ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2007, Vol. 36, No. 4

NEURAL NETWORK AS AN OPHTHALMOLOGIC DISEASE CLASSIFIER

Povilas Treigys, Vydūnas Šaltenis

Institute of Mathematics and Informatics
Akademijos Str. 4, LT-08663 Vilnius, Lithuania

Abstract. In this paper, we explore the neural network as a disease classifier. In our investigation, the sets of parameters describing glaucomatous and healthy eyes are taken. These sets represent the structure of the optical nerve disc which resides in a patient's eye fundus image. As a separate case, the excavation can be seen in the image as well. These two sets describe the elliptical shape of both structures and compound the initial data for analysis. Thus, the distinction of classes represented by the data sets becomes possible. In this article, a multi-layer neural network is explored. Selection of the optimal number of hidden neurons is taken into consideration. We also explore here the principal component analysis for feature reduction. The classification results are discussed as well.

Keywords: optic nerve disc, excavation parameters**,** multi-layer neural network, disease classification, glaucoma, number of hidden units, principal component analysis.

1. Introduction

Nowadays the information amount the medic has to deal with is huge. Thus, a careful analysis of such a data set is hardly possible. The problem arises while making the medical decision when the state of a patient has to be assigned to the initially known class. For clarity, the class can be defined as ailing or healthy. Almost each disease can be described by a set of quantitative parameters. Methods of data mining and analysis can be introduced for the decision support system development with a view of a preliminary medical diagnosis [1, 2, and 3]. However, the boundary between alternatives of diagnosis in most cases is not straightforward and the decision for the disease presence can be made very subjectively. In the medical context it is a topical problem. If an ailing patient is classified as healthy, the results could be unpredictable. Thus, it is of utmost importance to determine the boundary of transition from one class of a disease to another.

Eye fundus examination is one of the most important diagnostic procedures in ophthalmology. A high quality colour photograph of the eye fundus (Figure 1) helps in the accommodation and follow-up of the development of the eye disease. Evaluation of the eye fundus images is complicated because of the variety of anatomical structure and possible fundus changes in eye diseases.

The optic nerve disc appears in the normal eye fundus image as a yellowish disc with whitish central cupping (excavation) through which the central retinal artery and vein pass.

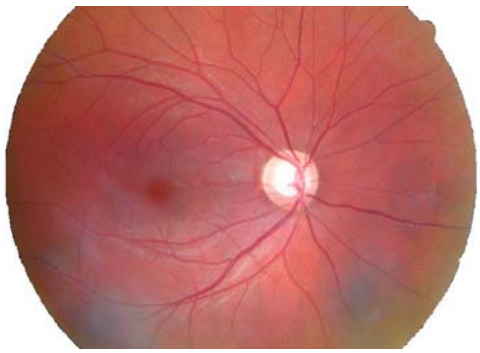

Figure 1. An example of fundus of the eye

Changes in the optic nerve disc can be associated with numerous vision threatening diseases such as glaucoma, optic neuropathy, swelling of the optic nerve disc, or related to some systemic disease.

Assume that some set of parameters characterizes the optic nerve disc and excavation. Hence, it becomes possible to construct an *n*-dimensional vector $X = (x_1, x_2, ..., x_n)$. Each *n*-dimensional vector corresponds to one fundus image and describes the disease.

The goal is to assign the vector X^m to one of the known classes, where $m=1...p$ and p is the number of patients.

