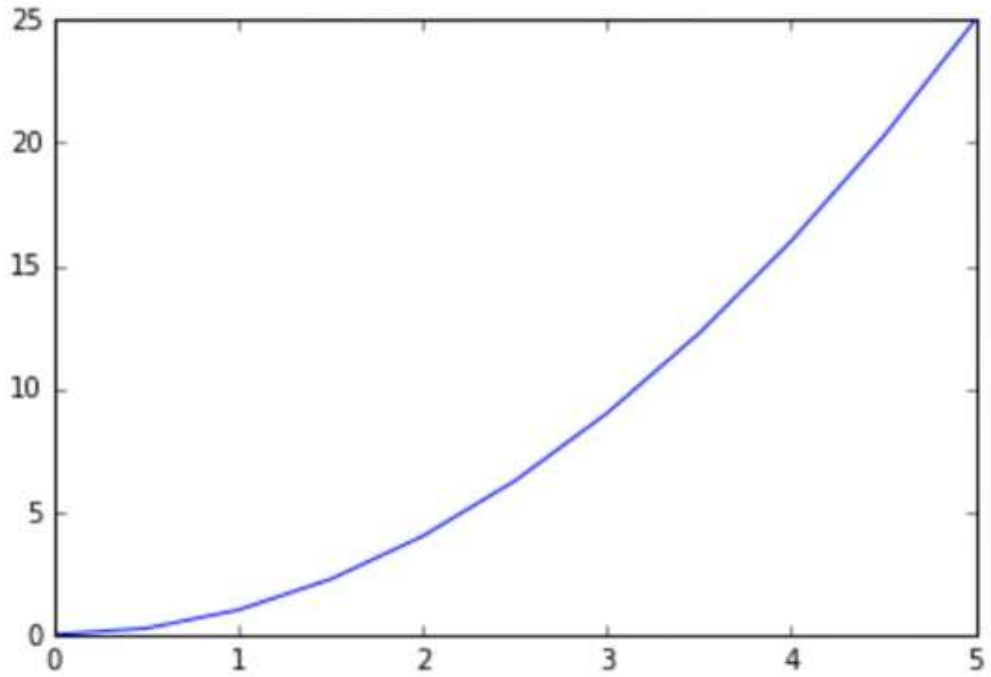
Python for Data Science

Matplotlib - Part 2

PIERIAN DATA
www.pieriandata.com

In [32]:
```
fig = plt.figure()
axes1 = fig.add_axes([0.1,0.1,0.8,0.8])
axes1.plot(x,y)
```

Out[32]: [<matplotlib.lines.Line2D at 0x164afa68828>]

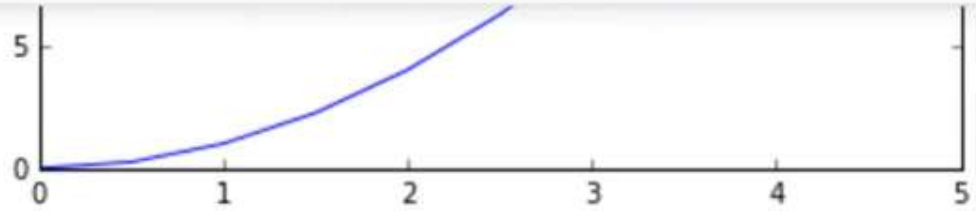


In [ ]:

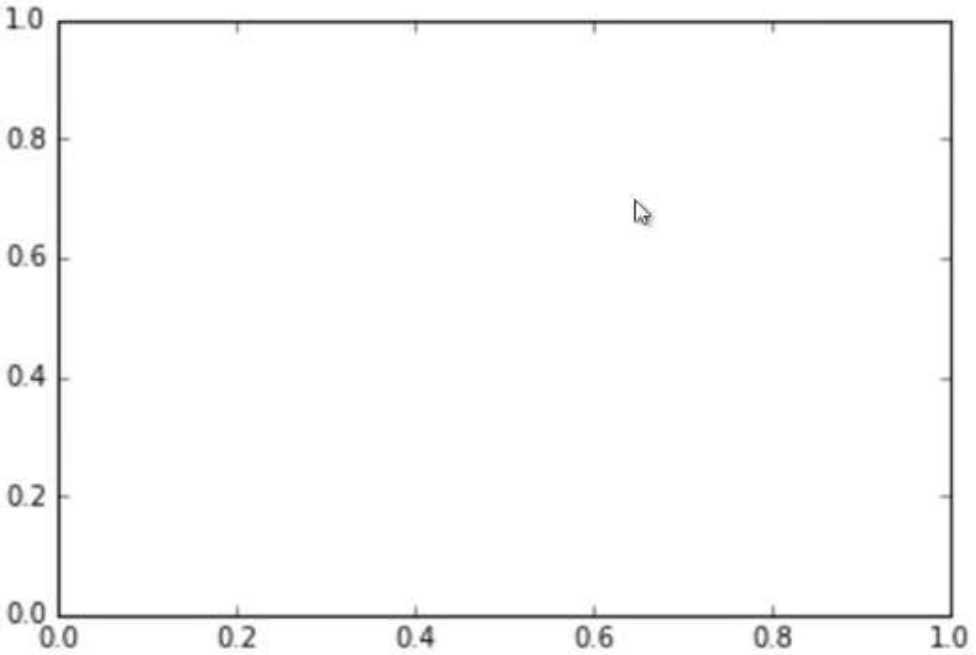
matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

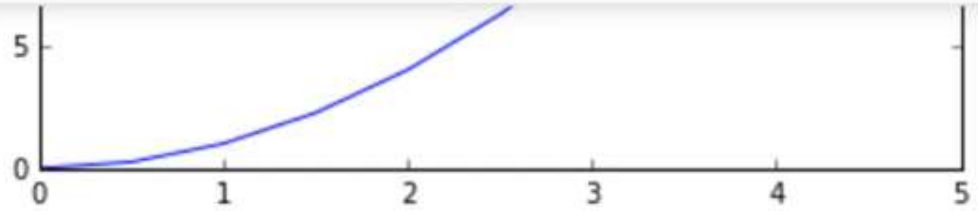


In [33]:
```
fig,axes = plt.subplots()
```



In [ ]:

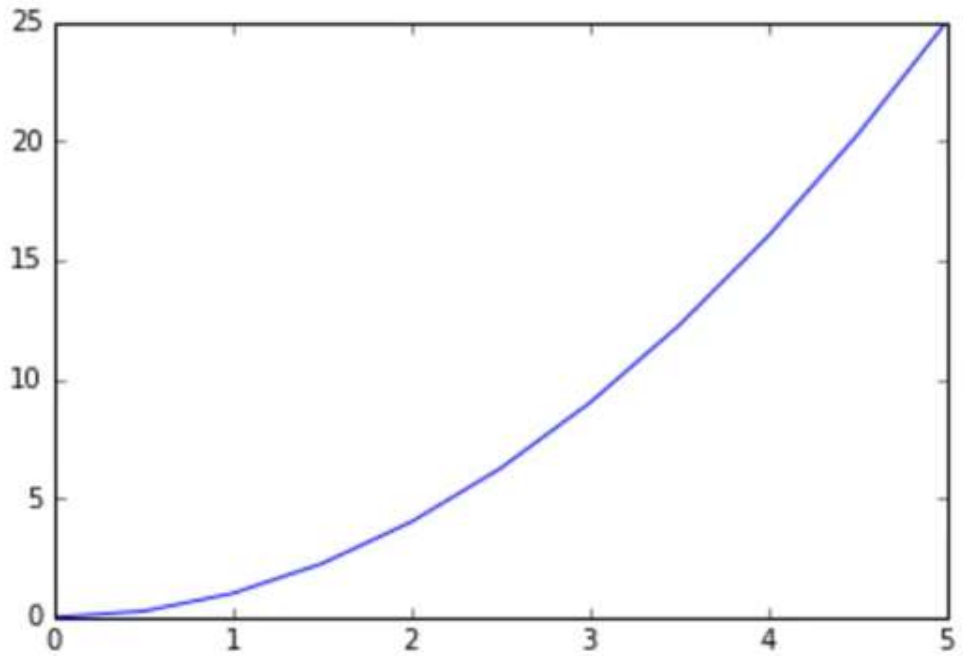
In [ ]:



