Assignment 1:
Neural Networks

Created By:
Khoa Diep
NetID: ktd170030

Maviya Yaseen
NetID: mxy200016

Farrel Raja
NetID: fjr190000

Summary of Neural Network:

For this assignment, we implemented a simple neural network with three different activation functions: Sigmoid, Tanh, and ReLu. While building the neural network, we assumed that the input layer has 4 neurons, the hidden layer has 6 neurons (but the layer size is adjustable), and the output layer has a size of 3 neurons. This was based on the number of attributes and categories in our dataset.

Results:

For each activation function, we trained and tested our neural network by varying parameters like learning rate and epochs. The sigmoid function we used on our dataset showed that there was a positive linear relationship between the learning rate and the accuracy of the function. We also noticed that the sigmoid function yielded better results with a higher epoch number. This indicates that the function takes a bit longer to converge and give accurate results. Our best result with the sigmoid function utilized a learning rate of 0.5 with 100 epochs to produce a testing accuracy of 96.7%.

The tanh activation function also yielded favorable results. Unlike the sigmoid function, the tanh function worked better with lower learning rates. Our highest accuracy with the tanh function used a learning rate of 0.001 and 100 epochs to produce a testing accuracy of 100%. This may be due to our relatively small dataset. If we had more data available for our neural network, the testing accuracy would have been slightly lower than 100%.

While using the ReLu activation function, we noticed that some changes in the learning rate did not result in a change in accuracy. A learning rate larger than 0.0018 resulted in a testing accuracy of 30% and a training accuracy of 46.7%. This activation function worked better with lower learning rates, the ideal rate being 0.001. With a learning rate of 0.001 and 100 epochs, the reLu function was able to produce a testing accuracy of 73.3%.

Conclusion:

Overall, the best parameters that we tested for the Iris data set would be the tanh activation function, learning rate = 0.001, and number of epochs = 100. The best activation function overall was the tanh function and the worst performing activation function overall was the reLu function. Our theory as to why tanh performed the best was due to the fact that tanh punishes wrong predictions the best, while sigmoid wasn't as powerful, and ReLu does not punish wrong predictions at all.

Optimizer:

Stochastic gradient descent performs a parameter update for each training example X and label y. The Neural Network created for this project updates the weights and the bias weights after every training instance, and SGD is the only optimizer implemented in this project.

Source: https://www.ruder.io/optimizing-gradient-descent/#gradientdescentvariants

Sigmoid Output Tables:

Learning Rate	Epochs	Training Accuracy	Testing Accuracy
0.001	10	0.35	0.267
0.01	10	0.556	0.5
0.1	10	0.692	0.8
0.2	10	0.792	0.9
0.5	10	0.767	0.733
0.001	100	0.575	0.433
0.01	100	0.675	0.8
0.1	100	0.958	0.9
0.2	100	0.942	0.9
0.5	100	0.942	0.967

Tanh Output Tables:

Learning Rate	Epochs	Training Accuracy	Testing Accuracy
0.001	10	0.792	0.933
0.01	10	0.9	0.933
0.1	10	0.858	0.833
0.2	10	0.725	0.7
0.5	10	0.492	0.733
0.001	100	0.933	1.0
0.01	100	0.975	0.9
0.1	100	0.916	0.9
0.2	100	0.908	0.6
0.5	100	0.45	0.466

ReLu Output Tables:

Learning Rate	Epochs	Training Accuracy	Testing Accuracy
0.001	10	0.633	0.7
0.01	10	0.3	0.467
0.1	10	0.3	0.467
0.2	10	0.3	0.467
0.5	10	0.3	0.467
0.001	100	0.642	0.733
0.01	100	0.3	0.467
0.1	100	0.3	0.467
0.2	100	0.3	0.467
0.5	100	0.3	0.467