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

def sigmoid(x):
    return 1 / (1 + np.exp(-x))

def back_prop_w1(g, y, x1):
    return (-2)*(g-y)*y*(1-y)*x1

def back_prop_w2(g, y, x2):
    return (-2)*(g-y)*y*(1-y)*x2

def back_prop_theta(g, y):
    return 2*(g-y)*y*(1-y)

def feedforward(x1, x2, w1, w2, theta):
    return sigmoid(x1*w1 + x2*w2 - theta)

w1 = 0.1
w2 = 0.1
theta = 0.5
lr = 0.1
epoch = 200

wb1 = list()
wb2 = list()

for i in range(epoch):
    if i % 4 == 0:
        x1 = 0
        x2 = 0
        g = 0
    elif i % 4 == 1:
        x1 = 1
```

```
        x2 = 0
        g = 0
    elif i % 4 == 2:
        x1 = 0
        x2 = 1
        g = 0
    elif i % 4 == 3:
        x1 = 1
        x2 = 1
        g = 1

    y=feedforward(x1, x2, w1, w2, theta)
    w1 = w1 - lr*back_prop_w1(g, y, x1)
    w2 = w2 - lr*back_prop_w2(g, y, x2)
    #theta = theta - lr*back_prop_theta(g, y)

    print("y\t:",y, "g\t:",g,"g-y\t:",g-y,"x1\t:",x1,"x2\t:", x2)
    print("w1\t:", w1, "w2\t:", w2, "theta\t:", theta)
    print("-------------------------------------------------")

    wb1.append(w1)
    wb2.append(w2)

x = np.arange(1, 200, 1)

plt.figure(1)
plt.plot(x, wb1[1:200], 'k')

plt.figure(2)
plt.plot(x, wb2[1:200], 'k')
```