

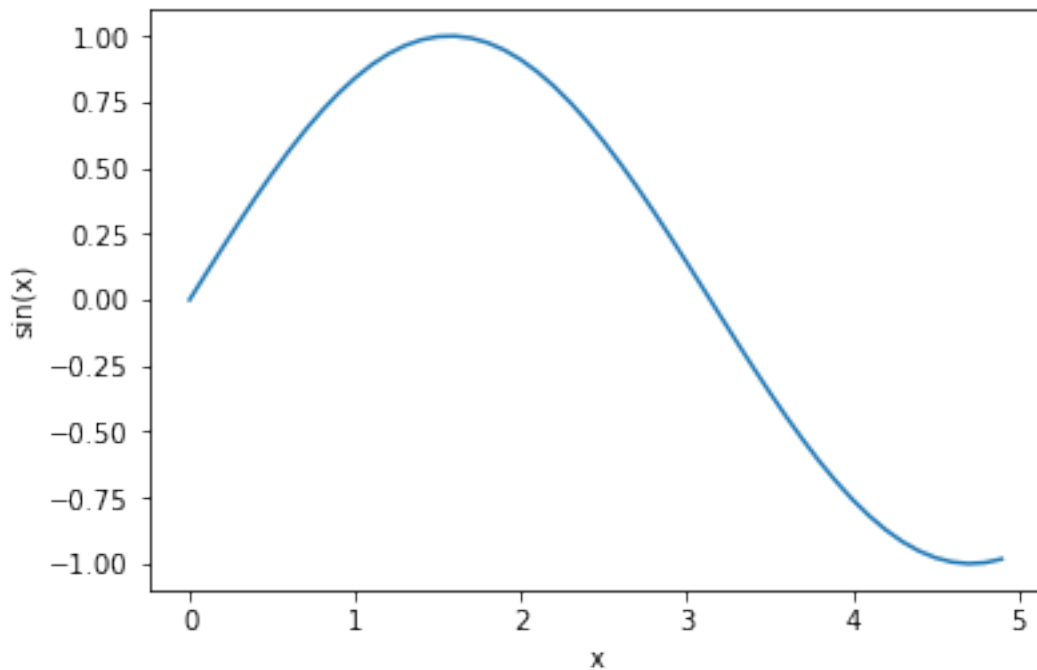
Astro-Session-6

October 16, 2018

```
In [93]: import numpy as np
import matplotlib.pyplot as plt
% matplotlib inline

x = np.arange(0, 5, 0.1)
y = np.sin(x)

plt.plot(x,y)
plt.xlabel('x')
plt.ylabel('sin(x)')
plt.show()
plt.savefig('sinx.png',bbox_inches="tight",dpi=600)
```



<Figure size 432x288 with 0 Axes>

```

In [94]: import numpy as np
import matplotlib.pyplot as plt
% matplotlib inline

x = np.linspace(0, 2*np.pi, 100)
print(x[-1], 2*np.pi)

y = np.sin(x)
z = np.cos(x)
w = np.sin(4*x)
v = np.cos(4*x)