In [34]:
```
fig,axes = plt.subplots()

axes.plot(x,y)
```

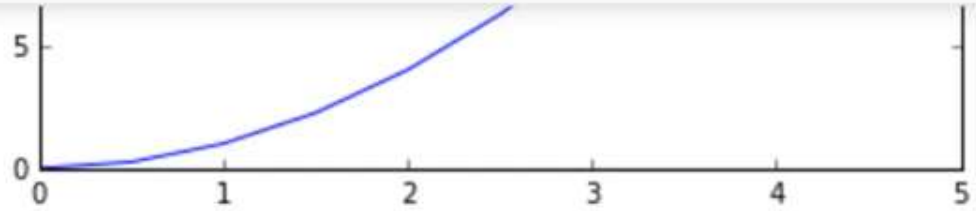
Out[34]:
```
[<matplotlib.lines.Line2D at 0x164ae9fdac8>]
```



matplotlib

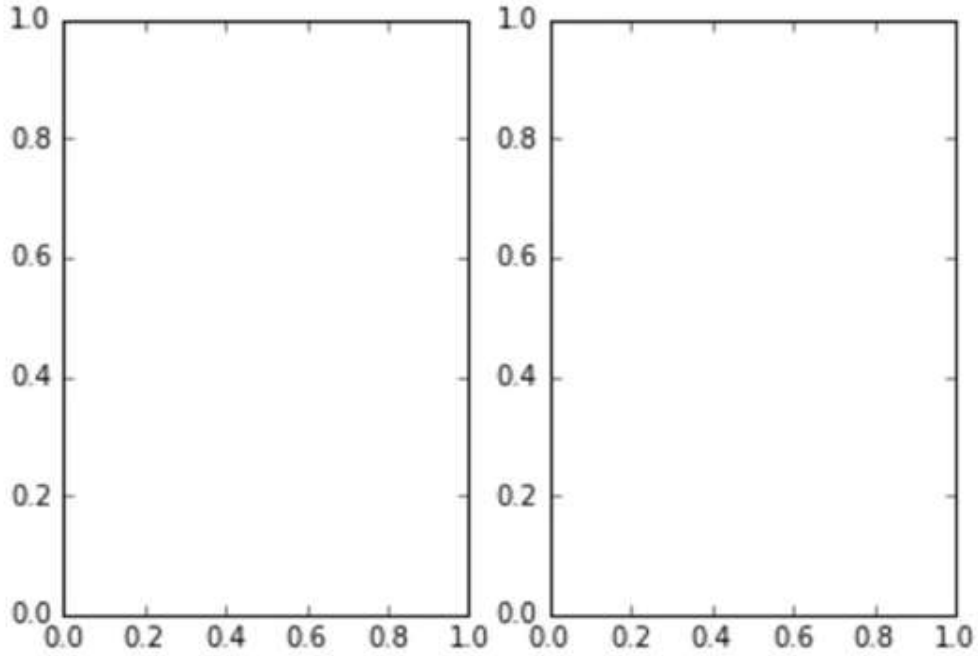
localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

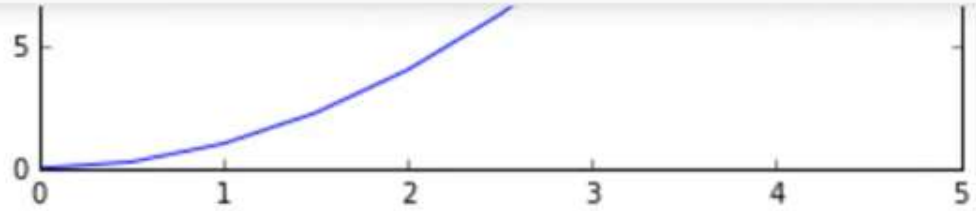


In [35]:
```
fig,axes = plt.subplots(nrows=1,ncols=2)

#axes.plot(x,y)
```

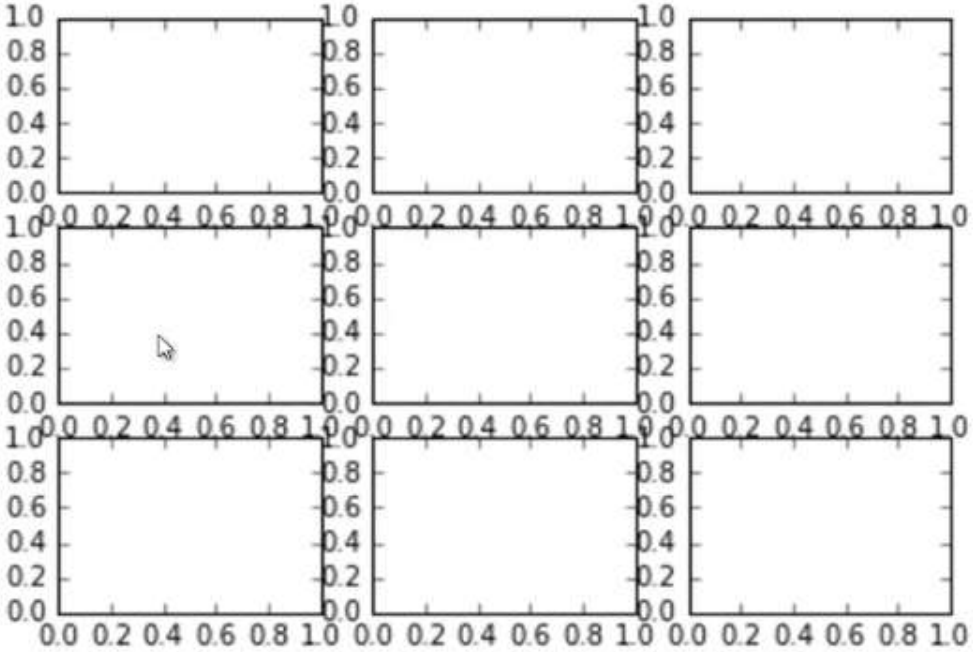


In [ ]:

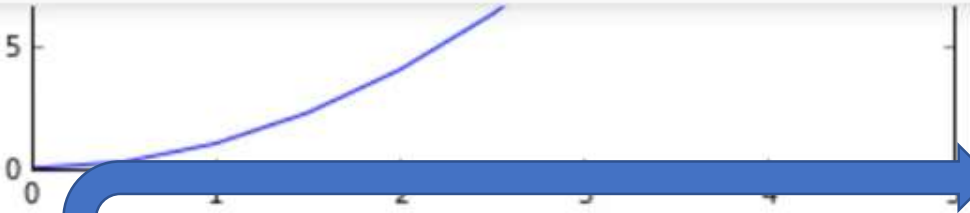


In [36]:
```
fig,axes = plt.subplots(nrows=3,ncols=3)

#axes.plot(x,y)
```



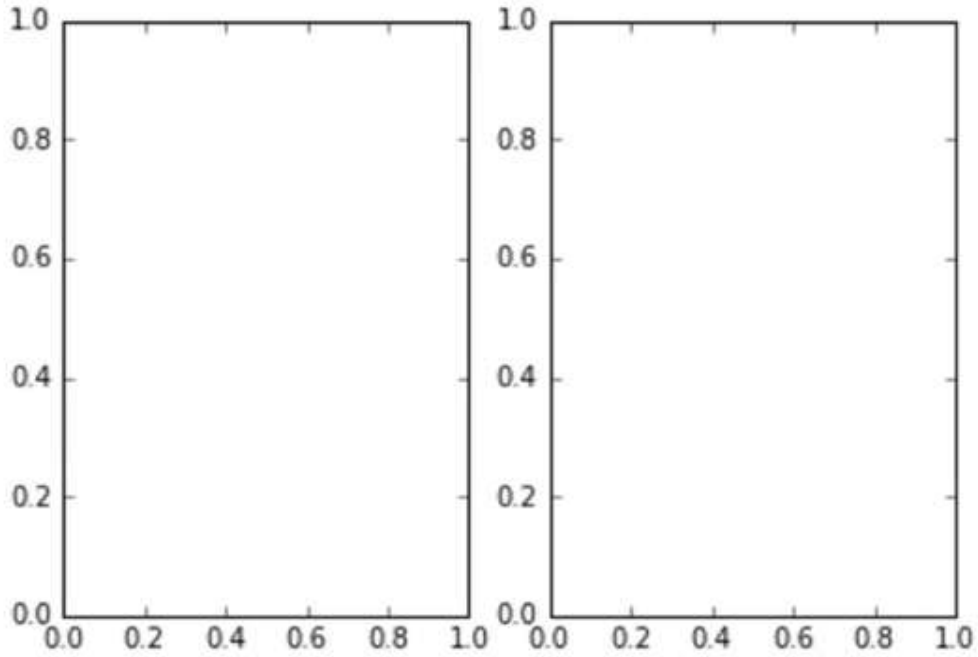
In [ ]:



This is called Tuple unpacking.

