

## **Report on Assignment-4**

**2021MCS2128**

**Bivas Ranjan Dutta**

Neural network is a very powerful tool for machine learning and deep learning. In this assignment we are supposed to make a Multilayer perceptron in C with the activation function of relu, tanh, and sigmoid. Two datasets have been given, one for classification and another for regression. For backpropagation algorithm we have to take three algorithms mainly Stochastic Gradient Descent, batch Gradient Descent and Mini batch Gradient Descent.

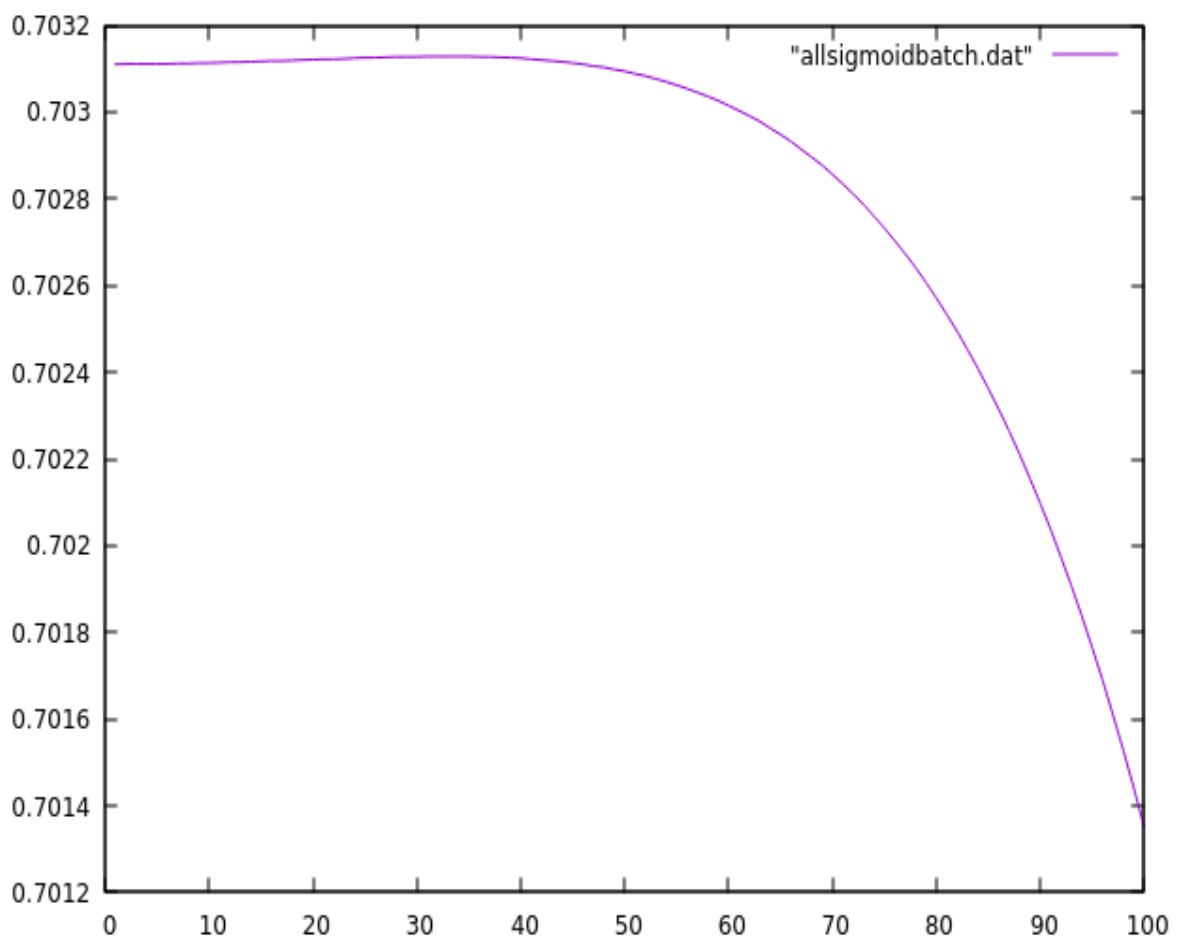
In this assignment I have used sigmoid activation function in the output layer in case of classification problem and used cross entropy error for error calculation.