```

6.283185307179586 6.283185307179586

```

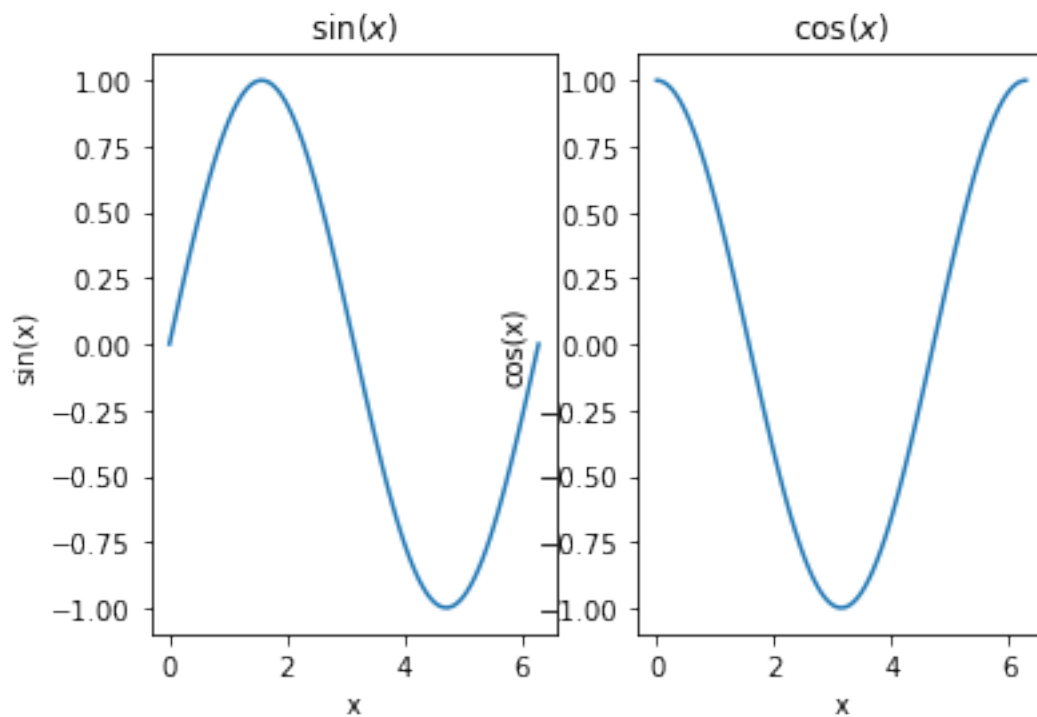
In [95]: f, axarr = plt.subplots(1, 2)

axarr[0].plot(x,y)
axarr[0].set_xlabel('x')
axarr[0].set_ylabel('sin(x)')
axarr[0].set_title(r'$\sin(x)$')

axarr[1].plot(x, z)
axarr[1].set_xlabel('x')
axarr[1].set_ylabel('cos(x)')
axarr[1].set_title(r'$\cos(x)$')

```

Out[95]: Text(0.5, 1.0, '\$\cos(x)\$')

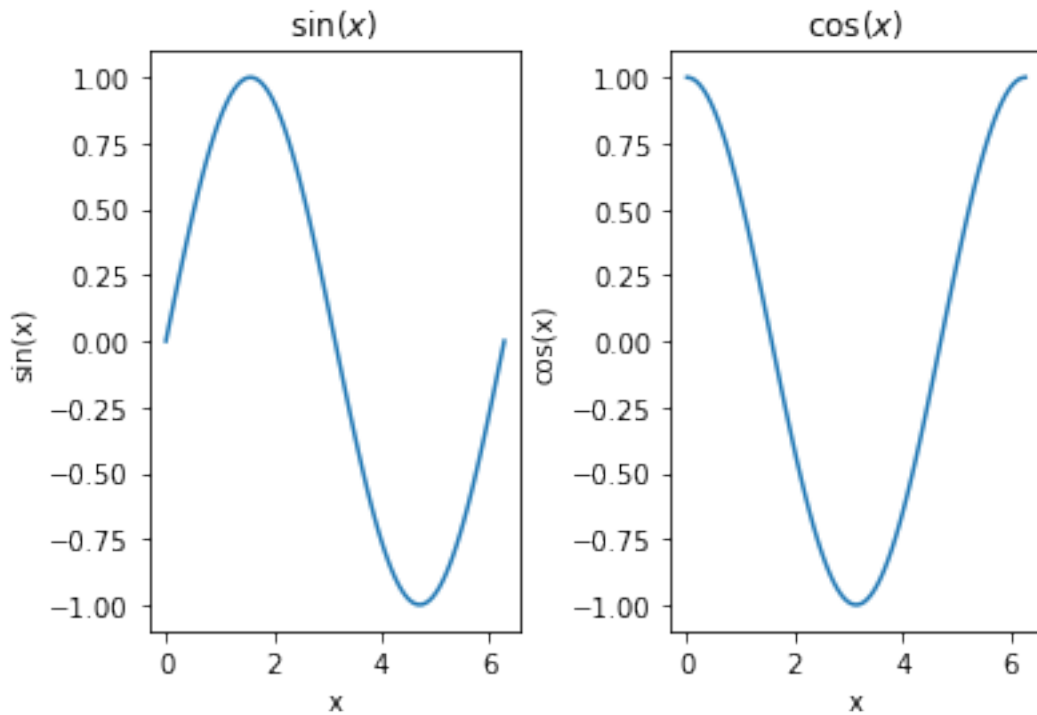


```
In [96]: f, axarr = plt.subplots(1, 2)

axarr[0].plot(x,y)
axarr[0].set_xlabel('x')
axarr[0].set_ylabel('sin(x)')
axarr[0].set_title(r'$\sin(x)$')

axarr[1].plot(x, z)
axarr[1].set_xlabel('x')
axarr[1].set_ylabel('cos(x)')
axarr[1].set_title(r'$\cos(x)$')

f.subplots_adjust(wspace=0.4)
```



```
In [97]: f, axarr = plt.subplots(1, 2)

axarr[0].plot(x,y)
axarr[0].set_xlabel('x')
axarr[0].set_ylabel('sin(x)')
axarr[0].set_title(r'$\sin(x)$')
```