In [38]:
```
fig,axes = plt.subplots(nrows=1,ncols=2)

#axes.plot(x,y)
```

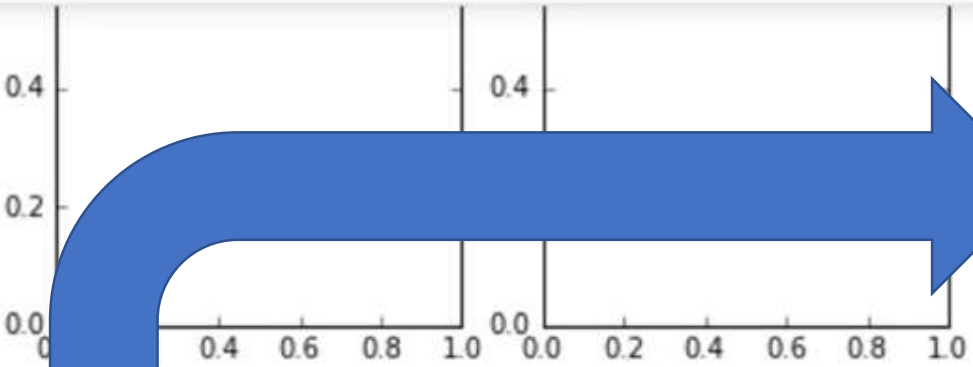


In [ ]:

matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]



Since it is an Array we can do the **Loops.**

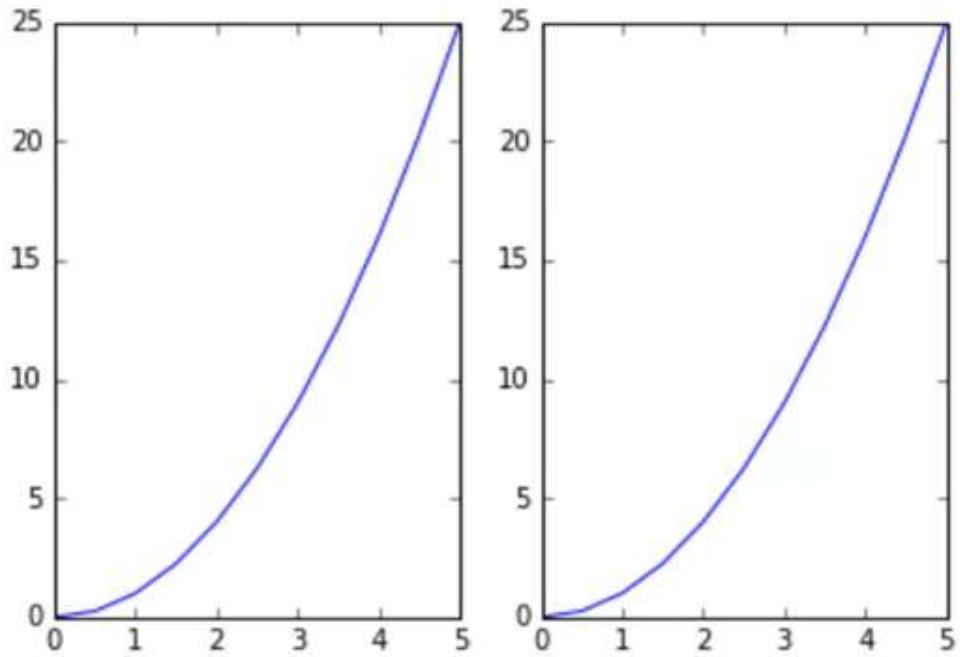
In [39]:
```
axe
```

Out[39]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000164B0139630>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x00000164B01A1240>], d
type=object)
```

In [ ]:

In [40]:
```
fig,axes = plt.subplots(nrows=1,ncols=2)

for current_ax in axes:
    current_ax.plot(x,y)
```

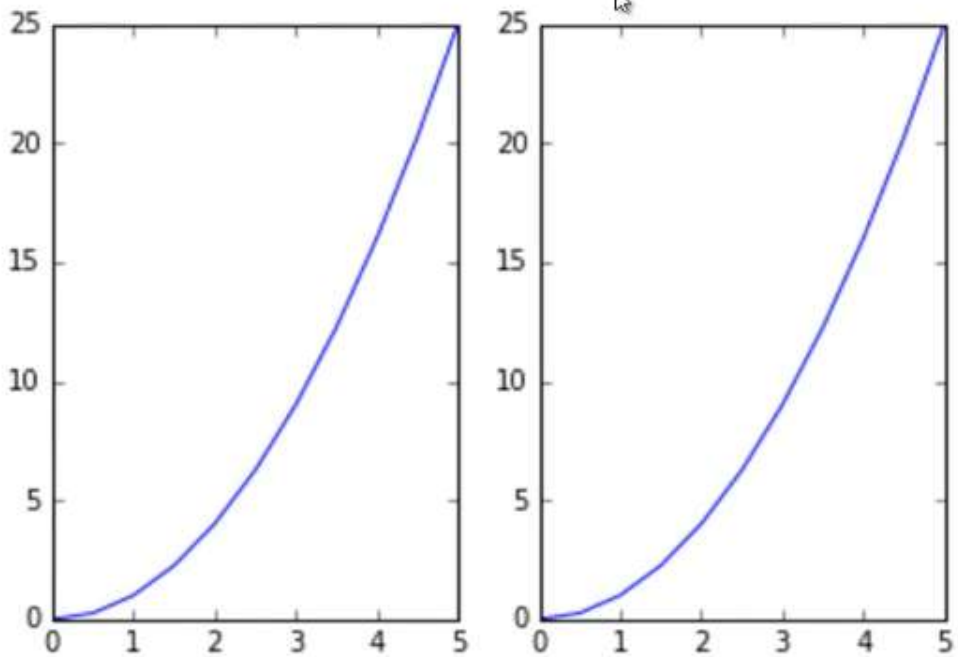


In [39]:
```
axes
```

Out[39]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000164B0139630>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x00000164B01A1240>], d
type=object)
```

In [40]:
```
fig,axes = plt.subplots(nrows=1,ncols=2)

axes[0].plot(x,y)
```



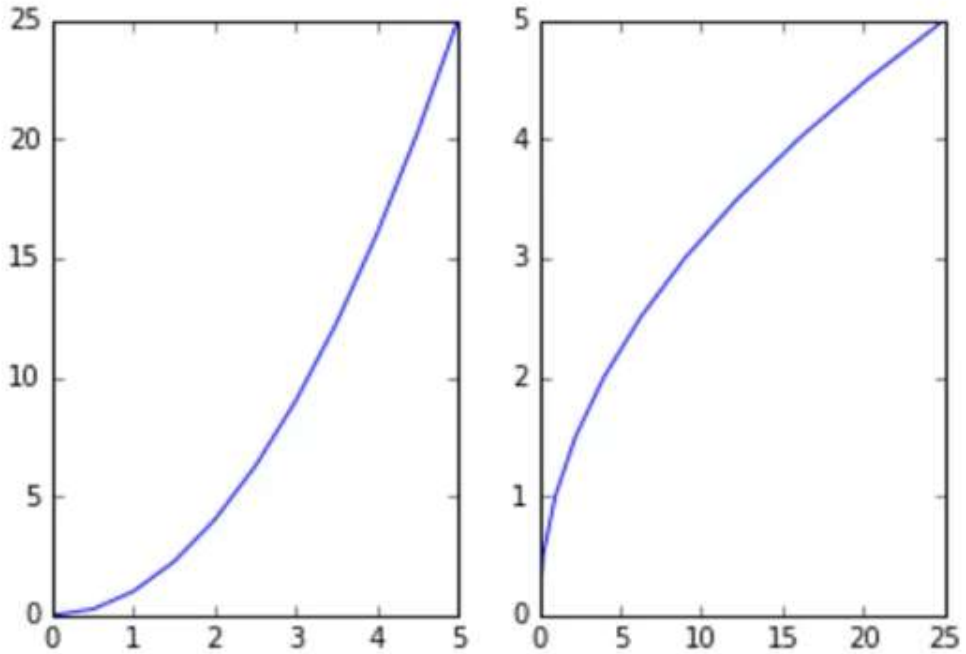
In [39]:
```
axes
```

Out[39]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000164B0139630>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x00000164B01A1240>], d
```

In [42]:
```
fig,axes = plt.subplots(nrows=1,ncols=2)

