1. Implementation challenges and solution approach

The implementation of neural network is challenging in all times. In our case, we are just implementing simple neural network with 1 hidden layer. Together with input and output layers, total number of layers become there with five hidden nodes. Neural network consists of forward and backward propagation for the better performance. Theory and code both are challenging for the implementation of neural network.

We have found layer structure and initialization difficult during the implementation. Each layer has several entities and nodes which are difficult in applying coding. The solution comes the structure data format in MATLAB. It gives to hold several data methods for layers. Also, forward and backward propagation are not easy to implement in MATLAB. Thanks to Professor that he has given us a template for it.

Steepest descent calculates the most optimal learning rate. To do so, it performs iterative search along line slopes. There are many lines search algorithm to perform the mentioned tasks. In our project, we have used line search bisection algorithm. Initially, I have found the algorithm challenging as well as implementation. Then, I have used divide and conquer rules to overcome the problem. By dividing the algorithm into several parts, finally I did able to successfully be implemented it.

2. Plots of error over iteration/epoch for each weight update method

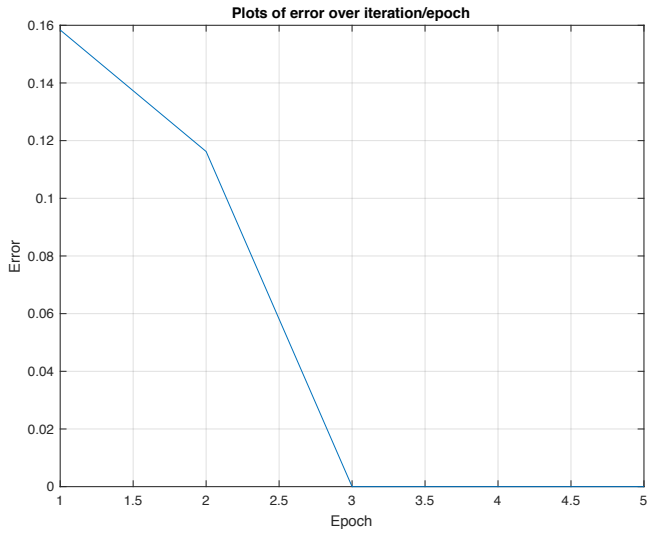

Figure 1: The batch update error plot over iteration. Initially, the error rate is larger value which decreases over the iteration. The algorithm has been tested for 10 trails. Although the instruction has been given to perform 100 trails which has suspended due to large computation time.

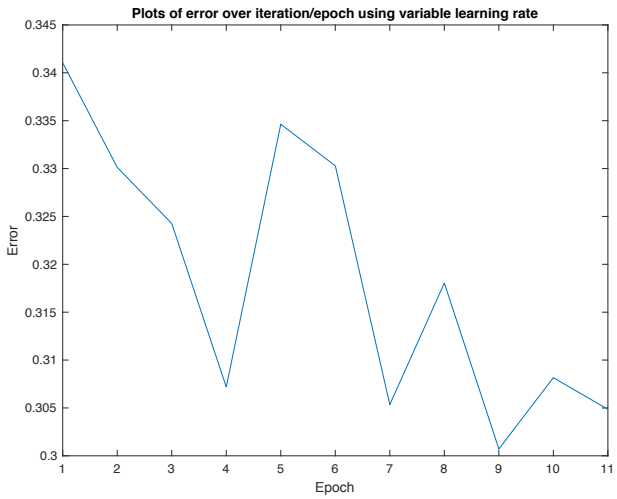

Figure 2: The variable learning method error plot over iteration. Initially, the error rate is larger value which shows periodical behavior over the iteration. The algorithm has been tested for 100 trails. It has required less computation times. Therefore, we have performed 100 trails.

