Multi-Layer Perceptron: Overcoming the Local Minima Problem: Hierarchical Binary Classifiers

Rama Murthy Garimella

September 22, 2024

MULTI-LAYER PERCEPTRON: OVERCOMING THE LOCAL MINIMA PROBLEM:

HIERARCHICAL BINARY CLASSIFIERS

Garimella Rama Murthy,

Professor, Deparment of Computer Science,

Mahindra University, Hyderabad, INDIA

ABSTRACT

It is well known that the gradient descent rule employed in training the Multi-Layer Perceptron (MLP) could get stuck in a local minima of the error/loss function (based on mean squared error). We reason that by realizing MLP using a cascade of binary classifiers (MLP with single neuron in the output layer), the Hierarchical classification approach overcomes the local minima problem (since the loss function of each binary classifier is a paraboloid). Several innovative ideas related to such Artificial Neural Network architecture are being proposed.

1. INTRODUCTION:

McCulloch, Pitts proposed a model of artificial neuron to emulate the classification function of “linearly Separable” patterns. But, such a model of artificial neuron doesnot have the training ability, since the synaptic weights are fixed and not variable. Rosenblatt proposed the “perceptron” model of neuron in which the synaptic weights are variable and thus it has training ability. Under the assumption that two classes of patterns are linerarly separable, Rosenblatt proved the “perceptron learning law”. It established that all patterns (belonging to two classes) are correctly classified in finitely many epochs i.e. the synaptic weights converge leading to a hyperplane which provides 100% classification accuracy. A naturallt question arises as to how classification needs to be done when the patterns are not linearly separable. A natural compromise is to relax the condition of 100% accuracy of classification. Towards this end, an error criteria such as mean square error (between the desired output for a training pattern and the actual output) is utilized. As explained below, the associated error measure corresponds to a paraboloid (parabolic error surface) when there is a single perceptron/linear neuron performing the classification. Such a classification approach is enabled by the so called “gradient descent rule “.

As natural generalization, Multi-Layer-Perceptron was proposed to classify patterns belonging to multiple classes that are separated by nonlinear decision boundaries. But the error criteria (loss function) associated with multiple neuronal units in the output layer) and all training patterns typically has multiple local minima. Hence the backpropagation algorithm (based on gradient descent rule) typically will get stuck in local minima. This research paper addresses the problem of modifying the Multi-Layer Perceptron architecture (a novel Artificial Neuron Architecture) which enables overcoming the multiple local minima problem.

This research paper is organized as follows. In Section 2, the local minima problem arising in arbitrary multi-layer perceptron with multiple neuronal units in the output layer is explained. In Section 3, a novel Artificial Neural Network (ANN) architecture with a cascade architecture of binary classifiers is proposed. It is reasoned that such a hierarchical classification architecture enables overcoming the multiple local minima problem.

2. Review of Known Research Literature:

Rosenblatt proposed the idea of varying the synaptic weights to train a perceptron. Our goal is to learn a weight vector which will classify all the training patterns (i.e. 100% accuracy) correctly into two classes. components. Weight vector components are modified at each step (when a new pattern is presented) according to the following PERCEPTRON LEARNING LAW

$$w_i(n+1) = w_i(n) + \Delta w_i(n), \qquad where$$
$$\Delta w_i(\text{n}) = \delta(t-o)x_i, \ with$$

$w_i(n+1)$: i^{th} *weight vector component at time* $'t+1'$

t : t is the target (desired) output for the current training example

o: output generated by the perceptron for the current training example

δ: *Positive constant called* "learning rate".