axes[0].plot(x,y)
axes[1].plot(y,x)
```

Out[42]: [<matplotlib.lines.Line2D at 0x164b041ebe0>]

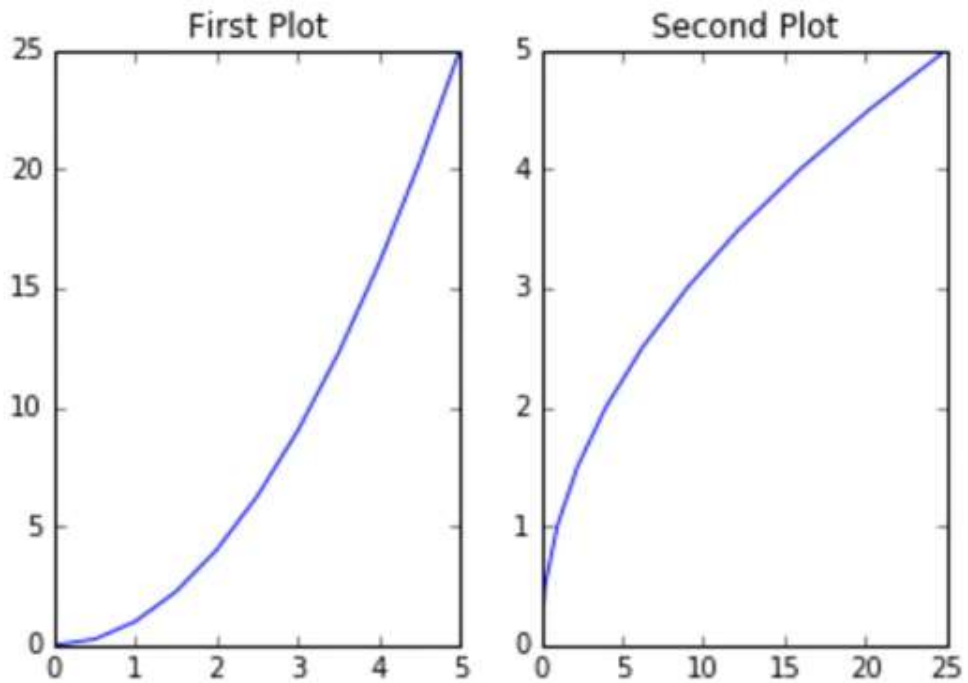


In [39]:
```
axes
```

```
axes[0].plot(x,y)
axes[0].set_title('First Plot')

axes[1].plot(y,x)
axes[1].set_title('Second Plot')
```

Out[43]: <matplotlib.text.Text at 0x164ae6ef320>



In [39]:
```
axes
```

Out[39]: array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000164B0139630>,

matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

Figure Size and DPI

In [ ]:

In [ ]:

matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

Figure Size and DPI

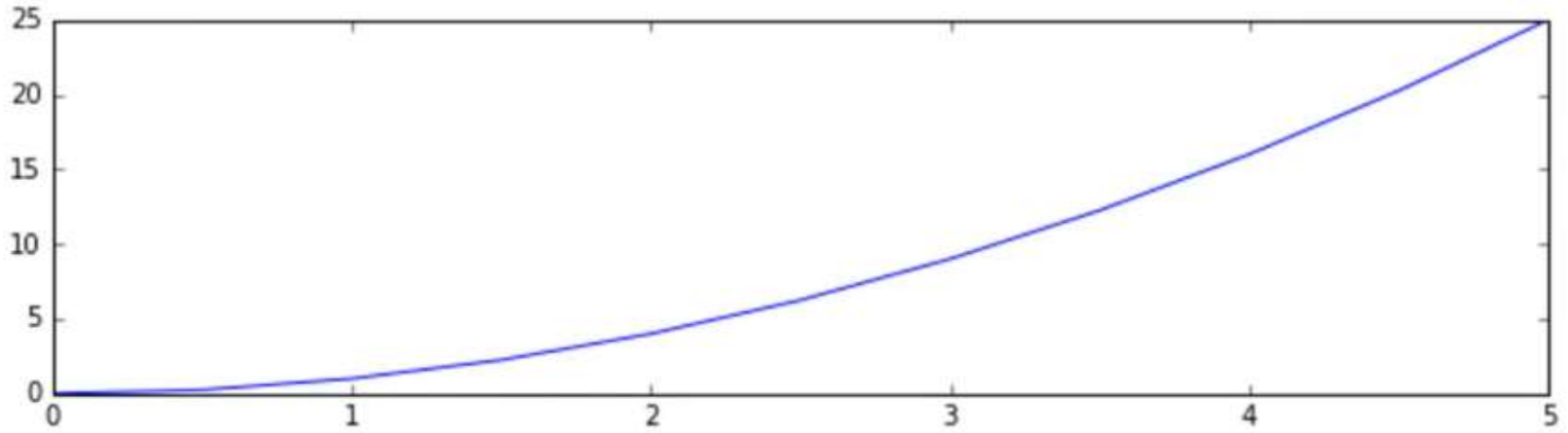
Width of the plot in inches

Height of the plot in inches

In [49]:
```
fig = plt.figure(figsize=(8,2))

ax = fig.add_axes([0,0,1,1])
ax.plot(x,y)
```

Out[49]:
```
[<matplotlib.lines.Line2D at 0x164afc5f9e8>]
```



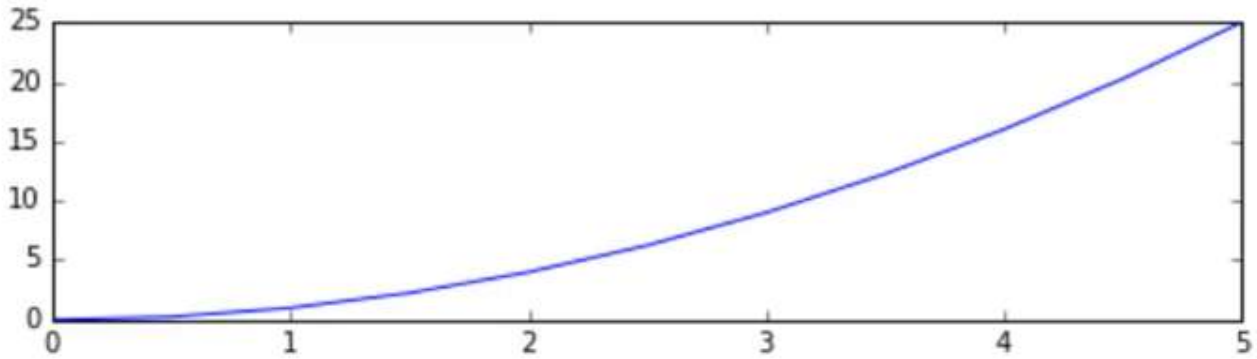
In [ ]:

Figure Size and DPI

In [50]:
```
fig,axes = plt.subplots(figsize=(8,2))

axes.plot(x,y)
```

Out[50]:
```
[<matplotlib.lines.Line2D at 0x164afc29518>]
```



In [ ]:

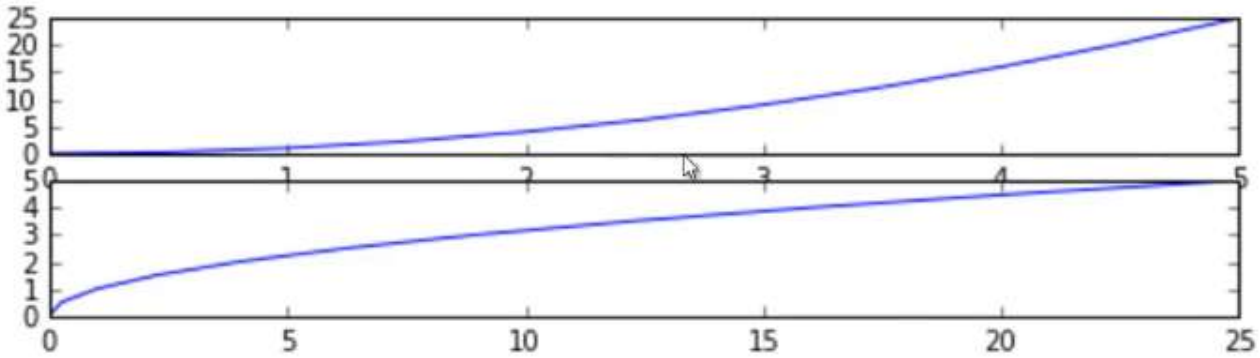
Figure Size and DPI

In [51]:
```
fig,axes = plt.subplots(nrows=2,ncols=1,figsize=(8,2))