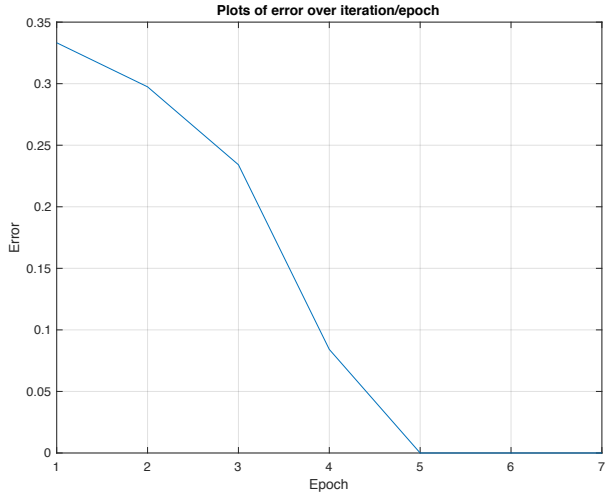

Figure 3: The Stochastic update error plot over iteration. Initially, the error rate is larger value which decreases over the iteration. The algorithm has been tested for 10 trails. Although the instruction has been given to perform 100 trails which has suspended due to large computation time.

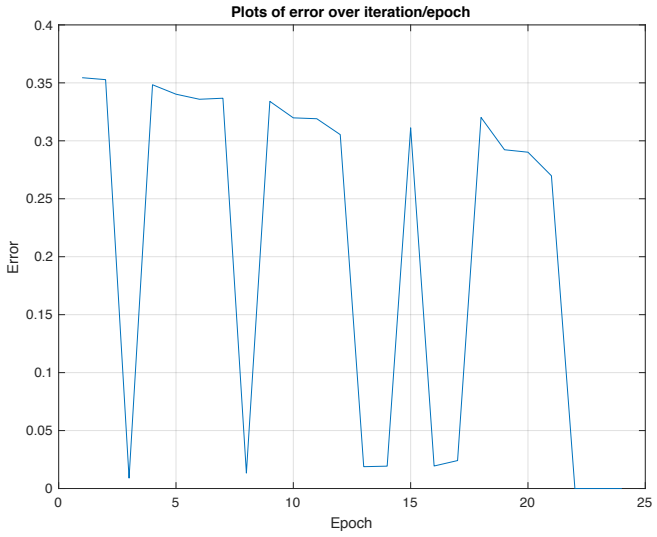

Figure 4: The Steepest update error plot over iteration. Initially, the error rate is larger value which periodic over the iteration. The algorithm has been tested for 10 trails. Although the instruction has been given to perform 100 trails which has suspended due to large computation time.

3. Average in-sample and test-sample error over 10 trials for each updated method

	In sample error, Ein	Out sample error, Eout
Variable learning rate	0.41210	0.37487
Batch Update	0.1612	0.11623
Stochastic Update	0.1116	0.18346
Steepest Descent	0.2698	0.24847

Table 1: Average in and out sample error rate over 10 trails for four different weight update method