The set of 27 parameters of each eye fundus image has been measured. Generally, these parameters fall into four groups [4]:

- parameters of optic nerve discs (OD): major axis of OD ellipse (x_1), minor axis (x_2), semi-major axis (x_3), semi-minor axis (x_4), horizontal diameter (x_5), vertical diameter (x_6), area (x_7), eccentricity (x_8), and perimeter (x_9);
- parameters of excavation (EKS) (excavation is a degenerated part of OD): major axis of EKS ellipse (x_{10}), minor axis (x_{11}), semi-major axis (x_{12}), semi-minor axis (x_{13}), horizontal diameter (x_{14}), vertical diameter (x_{15}), area (x_{16}), eccentricity (x_{17}), and perimeter (x_{18});
- ratios between various OD, EKS, NR parameters (neuroretinal rim (NR) is an OD part that is not degenerated): ratio between EKS and OD horizontal diameters (x_{19}), ratio between EKS and OD vertical diameters (x_{20}), NR area (x_{21}), ratio between NR and OD areas (x_{22}), ratio of EKS and OD (x_{23});
- thickness of NR parts: inferior disc sector (x_{24}), superior disc sector (x_{25}), nasal disc sector (x_{26}) and temporal disc sector (x_{27}).

Two groups of items are investigated: vectors, corresponding to the healthy eyes (24 items); vectors, corresponding to the eyes, damaged by glaucoma (24 items).

2. Neural network

2.1. Single-layer perceptron

A single-layer perceptron is a simplified mathematical representation of a biological neuron. The biological neuron fires an output signal only when the total strength of input signals exceeds a certain threshold. We model this phenomenon in a perceptron by calculating the weighted sum of the inputs to represent the total strength of the input signals, and applying a step (activation) function to the sum to determine its output [5].

In mathematical terms, we can state that for the input vector X (set of values representing the problem domain) and the weight vector W (set of weights describing how important each problem domain value is) the weighted sum can be found by:

$$s = w_0 + \sum_{i=1}^{n} x_i w_i , \qquad (1)$$

where n is the dimension of the input vector X, w_0 is the bias and w_i is the i – th weight.

Basically the activation function $f(s)$, shown in Figure 2, has to satisfy the following four criteria [6]:

- almost linear for small Δs,
- must have a limit as f(s)→-∞,
- must have a limit as f(s)→+∞,
- these limits have to be different.

In other words, this function has to produce value 1, when the input vector belongs to one class and 0 if the input vector belongs to another.

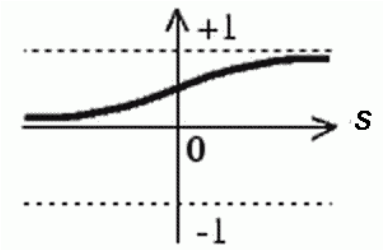

Figure 2. Activation function

In this research the *log* sigmoid function was used:

$$f(s) = \frac{1}{1 + e^{-s}} . \qquad (2)$$

The main disadvantage of the single-layer perceptron is that it can easily operate and show itself fine until the classes described by the vectors X are separable. But, as the dimensionality n of the vector X increases, in most cases it forms not linearly separable regions.

2.2. Multi-layer perceptron

As usual, a multi-layer perceptron consists of several single-layer perceptrons, which are arranged in some hierarchy. This hierarchy must satisfy the following characteristics [7]:

- The first layer is taking inputs with the number of perceptrons equal to the number of vectors X of the problem.
- The output layer produces outputs with the number of neurons equal to the desired number of quantities computed from the inputs.
- In-between those layers there are middle layers (it can be one or more layers) which have no connection to the external world. Hence, they are called hidden layers.
- Each single perceptron in one layer is connected to every perceptron in the next layer.
- There cannot be any connection among the perceptrons in the same layer.

As stated before, with no hidden layers, the perceptron can only perform linearly separable tasks. This scheme results in the separation of points into regions that are not linearly separable. Let us consider the network shown in Figure 3.

Here x, y are values representing a point on the plane. For the single-layer perceptron the output can be calculated as follows:

$$\begin{cases} 1, & f(s) > 0.5, \\ 0, & f(s) \le 0.5. \end{cases} \quad (3)$$

Formula (3) describes two regions on the plane *xy*. Actually a single-layer perceptron exposes the region to which the given point corresponds.

