Exercise 1
International Summer Course NLP

Prepared by Sneha Oram (23M2159)

June 26, 2024

Introduction

XOR stands for exclusive OR, is a classic problem in machine learning. It takes 2 binary inputs and returns true if exactly one of the inputs is true.

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Figure 1: Truth table for XOR network

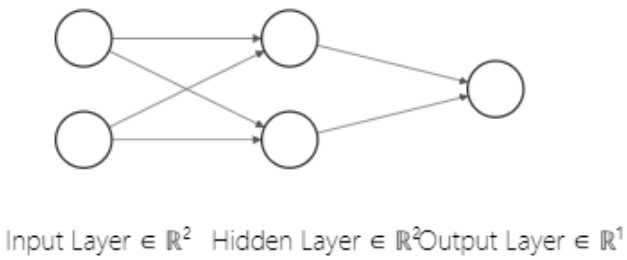

Figure 2: Typical neural network for XOR operation

Given

You are provided with a python code which is implementing the XOR network from scratch. Note the structure of the artificial neural network. There are 2 neurons in the input side, 2 neurons in the hidden layer, and 1 neuron in the output side. As mentioned in the above table,

for the given input pair, the output has to be correctly calculated. Following are the notations used in the code:
'W1' denotes the weight matrix between the input and hidden layer,
'b1' denotes bias terms in the hidden neurons,
'W1' denotes the weight matrix between input and hidden layer,
'b1' denotes bias term in the output neuron

You are advised to take up the following tasks:

- 1. Compute the values of net1, O1, net2 and O2 in the function forward_propagation(). The notations are kept consistent with the lecture.
- 2. Update the parameters in the function update_parameters()
- 3. Use different values of epoch (i.e., iterations) and learning_rate (to direct rate of convergence) from the Hyper-parameter space given. The correct output you should obtain is [[1. 0. 0. 1.]]
- 4. Implement the XOR network with no hidden layer, and use sin as the activation function (instead of the sigmoid function).

Few Instructions

Please write your code in the designated spaces provided. See for the text – 'Write your code here'. For training use different combinations of epochs and learning rate. Please do not edit in anywhere else in the code.

Reference

Neural Networks: Zero to Hero