In [1]:

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
```

Create data

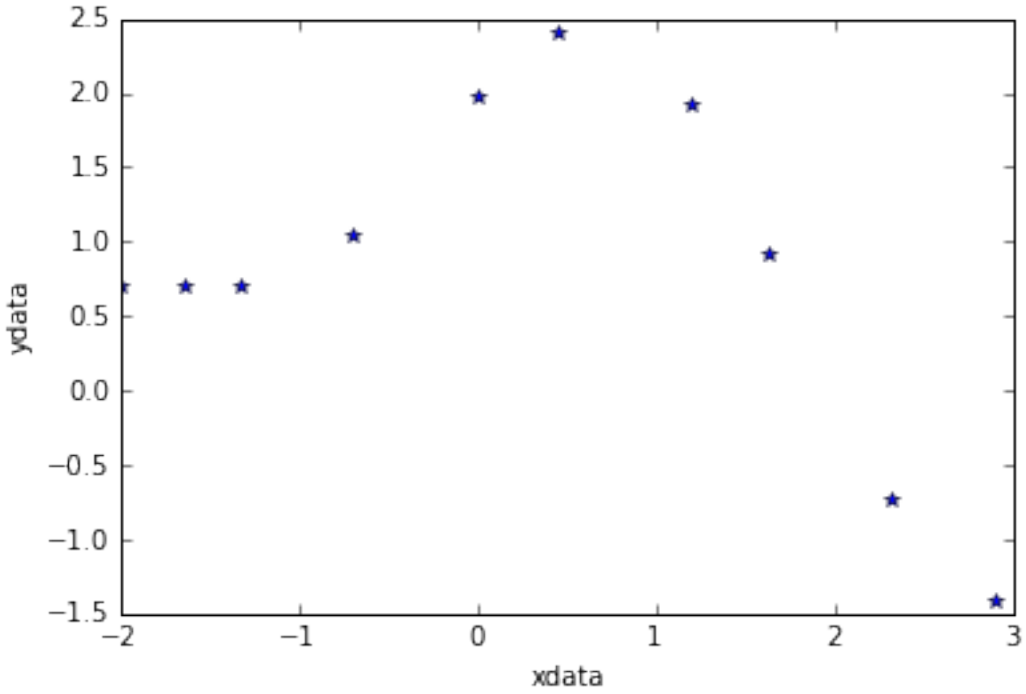
In [2]:

```
xdata = np.array([-2,-1.64,-1.33,-0.7,0,0.45,1.2,1.64,2.32,2.9])
ydata = np.array([0.699369,0.700462,0.695354,1.03905,1.97389,2.41143,1.91091,0.9
19576,-0.730975,-1.42001])
```

Show data points

In [3]:

```
plt.plot(xdata,ydata,'*')
plt.xlabel('xdata')
plt.ylabel('ydata');
```



Define fit function

In [4]:

```
def func(x, p1,p2):
  return p1*np.cos(p2*x) + p2*np.sin(p1*x)
```

Calculate and show fit parameters. Use a starting guess of p1=1 and p2=0.2

In [5]:

```
popt, pcov = curve_fit(func, xdata, ydata,p0=(1.0,0.2))
popt
```

Out[5]:

```
array([ 1.88185099,  0.70022986])
```

Calculate and show sum of squares of residuals since it's not given by the curve_fit function

In [6]:

```
p1 = popt[0]
p2 = popt[1]
residuals = ydata - func(xdata,p1,p2)
fres = sum(residuals**2)
fres
```

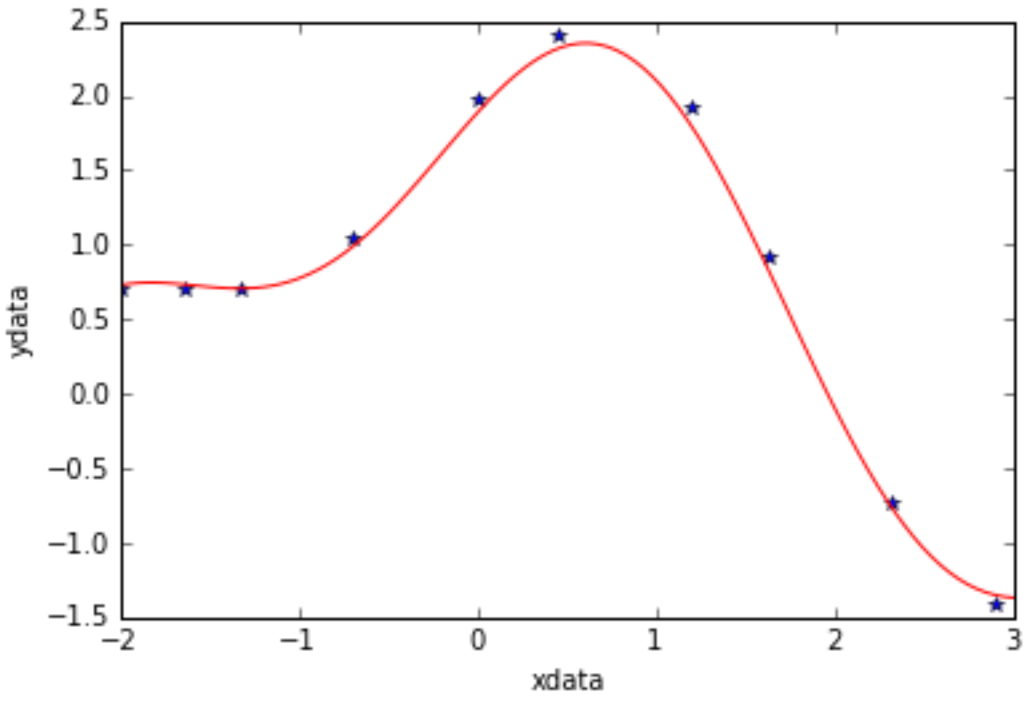
Out[6]:

```
0.05381269641876392
```

Plot fitted curve along with data

In [7]:

```
curvex=np.linspace(-2,3,100)
curvey=func(curvex,p1,p2)
plt.plot(xdata,ydata,'*')
plt.plot(curvex,curvey,'r')
plt.xlabel('xdata')
plt.ylabel('ydata');
```



In [8]:

```
print(p1, p2)
```

```
1.88185099417 0.700229857414
```

In [ ]:

```

```