
MSML 605 - Lecture 5



Matplotlib



Matplot library

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

```
data = np.linspace(-np.pi,np.pi,256,endpoint=True)
```

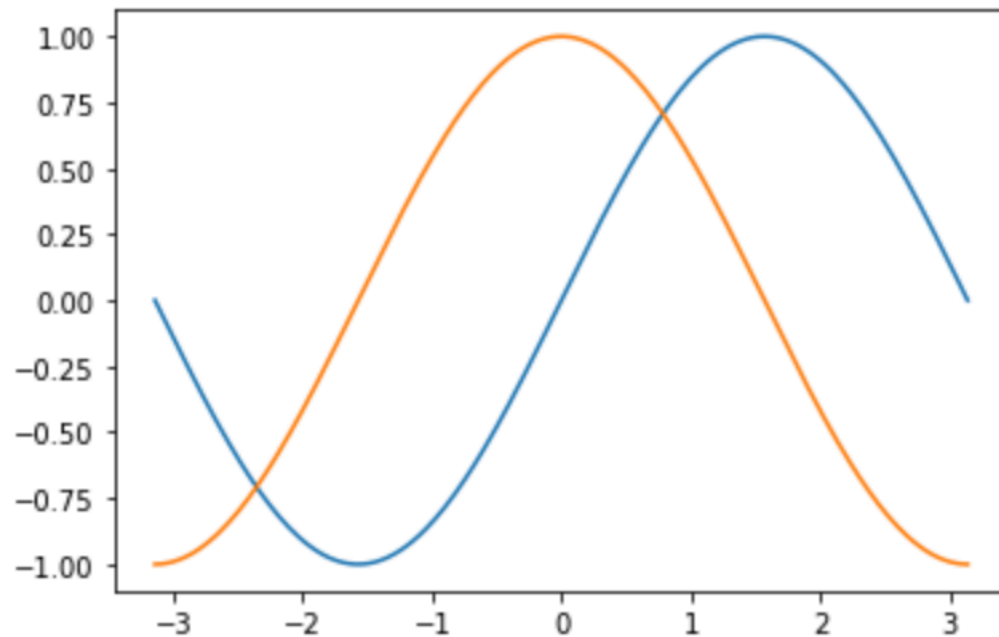
```
x,y,z = np.sin(data),np.cos(data),np.tan(data)
```

Matplotlib library - plot

```
data = np.linspace(-np.pi, np.pi, 256, endpoint=True)
```

```
x, y, z = np.sin(data), np.cos(data), np.tan(data)
```

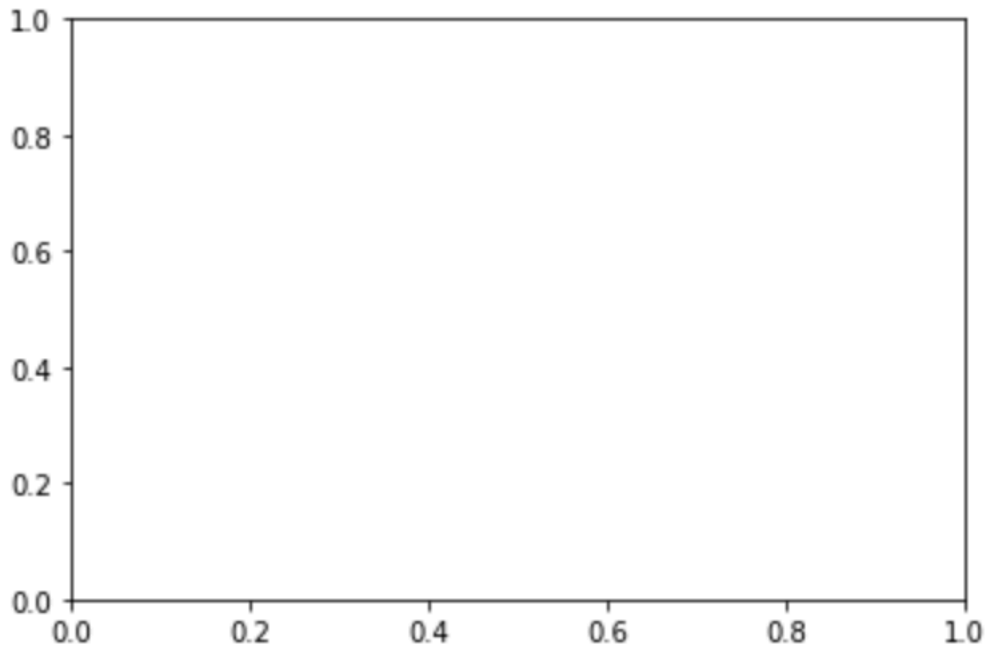
```
plt.plot(data, x)  
plt.plot(data, y)  
  
plt.show()
```



Matplot library - figure, subplot

```
# Create a figure of 8 x 6 inches, 80 dots per inch  
plt.figure(figsize=(8,6),dpi=80)
```

