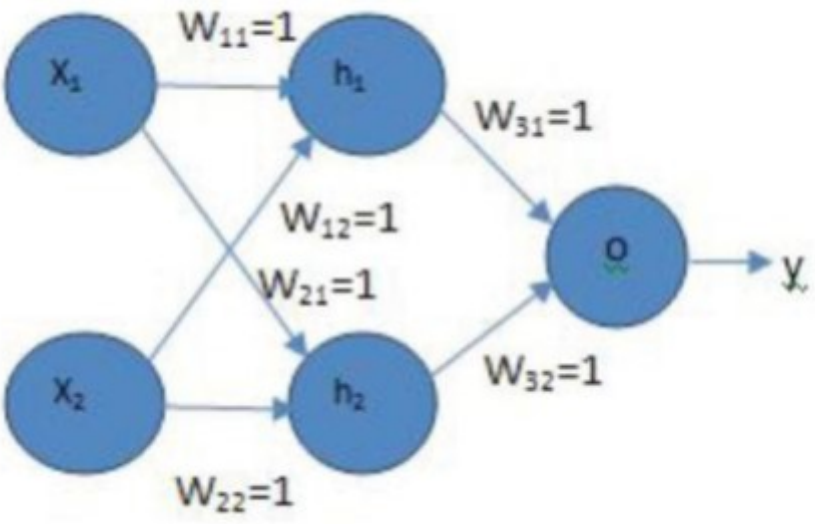
SCOA Assignment 6

Roll No. : 41163

Title : Single Hidden Layer Network

Problem Statement : The figure shows a single hidden layer neural network. The weights are initialized to 1‟s as shown in the diagram and all biases are initialized to 0‟s. Assume all the neurons have linear activation functions. The neural network is to be trained with stochastic (online) gradient descent. The first training example is [x1=1, x2=0] and the desired output is 1. Design the back-propagation algorithm to find the updated value for W11 after backpropagation.



Objective :

- To understand the concepts of forward propogation, backward propogation and stochastic gradient descent in neural networks.
- To understand how neural networks are trained.

Outcome :

- Understood and implemented single layer hidden neural network.

Software Requirements :

- Jupyter Notebook
- Python 3.8.5
- 64 bit OS

Hardware Requirements :

- Machine with 64 bit processor

Theory :

Artificial Neural Networks:

- Artificial Neural Networks(ANN) are computing systems vaguely inspired by the biological neural networks that constitute animal brains.
- It is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain.
- Each connection, like the synapses in a biological brain, can transmit a signal to other neurons, the neuron that receives a signal then processes it and can signal neurons connected to it.
- The signal at a connection is a real number, and the output of each neuron is computed by some non-linear function of the sum of its inputs.
- The connections are callededges. Neurons and edges typically have a weight that adjusts as learning proceeds.
- The weight increases or decreases the strength of the signal at a connection.
- Typically, neurons are aggregated into layers. Different layers may perform different transformations on their inputs.
- Signals travel from the first layer (the input layer), to the last layer (the output layer), possibly after traversing the layers multiple times.

Learning :

Learning is the adaptation of the network to better handle a task by considering sample observations. Learning involves adjusting the weights (and optional thresholds) of the network to improve the accuracy of the result. This is done by minimizing the observed errors. Learning is complete when examining additional observations does not usefully reduce the error rate.

Activation Function :

Activation function is the transformation operation performed over the input by a neuron before sending it to the next layer of neurons.
e.g. RELU, Sigmoid, tanh

Cost Function :

A function that measures the performance of a machine learning model for given data.

Back Propogation :

Backpropagation is a method used to adjust the connection weights to compensate for each error found during learning. The error amount is effectively divided among the connections. Technically, backprop calculates the gradient (the derivative) of the cost function associated with a given state with respect to the weights. The weight updates can be done via stochastic gradient descent or other methods.

Test Cases :

Description	**Expected O/P**	**Actual O/P**
Give a custom input to the model.	Display the values of all weights at the end of every epoch.	Successful
Give input in problem statement. i.e. X = [1,0] Y = [1] epoch = 10	W1= [[0.99, 0.99] [0.99, 0.99]] b1 = [[-0.01] [-0.01]]	Successful

Conclusion :

Therefore, successsfully understood the concept of Artificial Neural Networks and implemented single hidden layer neural network for given data.