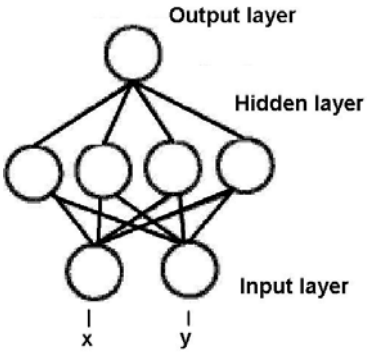

Figure 3. Multi-layer perceptron with one hidden layer

Let us assume that the same point *(x, y)* is introduced into the network (Figure 4) through the perceptrons in the input layer. Since there are four perceptrons in the hidden layer and they are independent of each other, the given point is classified into four pairs of linearly separable regions. Each region is described by a unique line produced by each perceptron in the hidden layer. Finally, the output layer performs some logical operation on the outputs of the hidden layer, so that the network ascribes the input point either to one region or another. These regions might not be linearly separable (Figure 4).

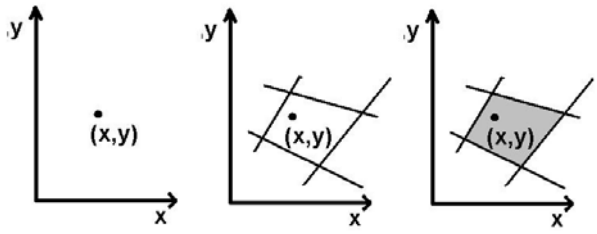

Figure 4. On the left, there is the initial point on the plane; in the center we see a region formed by four hidden layer perceptrons; on the right, - output layer logical operation

Thus, by varying the number of perceptrons in the hidden layer, the number of hidden layers, and output nodes, we can attribute the number of points of arbitrary dimensions into the required number of groups.

In our case, each input vector is in the space: $X_i \in \Re^{27}, i \in [1, 48]$. All the vectors fall into two classes: glaucomatous and healthy eyes.

2.3. Learning algorithm

Once the vector *W* has been initialized, the network is ready for training. The training process requires a subset from the training set *X* for proper network behaviour. During training the weights and biases of the network are iteratively adjusted to minimize the network performance function in the sense of sum of squared error [8]:

$$Err = \arg\min \sum_{j=1}^{m} (d_j - o_j)^2 . \quad (4)$$

Here *m* is number of samples taken from *X*, d_j is the *j*-th desired, initially known, value from the training sample, o_j is the *j*-th output value from the neural network.

Formula (4) can be easily expressed in terms of the input training sample X_j, the weight vector W_j, and the activation function.

Such a learning algorithm uses the gradient of the performance function with a view to determine how to adjust the weights in order to minimize the error. The gradient is determined using a back propagation technique [9]. Back propagation learning updates the network weights and biases (vector *W*) in the direction where the performance function decreases most rapidly, the gradient being negative. Such an iterative process can be expressed as:

$$W_{k+1} = W_k - \alpha \cdot g_k, \quad (5)$$

where W_k is the vector of current biases and weights, α is the learning rate, and g_k is the current gradient.

Note that a decrease in the weight value in the direction of the gradient and continuous form of the activation function (2) leads to a decrease of formula (4) result. Thus, formula (5) is known as Newton's method and can be rewritten as follows:

$$W_{k+1} = W_k - \alpha \cdot \frac{\partial Err}{\partial W_k}, \quad (6)$$

Here *k* is the iteration counter, ∂Err is differences between the desired and output values of the network.

In this research, the second order Levenberg-Marquardt [10] algorithm was used, since it is more robust and in many cases finds a solution even if it starts very far off the final minimum.