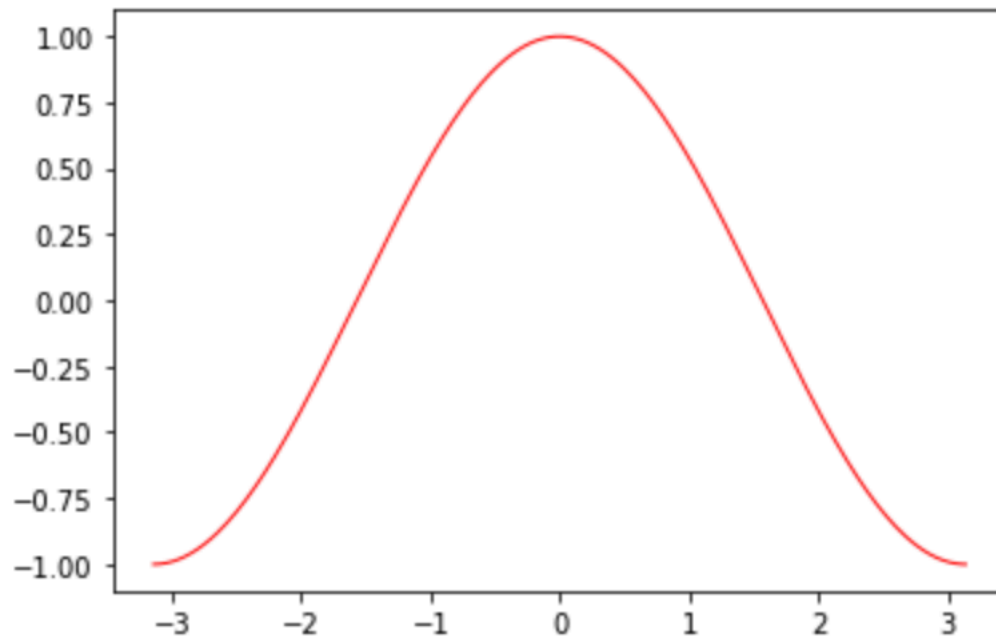
```
# Create a new subplot from a grid of 1 x 1  
plt.subplot(1,1,1)
```



Matplotlib library - plot data

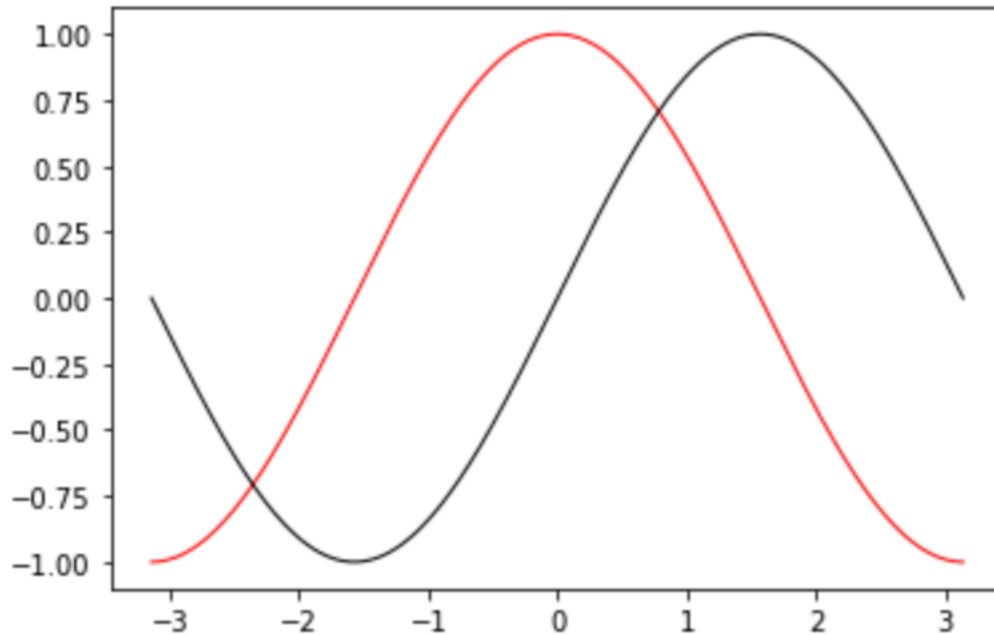
```
X = np.linspace(-np.pi,np.pi,256, endpoint=True)  
cos_data,sin_data = np.cos(X),np.sin(X)
```

```
# Plot cosine with a red continuous line of width 1 (pixels)  
plt.plot(X,cos_data,color="red",linewidth=1.0,linestyle="-")
```



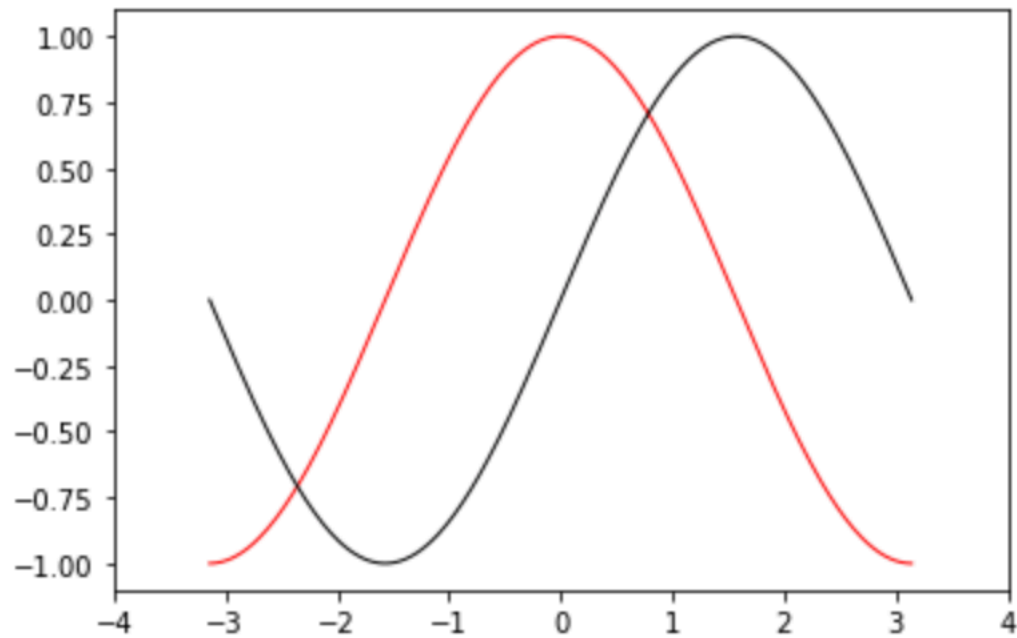
Matplotlib library - plot

```
# Plot sine with a black continuous line of width 1 (pixels)  
plt.plot(X,sin_data,color="black",linewidth=1.0,linestyle="-")
```



