10주. 신경망 학습			
학번	32170578	이름	김산

Q1 (2.5점) 강의 slide 15 에 있는 example 1을 python 코드를 작성하여 실행 결과를 보이시오. (repeat 는 10 까지 한다)

Source code :

```
import numpy as np

def ACTIVE_FUNC(x):
    return x

def delta_rule(x, w, d, alpha, repeat):
    for i in range(repeat):
        v = np.sum(w * x)
        y = ACTIVE_FUNC(v)
        e = d - y
        print("error",i,e)
        print(w)
        w = w + alpha * e * x

x = np.array([0.5, 0.8, 0.2])
w = np.array([0.4, 0.7, 0.8])
d = 1

delta_rule(x, w, d, 0.5, 10)
```

실행화면 캡쳐:

```
error 0 0.07999999999999996
error 1 0.04279999999999995
error 2 0.022897999999999863
error 3 0.012250430000000034
error 4 0.006553980049999963
error 5 0.00350637932675002
error 6 0.001875912939811153
error 7 0.0010036134227988658
error 8 0.0005369331811975186
error 9 0.0002872592519406192
```

Q2 (2.5점) 강의 slide 24 에 있는 Simple Delta rule 코드를 완성하여 실행 결과를 보이시오

Source code :

```
## simple delta rule
x = np.array([0.5,0.8,0.2]) # input
w = np.array([0.4,0.7,0.8]) # weight
d = 1 # 정답
alpha = 0.5

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

def simple_delta_rule(x, w, d, alpha, repeat):
    for i in range(repeat):
        v = np.sum(w * x)
        y = SIGMOID(v)
        e = d - y
        print("error",i,e)
        w = w + alpha * y * (1 - y) * e * x

x = np.array([0.5, 0.8, 0.2])
w = np.array([0.4, 0.7, 0.8])
d = 1

simple_delta_rule(x, w, d, 0.5, 50)
```

실행화면 캡쳐:

```
error 0 0.2849578942990102
error 1 0.2794887691927339
error 2 0.2742491010755598
error 3 0.26922614783872123
error 4 0.26440792063416385
error 5 0.25978315219123826
error 6 0.25534126252533806
error 7 0.25107232327280227
error 8 0.2469670215879135
error 9 0.24301662429965365
error 10 0.23921294283737404
error 11 0.23554829928650334
error 12 0.23201549382012487
error 13 0.22860777366327356
error 14 0.22531880367881096
error 15 0.222142638612413
error 16 0.21907369699602874
error 17 0.2161067366812902
error 18 0.21323683195453502
error 19 0.21045935217148426
error 20 0.20776994184079545
error 21 0.20516450208052606
error 22 0.20263917336909443
error 23 0.20019031951192723
error 24 0.19781451274606
error 25 0.1955085199071097
error 26 0.19326928958591072
error 27 0.19109394020546977
error 28 0.188979748952539
error 29 0.18692414150189518
error 30 0.18492468247524263
error 31 0.1829790665804365
error 32 0.18108511038041042
error 33 0.17924074464474082
error 34 0.17744400724015885
error 35 0.17569303651953594
error 36 0.17398606517187687
error 37 0.1723214144986963
error 38 0.1706974890847862
error 39 0.1691127718338511
error 40 0.16756581934176817
error 41 0.1660552575823464
error 42 0.16457977788241818
error 43 0.16313813316490478
error 44 0.16172913444016468
error 45 0.16035164752747189
error 46 0.15900458998988964
error 47 0.15768692826710384
error 48 0.15639767499198376
error 49 0.15513588647773924
```