It should be noted that if a training pattern is correctly classified, then the components of weight vector will not change [1]. The process of updating the synaptic weights (using the above perceptron learning law) is repeated, iterating through training examples as many times as needed until all the training examples are correctly classified. It was proved that the above learning procedure converges within a finite number of applications of perceptron learning law to a weight vector that correctly classifies all the training examples, provided the training examples are linearly separable and sufficiently small learning rate is used. In case the training patterns are not linearly separable, convergence of perceptron training rule is not assured. Hence, we are naturally led to the following alternative learning rule when the training patterns are not linearly separable (the following well known discussion is borrowed from Tom Mitchell's book)

- GRADIENT DESCENT AND DELTA RULE:-

  The delta rule converges toward a best fit approximation to the target concept if the training examples are not linearly separable. To derive the gradient descent rule, we consider the process of training an UNTHRESHOLDED PERCEPTRON i.e. a LINEAR UNIT (first stage of a perceptron without thresholding) for which the output is given by

  $$y = \bar{w}.\bar{x}$$

  where $\bar{x}$ is the training pattern vector and $\bar{w}$ is the weight vector (i.e. hypothesis) [2]

  To derive learning rule for the weights of a linear unit, we specify a measure of training error of a weight vector relative to the training examples. One widely used error measure is the MEAN SQUARE ERROR.

  $$E(\bar{w}) = \frac{1}{2}\sum_{d\in D}(t_d\text{-}o_d)^2$$

  Where D is the set of training examples, t_d is the target output for training example and o_d is the output of linear unit for the training example d . It should be noted that E is a function of weight vector and the set of training examples.

The gradient descent algorithm can be visualized by considering the hypothesis space of possible weight vectors and the associated error E values. It can easily be reasoned that the error E corresponds to a "paraboloid" (parabolic surface in 3 dimensions) with a single global minimum which is also a local minimum.

Gradient descent search determines a weight vector that minimizes E by starting with an arbitrary initial weight vector and repeatedly modifying it in small steps. At every step, the weight vector is modified in the direction that produces the steepest descent along the error hypersurface (i.e. in the direction of negative of gradient). The process continues until the global minimum (which is also the local minimum) is reached. Formally

$$\overline{w}(n+1) = \overline{w}(n) + \Delta\overline{W}(n)\,, where$$

$$\Delta\overline{W}(n) = -\delta\nabla E(\overline{w})$$

denotes $\nabla E(\overline{w})$ the gradient vector and δ is the learning rate.

For, the linear neuronal unit, it can be readily shown that

$$\Delta w_i(n) = -\delta\frac{\delta E}{\delta w_i}\ , where$$

$$\frac{\delta E}{\delta w_i} = \sum_{d\in D}(t_d\text{-}o_d)\,(\text{-}x_{id})$$

Note: Even though the delta rule is utilized as a method for learning weights of unthresholded linear units, it can easily be used to train thresholded perceptron units as well (Refer Tom Mitchell's book).

In the case of Multi Layer Perceptron (MLP) with multiple neuronqal units in the output layer, we redefine E by summing the error over all neuronal units in the output layer

$$E(\overline{w}) = \frac{1}{2}\sum_{d\in D}\ \sum_{k\in outputs}(t_{kd} - o_{kd})^2$$

where 'outputs' is the set of output units in the ANN and are the target (desired) output and actual output values associated with the output and training example d.

Note: With the above error measure, the error surface can have multiple local minima, in contrast to the case of single neuronal unit considered earlier. Hence, the gradient descent is guaranteed only to converge toward some local minimum and not necessarily the global minimum error.

3. Cascade of Binary Classifiers: Hierarchical Classification: Multi-Layer Perceptron:

The classification problem in which there are 'N' classes is converted into 'N-1" hierarchical binary classification problems using a cascade connection of 'N-1' binary classifiers.

It readily follows that the above ANN architecture is equivalent to the associated MLP architecture. In fact the architecture provides detailed information based on the associated balanced binary classification tree.

It is clear that each "binary classifier" performs gradient descent on an error surface which has unique global/local minima.

4. **Conclusion:**

The local minima problem associated with Multi Layer Perceptron (MLP) was studied by several researchers such as Levenberg. They suggested some solutions. In this research paper, we propose a CASCADE CONNECTION of BINARY CLASSIFIERS to overcome the problem of getting stuck in local minima (since the loss function associated with each binary classifier is a paraboloid). We are currently pursuing several innovative results associated with such ANN architecture such as the UNIQUENESS of such architecture

REFERENCES:

[1] Tom M. Mitchell, " Machine Learning", Mc Graw Hill Publishers

[2] B. Yegnanarayana,"Artificial Neural Networks," Prentice Hall India Publishers