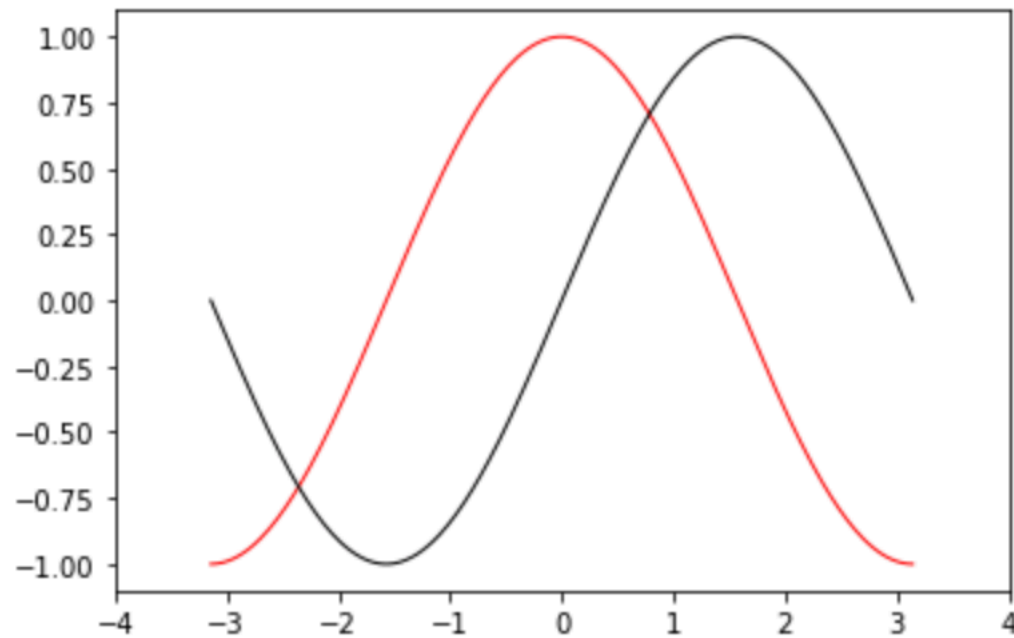
Matplotlib library - X limit

```
# Set x limits  
plt.xlim(-4.0,4.0)
```



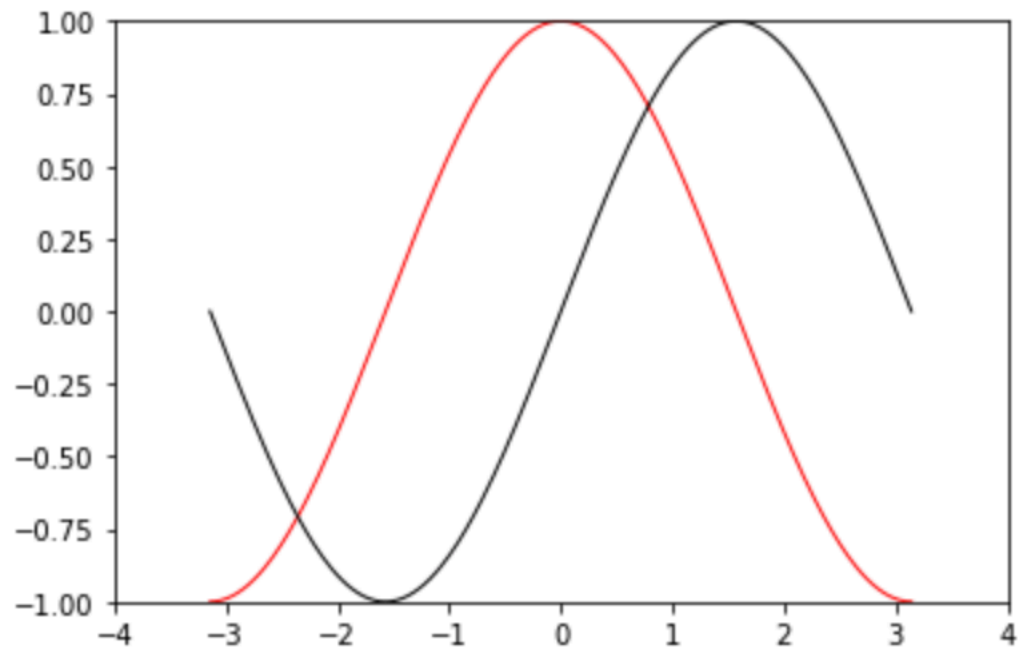
Matplotlib library - Xticks

```
# Set x ticks  
plt.xticks(np.linspace(-4,4,9,endpoint=True))
```



