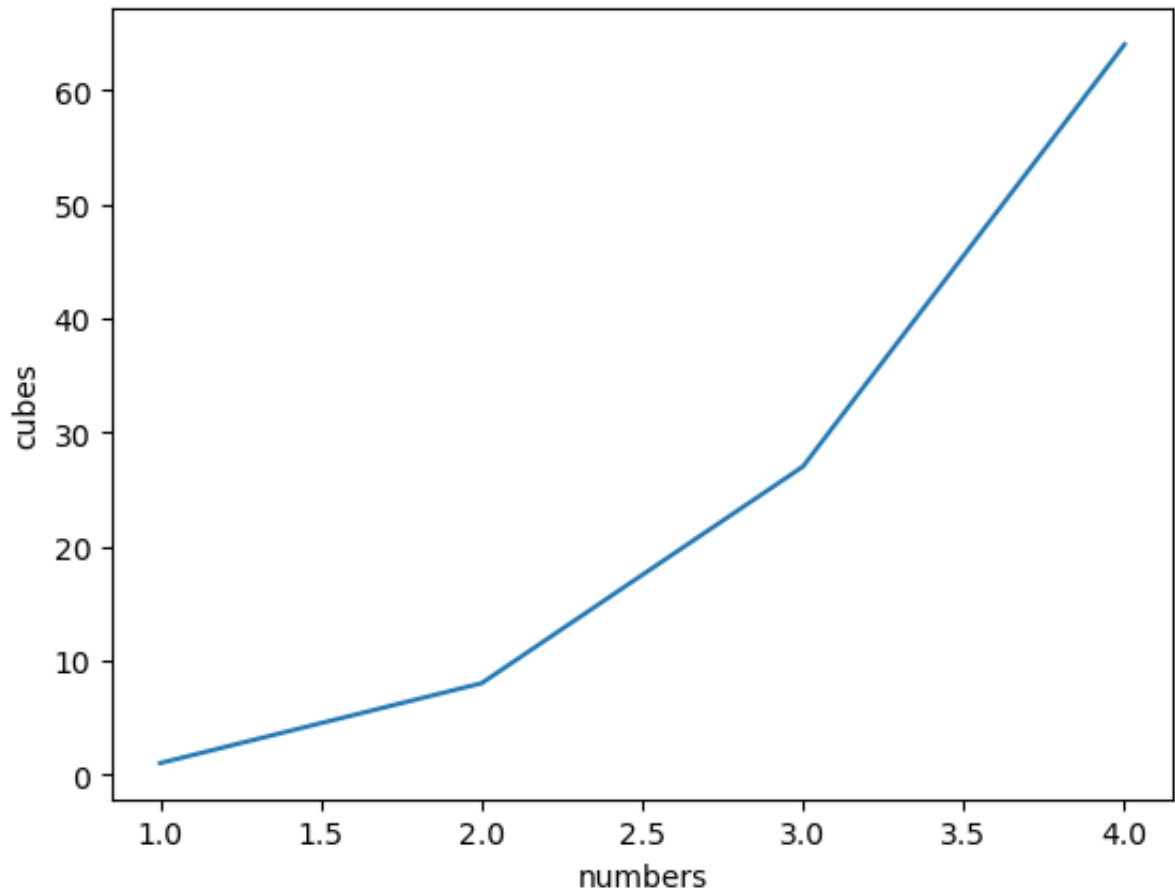
```
In [52]: #visualization with pyplot
```

visualization with pyplot

```
In [54]: plt.plot([1,2,3,4], [1,8,2,16])  
plt.ylabel('Numbers')  
plt.xlabel('scale')  
plt.show()
```



```
In [57]: plt.plot([1,2,3,4], [1,8,27,64])  
plt.ylabel('cubes')  
plt.xlabel('numbers')  
plt.show()
```



```
In [58]: x = np.linspace(0, 2, 100)

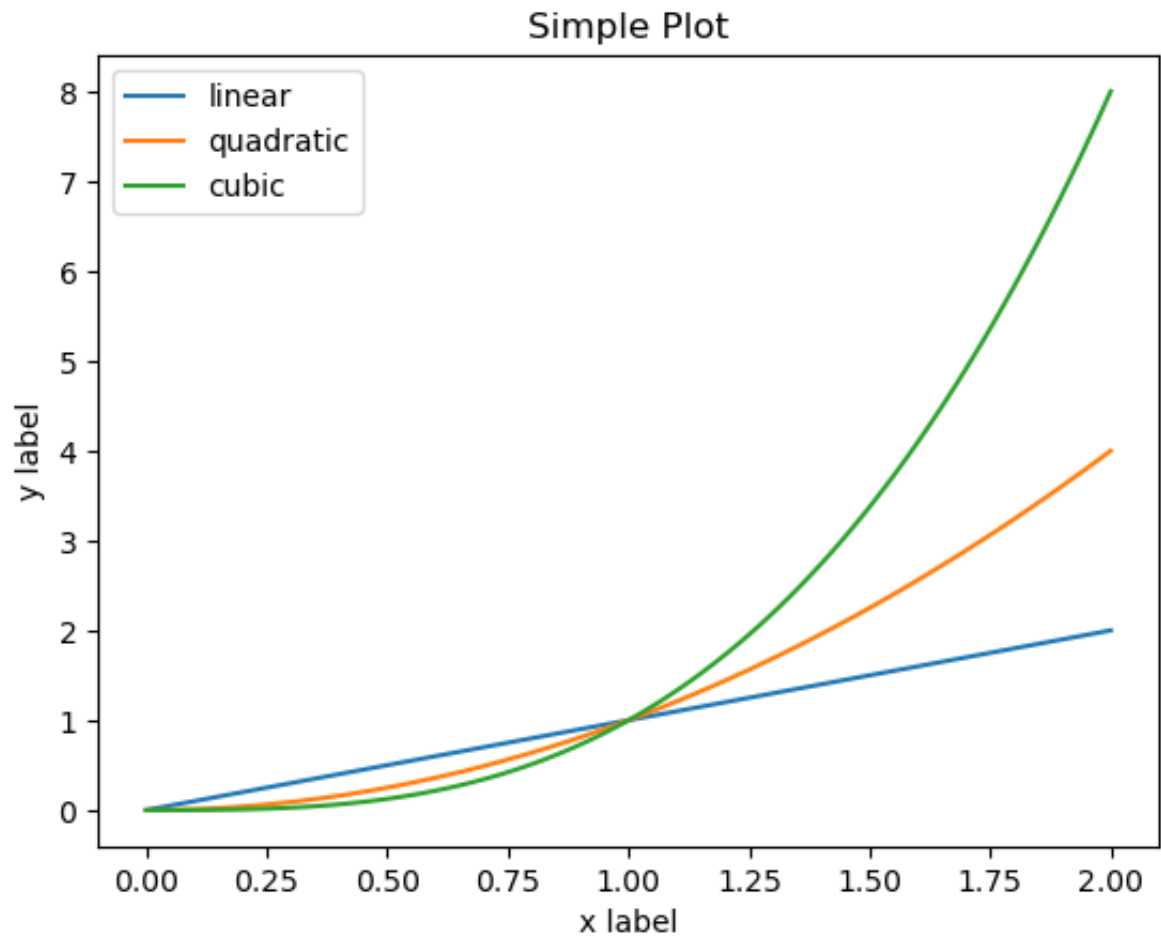
plt.plot(x, x, label='linear')