For regression problem I have used Mean Square Error as a metric for error calculation.

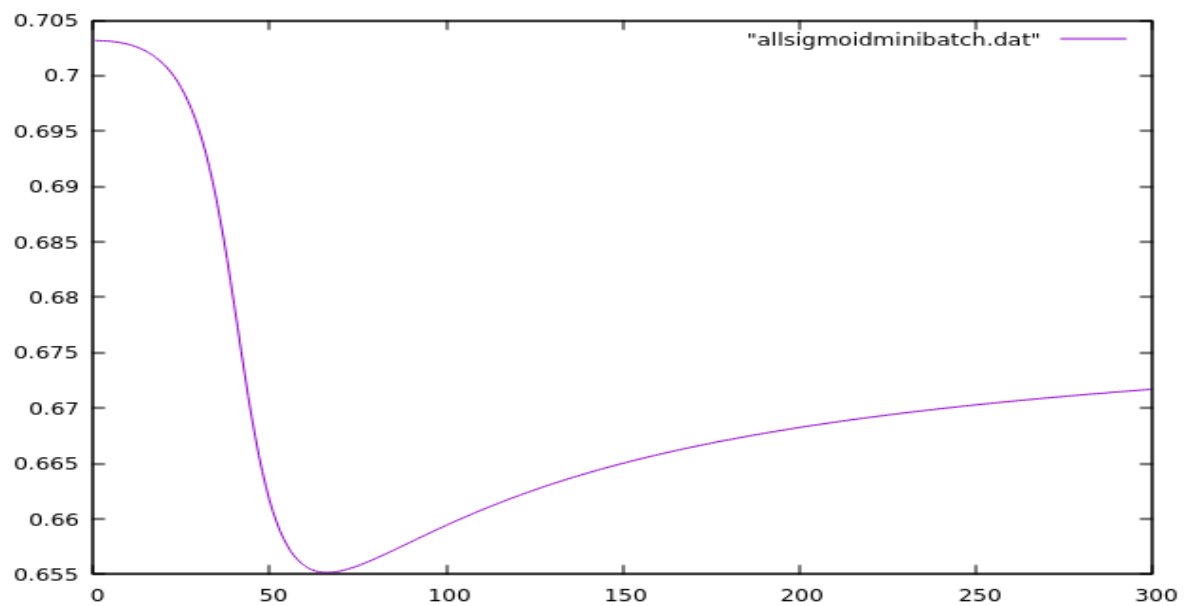
The following graph shows the Number of Iterations vs. Error calculation relationship.

Y-label indicating the error value and the x-label indicating the iteration number.

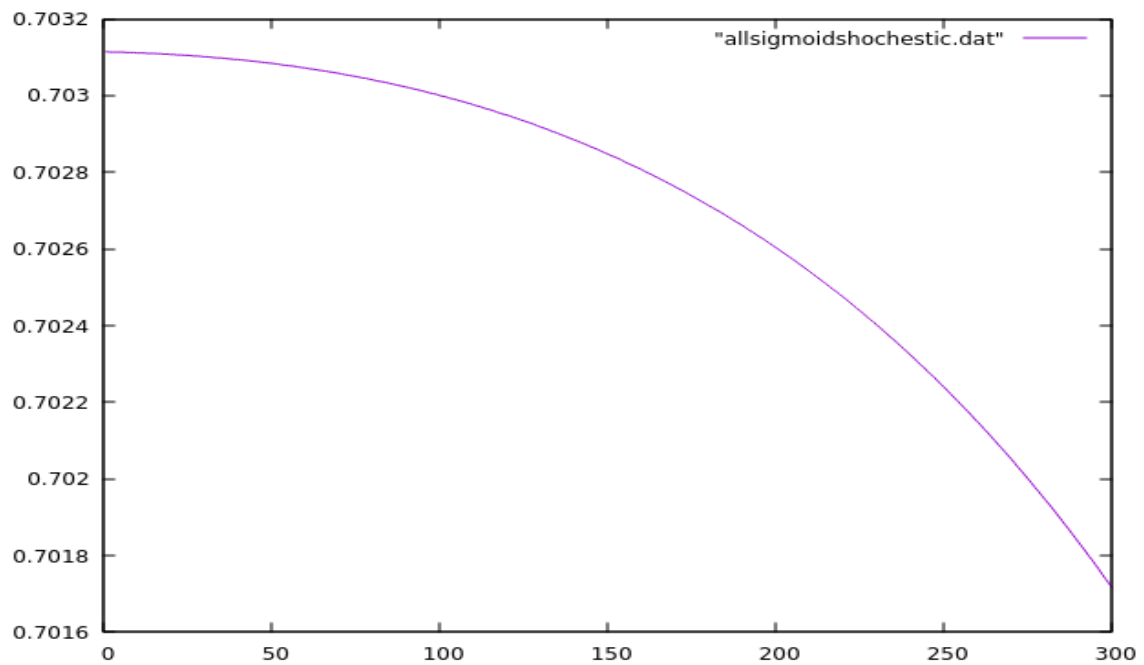
1. Batch gradient Descent and used sigmoid activation function in both output and hidden layer.