If the performance function has the form of a sum of squares, then it becomes possible to approximate the Hessian matrix by $H=J^TJ$ and the gradient by $g=J^TErr$. The approximation is performed since the Hessian matrix is often singular or ill – conditioned, also, computing a full matrix for the neural network is a memory and time consuming task [11]. Hence formula (6) is transformed to:

$$W_{k+1} = W_k - \left[J^T J + \alpha \cdot I\right]^{-1} J^T Err, \quad (7)$$

where *J* is the Jacobian matrix containing the first derivatives of the network errors with respect to the weights and biases, α is a regularization parameter, and *Err* is the vector of network errors [12].

It is well known that Newton's method (6) is faster and more accurate being closer to the error minimum [13]. Thus, the aim of the Levenberg-Marquardt learning algorithm is to shift towards Newton's method as soon as possible. In this case, α is decreased after each successful step, which results in the decrease of the performance function. The parameter is increased only if a tentative step increased the performance function [14]. This is due to the fact that when α is zero, the Levenberg-Marquardt becomes Newton's method, using the approximate Hessian matrix. When α is large, it becomes a gradient descent with a small step size [15].

3. Results

The set of 48 vectors was used for the investigation. Each vector was comprised of 27 parameters described in the introduction section. Unknown EKS parameters were assumed to be zeros. This happens when the excavation is absent in the eye fundus image.

For the neural network training, the set of 48 vectors was divided into 2 separable subsets of 24 vectors. One subset was used for the neural network training, and another for the cross validation procedure. Cross validation is performed to prevent the neural network from overtraining.

The class to which each vector belongs is known in advance. The proposed classifier has to be able to apart glaucomatous from healthy eyes by the given parameters corresponding to the one investigative eye fundus image. The neural network with one hidden layer was used for investigation.

First, the number of hidden neurons in the hidden layer was not known. In this case, to find the optimal number of units, the iterative procedure was used. The parameter vector consisted of 27 measured features of the optic nerve disc structure. Thus, in the first iteration, only one hidden unit was used. In the second iteration two hidden units were investigated, in the third iteration three hidden units were used, etc up to 27 hidden units. Each iteration was repeated 100 times because of initially selected random weights for neural network units. During the evaluation of network error, all the 48 parameter vectors were provided for the network classification task. The network error was measured in the sum of squared errors sense (SSE). The diagram of the best SSE produced for a particular number of units in a hidden layer is shown in Figure 5.