axes[0].plot(x,y)

axes[1].plot(y,x)
```

Out[51]: [<matplotlib.lines.Line2D at 0x164b00c4fd0>]



In [ ]:

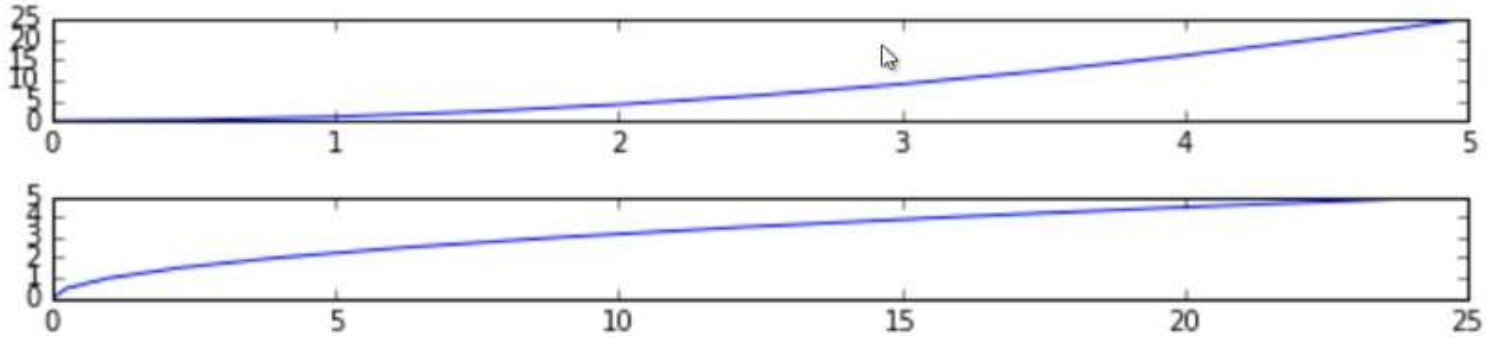
Figure Size and DPI

In [52]:
```
fig,axes = plt.subplots(nrows=2,ncols=1,figsize=(8,2))

axes[0].plot(x,y)

axes[1].plot(y,x)

plt.tight_layout()
```



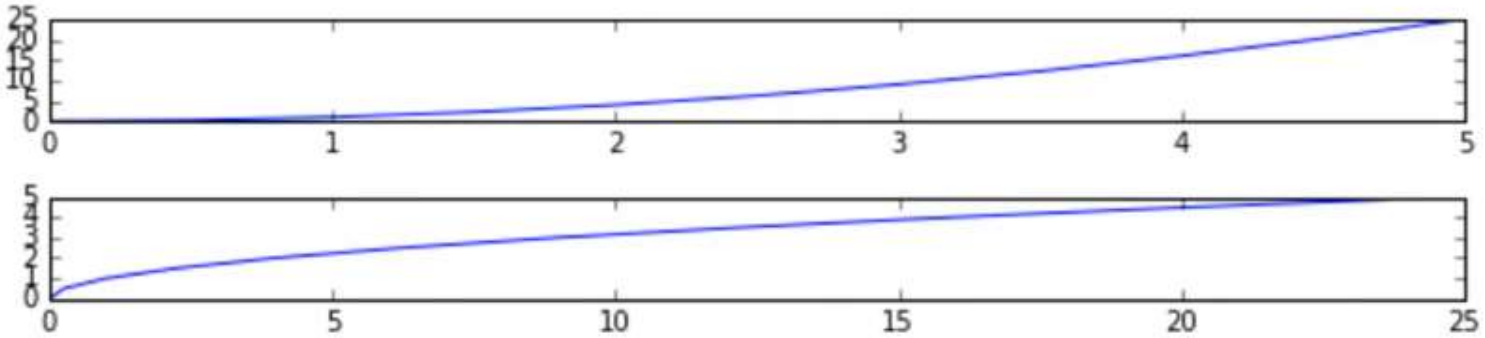
In [ ]:

matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

```
plt.tight_layout()
```



In [55]:
```
fig.savefig('my_picture.png',dpi=200)
```

In [ ]:

matplotlib

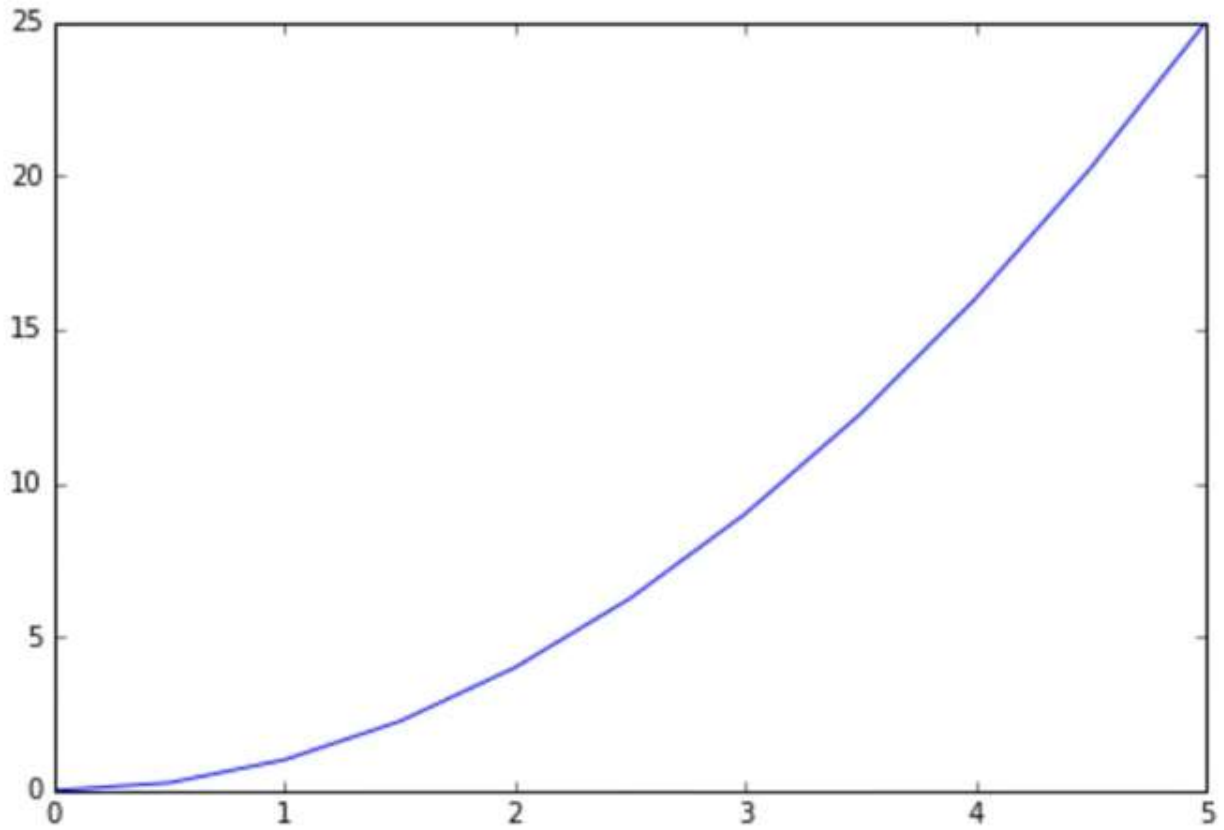
localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

In [57]:
```
fig = plt.figure()

