Application of Neural Networks for the Classification of Diffuse Liver Disease by Quantitative Echography

M.S. klein Gebbinck* J.T.M. Verhoeven‡ J.M. Thijssen‡
T.E. Schouten*

Abstract

Three different methods were investigated to determine their ability to detect and classify various categories of diffuse liver disease. A statistical method, i.e., discriminant analysis, a supervised neural network called backpropagation and a nonsupervised, self-organizing feature map were examined. The investigation was performed on the basis of a previously selected set of acoustic and image texture parameters. The limited number of patients was successfully extended by generating additional but independent data with identical statistical properties. The generated data were used for training and test sets. The final test was made with the original patient data as a validation set. It is concluded that neural networks are an attractive alternative to traditional statistical techniques when dealing with medical detection and classification tasks. Moreover, the use of generated data for training the networks and the discriminant classifier has been shown to be justified and profitable.

Key words: Classification; diffuse disease; discriminant analysis; liver; neural networks; ultrasound.

1 Introduction

Ultrasonic tissue characterization is investigated in the fields of both medicine and agriculture. An example of medical research is the detection of diffuse liver disease [1, 2, 3, 4, 5, 6, 7, 8, 9], while a possible agricultural application is meat quality grading [10]. Several acoustic and texture parameters can be extracted from echograms of the tissue on the basis of which characterization takes place [11]. Subsequent classification into one of a set of predefined classes requires a classifier, for which traditionally a statistical approach is chosen. In recent years, however, artificial neural networks have been shown to be a promising alternative [10, 12].

*Department of Real-Time Systems, Faculty of Mathematics and Computer Science, University of Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands.

‡Biophysics Laboratory of the Institute of Ophthalmologoy, University Hospital, P.O. Box 9101, 6500 HB Nijmegen, The Netherlands.

Address for reprints: Dr. Johan M. Thijssen, Biophysics Laboratory of the Institute of Ophthalmologoy, University Hospital, P.O. Box 9101, 6500 HB Nijmegen, The Netherlands.

An artificial neural network [13, 14] is a system of highly connective but simple processing units, inspired by the architecture of the human brain. In contrast to classical discriminant analysis, which is limited to linear or quadratic separation functions [15], neural networks are capable of nonlinear strategies. A neural network is trained by repeatedly presenting a set of training vectors to it. If during training for each vector, its classification is used as feed-back, the neural network is said to be supervised, while unsupervised otherwise.

In this paper, earlier work regarding the detection of diffuse liver disease [9] is extended by comparing three different classifiers. Apart from the statistical *discriminant analysis* method, the unsupervised *self-organizing feature map* as well as the supervised neural network with *backpropagation* have been investigated. The goal of the study was to find the classifier by which various disease categories can best be discriminated from a group of normals. For this purpose, a set of three acoustic and two texture parameters was selected for the classifications [11]. For each disease, the design and specifications of a semi-optimal network were determined experimentally [16].

2 Methods and Material

For an extensive description of the methods employed for data acquisition and analysis is referred to previous publications [7, 8, 9, 11]. The five parameters used in this paper have been chosen because they are not strongly correlated with each other [11] and have been shown to possess high discriminative capabilities [9]. However, in contrast to this earlier work, classification was performed on the basis of all five parameters, instead of application of the stepwise parameter selection method [15].

The selected parameters comprise the acoustic parameters β, α_1 and S, and the texture parameters μ and SNR. Parameter β represents the slope of the attenuation coefficient determined with the quasi-multinarrowband method, where the linear regression line is forced to have zero intercept at zero frequency. The attenuation coefficient slope α_1 was determined with the same method as β, but the fit was not forced to have zero intercept. After correction of the data for the frequency dependent attenuation by using β [8], a single backscattering spectrum was calculated for each region-of-interest acquired. The slope of the linear regression line of the backscattering spectrum between 1.2 and 3.2 MHz is symbolized by S. The texture parameters used are derived by first-order statistical image analysis: μ represents the estimated mean grey level, while SNR is the signal-to-noise ratio, equal to the mean divided by the standard deviation of the grey level histogram.