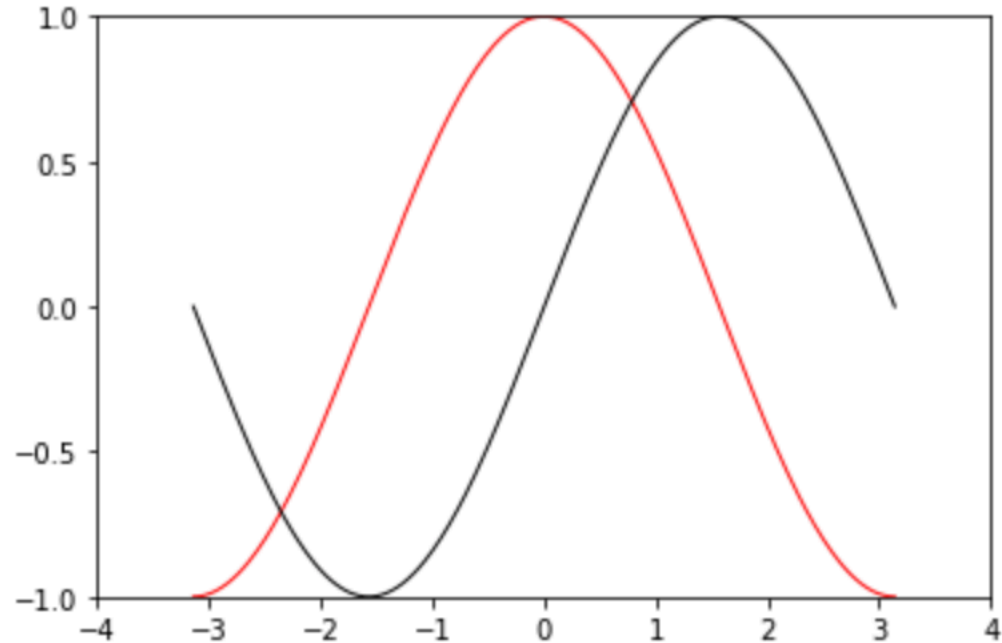
Matplotlib library - y limit

```
# Set y limits  
plt.ylim(-1.0,1.0)
```



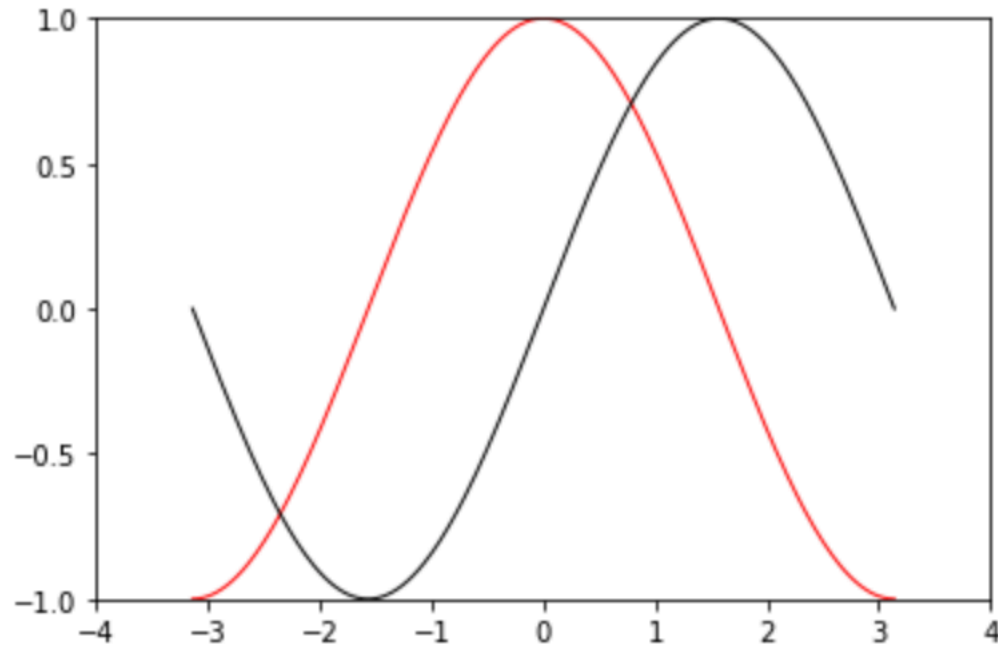
Matplotlib library - y ticks

```
# Set y ticks  
plt.yticks(np.linspace(-1,1,5,endpoint=True))
```



Matplotlib library - show and save

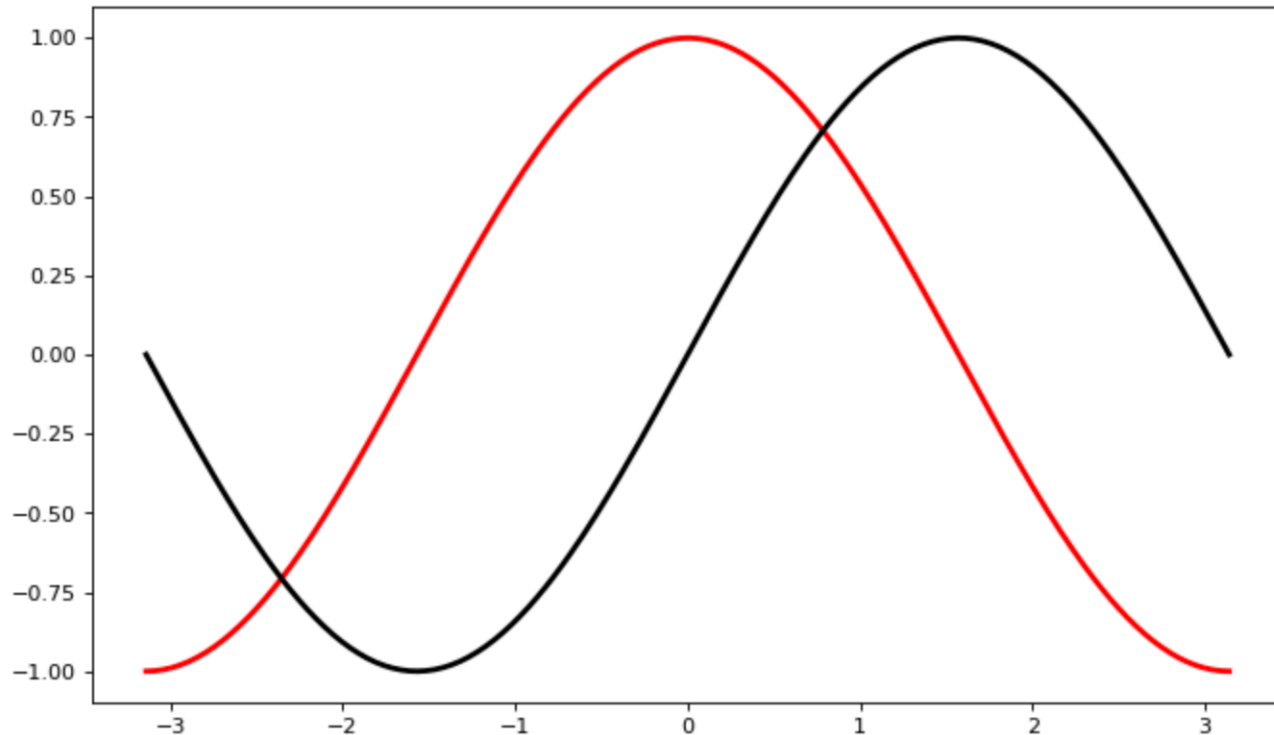
```
# Show result on screen  
plt.show()
```



```
# Save figure using 72 dots per inch  
# plt.savefig("sines-cosines",dpi=72)
```

