1 Gaussian processes

1.1 1

```
from pylab import *
from numpy import *

close('all')
#  q1

# kernel = lambda  x1, x2, theta : \
# theta[0] * exp(- theta[1] / 2 * (x1 - x2)**2) + theta[2] + theta[3] * x1.T.dot
def kernel(x1, x2, theta):
    """
    Code for equation 3
    """
    sigma = np.zeros((len(x1), len(x2)))
    for idx, i in enumerate(x1):
        for jdx, j in enumerate(x2):
            sigma[ idx, jdx ] = \
                theta[ 0 ] * np.exp(- .5 * theta[ 1 ] * (i - j)**2) + \
                theta[ 2 ] + theta[ 3 ] * i * j
    return sigma
```