In this paper, four disease classes are to be discriminated against a normal population (NO, n=129). The first class are women with primary biliary cirrhosis (PBC, n=10). Patients with acute hepatitis (AH, n=13) constitute the next class. The third class consists of patients with alcoholic hepatitis/cirrhosis (AHC, n=11). All patients with other types of hepatitis/cirrhosis (HC, n=29) are placed in the last class. For all patients, their disease was proven by a biopsy, except for those suffering from acute hepatitis, which was assessed by

Correlations between parameters

x	$\rho_{\alpha_1 x}$	$\rho_{\mathrm{S}x}$	$\rho_{\mu_A x}$	$\rho_{\mathrm{SNR}x}$
β	0.212 (0.241)	-0.055 (-0.063)	0.568 (0.613)	0.290 (0.303)
α_1	–	0.132 (0.129)	0.440 (0.461)	-0.137 (-0.144)
S	–	–	-0.075 (-0.123)	-0.425 (-0.470)
μ_A	–	–	–	0.098 (0.122)

Table I: Correlations between the selected parameters for the normal population after extension with generated cases. The correlations between the parameters before extension of this group are indicated between parentheses.

clinical diagnosis. For both the normals as well as the patients, the parameters were corrected for age effects [17].

To ensure that a neural network classifies previously unseen data correctly, the training set must be sufficiently large. Therefore, each class has been extended to contain 1000 cases, using the optimal kernel estimated from the probability density function (pdf) of the original data [18]. Since it can be shown [16] that all five parameters are approximately distributed according to a normal probability density function, the Bartlett kernel with a smoothing parameter $h_n = 1.8218\sigma n^{-1/5}$ can be shown to be optimal for the estimation of the pdf of the data [18]. The new data were generated by making a random selection in Bartlett windows around the original parameters. In this way the correlations between the parameters of the generated data set are approximately equal to those of the original data set as can be seen in table I. The mean values and standard deviations differed by less than one percent and five percent, respectively. For the disease classes, similar results were obtained.

Further processing of the data includes scaling and splitting, necessary for use with neural networks. All parameters are scaled separately, where the mean minus two times the standard deviation is projected onto 0.1 and the mean plus two times the standard deviation onto 0.9. Some of the information is lost in this procedure, but the influence of outliers is reduced. After scaling, three sets are constructed: the *training set*, containing 500 generated cases for each class, the *test set*, containing the remainder of the generated data completed with the — scaled — original data, and the *validation set*, consisting only of original data. The need for this three sets will be explained in sections 5 and 6.

Discriminant analysis was performed using the individual within-group covariance matrices, resulting in a quadratic discriminant function [19]. Equal prior probabilities were assumed in the calculation of this function. The discriminant function was determined on the basis of the training set, while classification was performed on the validation set.

3 Self-organizing Feature Mapping

A feature map (Fig. 1) consists of a number of simple processing units, which in biological systems are called neurons. These processing units are arranged

Feature mapping network

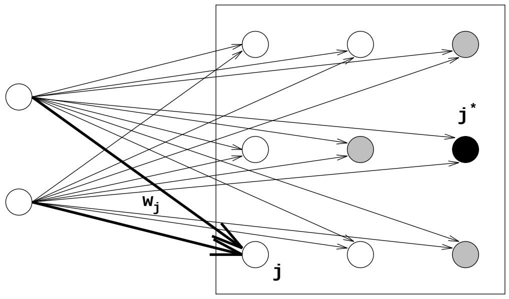

Figure 1: Possible architecture of a feature map, consisting of two units in the input layer and nine units in the output layer, with the input layer fully connected to the output layer. Weight vector $\vec{w_j}$ consists of the two bold arrows. Grey units are within the circular neighborhood (radius 1) of the winning unit j^* (in black).

in an input layer and an output layer. The units in the input layer are fully connected to the units in the output layer, while between units in the same layer no connections exist. Usually, the units in the output layer are organized in a two-dimensional grid, but other organizations are possible as well [13, 14, 21].