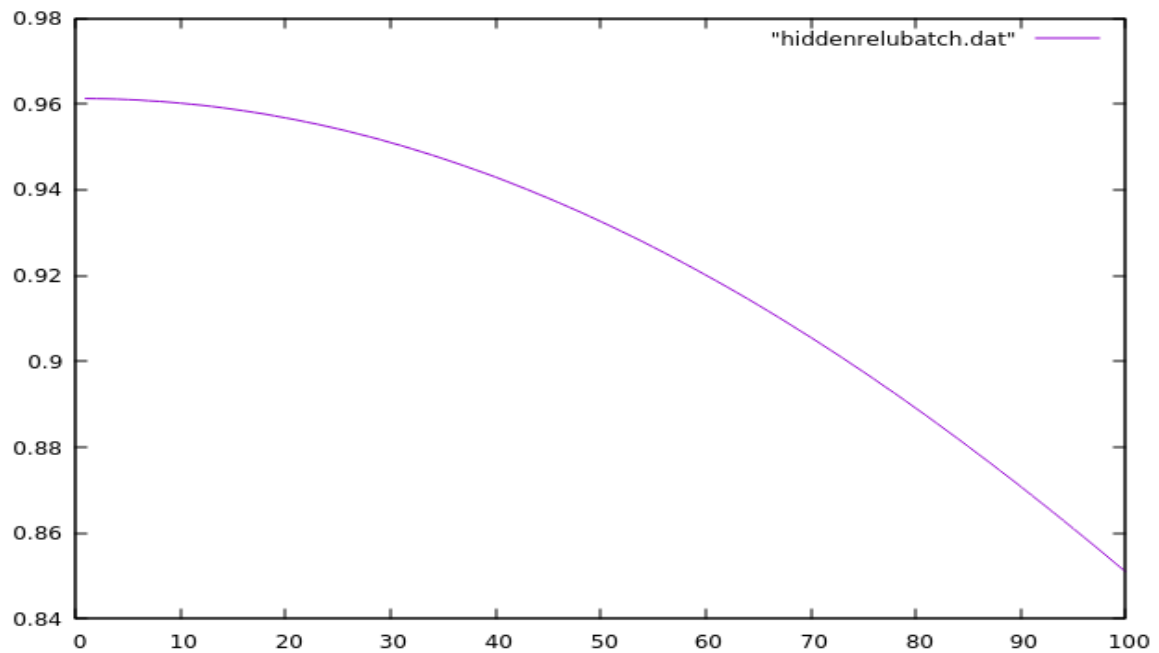
2. Mini Batch gradient Descent and used sigmoid activation function in both output and hidden layer.