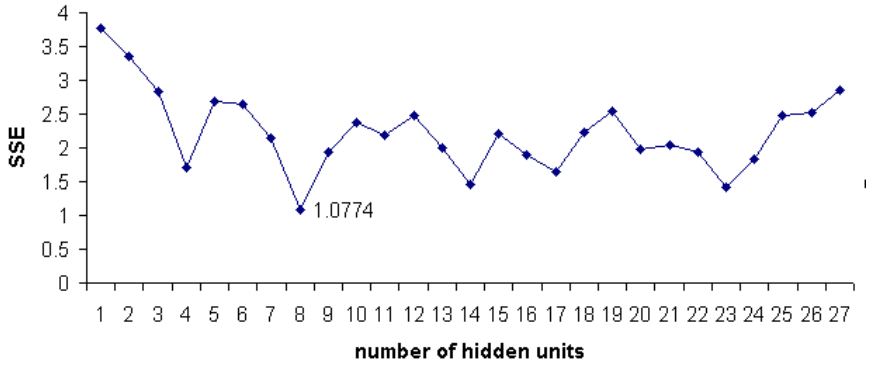

Figure 5. SSE according to the number of hidden units

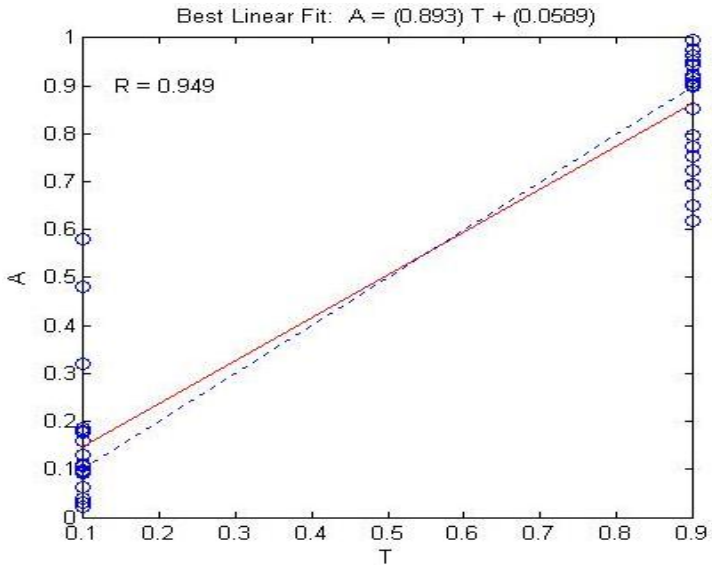

Figure 6. Classification results of non-scaled data

Figure 5 shows that the optimal number of hidden units is 8 for this problem statement in the SSE sense.

Second, the network was tested as a disease classifier. The results achieved are provided in Table 1.

Table 1. Classification performances

Number of hidden units	True Glaucoma	False Glaucoma	False Norma	True Norma	SSE
1	20	4	1	23	3.775
2	21	3	3	21	3.3615
3	22	2	4	20	2.8297
4	21	3	1	23	1.7058
5	22	2	2	22	2.6976
6	22	2	2	22	2.6422
7	23	1	0	24	2.1412
8	**23**	**1**	**0**	**24**	**1.0774**
9	22	2	0	24	1.9352
10	22	2	2	22	2.371
11	22	2	1	23	2.1944
12	20	4	0	24	2.4877
13	22	2	0	24	2.0048
14	23	1	1	23	1.4636
15	23	1	3	21	2.2086
16	22	2	0	24	1.8904
17	22	2	0	24	1.6364
18	20	4	1	23	2.2304
19	21	3	1	23	2.5361
20	23	1	1	23	1.9713
21	22	2	0	24	2.048
22	22	2	1	23	1.9325
23	22	2	0	24	1.4258
24	22	2	2	22	1.8289
25	21	3	4	20	2.4806
26	23	1	4	20	2.5277
27	22	2	3	21	2.8622

We can see from the table that the minimum of SSE is when the number of hidden units is 8. All the healthy eyes were identified correctly and only one case of glaucomatous eyes was classified as healthy. The graphical result of classification is provided in Figure 6.

However, the result shown in Figure 5 is questionable, since the curve has a lot of peaks. To explain such curvature, the standard deviation of network error was explored. As stated earlier in this section, the initial weights are selected randomly. Thus, it becomes possible to measure what influence those starting values have on the sum of squared error. In the case of 8 hidden units, where the SSE was minimal, the standard deviation was 3.257.

Figure 6 shows a linear regression between the network response and the target. In this picture, the parameter vector is visualized as circle-shaped, the line shows the regression between network responses and the dashed line shows the regression between the desired targets. *R* stands for the regression value.

It can be seen that, according to formula (3) on the left-hand side of Figure 6 only one parameter vector steps out the decision boundary.

Also, the influence of derivative parameters was investigated. In this case, the principal component analysis (PCA) was made as the initial step for pre-processing data vectors, which retain only those components that contribute more than 0.01% to the variance in the given data set. Such an analysis has showed that in the case of the given variance it is enough to take 10 parameters in each feature vector. Further, the neural network was trained under the same conditions as described earlier with the pre-processed data set. The sum of squared errors is shown in Figure 7.

One can see that opposite to Figure 5, the network SSE response variance is smoother. However, the overall error is greater than that with not pre-processed data.

Classification results are presented in Table 2.