Training a feature map is done unsupervised, meaning that no classification feedback is required to train the network. The error that is to be minimized is given by

$$\mathrm{E}(w) = 1/2 \sum_p \min_j |\vec{w_j} - \vec{v_p}| \tag{1}$$

where p ranges over the training vectors and j over the output units, $|\cdots|$ represents the Euclidean distance function, $\vec{w_j}$ is the weight vector of unit j and $\vec{v_p}$ is the currently presented training vector. A competitive training procedure is involved, which is based on the principle of similarity matching: the output unit whose weight vector is most similar to the presented input vector is declared to be the winner. The winning unit (index j^*) meets the condition

$$\forall j \qquad |\vec{w_{j^*}} - \vec{v}| \leq |\vec{w_j} - \vec{v}| \tag{2}$$

Usually, the weight vectors are adjusted after presentation of a single, randomly drawn training vector, according to the rule

$$\Delta \vec{w_j} = \begin{cases} \eta(\vec{v} - \vec{w_j}) & \text{if } j \in N_{j^*} \\ \vec{0} & \text{otherwise} \end{cases} \tag{3}$$

with η — the learning rate — controlling the speed of learning. The topological neighbourhood N_{j^*} contains all units being within a certain radius centered at the winning unit j^*. The application of such a neighborhood function accomplishes the goal that the map organizes itself in a topologically preserving way. This means that vectors that are located closely in feature space have winning units that are located closely on the feature map. At the start of the

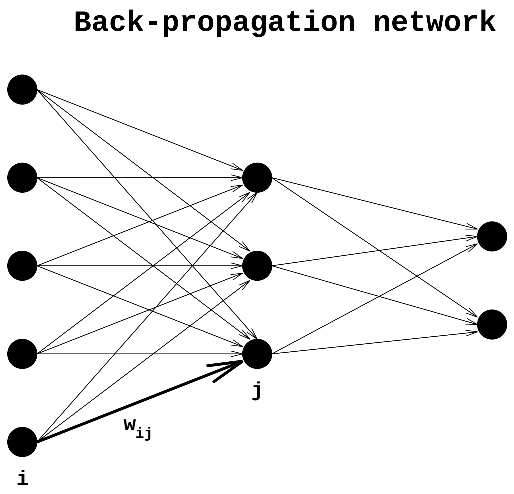

Figure 2: Possible back-propagation network, containing five units in the input layer, three units in its single hidden layer and two units in the output layer, with all successive layers fully connected. Weight w_{ij} has been drawn bold.

training process the weights are set at random values, resulting in a totally disorganized map. To get a rough organization of the feature map, the radius of the neighborhood has to be relatively large. During training, however, the neighborhood's radius should decrease to allow the network to tune itself to the training vectors. Therefore, both the learning rate as well as the neighborhood radius are decreased linearly in time.

After the map has been trained sufficiently, each unit in the output layer is classified. Although training the feature map itself is unsupervised, subsequent classification requires the vectors in the training set to be preclassified. The class that is assigned to a unit is equal to the class of the training vectors that have that unit as a winner most frequently.

Of the multiple variations on the training procedure only one — the conscience mechanism — has been investigated by the authors. A description of this mechanism can be found in [20, 16].

4 Back-propagation

A backpropagation network (Fig. 2) consists of units spread over an input layer, possibly some hidden layers, and an output layer. Between units in successive layers, weighted connections exist, which determine the function executed by the network [13, 14, 22].

Each unit has an internal state or activation level (a_j), depending on the input received from units in a previous layer (x_i), the associated weights of the connections (w_{ij}) and value of its threshold (θ_j):

$$a_j = \sum_i x_i w_{ij} - \theta_j \tag{4}$$

For most units, the output (x_j) is calculated by passing its activation through a sigmoidal function, like

$$x_j = \frac{1}{1 + e^{-a_j}} \tag{5}$$

However, the output of a unit belonging to the input layer is equal to the corresponding element of the data vector currently presented to the network. Using this procedure, a vector presented to the input layer is fed forward to the output layer.