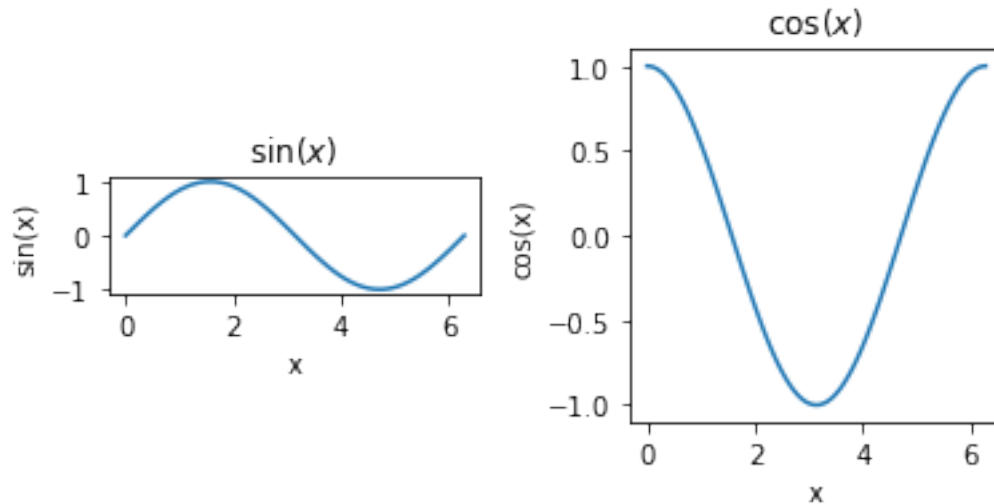
```

axarr[1].plot(x, z)
axarr[1].set_xlabel('x')
axarr[1].set_ylabel('cos(x)')
axarr[1].set_title(r'$\cos(x)$')

f.subplots_adjust(wspace=0.4)

axarr[0].set_aspect('equal')
axarr[1].set_aspect(np.pi)

```



```

In [98]: import numpy as np
import matplotlib.pyplot as plt
% matplotlib inline

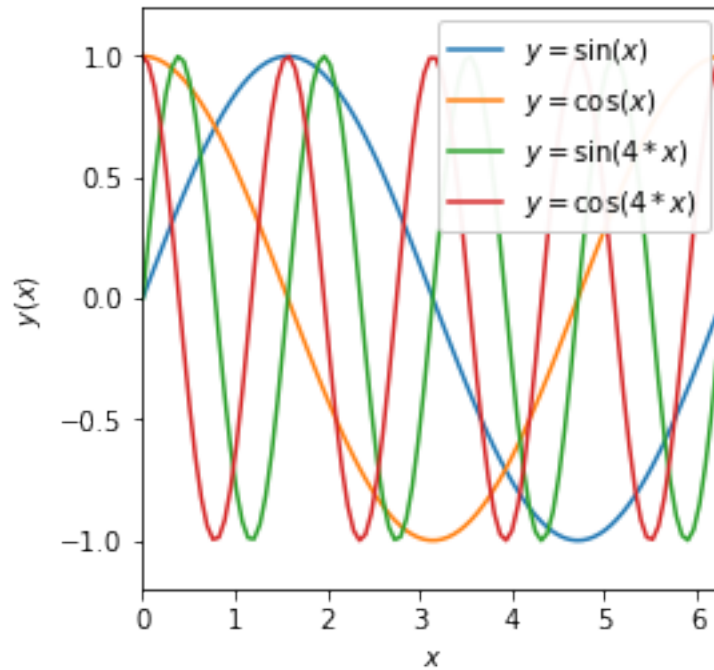
fig = plt.figure()

plt.plot(x, y, label=r'$y = \sin(x)$')
plt.plot(x, z, label=r'$y = \cos(x)$')
plt.plot(x, w, label=r'$y = \sin(4*x)$')
plt.plot(x, v, label=r'$y = \cos(4*x)$')

plt.xlabel(r'$x$')
plt.ylabel(r'$y(x)$')
plt.xlim(0, 2*np.pi)
plt.ylim(-1.2, 1.2)
plt.legend(loc=1, framealpha=0.95)

plt.gca().set_aspect(np.pi/1.2)