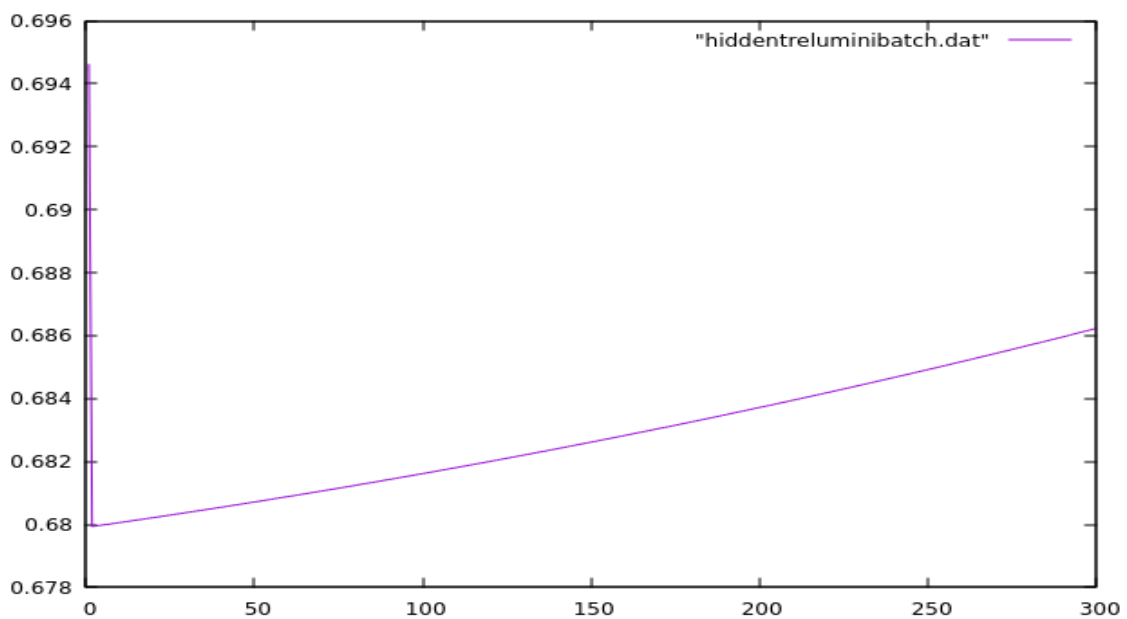
3. Stochastic gradient Descent and used sigmoid activation function in both output and hidden layer.



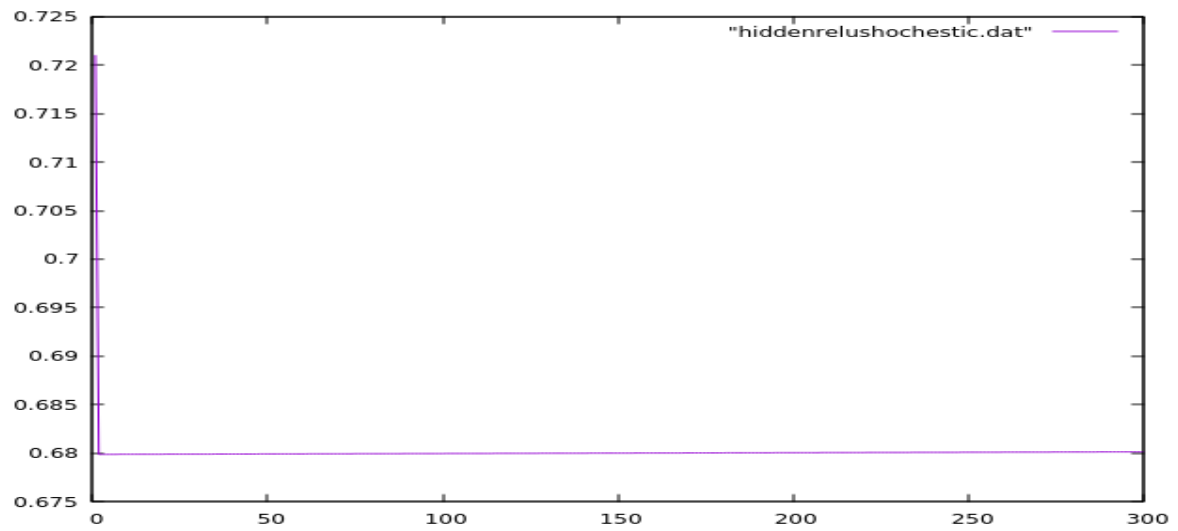
4. Batch gradient Descent and used sigmoid activation function in output layer and relu activation function in hidden layer.



