Cooper Tush

Dr. Sathyanarayanan Aakur

CS-4783-65788

October 7th, 2022

CS 4783 Assignment 2

Question 1

To know how we derive the update rule, we first must know how a neural network works.

There are a couple steps we need to know for any neural network: summation of the inputs and weights, the activation function, calculating all the hidden layers, and then calculating the total error. To go into more detail, the summation of the inputs and weights is done first before the activation function. With the activation function, we will be using Sigmoid which is just 1 / ( 1 + (e^(-x)) ) where x is the summation of the inputs and weights. The Sigmoid activation function is also used in each hidden layer, so we use the function two more times. Afterwards, based on what we get from each hidden layer we then use them to solve the error.

This way of using the neural network is also called Forward Pass or Forward Propagation. As it goes from the input to the output. But there is another way we can do this using Backprop or Backwards Propagation. Like Forward Pass, it uses the same elements to solve the neural network. But in this case, it goes backwards from the output to the activation function and hidden layers to try to solve the network.

Question 2

1.

I used the sigmoid activation function for the output layer. Sigmoid activation is better for classification so I used this function for the output data.

2.

The number of neurons should equal the number of outputs, so there should be only 1 neuron per 1 output layer.

5.

The learning rates control how fast the model adapts to the problem. Smaller learning rates require more training and larger rates result in rapid changes.

6. a.

As the number of nodes increases the time it takes to train the neural network also increases. But this does not change how we use the update rule.

7.

I tested the neural network with the tanh and ReLU function. Tanh is like sigmoid function, yet the function takes any value as input and gives us a output in the range of -1 to 1. A larger input will give us a more positive output and make it closer to 1. The smaller output will be -1. ReLU activation function is less likely to have vanishing gradients which function like sigmoid and Tanh can have more of. ReLU is also easy to implement and effective at overcoming these problems yet doesn’t do well with dead units.