

Assignment 9

```
In [1]: import numpy as np

class NeuralNetwork:
    def __init__(self, num_inputs, num_hidden, num_outputs):
        self.num_inputs = num_inputs
        self.num_hidden = num_hidden
        self.num_outputs = num_outputs

        self.weights1 = np.random.randn(self.num_inputs, self.num_hidden)
        self.weights2 = np.random.randn(self.num_hidden, self.num_outputs)

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

    def sigmoid_derivative(self, x):
        return x * (1 - x)

    def feed_forward(self, inputs):
        self.hidden_layer = self.sigmoid(np.dot(inputs, self.weights1))
        self.output_layer = self.sigmoid(np.dot(self.hidden_layer, self.weights2))
        return self.output_layer

    def back_propagation(self, inputs, targets, learning_rate):
        output_error = targets - self.output_layer
        output_delta = output_error * self.sigmoid_derivative(self.output_layer)

        hidden_error = np.dot(output_delta, self.weights2.T)
        hidden_delta = hidden_error * self.sigmoid_derivative(self.hidden_layer)

        self.weights2 += learning_rate * np.dot(self.hidden_layer.T, output_delta)
        self.weights1 += learning_rate * np.dot(inputs.T, hidden_delta)

    def train(self, inputs, targets, num_epochs, learning_rate):
        for i in range(num_epochs):
            self.feed_forward(inputs)
            self.back_propagation(inputs, targets, learning_rate)
```

```
In [2]: # Create a neural network with 2 inputs, 3 hidden neurons, and 1 output
nn = NeuralNetwork(2, 3, 1)

# Train the neural network with some training data
inputs = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])
targets = np.array([[0], [1], [1], [0]])
nn.train(inputs, targets, num_epochs=10000, learning_rate=0.1)

# Test the neural network on some new input data
test_inputs = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])
for input_ in test_inputs:
    print(input_, nn.feed_forward(input_))
```

```
[0 0] [0.08358402]
[0 1] [0.49692098]
[1 0] [0.90615264]
[1 1] [0.11919635]
```