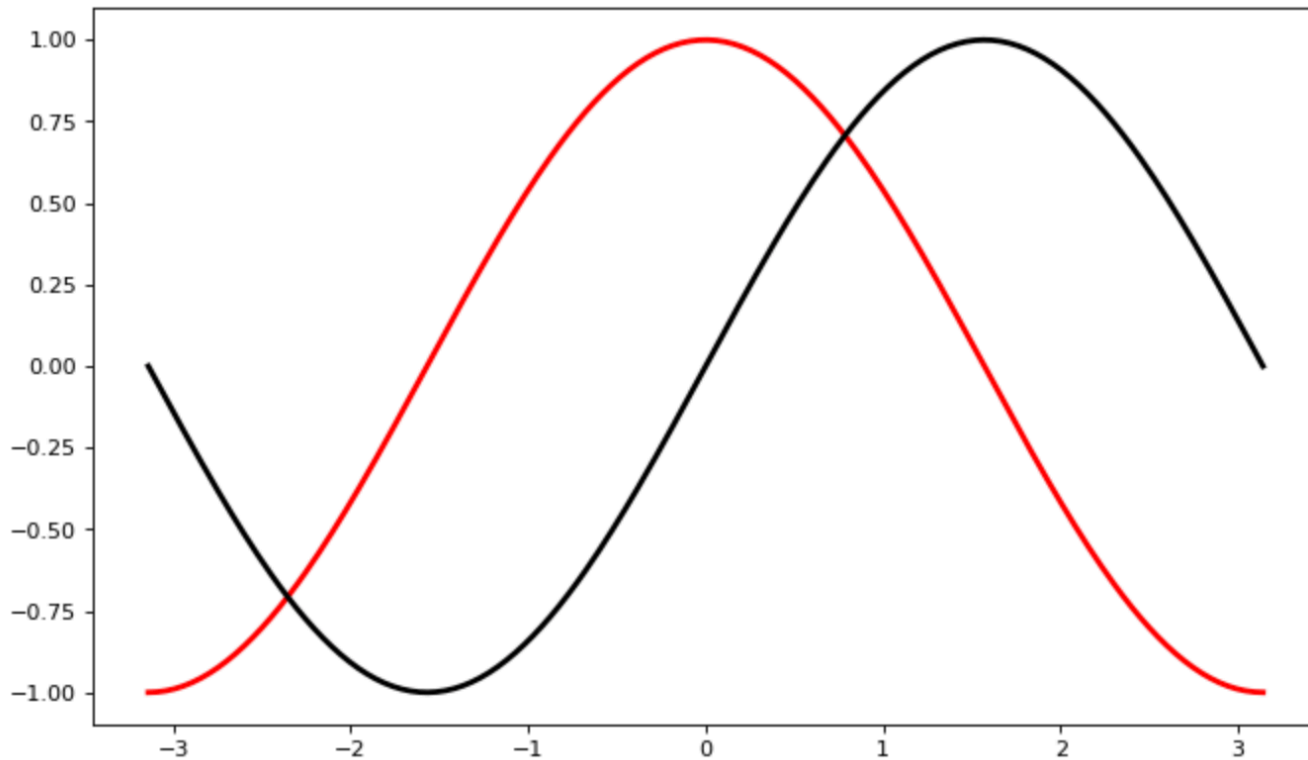
Matplotlib library - plot in a figure

```
plt.figure(figsize=(10,6), dpi=80)
# Change the line width
plt.plot(X,cos_data,color="red", linewidth=2.5, linestyle="-")
plt.plot(X,sin_data,color="black", linewidth=2.5,linestyle="-")
```