Table 2. Classification performance with scaled data

Number of hidden units	True Glaucoma	False Glaucoma	False Norma	True Norma	SSE
1	24	0	5	19	3.175
2	22	2	3	21	3.129
3	22	2	3	21	2.867
4	17	7	1	23	3.17
5	21	3	2	22	2.642
6	24	0	3	21	2.936
7	22	2	1	23	2.671
8	22	2	1	23	2.856
9	**24**	**0**	**1**	**23**	**1.825**
10	22	2	1	23	2.621

In the case of data pre-processed by PCA algorithm, the minimal sum of squared error was obtained with nine hidden units. The best classification error achieved was 1.825 with the standard deviation of 2.13. Linear regression is presented in Figure 8.

By comparing Figures 8 and 6, one can see that a better regression identified by the parameter R is achieved with the not pre-processed data. However, by analyzing those figures, we can state that in the case of pre-processed data one healthy eye has been classified as glaucomatous and the error has stepped out far beyond the decision boundary. Other than in Figure 6, there are two cases where the diagnosis of illness is questionable (circles are near to the decision boundary).

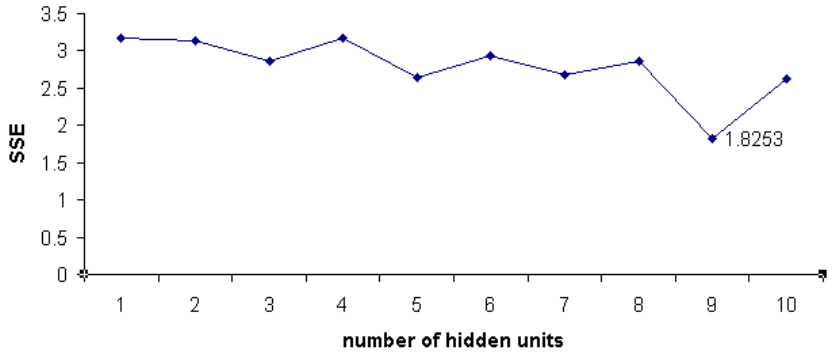

Figure 7. SSE according to the number of hidden units with scaled data after PCA analysis

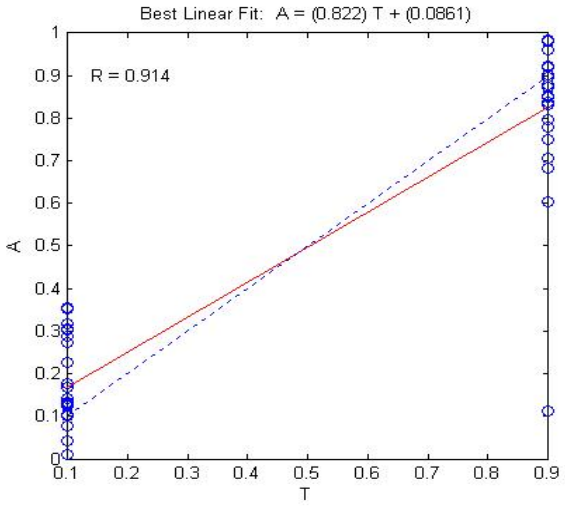

Figure 8. Linear regression of scaled data

4. Conclusions

In this paper, the neural network was investigated as a disease classifier. The neural network with one hidden layer was used. The network activation function logsig and the Levenberg-Marquardt learning algorithm have been applied and the results achieved are satisfactory.

The optimal number of hidden neurons for this kind of problem solving is 8 in the sense of SSE of non-pre-processed initial data. The neural network produced a classification result that contained only one false unit of a glaucomatous eye classified as healthy.

In addition, in the results section, we have shown that the initial data pre-processed by the principal component analysis reduce quantity of features from 27 to 10. This yields the best network performance with 9 hidden units. Besides, the network SSE response is much smoother than that of non-scaled data. After the classification such a network has produced the output, which consists of one false result where the healthy eye has been classified as glaucomatous.