plt.plot(x, x**2, label='quadratic')
plt.plot(x, x**3, label='cubic')

plt.xlabel('x label')
plt.ylabel('y label')

plt.title("Simple Plot")

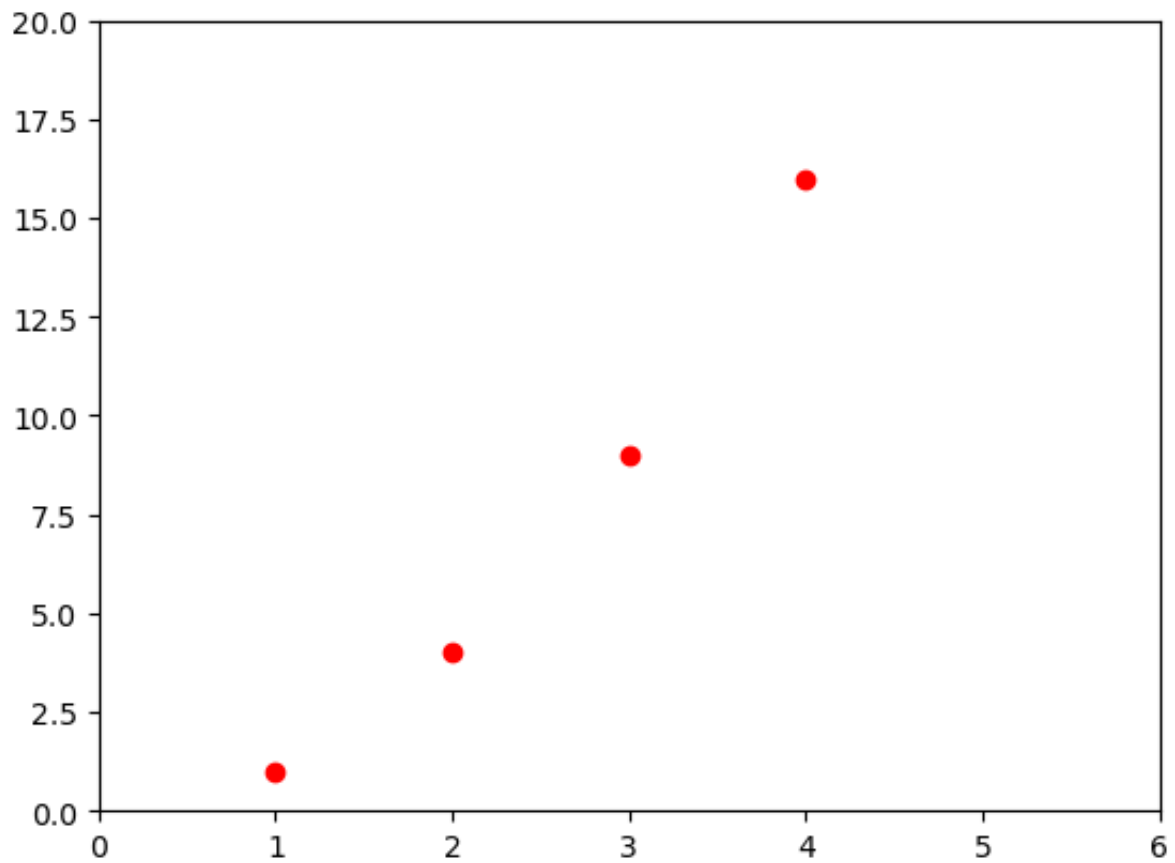
plt.legend()

plt.show()
```



```
In [59]: # forming the style of plot

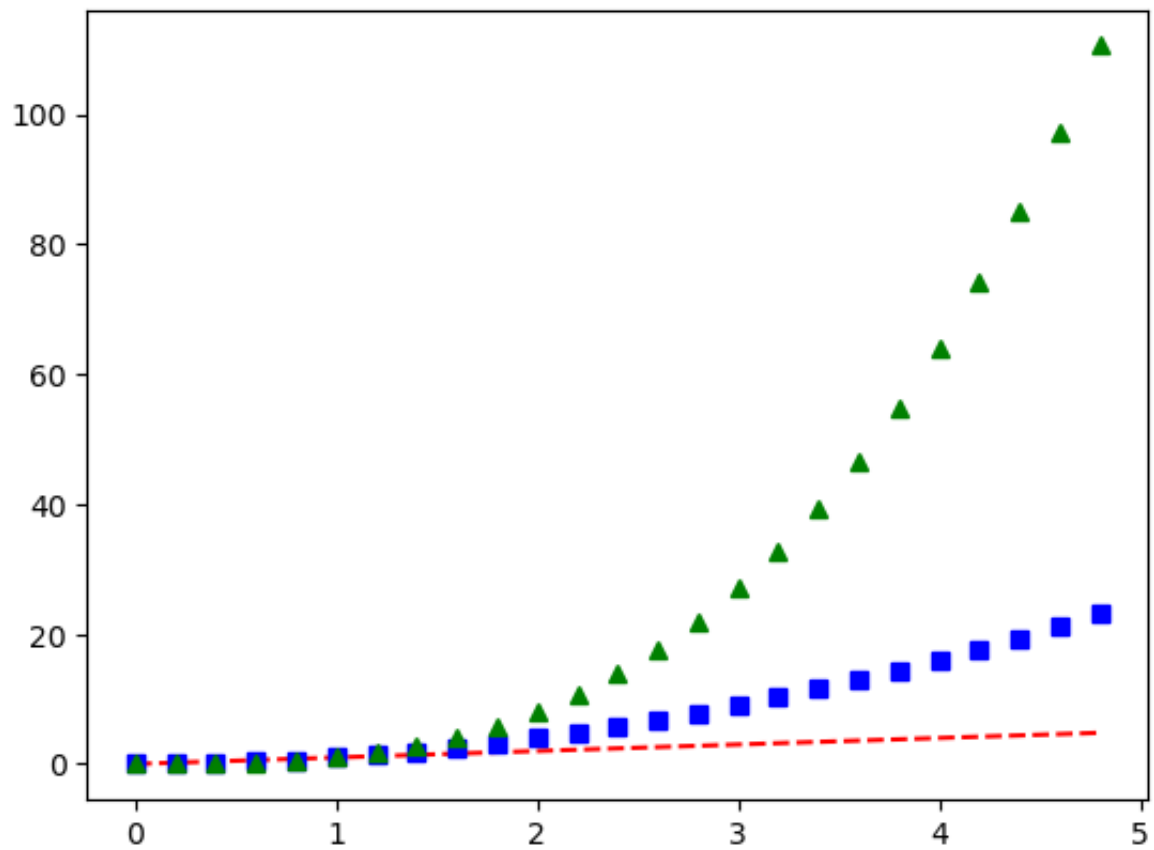
plt.plot([1, 2, 3, 4], [1, 4, 9, 16], 'ro')
plt.axis([0, 6, 0, 20])
plt.show()
```



numpy arrays

```
In [60]: # evenly sampled time at 200ms intervals
t = np.arange(0., 5., 0.2)

# red dashes, blue squares and green triangles
plt.plot(t, t, 'r--', t, t**2, 'bs', t, t**3, 'g^')
plt.show()
```



```
In [65]: #First create a grid of plots  
# ax will be an array of two Axes objects  
fig, ax = plt.subplots(2)  
  
# Call plot() method on the appropriate object  
ax[0].plot(x1, np.sin(x1), 'b-')  
ax[1].plot(x1, np.cos(x1), 'b-');
```

```
In [ ]:
```

```
In [ ]:
```