Homework # 2

Final Report

On

Training

Multilayer Neural

Networks

Submitted by:- To:-

Gurpreet Singh Dr. Changhe Yuan

CS 3813(Machine Learning)

**Contents**

Intro: Multilayer Neural Networks

Total Output Error vs. Number of Epochs (Tables + Graphs + Analysis) when

1) Learning rate = 0.1 and Threshold = 0.00155

2) Learning rate = 0.01 and Threshold = 0.00155

3) Learning rate = 0.1 and Threshold = 0.01

4) Learning rate = 0.01 and Threshold = 0.01

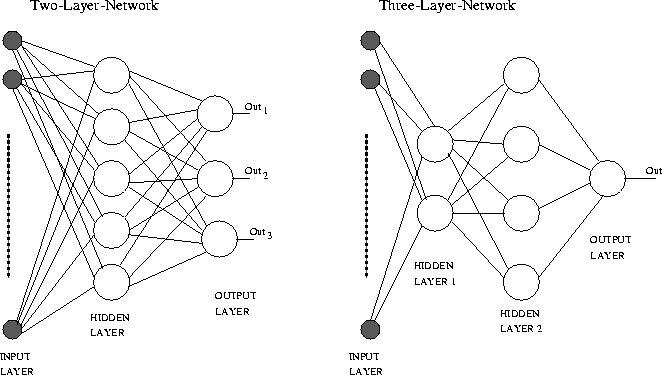
5) Learning rate = 0.001 and Threshold = 0.00155

6) Learning rate = 0.001 and Threshold = 0.01

**Intro: Multilayer Neural Network**

Multilayer networks solve the classification problem for non-linear sets by employing hidden layers, whose neurons are not directly connected to the output. The additional hidden layers can be interpreted geometrically as additional hyper-planes, which enhance the separation capacity of the network.

This new architecture introduces a new question: how to train the hidden units for which the desired output is not known. The **Back propagation** algorithm offers a solution to this problem.

The training occurs in a supervised style. The basic idea is to present the input vector to the network; calculate in the forward direction the output of each layer and the final output of the network. For the output layer the desired values are known and therefore the weights can be adjusted as for a single layer network; in the case of the BP algorithm according to the gradient decent rule.

To calculate the weight changes in the hidden layer the error in the output layer is back-propagated to these layers according to the connecting weights. This process is repeated for each sample in the training set. One cycle through the training set is called an epoch. The number of epochs needed to train the network depends on various parameters, especially on the error calculated in the output layer.

For forecasting, the most popular error function is the sum-of-squared errors, or one of its scaled versions.  This is analogous to using the minimum least squares optimization criterion in linear regression.  Like least squares, the sum-of-squared errors is calculated by looking at the squared difference between what the network predicts for each training pattern and the target value, or observed value, for that pattern.  Formally, the equation is the same as one-half the traditional least squares error.

**Experiment Results**

1) Learning rate = 0.1 and Threshold = 0.00155

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.00155 Learning Rate: 0.1 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.03058 |
| 2 | 50 | 0.00329 |
| 3 | 200 | 0.00188 |
| 4 | 400 | 0.00165 |
| 5 | 600 | 0.00157 |
| 6 | 706 | 0.00154 |

**Table 1: Total Output Error against the number of Epochs when Learning rate = 0.1 and Threshold = 0.00155**

**Graph 1🡪Table1: Total Output Error against Number of Epochs when Learning rate = 0.1 and Threshold = 0.00155**

One can see in Graph 1, how Total Output Error decreases abruptly in the beginning, and then very slowly thereafter. This also tells the nature of the Total Output Error Function that there is a **steep fall in the function** and also **Much Lower Threshold** **leads to higher number of Epochs.**

2) Learning rate = 0.01 and Threshold = 0.00155

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.00155 Learning Rate: 0.01 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.0837 |
| 2 | 500 | 0.0035 |
| 3 | 1000 | 0.00225 |
| 4 | 2000 | 0.00182 |
| 5 | 3000 | 0.00176 |
| 6 | 4000 | 0.00168 |
| 7 | 5000 | 0.00161 |
| 8 | 6000 | 0.00155 |

