

## EX 4

### CODING:

```
import math,random
```

```
X=[[0,0],[0,1],[1,0],[1,1]]
```

```
Y=[0,1,1,0]
```

```
sig=lambda x:1/(1+math.exp(-x))
```

```
ds=lambda y:y*(1-y)
```

```
w=[random.random() for _ in range(6)]
```

```
lr=0.5
```

```
for _ in range(8000):
```

```
    for x,y in zip(X,Y):
```

```
        h1=sig(x[0]*w[0]+x[1]*w[1])
```

```
        h2=sig(x[0]*w[2]+x[1]*w[3])
```

```
        o=sig(h1*w[4]+h2*w[5])
```

```
        e=y-o
```

```
        w[4]+=lr*e*ds(o)*h1
```

```
        w[5]+=lr*e*ds(o)*h2
```

```
        w[0]+=lr*e*ds(o)*w[4]*ds(h1)*x[0]
```

```
        w[1]+=lr*e*ds(o)*w[4]*ds(h1)*x[1]
```

```
        w[2]+=lr*e*ds(o)*w[5]*ds(h2)*x[0]
```

```
        w[3]+=lr*e*ds(o)*w[5]*ds(h2)*x[1]
```

```
for x in X:
```

```
    h1=sig(x[0]*w[0]+x[1]*w[1])
```

```
    h2=sig(x[0]*w[2]+x[1]*w[3])
```

```
    print(x,"->",round(sig(h1*w[4]+h2*w[5])))
```

## OUTPUT:

The screenshot shows a Jupyter Notebook cell with the following content:

```
main.py
```

```
1 import math,random
2
3 X=[[0,0],[0,1],[1,0],[1,1]]
4 Y=[0,1,1,0]
5
6 sig=lambda x:1/(1+math.exp(-x))
7 ds=lambda y:y*(1-y)
8
9 w=[random.random() for _ in range(6)]
10 lr=0.5
11
12 for _ in range(8000):
13     for x,y in zip(X,Y):
14         h1=sig(x[0]*w[0]+x[1]*w[1])
15         h2=sig(x[0]*w[2]+x[1]*w[3])
16         o=sig(h1*w[4]+h2*w[5])
17         e=y-o
18         w[4]+=lr*e*ds(o)*h1; w[5]+=lr*e*ds(o)*h2
19         w[0]+=lr*e*ds(o)*w[4]*ds(h1)*x[0]
20         w[1]+=lr*e*ds(o)*w[4]*ds(h1)*x[1]
21         w[2]+=lr*e*ds(o)*w[5]*ds(h2)*x[0]
22         w[3]+=lr*e*ds(o)*w[5]*ds(h2)*x[1]
23
24 for x in X:
25     h1=sig(x[0]*w[0]+x[1]*w[1])
26     h2=sig(x[0]*w[2]+x[1]*w[3])
27     print(x,"->",round(sig(h1*w[4]+h2*w[5])))
28
```

The output pane shows the results of the code execution:

```
{0, 0} -> 0
{0, 1} -> 1
{1, 0} -> 1
{1, 1} -> 0
*** Code Execution Successful ***
```