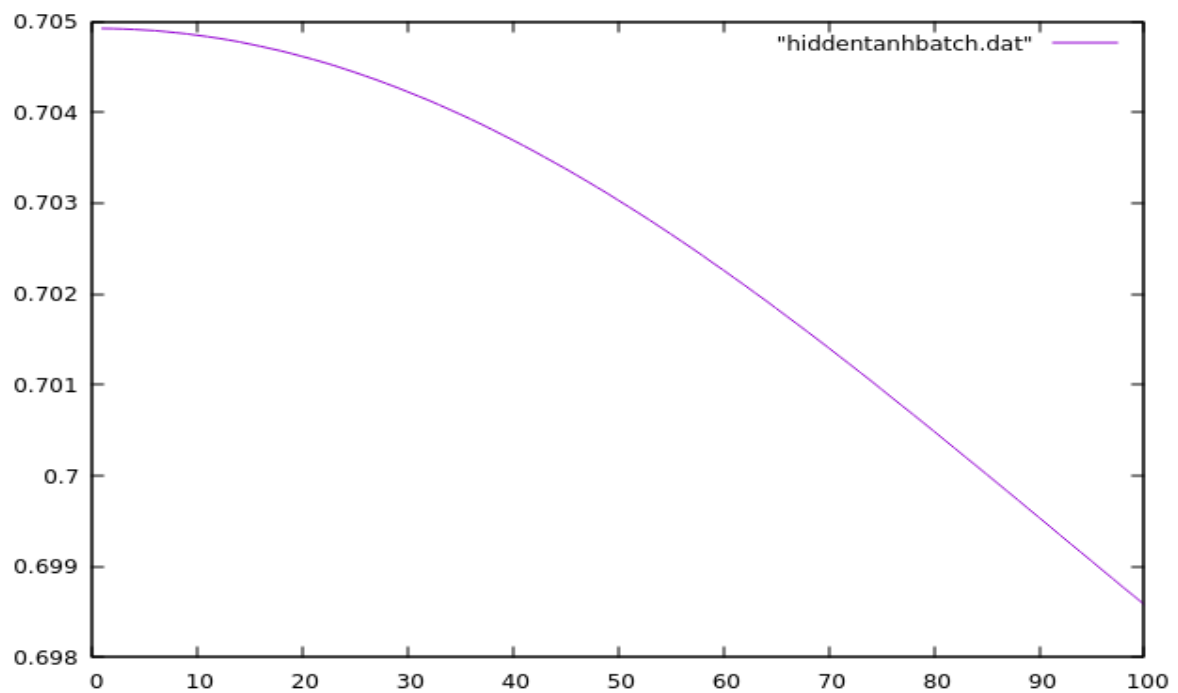
5. Mini Batch gradient Descent and used sigmoid activation function in output layer and relu activation function in hidden layer.



