|  |  |
| --- | --- |
| Name: | Edward Eisenberger |
| ID#: | 1066164 |
| Assignment 3 | |
| Due Date: | March 1, 2019 |
| Date of Submission: | February 27, 2019 |

Table of Contents

[Part A 3](#_Toc2189928)

[Part B 3](#_Toc2189929)

[Part C 3](#_Toc2189930)

[Part D 3](#_Toc2189931)

[Part E 3](#_Toc2189932)

[Output 3](#_Toc2189933)

# Part A

Show the matrix equations for the forward pass:

# Part B

Show the matrix equations (partial derivatives) for the backpropagation algorithm:

# Part C

The Neural Network (NeuralNetwork.py), Hidden Layer (NeuronLayer.py), and Backpropagation algorithm have been implemented in the NeuralNetworkExample project located in the Assignment03 directory.

# Part D

The Neural Network Example project implemented both the Stochastic Gradient Descent (SGD) algorithm and the Mini Batch Gradient Descent algorithms.

# Part E

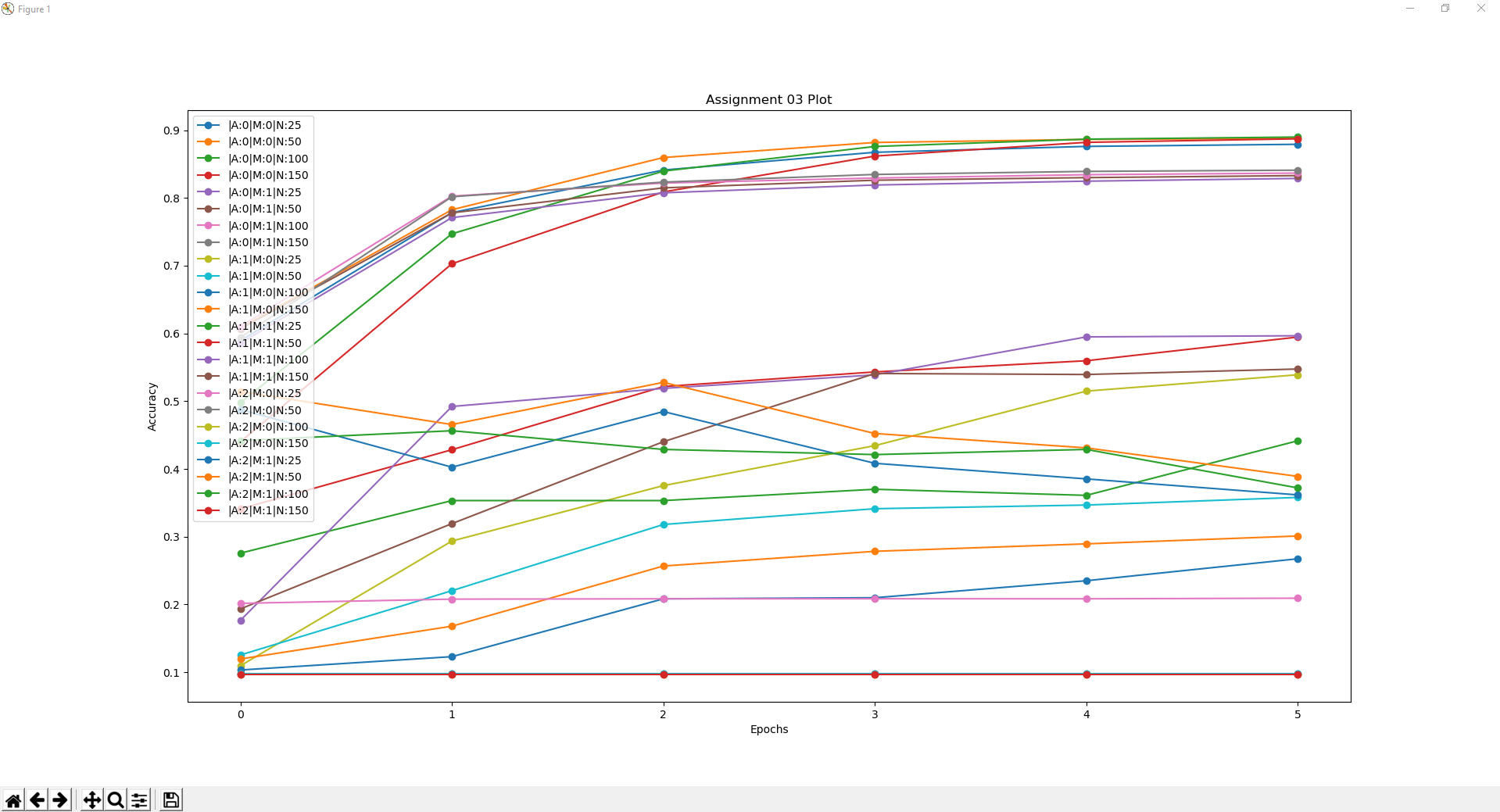
The Neural Network Example project implemented the tanh and RELU activation function.