Backpropagation is a supervised technique of training, which means that the training procedure needs the vectors in the training set to be classified. Usually, the classification of an input vector is determined by the element of the output vector that has the highest value. Therefore, when a backpropagation network is used as classifier for n classes, for class i an n-dimensional output vector is constructed where all but one elements are low and the i-th element is high. Training the network involves adjusting the weights, which are initialized at small random values, until the network approximates the desired output closely enough.

A measure of how much the network is deviating from the desired performance is expressed by the error function $\mathrm{E}(w)$

$$\mathrm{E}(w) = 1/2 \sum_{i,p} (d_{i,p} - x_{i,p})^2 \tag{6}$$

where d represents the desired output vector, i ranges over the output units and p over the training vectors. In this view, training coincides with minimizing the error function. This is achieved by iteratively changing the weights using gradient descent

$$\Delta w_{ij} = -\eta \partial \mathrm{E}(w) / \partial w_{ij} \tag{7}$$

where the learning rate η controls the speed of learning. While in Eq. (7) the weights are adjusted after all training vectors have been presented, i.e., after an entire iteration, usually the weights are changed after presentation of a single vector. If the vectors are drawn at random from the training set, the training procedure is extended with stochastic noise, which improves the performance of the network [14]. For this strategy, it can be derived [14, 16] that

$$\Delta w_{ij} = \eta x_i \delta_j,\ \delta_j = \begin{cases} (1 - x_j^2)(d_j - x_j) & \text{for output units} \\ (1 - x_j^2) \sum_k \delta_k w_{jk} & \text{otherwise} \end{cases} \tag{8}$$

It is this mechanism that has provided the name (error) backpropagation for this type of neural networks.

Sometimes, to speed up training, a momentum term is added

$$\Delta w_{ij}(t+1) = -\eta \partial \mathrm{E}(w) \partial w_{ij} + \alpha \Delta w_{ij}(t) \tag{9}$$

where t represents the time at which the network is adjusted. There exist many more variations on the training procedure, but they are not described in this paper. For an explanation of the variations investigated by the authors — weight decay and distortion of the training vectors — the reader is referred to [16].

5 Designing a Neural Network

In the design of a neural network, two features have to be optimized: *convergence* and *generalization*. A network is said to converge on the training data if the error function it tries to minimize decreases during training, i.e., the training set is memorized increasingly well. Convergence is not inherent behavior, because the network may be too small to be able to solve the problem in the first place. Furthermore, the surface of the error function can contain local minima, causing the gradient descent procedure to get trapped. Addition of more units or, in case of backpropagation, another hidden layer solves the first problem [14], while choosing different initial weights may overcome the second problem. Convergence can only be determined experimentally by monitoring the network's error function given by Eqs. (1) and (6) during training.

The second and most important feature of a neural network is its ability to generalize over the problem. It is possible that a network perfectly memorizes all vectors of the training set, but fails on previously unseen vectors. An explanation for this behavior is overlearning, which means that the network focuses too much on the training set and disregards the general characteristics of the problem itself. Good generalization is more likely if the number of weights is reduced, of course without reducing the ability to solve the problem. Unfortunately however, the relation between the settings of network parameters and good generalization is not exactly understood. The only way to determine the generalization capabilities and to avoid overlearning is by monitoring an error function over an independent test set during training. As soon as the error over the test set starts to increase, while the error over the training set still decreases, training should be stopped. A suitable function used by the authors to detect overlearning is based on the correct fractions (CF) of classification, the "specificity" (P_{sp}) and "sensitivity" (P_{se}):

$$\mathrm{CF}(w) = 1 - \sqrt{\frac{(1-P_{sp})^2 + (1-P_{se})^2}{2}} \tag{10}$$

The specificity here represents the fraction of true negatives, i.e., correctly classified normals, while the sensitivity indicates the fraction of true positives, i.e. correctly detected disease cases. For a problem where i classes are to be discriminated simultaneously, Eq. (10) can easily be extended to:

$$\mathrm{CF}(w) = 1 - \sqrt{1/i \sum_i (1-P_i)^2} \tag{11}$$

Here i ranges over all classes and P_i represents the correct classification fraction of class i. An advantage of the CF function over a normal error function is not only that the network can be prevented from overlearning, but also correct classification, in our opinion the most interesting property, can be examined. In figure 3, convergence and generalization for several (backpropagation) network architectures are presented. It can be found that convergence improves with increasing training (and more weights), while, on the other hand, generalization deteriorates after 3,000 to 5,000 iterations.

