**2802ICT Programming Assignment 2**

Overview:

Implement a machine learning algorithm, Neural Networks, using Python. Train and test a simple neural network with datasets and experiment different settings of hyper parameters.

**Part 1: Calculation for a smaller neural network**

**Forward Propagation**

The “feedforward” or forward propagation is the structure used to calculate the neural network. It is a three-layered network with two neurons in the input, hidden and output layers. The initial weights are between 0 and 1.

*Layer One:*

W1 = W2 = B1= = Bterm \* Bweight

Finding the hidden layer values:

The activation function (sigmoid) = is then applied to the hidden layer sums:

*Layer two:*

To find the output (y1, y2) we substitute the hidden layer values as inputs for the output layer:

Apply the sigmoid function:

In forward propagation, a set of weights are applied to the input and these are used to calculate the output. In backward propagation, the margin of error is measured on the output and the weights are adjusted accordingly to decrease the error.

To improve the current model, we will use stochastic gradient descent with backward propagation.

**Backpropagation**

Formulas:

Activation Function:

*Layer Two:*

We want to know how much a change in W(weight) affects the total error, aka (the gradient with respect to the weight).

*Layer One:*

**Updating!**

(Learning Rate)

**Part 2: Python Implementation**

Parameters:

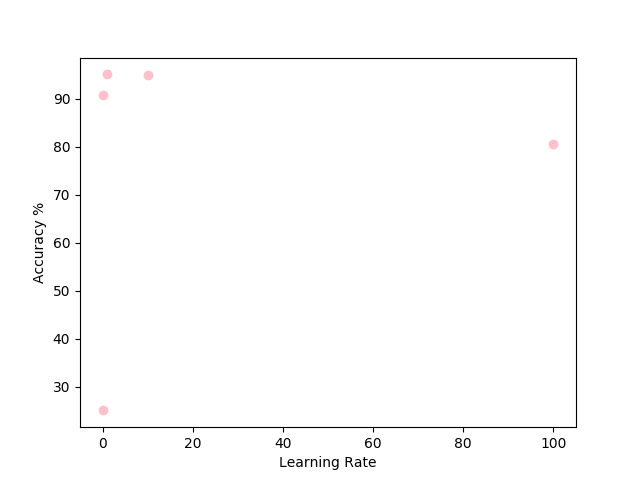
Neural net size: [784, 30, 10]

Settings:

Epoch = 30, Mini-batch size = 20, = 3.0

***Accuracy Achieved: 95.39%***

**Figure 1: Learning Rate VS. Accuracy**

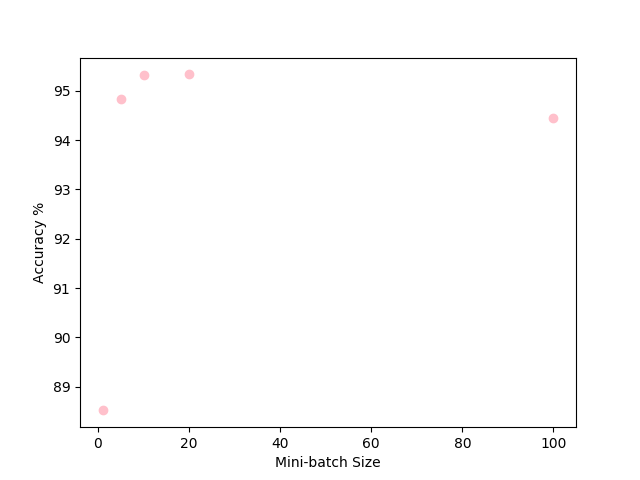
****

**Table 1: Learning Rate and Accuracy**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Learning Rate | 0.001 | 0.1 | 1.0 | 10 | 100 |
| Accuracy | 25.1% | 90.63% | 95.02% | 94.84% | 80.58% |

The maximum test accuracy achieved when updating the learning rates within this sample range, was when = 0.1. When = 1.0, the neural network achieved 95.02% accuracy which is greater than any other in the sample. However, the original = 3.0 achieved a 95.39% accuracy rate where the settings were the same. Further testing may change the optimal learning rate.

**Figure 2: Mini-batch size VS. Accuracy**



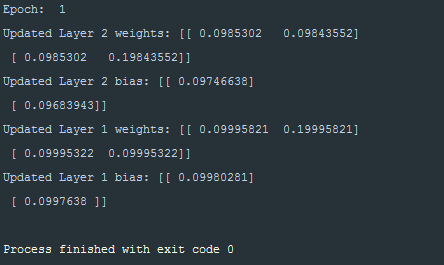
**Table 2: Mini-batch size and Accuracy**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mini-batch Size | 1 | 5 | 10 | 20 | 100 |
| Accuracy | 88.52% | 94.84% | 95.32% | 95.39% | 94.44% |

The mini-batch size sampling varied in a range of 1 to 100, the maximum accuracy achieved was when the mini-batch size was 20. Coincidentally, the original mini-batch size provided a higher accuracy of results. In figure 2, the range between mini-batch size 10 and 15 is where the accuracy is the highest and with further testing should lead to a better result.

**Verifying Part 1 Manual Calculations**

**Epoch: 1**



**Epoch: 3**