ax = fig.add_axes([0,0,1,1])
ax.plot(x,y)
```

Out[57]: [<matplotlib.lines.Line2D at 0x164aff0f940>]



matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

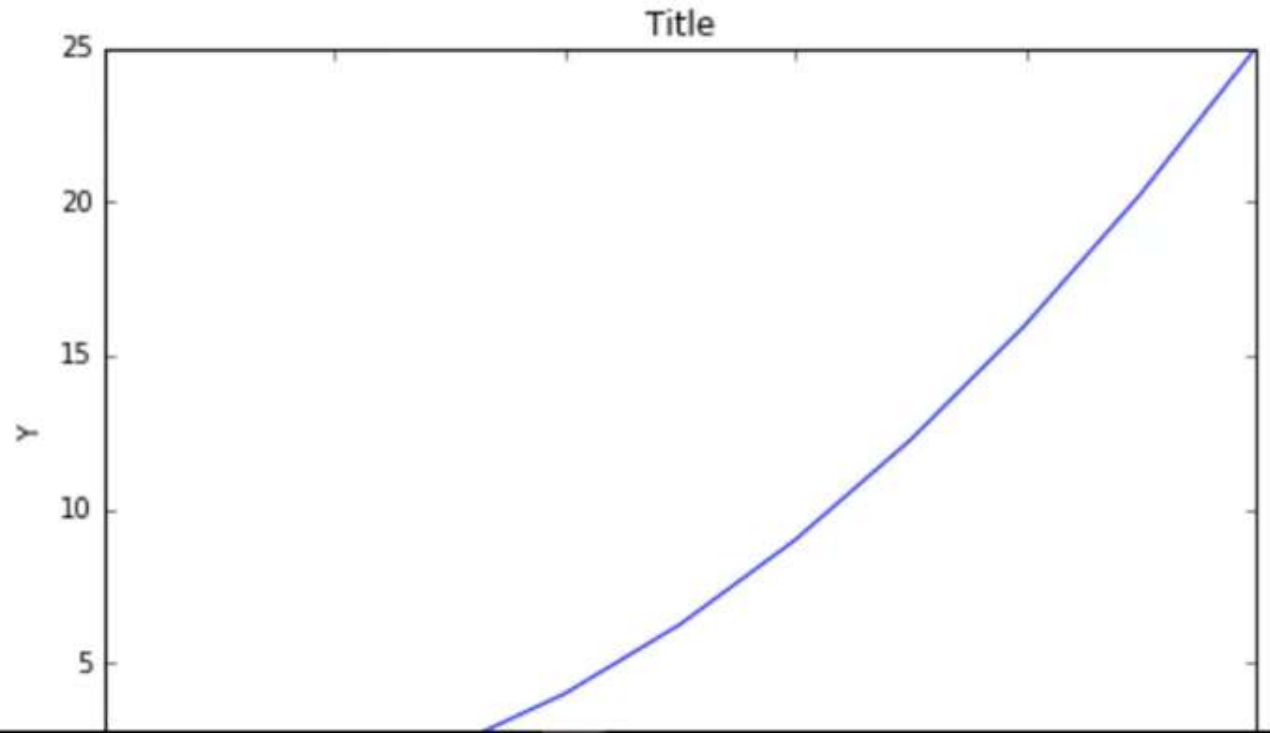
File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

In [58]:
```
fig = plt.figure()

ax = fig.add_axes([0,0,1,1])
ax.plot(x,y)

ax.set_title('Title')
ax.set_ylabel('Y')
ax.set_xlabel('X')
```

Out[58]: <matplotlib.text.Text at 0x164b0046c50>



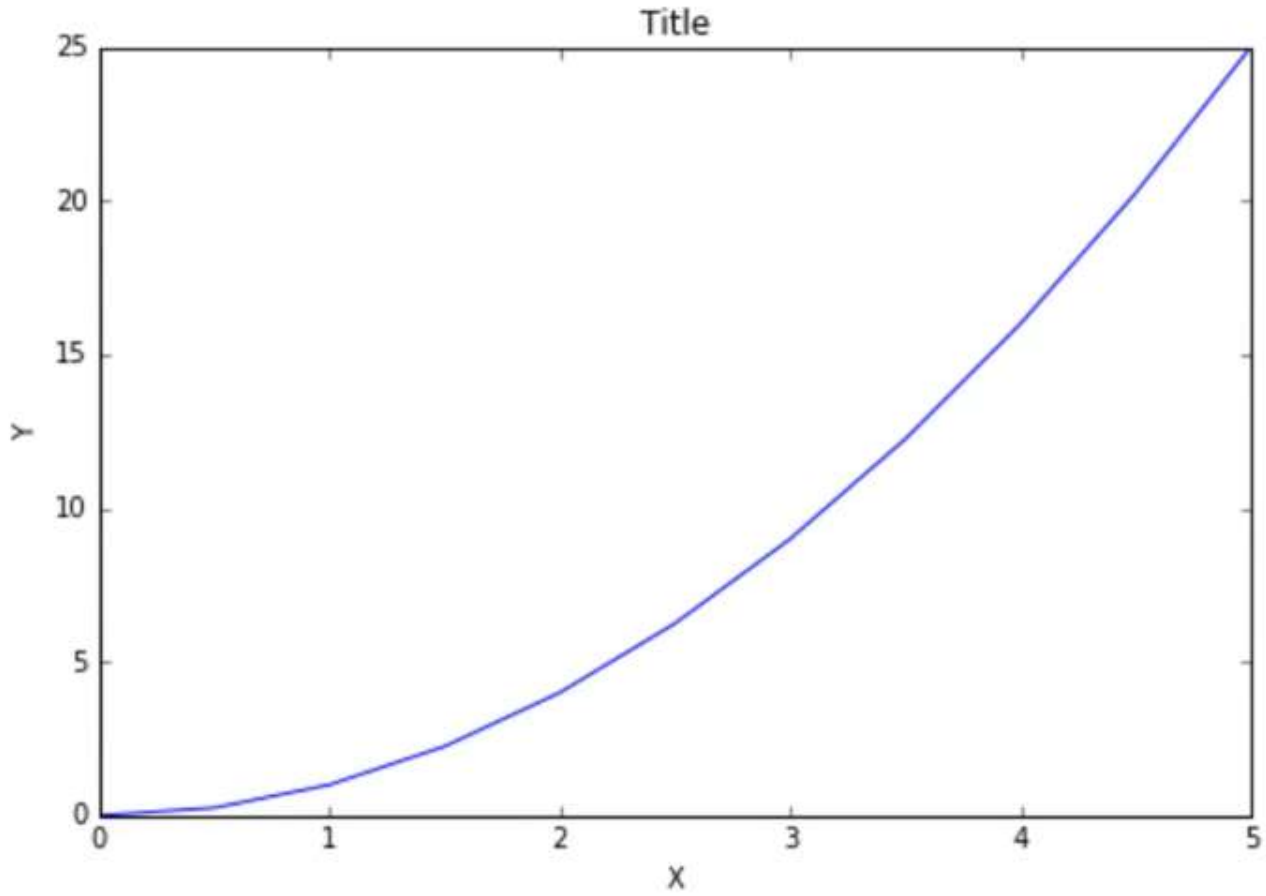
matplotlib

localhost:8888/notebooks/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-Data-Science-and-Machine-Learning-Bootcamp-master/Python-for-Data-Visualization/Matplotlib/matplotlib.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