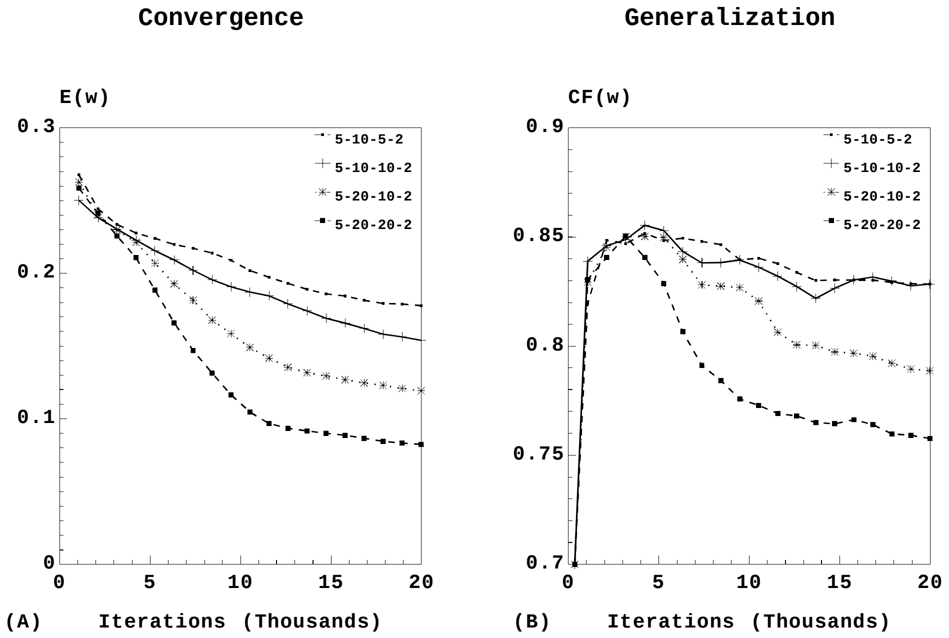

Figure 3: Convergence (A) and generalization (B) of several back-propagation networks with two hidden layers. E(w) is normalized for the number of training vectors and the number of classes. Each data point is the average over 25 successive iterations.

Designing a good neural network is most of all a time-consuming task, because of the many design choices that have to be made. These choices include basic settings like the network architecture, learning rate, initial weight distribution etc., but also whether or not variations on the training procedure like distorting the training vectors, weight decay or the conscience mechanism [16] are to be applied. For every possible combination, a neural network has to be simulated to assess convergence and generalization.

6 Results

For self-organizing feature mapping as well as backpropagation, the number of neural networks that had to be implemented according to the methods described in section 5 would be too extensive. To reduce this vast number of time-consuming simulations, the parameters of the network were optimized in order of — assumed — decreasing importance, after which each variation on the training procedure was assessed separately. For parameters that had not yet been optimized, values were chosen following certain rules of thumb found in literature. The network with the best generalization, i.e., highest CF on the test set, was chosen for further optimization. In figure 3, convergence and generalization of several backpropagation networks with two hidden layers are shown. Since the 5-10-10-2 network achieved the best generalization, although certainly not the best convergence, this network was optimized. After the semi-optimal network configuration had been established, specificity and sensitivity of the network regarding the validation set were determined.

In the case of detection of the various disease classes separately vs. normals, the initial weights of the feature mapping network were drawn from the feature space covered by the training set. The parameters that were successively optimized are the network architecture (5-3x3, 5-4x4, 5-5x5, 5-6x6, 5-7x7, 5-8x8, 5-9x9, 5-10x10) and the learning rate (0.00025, 0.005, 0.1). The only variation on the training procedure that was examined was the conscience mechanism. Each feature map was trained for 1,000 iterations. The optimal feature mapping network for the detection tasks was found to have the following configurations: architecture 5-7x7 for classes PBC and AH vs. NO and 5-8x8 for AHC and HC vs. NO, learning rate 0.005, no conscience.