**Table 2: Total Output Error against the number of Epochs when Learning rate = 0.01 and Threshold = 0.00155**

**Graph🡪Table 2: Total Output Error against Number of Epochs when Learning rate = 0.01 and Threshold = 0.00155**

As one can see in Graph 2, Much Lower Threshold and Low Learning Rate increases the Number of Epochs very faster. Also, Total Output Error decreases abruptly in the beginning, and then very slowly thereafter. **Too Lower Learning Rates can lead to Multiple days of Training.**

3) Learning rate = 0.1 and Threshold = 0.01

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.01 Learning Rate: 0.1 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.3058 |
| 2 | 2 | 0.01722 |
| 3 | 3 | 0.01284 |
| 4 | 4 | 0.01086 |
| 5 | 5 | 0.00977 |

**Table 3: Total Output Error against the number of Epochs when Learning rate = 0.1 and Threshold = 0.01**

**Graph🡪Table 3: Total Output Error against Number of Epochs when Learning rate = 0.1 and Threshold = 0.01**

**Observation** in **Graph 3**, High Learning Rate leads to lower Number of Epochs to train the network for certain threshold.

4) Learning rate = 0.01 and Threshold = 0.01

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.01 Learning Rate: 0.01 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.083 |
| 2 | 4 | 0.055 |
| 3 | 8 | 0.035 |
| 4 | 10 | 0.02967 |
| 5 | 20 | 0.0165 |
| 6 | 30 | 0.0123 |
| 7 | 40 | 0.0105 |
| 8 | 50 | 0.0096 |

**Table 4: Total Output Error against the number of Epochs**

**Graph🡪Table 4: Total Output Error against Number of Epochs when Learning rate = 0.01 and Threshold = 0.01**

Observation in Graph 4, Average Learning Rate and Average Threshold Value leads to average number of Epochs.

5) Learning rate = 0.001 and Threshold = 0.00155

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.00155 Learning Rate: 0.001 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.09556 |
| 2 | 1000 | 0.00786 |
| 3 | 5000 | 0.00368 |
| 4 | 10000 | 0.00230 |
| 5 | 20000 | 0.00188 |
| 6 | 30000 | 0.00172 |
| 7 | 40000 | 0.00164 |
| 8 | 50000 | 0.00159 |
| 9 | 60000 | 0.00155 |

**Table 5: Total Output Error against the number of Epochs when Learning rate = 0.001 and Threshold = 0.00155**

**Graph🡪Table 5: Total Output Error against Number of Epochs when Learning rate = 0.001 and Threshold = 0.00155**

Observation in Graph 5: Too Low Learning Rate and Too Low Threshold Value can lead to thousands and even millions of Epochs to train a Multilayer Network. Initial fall in the Error is very fast, but later due to lower threshold and lower learning rate, epochs increase much faster.

6) Learning rate = 0.001 and Threshold = 0.01

|  |  |  |
| --- | --- | --- |
| Threshold Value: 0.01 Learning Rate: 0.001 | | |
| S.no. | Average  # of Epochs | Average  Total Output Error |
| 1 | 1 | 0.0955 |
| 2 | 100 | 0.0307 |
| 3 | 200 | 0.0172 |
| 4 | 300 | 0.0128 |
| 5 | 400 | 0.0108 |
| 6 | 500 | 0.010 |

**Table 6: Total Output Error against the number of Epochs when Learning rate = 0.001 and Threshold = 0.01**

**Graph🡪Table 6: Total Output Error against Number of Epochs when Learning rate = 0.001 and Threshold = 0.01**

Observation in Graph 6: Too Low Learning Rate and Moderately Low Threshold Value can lead to Moderately high number of Epochs to train a Multilayer Network.

**Final Observations:-**

* Too Low Learning Rate can lead to enormously high number of Epochs.
* **How the stopping threshold value affects the training and testing error?**

High Stopping threshold value leads to high training and testing error.

Lower threshold value leads to lower training and testing error.