Journal of Advances and Scholarly Researches in Allied Education

Vol. VIII, Issue No. XVI, Oct-2014, ISSN 2230-7540

LOCAL LEARNING ALGORITHMS WITH NEURAL COMPUTATION

www.ignited.in

Journal of Advances and Scholarly Researches in Allied Education
Vol. VIII, Issue No. XVI, October-2014, ISSN 2230-7540

Local Learning Algorithms with Neural Computation

Chawda Shyam Navinchandra

Research Scholar, Bhagwant University, Ajmer

Abstract* – *This paper introduces a learning method for two-layer feedforward neural networks based on sensitivity analysis, which uses a linear training algorithm for each of the two layers. First, random values are assigned to the outputs of the first layer; later, these initial values are updated based on sensitivity formulas, which use the weights in each of the layers; the process is repeated until convergence. Since these weights are learnt solving a linear system of equations, there is an important saving in computational time. The method also gives the local sensitivities of the least square errors with respect to input and output data, with no extra computational cost, because the necessary information becomes available without extra calculations. This method, called the Sensitivity-Based Linear Learning Method, can also be used to provide an initial set of weights, which significantly improves the behavior of other learning algorithms.

Keywords: Supervised Learning, Neural Networks, Linear Optimization, Least-Squares, Initialization Method, etc.

- X -

INTRODUCTION

The theoretical basis for the method is given and its performance is illustrated by its application to several examples in which it is compared with several learning algorithms and well known data sets [1]. The results have shown a learning speed generally faster than other existing methods. In addition, it can be used as an initialization tool for other well-known methods with significant improvements [2].

Modelling nonlinear dynamic systems from observed data and a priori engineering knowledge is a major area of science and engineering. In recent years a great deal of work has appeared in new areas like Fuzzy Modelling and Neural Networks to complement the previous work in statistics and system identification. The recent work, however, often lacks a solid engineering [3]. Neural networks can learn to reproduce the behaviour evident in their training set, but they are usually unable to benefit directly from a priori knowledge, or to provide good estimates of their accuracy and robustness. Fuzzy systems have also been heavily used for modelling non-linear systems with their structure provided by experts for the system, but often lacked the ability to refine their structure (membership functions, rules) in a data driven manner, meaning that much of the development time is then spent 'tweaking' parameters [4]. The Local Model Net is proposed as a useful hybrid method incorporating the advantages of the various paradigms. This paper gives an outline of the architecture, with an overview of the literature and discusses a variation in the learning algorithm for the networks local model parameters [5].

REVIEW OF LITERATURE:

There are many alternative learning methods and variants for neural networks. In the case of feedforward multilayer networks the first successful algorithm was the classical backpropagation [6]. Although this approach is very useful for the learning process of this kind of neural networks it has two main drawbacks:

- Convergence to local minima.
- Slow learning speed.

In order to solve these problems, several variations of the initial algorithm and also new methods have been proposed. Focusing the attention on the problem of the slow learning speed, some algorithms have been developed to accelerate it:

1. from Basis Function Nets to Local Model Nets:

➢ **Basis Function Nets:**

The output is a linear combination of many locally active non-linear basis functions. Each unit's centre is a point in the input space, and the receptive field of the unit (the volume of the input space to which it reacts) is defined by its distance metric. The basis or activation function (similar to the membership function of a fuzzy set) of the unit is composed of two elements: The distance metric d(x), which can scale and shape the spread of the basis function relative to its centre C, and the basis function itself (d(x)), which takes the distance metric as its input [7]. They are usually chosen so that the activation monotonically decreases towards zero as the input point moves away from the unit's centre, e.g. B-Splines or Gaussian bells are common choices. Radial Basis Function (RBF) nets are the most straightforward, and most commonly used types of Basis Function networks [8].

2. Modifications of the standard algorithms:

Some relevant modifications of the backpropagation method have been proposed. Extend the backpropagation framework by adding a gradient descent to the sigmoid steepness parameters. Present a novel fast learning algorithm to avoid the slow convergence due to weight oscillations at the error surface narrow valleys. To overcome this difficulty they derive a new gradient term by modifying the original one with an estimated downward direction at valleys. Also, stochastic backpropagation—which is opposite to batch learning and updates the weights in each iteration often decreases the convergence time, and is specially recommended when dealing with large data sets on classification problems [9].