The weights of the backpropagation network were initialized with values randomly chosen from the interval [-0.5,0.5]. The parameters of the network that were optimized are, in order of decreasing importance, the network architecture (5-5-2, 5-10-2, 5-20-2, 5-40-2, 5-10-5-2, 5-10-10-2, 5-20-10-2, 5-20-20-2), the learning rate (0.0005, 0.001, 0.2) and the momentum (0.0, 0.4, 0.8, 0.95). Furthermore, two variations on the basic training procedure were examined: distorting the training patterns with noise (2%, 5%, 10%) and weight decay (0.001, 0.01). Each network was trained for 10,000 iterations. The optimal backpropagation network configuration for the detection tasks was: architecture 5-40-2 for classes PBC and AH vs. NO and 5-20-2 for AHC and HC vs. NO, learning rate 0.01, momentum 0.8, no weight decay, no distortion of training vectors.

Detection by discriminant analysis

disease	n	P_{sp} (%)	P_{se} (%)	CF (%)
PBC	10	93.0 (96.9)	90.0 (70.0)	91.4 (78.7)
AH	13	89.9 (88.4)	76.9 (61.5)	82.2 (71.6)
AHC	11	96.1 (95.4)	72.7 (27.3)	80.5 (48.5)
HC	29	95.3 (93.8)	62.1 (58.6)	73.0 (70.4)

Table II: Detection of diffuse liver diseases with respect to normal liver (NO, n=129) for the validation set when discriminant analysis is applied after calibrating with generated data. Between parentheses the results of the discriminant analysis by the "Jack-knife" (cross-validation) method. Patient categories: PBC=primary biliary cirrhosis, AH=acute hepatitis, AHC=alcoholic hepatitis/cirrhosis, HC=hepatitis/cirrhosis.

Detection by feature mapping

disease	n	P_{sp} (%)	P_{se} (%)	CF (%)
PBC	10	95.3	100.0	96.7
AH	13	93.0	92.3	92.6
AHC	11	88.4	90.9	89.6
HC	29	82.2	82.8	82.5

Table III: Detectability of diffuse liver diseases by a self-organizing feature map. Patient categories are as in table II. PBC and AH: 5-7x7 network; AHC and HC: 5-8x8 network.

The detection results of the validation set for the various categories of diffuse liver disease vs. the group of normals can be found in table II for discriminant analysis, in table III for self-organizing feature mapping and in table IV for backpropagation. The correct fractions (CF) of detection are listed in the rightmost column of tables II to IV. Both neural networks yield a considerably improved correctness.

The optimal feature mapping network configuration for the discrimination task, i.e., classification of all the groups simultaneously, was found to be 5-7x7, learning rate 0.005, no conscience. The optimal backpropagation network was: 5-40-5, learning rate 0.01, momentum 0.8, no weight decay, no distortion of training vectors. The classification results of the validation set are presented, for discriminant analysis, feature mapping and backpropagation in table V. Also the classification (or differentiation) of the various disease classes is improved significantly by using neural networks as compared to discriminant analysis. The backpropagation network stands out as the optimal one.

7 Discussion

The three methods that have been investigated for their ability to detect and classify the various categories of diffuse liver disease can be compared fairly,

Detection by back-propagation

disease	n	P_{sp} (%)	P_{se} (%)	CF (%)
PBC	10	99.2	100.0	99.4
AH	13	99.2	92.3	94.5
AHC	11	96.1	81.8	86.8
HC	29	86.8	82.8	84.7

Table IV: Detectability of diffuse liver diseases by a back-propagation network. Patient categories are as in table II. PBC and AH: 5-40-2 network; AHC and HC: 5-20-2 network.