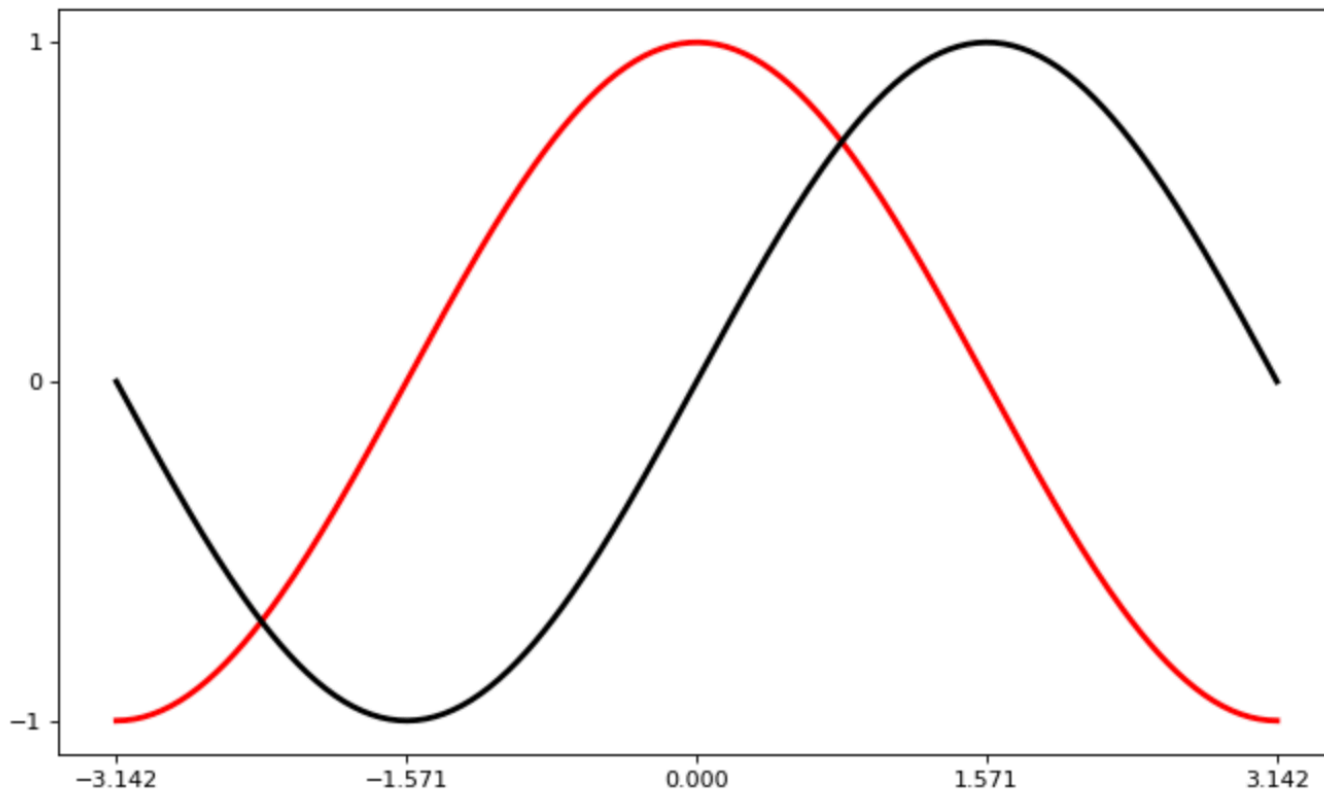
Matplotlib library - plot

```
# Make some space in the x width and y width  
plt.xlim(X.min()*1.1,X.max()*1.1)  
plt.ylim(cos_data.min()*1.1,cos_data.max()*1.1)
```



Matplotlib library - update ticks

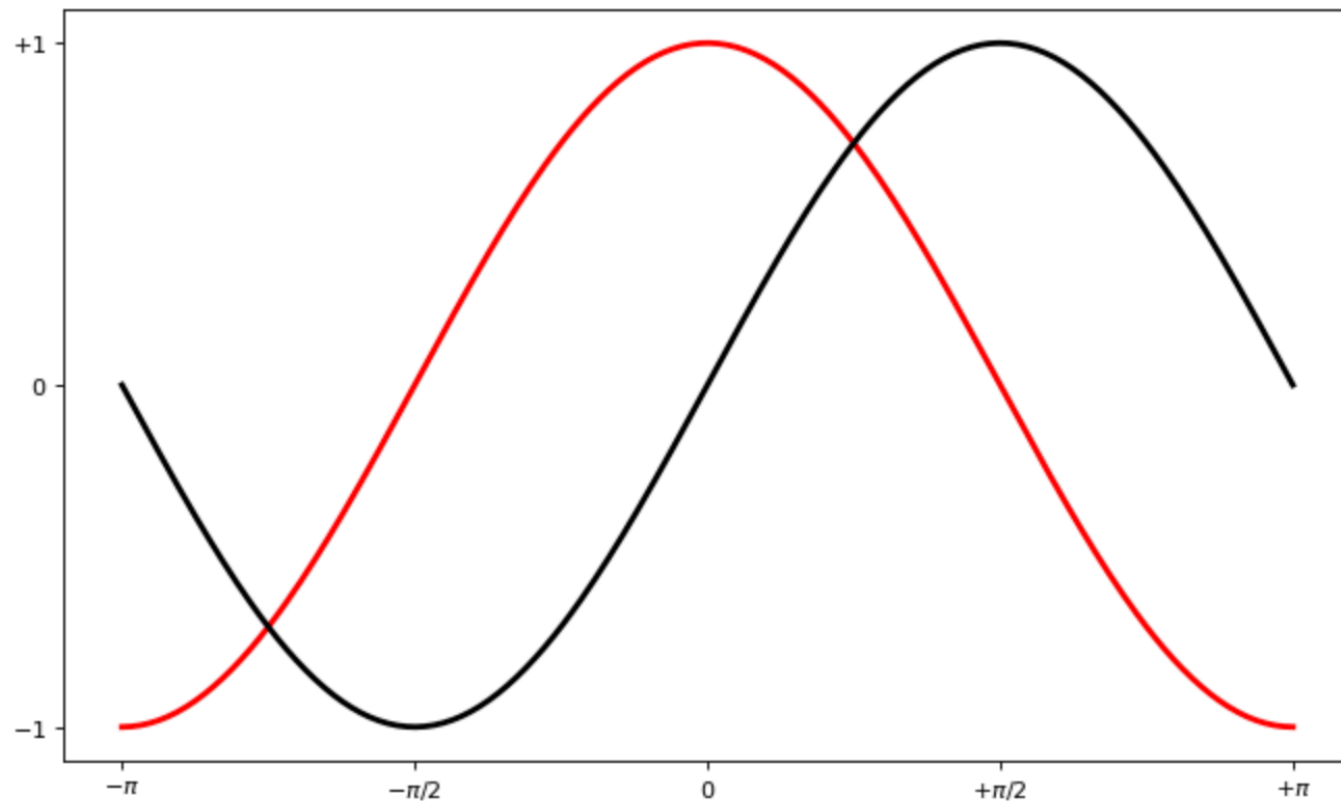
```
# Adjust ticks so that we can see pi/2 and -pi/2  
plt.xticks([-np.pi, -np.pi/2, 0, np.pi/2, np.pi])  
plt.yticks([-1, 0, 1])
```



