

```
import numpy as np
X = np.array([[0,0],[0,1],[1,0],[1,1]])
Y = np.array([[0],[1],[1],[0]])
```

```
np.random.seed(0)
W1 = np.random.uniform(size=(2,2))
b1 = np.random.uniform(size=(1,2))
W2 = np.random.uniform(size=(2,1))
b2 = np.random.uniform(size=(1,1))
```

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

def sigmoid_derivative(x):
    return x * (1 - x)
```

```
lr = 0.1
epochs = 10000

for _ in range(epochs):
    hidden_input = np.dot(X, W1) + b1
    hidden_output = sigmoid(hidden_input)
    final_input = np.dot(hidden_output, W2) + b2
    final_output = sigmoid(final_input)

    error = Y - final_output
    d_output = error * sigmoid_derivative(final_output)
    d_hidden = d_output.dot(W2.T) * sigmoid_derivative(hidden_output)

    W2 += hidden_output.T.dot(d_output) * lr
    b2 += np.sum(d_output, axis=0, keepdims=True) * lr
    W1 += X.T.dot(d_hidden) * lr
    b1 += np.sum(d_hidden, axis=0, keepdims=True) * lr
```

```
print(np.round(final_output))
```

```
[[0.]
 [1.]
 [1.]
 [0.]]
```