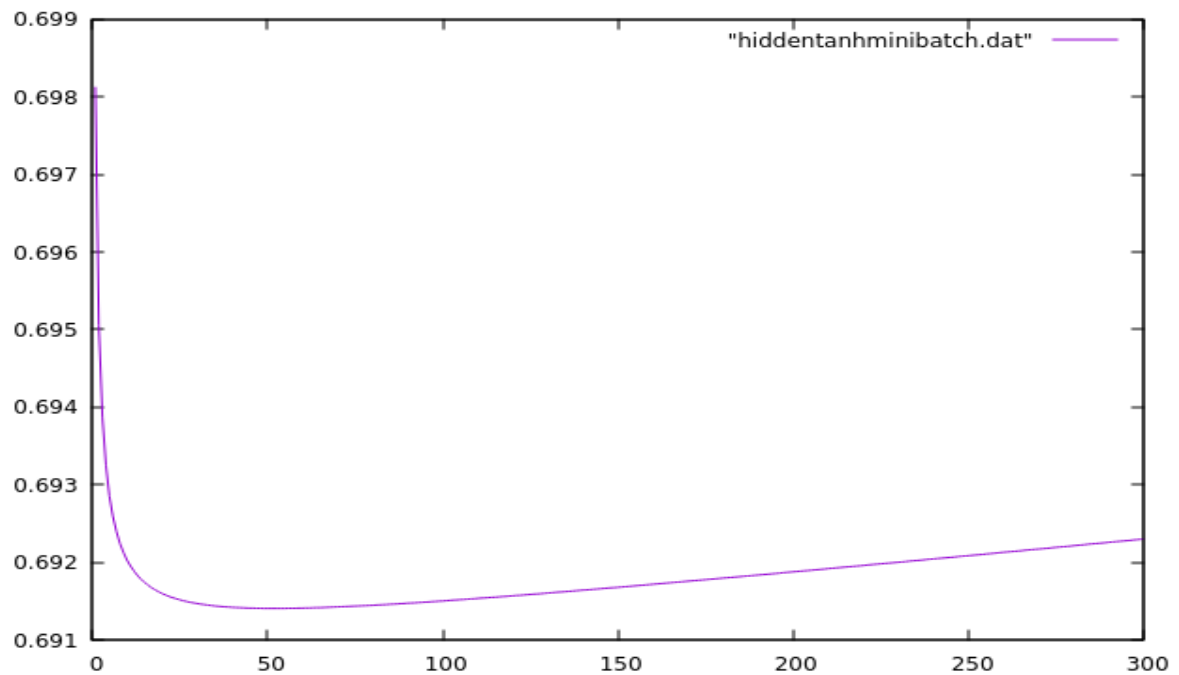
6. Stochastic gradient Descent and used sigmoid activation function in output layer and relu activation function in hidden layer.



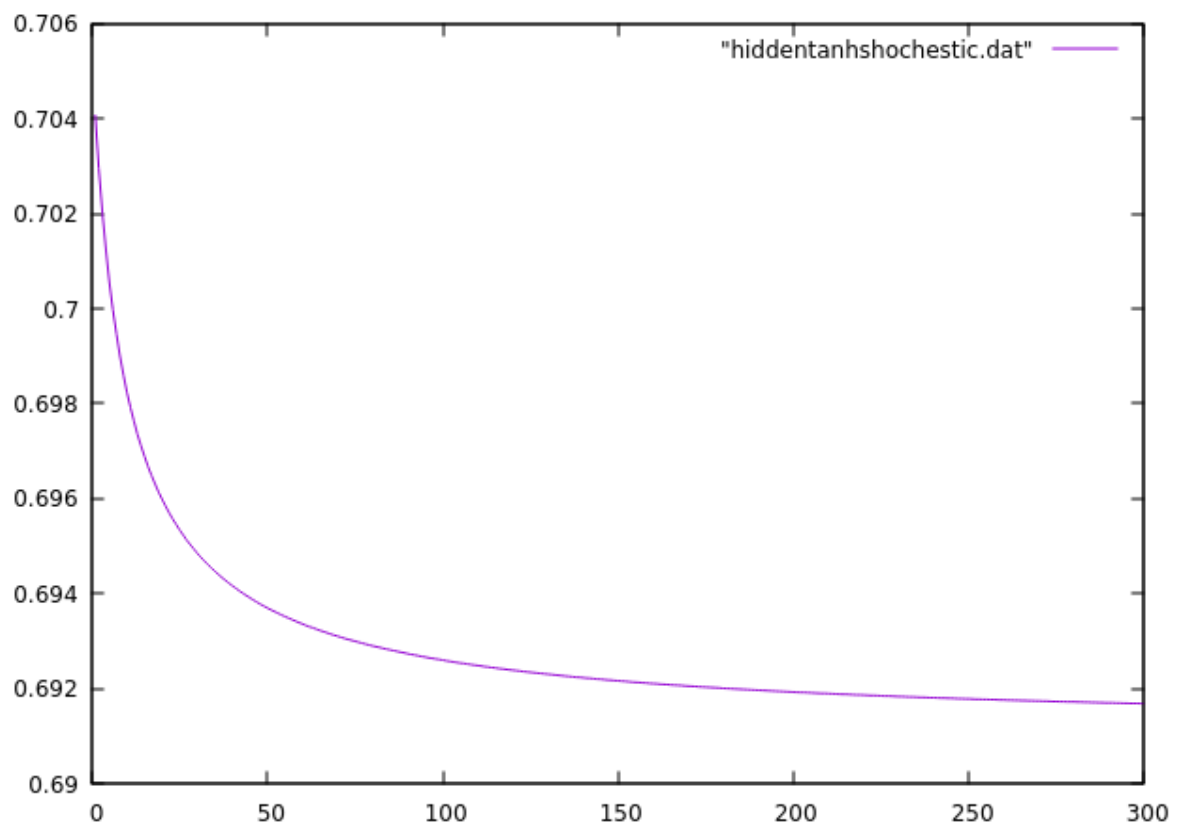
7. Batch gradient Descent and used sigmoid activation function in output layer and tanh activation function in hidden layer.