```



```
In [99]: import numpy as np
import matplotlib.pyplot as plt
% matplotlib inline

np.random.seed(119)

npoints = 50

x = np.linspace(0,10.,npoints)

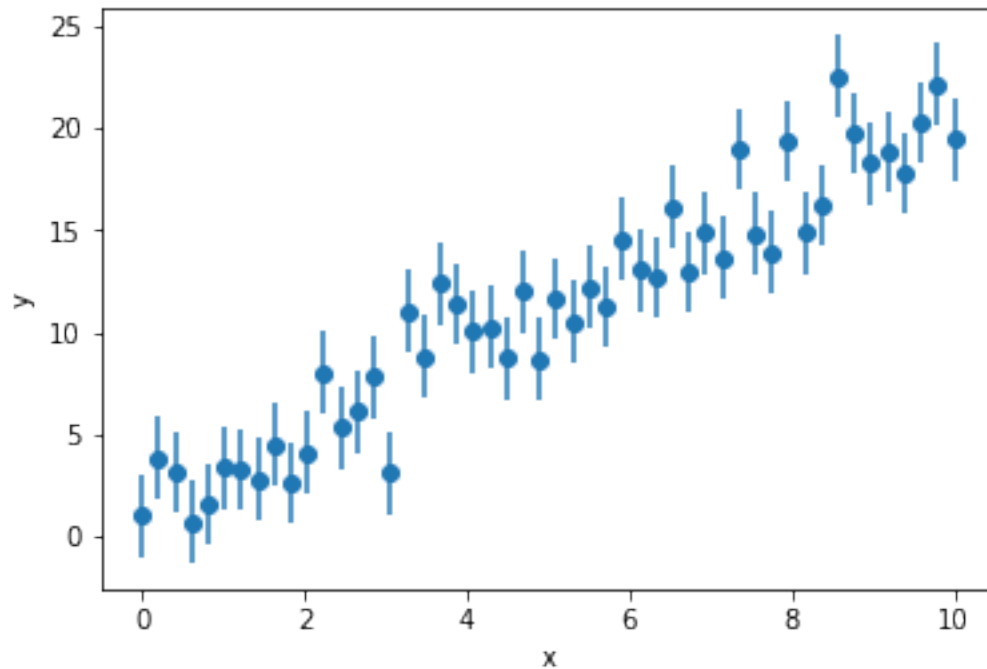
m = 2.0
b = 1.0
sigma = 2.0
y_err = np.linspace(2,2,100)

y = m*x + b + np.random.normal(scale=sigma,size=npoints)

f = plt.figure()
plt.errorbar(x,y,sigma,fmt='o')
plt.xlabel('x')
plt.ylabel('y')

print(y_err)
```

```
2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2.
2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2.
2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2.
2. 2. 2. 2.]
```



```
In [100]: m_fit, b_fit = np.poly1d(np.polyfit(x, y, 1, w=1.0/y_err))
          print(m_fit, b_fit)

          y_fit = m_fit * x + b_fit
```

TypeError

Traceback (most recent call last)

```
<ipython-input-100-ef0db1f69a07> in <module>()
----> 1 m_fit, b_fit = np.poly1d(np.polyfit(x, y, 1, w=1.0/y_err))
      2 print(m_fit, b_fit)
      3
      4 y_fit = m_fit * x + b_fit
```

```
/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/numpy/lib/
568         raise TypeError("expected a 1-d array for weights")
```

```

569         if w.shape[0] != y.shape[0]:
--> 570             raise TypeError("expected w and y to have the same length")
571         lhs *= w[:, NX.newaxis]
572         if rhs.ndim == 2:

```

TypeError: expected w and y to have the same length

```
In [ ]: f = plt.figure()
```

```

plt.errorbar(x,y,yerr=y_err,fmt='o',label='data')
plt.plot(x,y_fit,label='fmt')
plt.xlabel('x')
plt.ylabel('y')
plt.legend(loc=2,frameon=False)

```

```
In [ ]:
```