# Output

Neural Networks using hidden layers with 25, 50, 100, and 150 neurons were each trained using every permutation of Sigmoid, Tan Hyperbolic, and Rectified Linear Unit activation functions with either the Stochastic Gradient Descent (SGD) or Mini Batch Gradient algorithm. The following table provides the description for each.

| **Key** | **Activation** | **Method** | **Neurons** |
| --- | --- | --- | --- |
| A:0|M:0|N:25 | Sigmoid | Stochastic Gradient Descent (SGD) | 25 |
| A:0|M:0|N:50 | Sigmoid | Stochastic Gradient Descent (SGD) | 50 |
| A:0|M:0|N:100 | Sigmoid | Stochastic Gradient Descent (SGD) | 100 |
| A:0|M:0|N:150 | Sigmoid | Stochastic Gradient Descent (SGD) | 150 |
| A:0|M:1|N:25 | Sigmoid | Mini Batch Gradient Descent | 25 |
| A:0|M:1|N:50 | Sigmoid | Mini Batch Gradient Descent | 50 |
| A:0|M:1|N:100 | Sigmoid | Mini Batch Gradient Descent | 100 |
| A:0|M:1|N:150 | Sigmoid | Mini Batch Gradient Descent | 150 |
| A:1|M:0|N:25 | Tan Hyperbolic | Stochastic Gradient Descent (SGD) | 25 |
| A:1|M:0|N:50 | Tan Hyperbolic | Stochastic Gradient Descent (SGD) | 50 |
| A:1|M:0|N:100 | Tan Hyperbolic | Stochastic Gradient Descent (SGD) | 100 |
| A:1|M:0|N:150 | Tan Hyperbolic | Stochastic Gradient Descent (SGD) | 150 |
| A:1|M:1|N:25 | Tan Hyperbolic | Mini Batch Gradient Descent | 25 |
| A:1|M:1|N:50 | Tan Hyperbolic | Mini Batch Gradient Descent | 50 |
| A:1|M:1|N:100 | Tan Hyperbolic | Mini Batch Gradient Descent | 100 |
| A:1|M:1|N:150 | Tan Hyperbolic | Mini Batch Gradient Descent | 150 |
| A:2|M:0|N:25 | Rectified Linear Unit | Stochastic Gradient Descent (SGD) | 25 |
| A:2|M:0|N:50 | Rectified Linear Unit | Stochastic Gradient Descent (SGD) | 50 |
| A:2|M:0|N:100 | Rectified Linear Unit | Stochastic Gradient Descent (SGD) | 100 |
| A:2|M:0|N:150 | Rectified Linear Unit | Stochastic Gradient Descent (SGD) | 150 |
| A:2|M:1|N:25 | Rectified Linear Unit | Mini Batch Gradient Descent | 25 |
| A:2|M:1|N:50 | Rectified Linear Unit | Mini Batch Gradient Descent | 50 |
| A:2|M:1|N:100 | Rectified Linear Unit | Mini Batch Gradient Descent | 100 |
| A:2|M:1|N:150 | Rectified Linear Unit | Mini Batch Gradient Descent | 150 |

The accuracy of each Neural Network is plotted in the following figure:



It appears that the Neural Network with 100 Neurons in the hidden layer using the SGD algorithm and Sigmoid activation function (A:0|M:0|N:100) reached the highest accuracy after 5 epochs. All of the Neural Networks using the Mini Batch Gradient algorithm appear to reach their peak accuracy after the least number of epochs, however they do not reach the maximum accuracy of the Neural Networks using the SGD algorithm.