3. Methods based on linear least-squares:

Some algorithms based on linear least-squares methods have been proposed to initialize or train feedforward neural networks. These methods are mostly based on minimizing the mean squared error (MSE) between the signal of an output neuron, before the output nonlinearity, and a modified desired output, which is exactly the actual desired output passed through the inverse of the nonlinearity [10]. Specifically, in a method for learning a single layer neural network by solving a linear system of equations is proposed. This method is also used in to learn the last layer of a neural network, while the rest of the layers are updated employing any other non-linear algorithm (for example, conjugate gradient). Again, the linear method in the basis for the learning algorithm proposed in this article, although in this case all layers are learnt by using a system of linear equations [11].

4. Structure Initialization & Optimization:

Structure optimization is the most important aspect of the learning process for local model networks, but is not emphasized in this paper due to lack of space. The most relevant aspects will be briefly described. The advantage of Basis Function networks is that the nonlinearity is a localized one. This provides advantages for learning efficiency, generalization and transparency. It is, however, very difficult to automatically find the 'correct' level of locality for a given subspace of an arbitrary problem, because he concept of 'locality' is obviously relative, depending on the complexity of the system, the availability of training data, the importance of the given area of the input space, and, importantly for models, a priori knowledge of internal structures within the given system [12]. If the user already has a priori knowledge about the system being modelled, this could be used to define or initialize the basis function partition. (The use of such physically based knowledge makes the model more easily interpretable and also makes on-line adaptation of the system's parameters much more feasible). As mentioned earlier, some classes of fuzzy systems can also be viewed as Basis Function models. The relationship to fuzzy systems is interesting, as an initial partition of the input space can then be supplied in the form of linguistic rules and membership functions. Once this initial partition of the input space has been completed, the consequence of each node can be a local model network which learns the remaining structural details by a data-driven structure adaptation algorithm.

CONCLUSION:

This paper investigated two ways of optimizing the local model parameters, given a particular basis structure. Global and Local singular value decomposition algorithms were used. The analysis of the computational complexity shows that local learning is faster than global learning. The structure also allows more flexibility in the use of optimization algorithms, which will be especially useful with hybrid model structures which require nonlinear local optimization, or on-line learning. A further point is that the locally trained networks are more interpretable than the globally trained ones, and although producing worst least squares statistics, deliver smoother models without having to resort to expensive cost functional.

REFERENCES:

1. L. B. Almeida, T. Langlois, J. D. Amaral, and A. Plakhov. Parameter adaptation in stochastic optimization. In D. Saad, editor, On-line Learning in Neural Networks, chapter 6, pages 111– 134. Cambridge University Press, 1999.

2. R. Battiti. First and second order methods for learning: Between steepest descent and Newton's method. Neural Computation, 4(2):141–166, 1992.

3. E. M. L. Beale. A derivation of conjugate gradients. In F. A. Lootsma, editor, Numerical methods for nonlinear optimization, pages 39–43. Academic Press, London, 1972.

4. F. Biegler-Konig and F. B ¨ armann. A learning algorithm for multilayered neural networks based on ¨ linear least-squares problems. Neural Networks, 6:127–131, 1993.

5. W. L. Buntine and A. S. Weigend. Computing second derivatives in feed-forward networks: A review. IEEE Transactions on Neural Networks, 5(3):480–488, 1993.

6. E. Castillo, J. M. Gutierrez, and A. Hadi. Sensitivity analysis in discrete bayesian networks. ´ IEEE Transactions on Systems, Man and Cybernetics, 26(7):412–423, 1997.

7. E. Castillo, A. Cobo, J. M. Gutierrez, and R. E. Pruneda. Working with differential, functional and ´ difference equations using functional networks. Applied Mathematical Modelling, 23(2):89–107, 1999.

8. A. Chella, A. Gentile, F. Sorbello, and A. Tarantino. Supervised learning for feed-forward neural networks: a new minimax approach for fast convergence. Proceedings of the IEEE International Conference on Neural Networks, 1:605 – 609, 1993.

9. V. Cherkassky and F. Mulier. Learning from Data: Concepts, Theory, and Methods. Wiley, New York, 1998.

10. R. Collobert, Y. Bengio, and S. Bengio. Scaling large learning problems with hard parallel mixtures. International Journal of Pattern Recognition and Artificial Intelligence, 17(3):349–365, 2003.

11. J. E. Dennis and R. B. Schnabel. Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Prentice-Hall, Englewood Cliffs, NJ, 1983.

12. G. P. Drago and S. Ridella. Statistically controlled activation weight initialization (SCAWI). IEEE Transactions on Neural Networks, 3:899–905, 1992.