4. Map of classification space with overlay of data-points

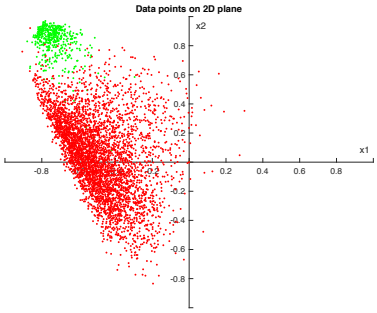

Figure 5: Original dataset

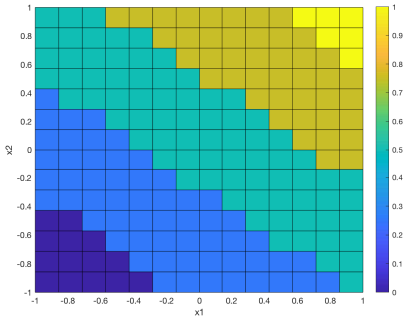

Figure 6: 2D view of Variable length update surface plot

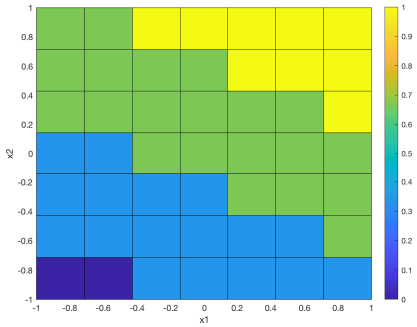

Figure 7: 2D view of Stochastic weight update surface plot

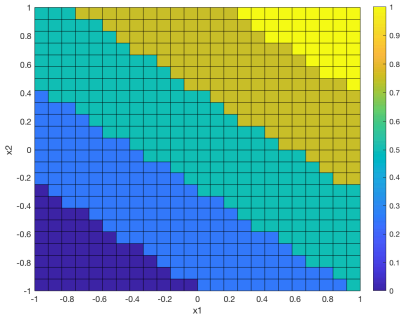

Figure 8: 2D view of Steepest descent weight update surface plot

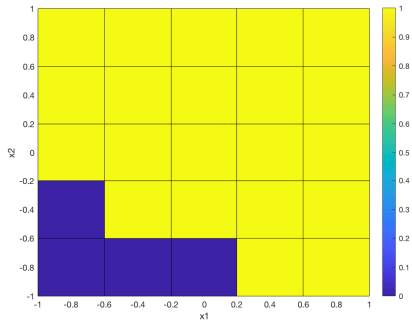

Figure 9: 2D view of Batch weight update surface plot

5. Result Discussion

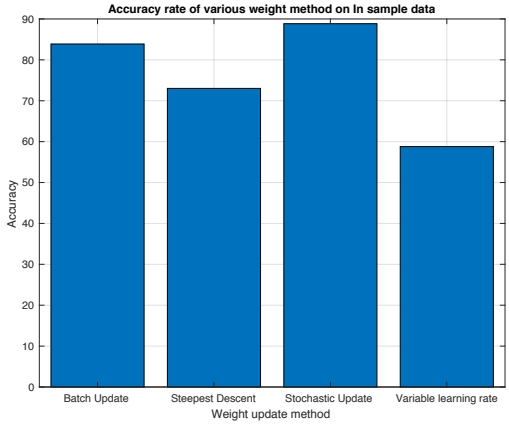

Figure 10: The plot shows the accuracy rate of neural network for various weight update method on training dataset.

The USPS dataset total average accuracy results for ‘digit 1’ can be seen in figure 10-11. The accuracy percentage from the highest to the lowest are USPS data set with 88% to 58 % for in sample dataset with our implemented neural network.

The Stochastic update generates less in sample error among all four methods. The accuracy rate is almost 85%. On the other hand, variable learning rate method depicts the less accuracy rate (below 60%) among all methods. But the variable learning method takes less convergence time. In term of convergency, batch update method requires highest time because it works on single datapoints. Steepest weight update method exhibits the moderate in sample accuracy rate as well as convergence time.

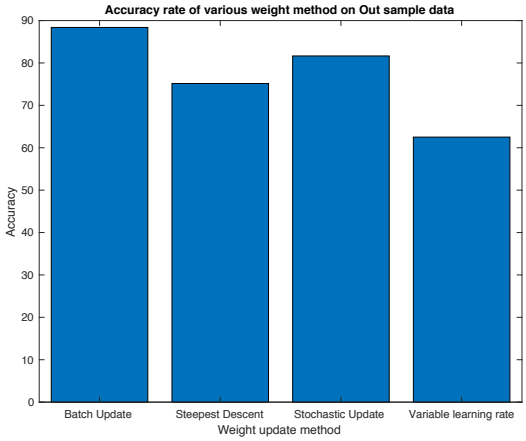

Figure 11: The plot shows the accuracy rate of neural network for various weight update method on testing dataset.

The Batch update generates less out sample error among all four methods. The accuracy rate is almost 88%. On the other hand, variable learning rate method depicts the less accuracy rate (below 70%) among all methods. But the variable learning method takes less convergence time. In term of convergency, batch update method requires highest time because it works on single datapoints. Steepest weight update method exhibits the moderate out sample accuracy rate as well as convergence time. The stochastic weight update method shows the 2nd highest out sample accuracy rate.

In conclusion, our implemented neural network exhibits 88% accuracy rate to detect ‘digit 1’ using Batch weight method.