```
ax.set_xlabel('X')
```

Out[58]: <matplotlib.text.Text at 0x164b0046c50>



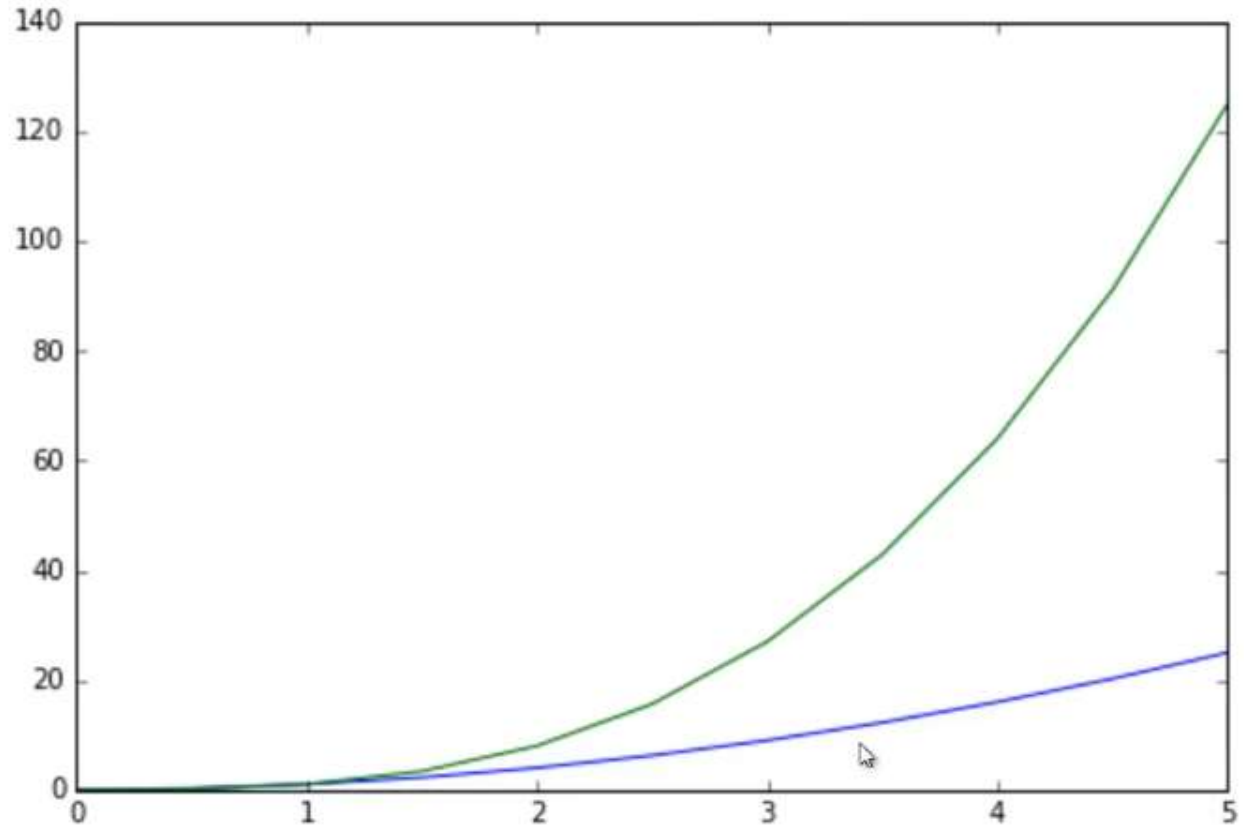
In [56]:
```
fig.savefig('my_picture.png',dpi=200)
```

In [ ]:

In [60]:
```
fig = plt.figure()

ax = fig.add_axes([0,0,1,1])
ax.plot(x,x**2)
ax.plot(x,x**3)
```

Out[60]: [<matplotlib.lines.Line2D at 0x164afde1940>]

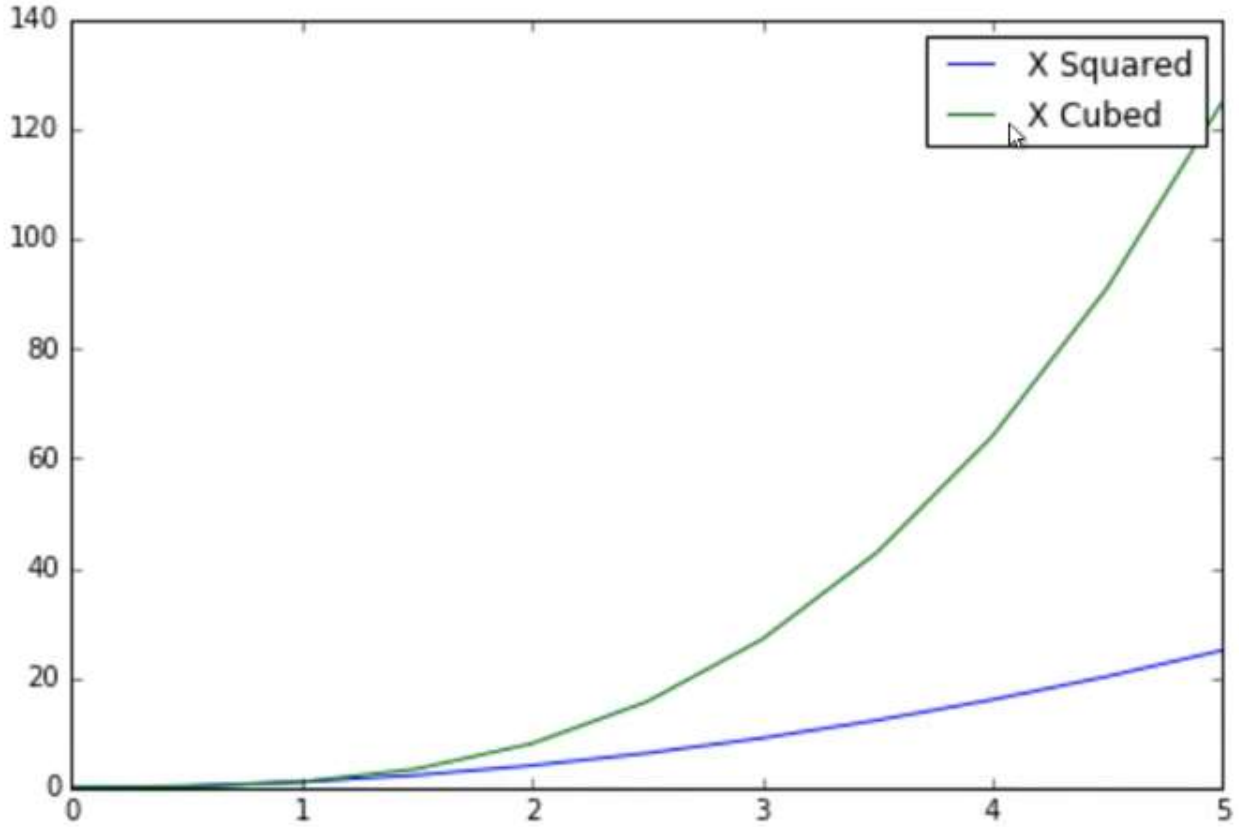


```
ax = fig.add_axes([0,0,1,1])

ax.plot(x,x**2,label='X Squared')
ax.plot(x,x**3,label='X Cubed')

ax.legend()
```

Out[62]: <matplotlib.legend.Legend at 0x164b0071828>



Parameters:

loc : int or string or pair of floats, default: 'upper right'

The location of the legend. Possible codes are:

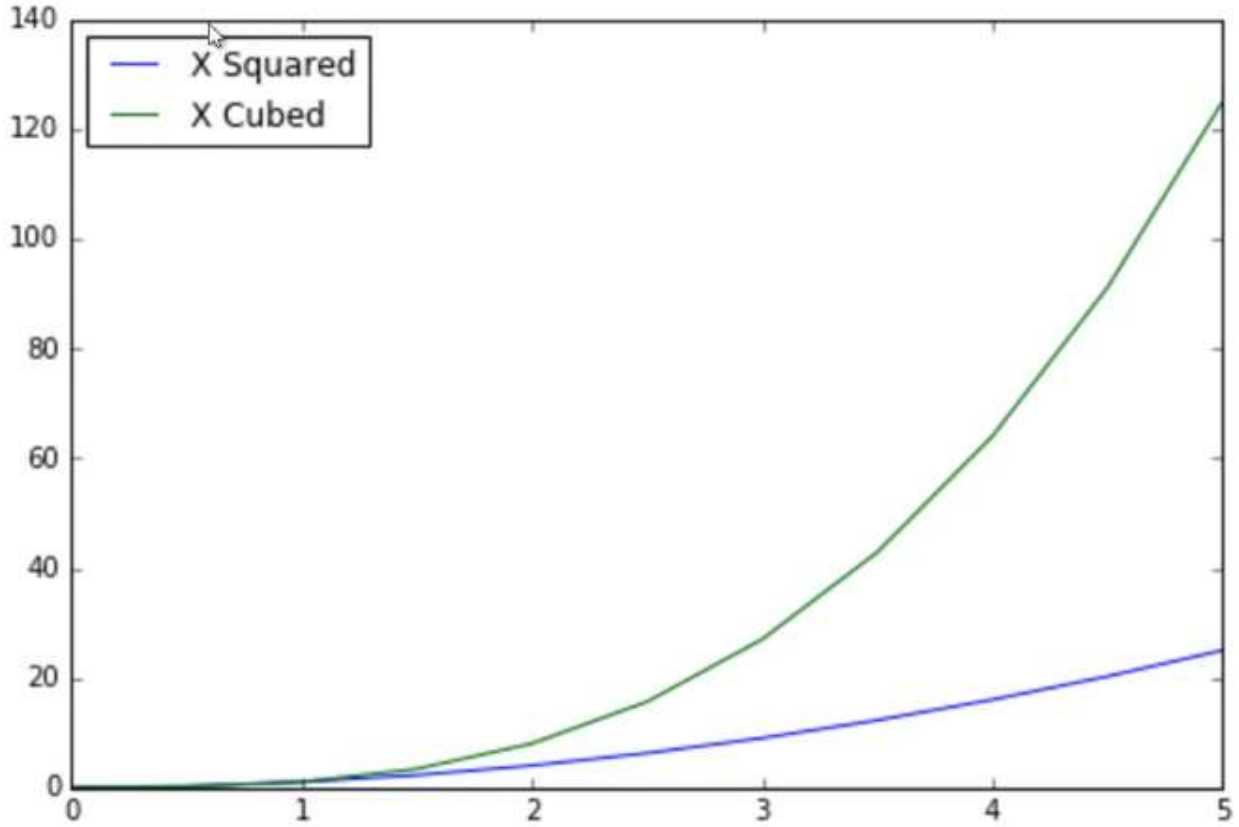
Location String	Location Code
'best'	0
'upper right'	1
'upper left'	2
'lower left'	3
'lower right'	4
'right'	5
'center left'	6
'center right'	7
'lower center'	8
'upper center'	9
'center'	10