Matplotlib library - tick labels using latex

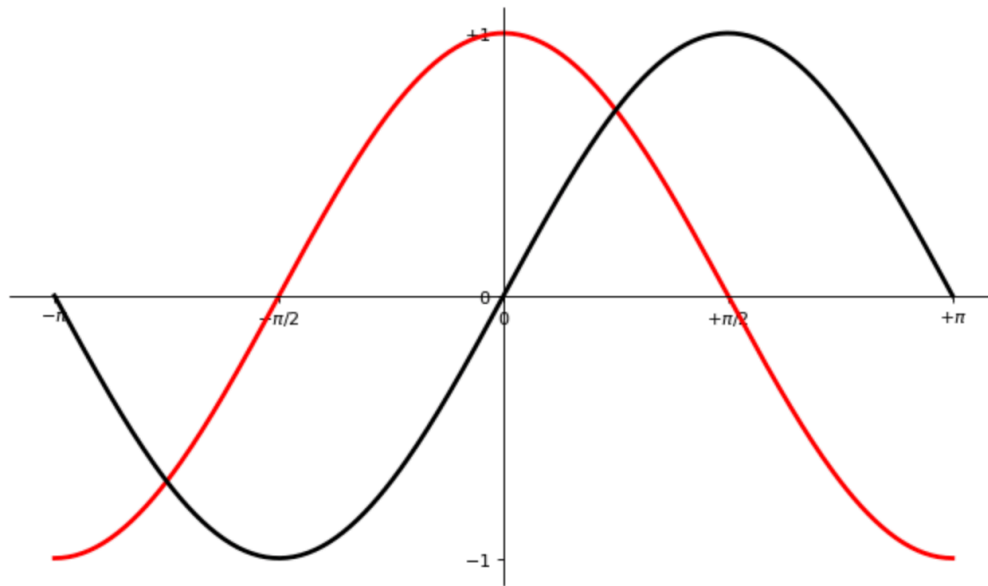
```
# Set tick labels
```

```
plt.xticks([-np.pi, -np.pi/2, 0, np.pi/2, np.pi],  
           [r'$-\pi$', r'$-\pi/2$', r'$0$', r'$+\pi/2$', r'$+\pi$'])  
plt.yticks([-1, 0, 1], [r'$-1$', r'$0$', r'$+1$'])
```



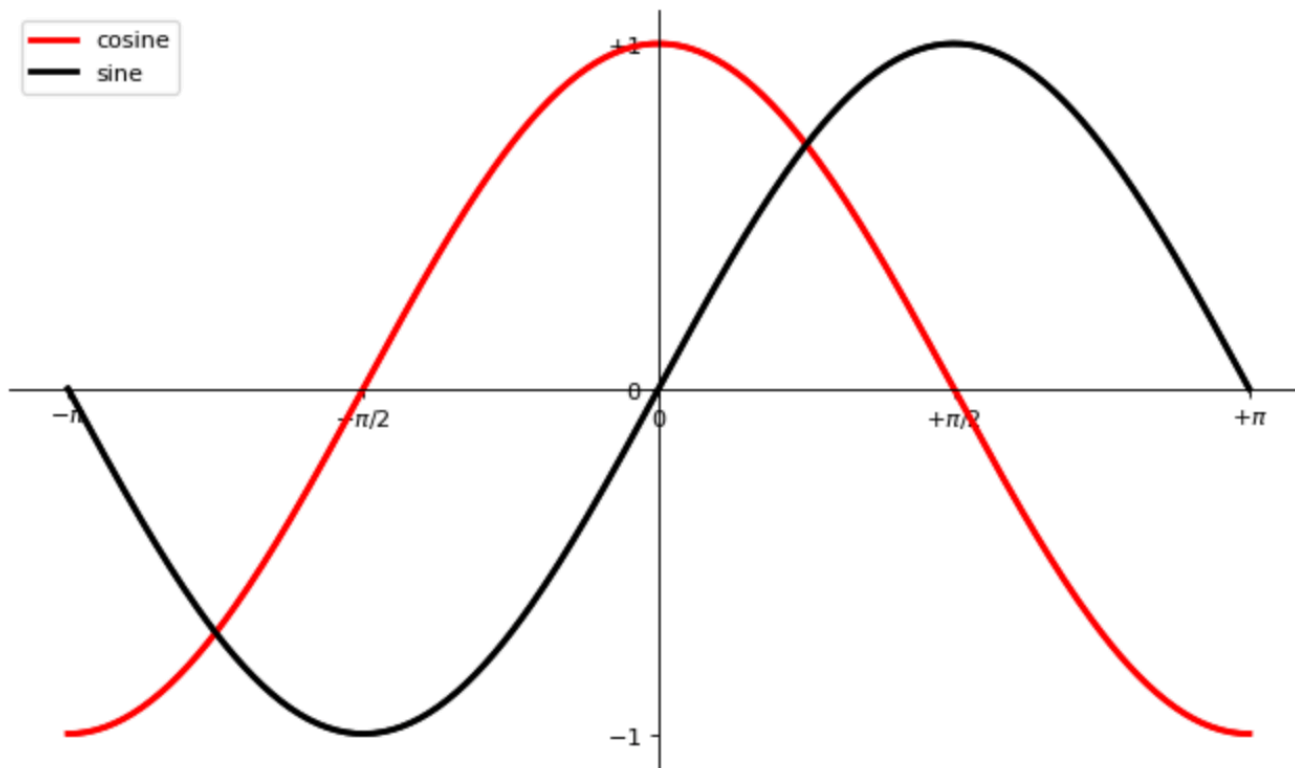
Matplotlib library - Update axis

```
# get current axis (gca)
ax = plt.gca()
# Get rid of the right and the top
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.spines['bottom'].set_position(('data',0))
ax.yaxis.set_ticks_position('left')
ax.spines['left'].set_position(('data',0))
```