8. Mini Batch gradient Descent and used sigmoid activation function in output layer and tanh activation function in hidden layer.



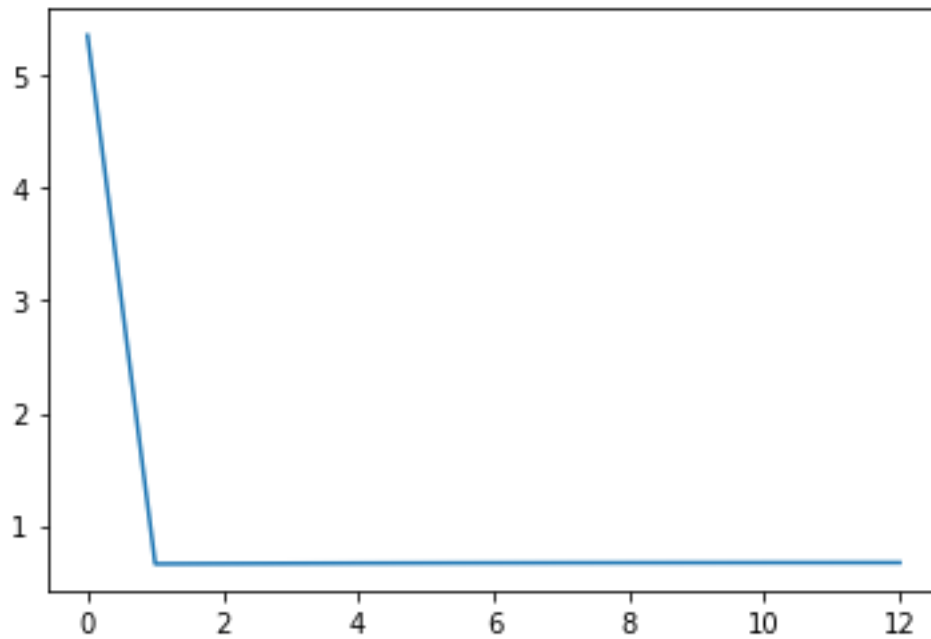
9. Stochastic gradient Descent and used sigmoid activation function in output layer and tanh activation function in hidden layer.