```
ax = fig.add_axes([0,0,1,1])

ax.plot(x,x**2,label='X Squared')
ax.plot(x,x**3,label='X Cubed')

ax.legend(loc=0)
```

Out[63]: <matplotlib.legend.Legend at 0x164aff35908>

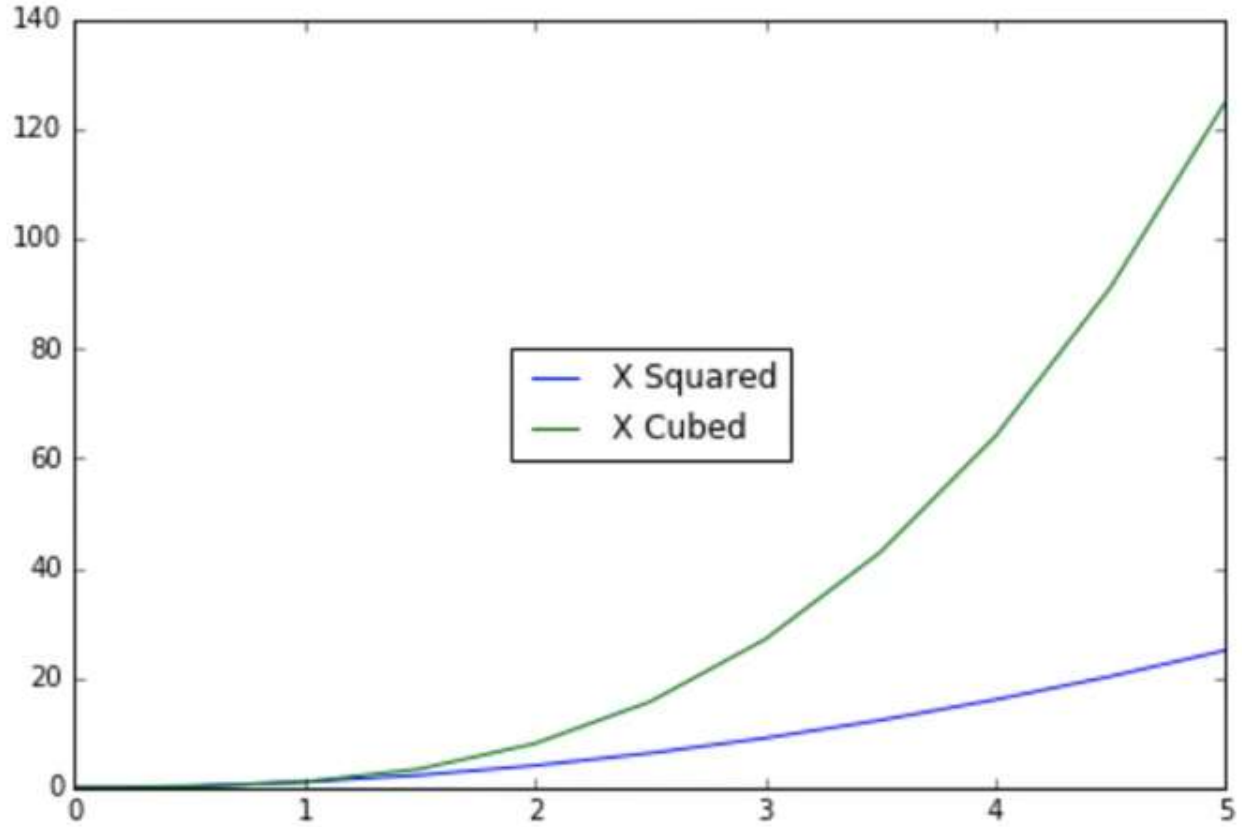


```
ax = fig.add_axes([0,0,1,1])

ax.plot(x,x**2,label='X Squared')
ax.plot(x,x**3,label='X Cubed')

ax.legend(loc=10)
```

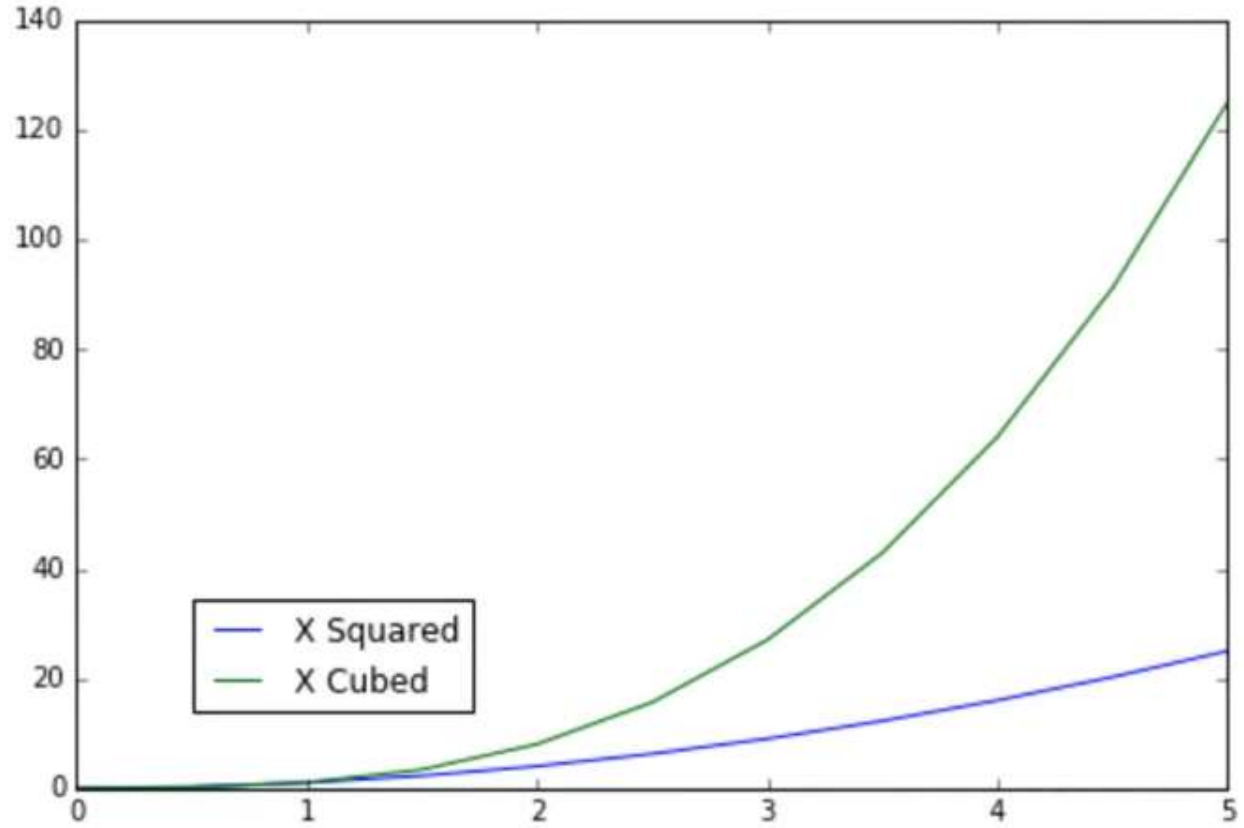
Out[64]: <matplotlib.legend.Legend at 0x164b19919b0>



```
ax.plot(x,x**3,label='X Cubed')

ax.legend(loc=(0.1,0.1))
```

Out[65]: <matplotlib.legend.Legend at 0x164b1a0d588>



In [56]:
```
fig.savefig('my_picture.png',dpi=200)
```