However, the network should be properly evaluated on a larger set of input vectors used for network training and validation. Also, in the future it should be investigated which of the parameter vector features are most informative.

Acknowledgement

The research is partially supported by the Lithuanian State Science and Studies Foundation project "Information technology tools of clinical decision support and citizens wellness for e.Health system (No. B-07019)".

References

[1] **D. Jegelevicius, A. Lukosevicius, A. Paunksnis, V. Barzdziukas.** Application of Data Mining Technique for Diagnosis of Posterior Uveal Melanoma. *Informatica* 13 (4), 2002. 455-464.

[2] **N. Lavrac, E. Keravnou, B. Zupan (editors).** Intelligent Data Analysis in Medicine and Pharmacology. *Springer Verlag*. ISBN: 978-0792380009, 1997, 175.

[3] **K.P. Adlassnig (editor).** Artificial intelligence in medicine. *Elsevier*, 2003, ISSN:09333657.

[4] **J. Bernataviciene, G. Dzemyda, O. Kurasova, V. Marcinkevicius, V. Medvedev.** The Problem of Visual Analysis of Multidimensional Medical Data. Models and Algorithms for Global Optimization. *Springer Optimization and its Applications (A. Torn*, J.

Žilinskas editors), *Vol*. 4, *Springer*, 2007, 277-298. ISSN 1931-6828, ISBN 0-387-36720- 9.

[5] F. Rosenblatt. Principles of Neurodynamics; perceptrons and the theory of brain mechanisms. *Washington, Spartan Books,* 1962.

[6] E.M. Corwin, A.M. Logar, W.J.B. Oldham. An Iterative Method for Training Multilayer Networks with Threshold Functions. *IEEE Transactions on Neural Networks*, 1995, 507–508.

[7] K.D. Maier, C.Beckstein, R.Blickhan, W. Erhard, D. Fey. A Multi-Layer-Perceptron Neural Network Hardware Based on 3D Massively Parallel Opto-electronic Circuits. *In Proceedings of the* 6*th International Conference on Parallel Interconnects*, 1999, 73-80.

[8] L.M. Bruce, N. Shanmugam. Using neural networks with wavelet transforms for an automated mammographic mass classifier Engineering in Medicine and Biology Society. *Proceedings of the* 22*nd Annual International Conference of the IEEE Vol*.2. 2000, 985 – 987.

[9] D.E. Rumelhart, G.E. Hinton, R.J. Williams. Learning Internal Representations by Error Propagation. (*D.E. Rumelhart., J.L. McClelland editors*) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition Vol*.1, *MIT Press* 1986, 318-362.

[10] D. Marquardt. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM J. Appl. Math*. 11, 1963, 431-441.

[11] S. Saarinen, R. Bramley, G. Cybenko. Ill-conditioning in Neural Network Training Problem. *SIAM J. Sci. Comp*. 14, 1993, 693-714.

[12] A. Abraham, B. Nath. ALEC: An Adaptive Learning Framework for Optimizing Artificial Neural Networks. *Lecture Notes in Computer Science, Vol*. 2074/2001, *Springer Berlin / Heidelberg*, 2001, 171. ISSN 1611-3349.

[13] H.T. He, H.M. Liu. The research on integrated neural networks in rolling load prediction system for temper mill Machine Learning and Cybernetics. *Proceedings of* 2005 *International Conference Vol*.7, *Issue* 18-21. 2005, 4089 - 4093

[14] A. Weigend, B. Huberman, D. Rumelhart. Predicting Sunspots and Exchange Rates with Connectionist Networks. *Nonlin. Modeling and Forecasting, Addison-Wesley*, 1991, 395-432.

[15] S. İçer, S. Kara, A. Guvenc. Comparison of multilayer perceptron training algorithms for portal venous Doppler signals in the cirrhosis disease. *Expert Systems with Applications Vol*.31, *Issue* 2. 2006, 406-413, ISSN: 0957-4174.

Received August 2007.