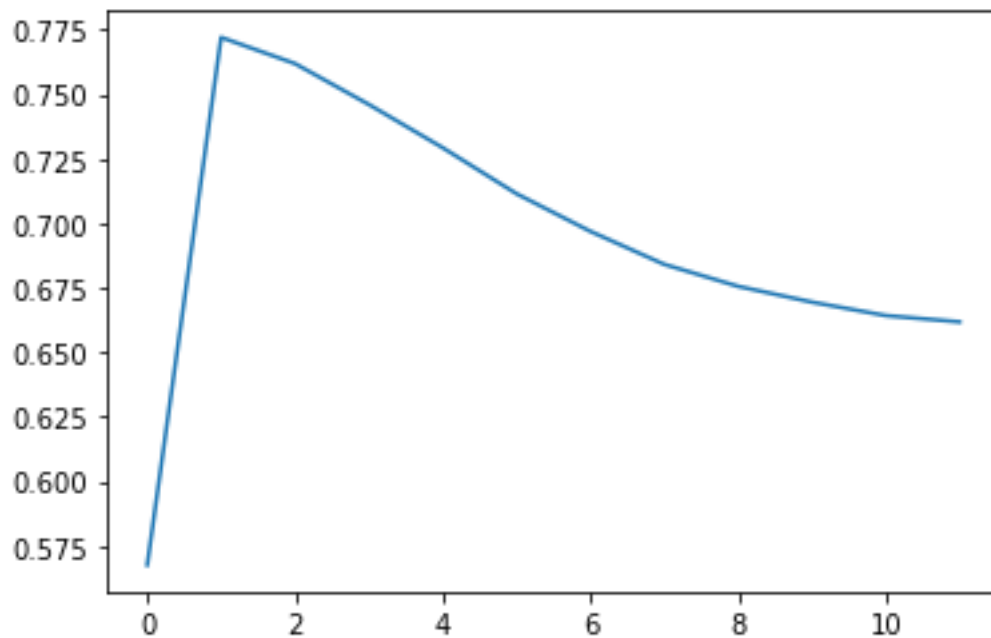
The above graphs are drawn from the data generated by a MLP classifier with 1 hidden node with 2 nodes and 30 input node and one output node. The number of iteration is 300.

The following graphs are taken from the sklearn MLPClassifier.

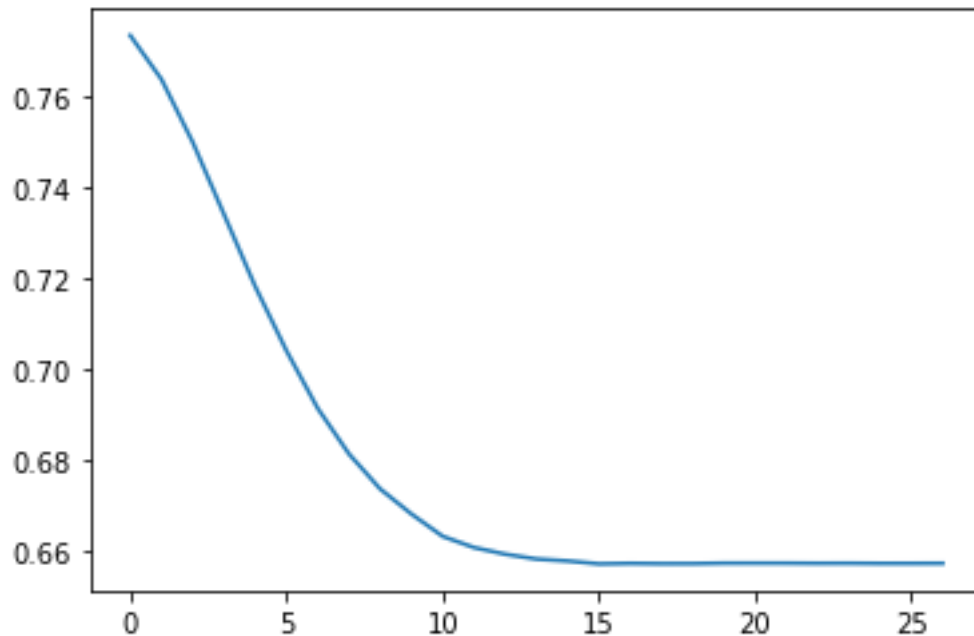
1. Stochastic gradient descent with 1 hidden node and one output node and relu is used as activation function.



2. Stochastic gradient descent with 1 hidden node and one output node and tanh is used as activation function.



3. Stochastic gradient descent with 1 hidden node and one output node and sigmoid is used as activation function.



### Vanishing Gradient Descent

Vanishing Gradient Descent is the problem when the slope become zero. When the slope became zero, the change in the weight is 0. So, the error became constant in every iteration. I have faced this problem in Stochastic gradient Descent and used sigmoid activation function in output layer and relu activation function in hidden layer.