Matplot library - legend

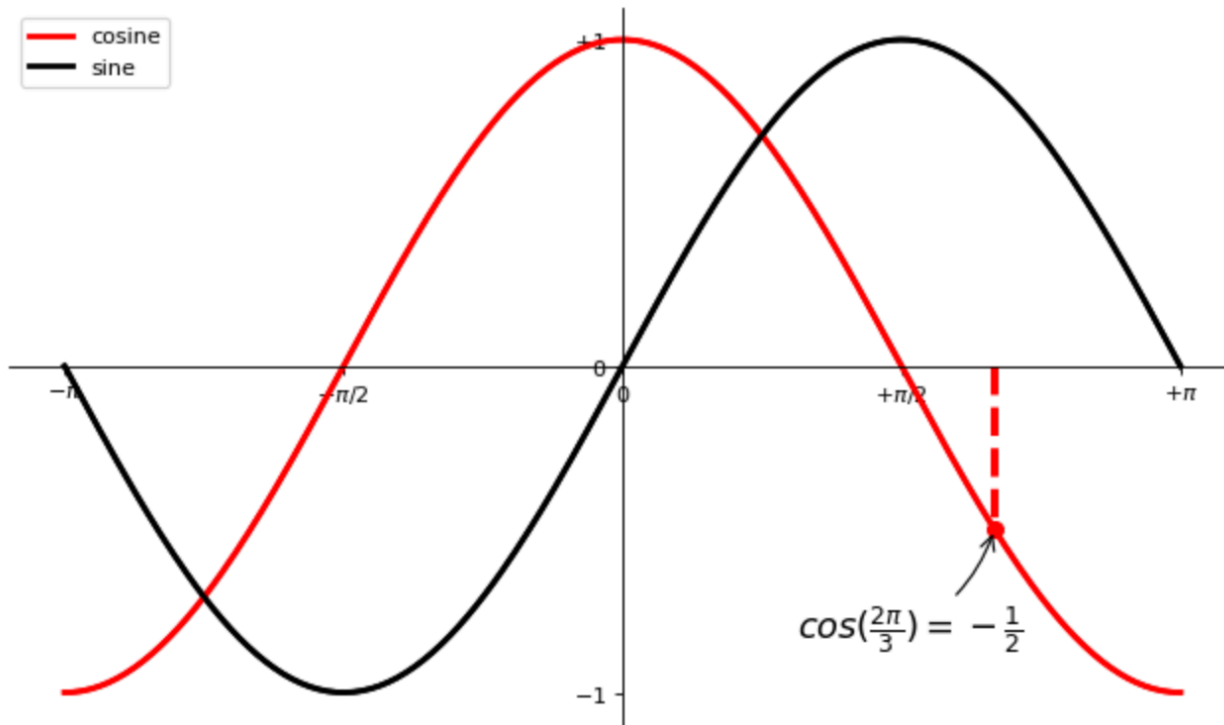
```
# Add a legend in the upper left corner  
plt.plot(X,cos_data,color="red", linewidth=2.5, linestyle="-",label="cosine")  
plt.plot(X,sin_data,color="black", linewidth=2.5,linestyle="-",label="sine")  
plt.legend(loc='upper left')
```



Matplot library - annotate

```
t = 2 * np.pi/3
```

```
plt.annotate(r'$\cos(\frac{2\pi}{3})=-\frac{1}{2}$',  
            xy = (t,np.cos(t)),xycoords='data',  
            xytext=(-90,-50),textcoords='offset points', fontsize=16,  
            arrowprops=dict(arrowstyle="->",connectionstyle="arc3,rad=.2"))
```



Matplotlib library - annotate

```
plt.plot([t,t],[0, np.sin(t)],color='black',linewidth=3.5,linestyle="--")
plt.scatter([t, ],[np.sin(t), ],50,color='black')

plt.annotate(r'$\sin(\frac{2\pi}{3})=\frac{\sqrt{3}}{2}$',
            xy = (t,np.sin(t)),xycoords='data',
            xytext=(+10,30),textcoords='offset points', fontsize=16,
            arrowprops=dict(arrowstyle="->",connectionstyle="arc3,rad=.2"))
```

