Task for Lab 3

February 7, 2018

In [1]:
```
x1 = [0, 0, 1, 1]
x2 = [0, 1, 0, 1]
y = [0, 0, 0, 1]
w1 = 0.3
w2 = -0.1
n = 0.1 # Learning Rate
th = 0.2 # Threshold
```

In [2]:
```
for i in range(5):
    print('='*36, 'Epoch:', i+1, '='*36)
    error = []
    temp = []

    for j in range(len(x1)):
        y_pred =x1[j] * w1 + x2[j] * w2 # Calculating the Y prediction
                                        # h(x) = theta1 * x1 + theta2 * x2
        if y_pred < th:
            # If the predicted value is lesser than the threshold it will assign 0, el
            y_pred = 0
        else:
            y_pred = 1

        cost=y[j]-y_pred # Cost Function
        temp.append(y_pred)
        error.append(cost)

        if temp == y:
#             print(w1, w2)
            print('Final Result: ')
            print( 'Inputs:', x1[j], x2[j],'Outputs:', y[j], 'Old Weight:', w1_temp, w
                  'Output: ', y_pred, 'Cost:', cost, 'New Weight:', w1, w2)
            break
        else:
            w1_temp = w1
            # Updating the weights w1 and w2
            # Rule: Old weight + leraning rate * input * cost
            w1 = w1 + n * x1[j] * cost
            w1 = float("{0:.2f}".format(w1))
```

```
                    w2_temp = w2
                    w2 = w2 + n * x2[j] * cost
                    w2 = float("{0:.2f}".format(w2))

                print( 'Inputs:', x1[j], x2[j],'Outputs:', y[j], 'Old Weight:', w1_temp, w2_ter
                      'Output: ', y_pred, 'Cost:', cost, 'New Weight:', w1, w2)
```

```
================================== Epoch: 1 ==================================
Inputs: 0 0 Outputs: 0 Old Weight: 0.3 -0.1 Output:  0 Cost: 0 New Weight: 0.3 -0.1
Inputs: 0 1 Outputs: 0 Old Weight: 0.3 -0.1 Output:  0 Cost: 0 New Weight: 0.3 -0.1
Inputs: 1 0 Outputs: 0 Old Weight: 0.3 -0.1 Output:  1 Cost: -1 New Weight: 0.2 -0.1
Inputs: 1 1 Outputs: 1 Old Weight: 0.2 -0.1 Output:  0 Cost: 1 New Weight: 0.3 0.0
================================== Epoch: 2 ==================================
Inputs: 0 0 Outputs: 0 Old Weight: 0.3 0.0 Output:  0 Cost: 0 New Weight: 0.3 0.0
Inputs: 0 1 Outputs: 0 Old Weight: 0.3 0.0 Output:  0 Cost: 0 New Weight: 0.3 0.0
Inputs: 1 0 Outputs: 0 Old Weight: 0.3 0.0 Output:  1 Cost: -1 New Weight: 0.2 0.0
Inputs: 1 1 Outputs: 1 Old Weight: 0.2 0.0 Output:  1 Cost: 0 New Weight: 0.2 0.0
================================== Epoch: 3 ==================================
Inputs: 0 0 Outputs: 0 Old Weight: 0.2 0.0 Output:  0 Cost: 0 New Weight: 0.2 0.0
Inputs: 0 1 Outputs: 0 Old Weight: 0.2 0.0 Output:  0 Cost: 0 New Weight: 0.2 0.0
Inputs: 1 0 Outputs: 0 Old Weight: 0.2 0.0 Output:  1 Cost: -1 New Weight: 0.1 0.0
Inputs: 1 1 Outputs: 1 Old Weight: 0.1 0.0 Output:  0 Cost: 1 New Weight: 0.2 0.1
================================== Epoch: 4 ==================================
Inputs: 0 0 Outputs: 0 Old Weight: 0.2 0.1 Output:  0 Cost: 0 New Weight: 0.2 0.1
Inputs: 0 1 Outputs: 0 Old Weight: 0.2 0.1 Output:  0 Cost: 0 New Weight: 0.2 0.1
Inputs: 1 0 Outputs: 0 Old Weight: 0.2 0.1 Output:  1 Cost: -1 New Weight: 0.1 0.1
Inputs: 1 1 Outputs: 1 Old Weight: 0.1 0.1 Output:  1 Cost: 0 New Weight: 0.1 0.1
================================== Epoch: 5 ==================================
Inputs: 0 0 Outputs: 0 Old Weight: 0.1 0.1 Output:  0 Cost: 0 New Weight: 0.1 0.1
Inputs: 0 1 Outputs: 0 Old Weight: 0.1 0.1 Output:  0 Cost: 0 New Weight: 0.1 0.1
Inputs: 1 0 Outputs: 0 Old Weight: 0.1 0.1 Output:  0 Cost: 0 New Weight: 0.1 0.1
Final Result:
Inputs: 1 1 Outputs: 1 Old Weight: 0.1 0.1 Output:  1 Cost: 0 New Weight: 0.1 0.1
```

0.1 Thank You

Shahariar Rabby
151 - 15 - 5424