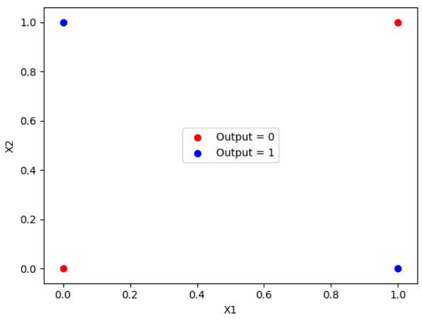
BACKPROPAGATION - IRM2016502

Question:

Using two input one output X-NOR data , train a neural network using Back Propagation Algorithm.

Procedure:

The XOR and XNOR gates cannot be learned by a simple perceptron so the idea of backpropagation and hidden layer is used to overcome the problem.



XOR GATE

- The weights are initialised randomly.
- Iterating over the data for every epoch

    i. Compute the predicted output using the sigmoid function

    ii. Compute the loss using the square error loss function

    iii. W(new) = W(old) — α ∆W

    iv. B(new) = B(old) — α ∆B

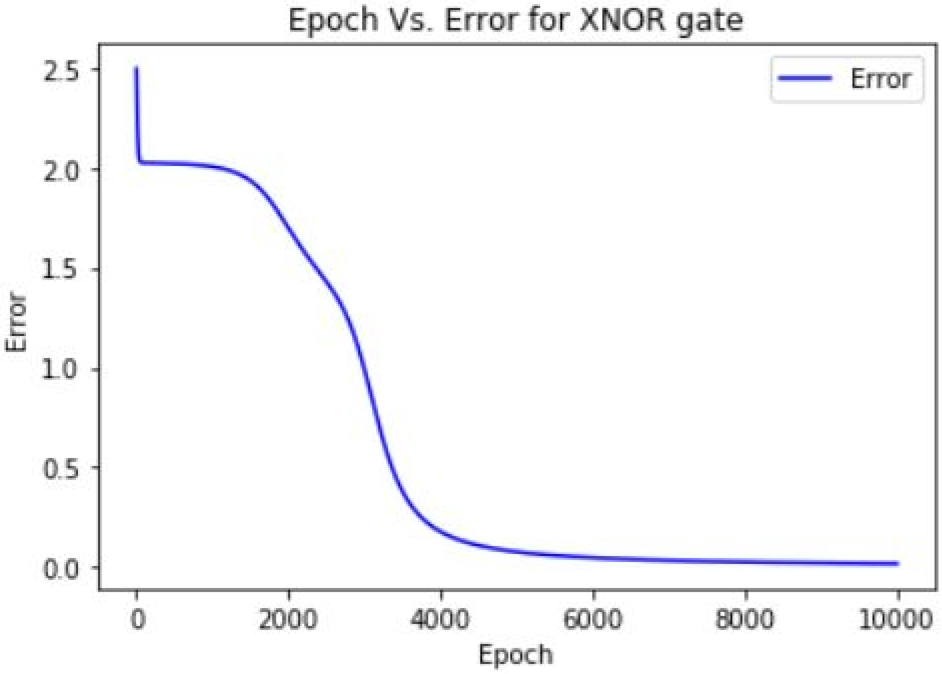
    Where α is the learning rate.

Repeat until the error is minimal

This is a fairly simple learning algorithm consisting of only arithmetic operations to update the weights and biases. The algorithm can be divided into two parts: the forward pass and the backward pass. As in the code described by two functions **forward_propagate and backward_propagate.**

In the code I’ve used a single hidden layer with 2 neurons.

The epoch vs error graph is shown below.



The correctness of the network is assumed by the predicted values of the given input being correct and the above graph representing the decrease in error as the number of epochs increased.

Reference

https://machinelearningmastery.com/implement-backpropagation-algorithm-scratch-python/