Classification of all diseases

method	P_{NO} (%)	P_{PBC} (%)	P_{AH} (%)	P_{AHC} (%)	P_{HC} (%)	CF (%)
JK	86.8	40.0	30.8	0.0	10.3	27.0
DA	87.6	90.0	61.5	18.2	24.1	46.6
FM	65.9	90.0	46.2	63.6	58.6	62.1
BP	82.9	100.0	92.3	45.5	75.9	72.0

Table V: Classification of all diffuse liver disease categories simultaneously. Method acronyms: JK=discriminant analysis by the Jack-knife method applied to the original data set, DA=discriminant analysis by using generated data for calibration, FM=feature mapping, BP=back-propagation. Patient categories are as in table II.

since the procedure they follow basically is the same. First, the classifier is calibrated using a training set, after which its performance is assessed by classifying a validation set. Considering the values of the correct fraction for each class, weighted by using the root-mean-squares method, the best performance is achieved by backpropagation, followed by self-organizing feature mapping and discriminant analysis. On the other hand, the time needed to calibrate the classifier ranged from short for discriminant analysis to very long for backpropagation, with feature mapping in between. However, each classifier needs to be calibrated only once, while classification, which must be performed many times, is equally fast for all methods.

In tables II and V, a comparison can be made between the results after using generated data as a training set in the discriminant procedure and the Jack-knife method of discriminant analysis. In the latter method the validation set is used for both the training and the validation. The detectability (Table II) of diseases is improved to some extent by using the generated data. In addition to the theoretical argument (optimal estimate of the pdf by using the Bartlett kernel) and the preservation of the correlations between the parameters (Table I), it can be stated that generating data is not only useful for training, but that it also improves the classification procedure. Since the generated data for each parameter are statistically independent from the original data set (random drawing according to pdf), it is permitted to generalize to the conclusion that a significant improvement of the employment of small data sets for diagnostic tasks has been achieved. This conclusion is strongly supported by the results of table V for the classification task (first two rows): the overall fraction of correct classifications has increased by a factor of two. The same arguments justify and support the use of the generated data for the classification by neural networks. As becomes evident in table V, the classification is considerably further enhanced (rightmost column) by using neural networks. As compared to the Jack-knife discriminant analysis, the CF value improves even by a factor of 2.6 by the back-propagation network.

The results of the discriminant analysis method in tables II and V can be compared to the results in recent papers by Hartman *et al* [23] and Oosterveld *et al* [9]. Two differences should be noted, however: the parameters used are not identical and a slight difference occurs between the number of patients of the various categories. The conclusion is that the optimal set (two or three parameters) of parameters yielded better results than the fixed set of five parameters used in the present study, especially in the detection task [9, 23]. Furthermore, the results of the discrimination analysis by using the generated data as a training set are comparable to the results of the two by two classification (differentiation would be more appropriate) as reported in another study [23]. So the conclusions formulated in the previous paragraph are supported here.

From this study, it can be concluded that for both the detection and the discrimination of diffuse liver disease neural networks are an attractive alternative to discriminant analysis. The generation of data has positively influenced the results, but it has no effect on the order of the classifiers, because all methods use the same data. An extension of the data set with real patients' material could be used to neutralize a few clear outliers. Further improvement of the classification might be found in the employment of data sets based on other selections of parameters. The dependence of neural networks on the number as well as the selection of parameters could be further investigated. Other research topics could comprise the preprocessing of data and the application of different classifiers.

References

[1] Maklad, N.F., Ophir, J., Balsara, V., Attenuation of ultrasound in normal liver and diffuse liver disease in vivo, *Ultrasonic Imaging 6*, 117–125 (1984).

[2] Nicholas, D., Nassiri, D.K., Garbutt, P., Hill, C.R., Tissue characterization from ultrasound B-scan data, *Ultrasound Med. Biol. 12*, 135–143 (1986).

[3] Taylor, K.J.W., Riely, C.A., Hammers, D.O. et al., Quantitative US attenuation in normal liver and in patients with diffuse liver disease: importance of fat, *Radiology 160*, 65–71 (1986).

[4] Schlaps, D., Zuna, J., Walz, M. et al., Ultrasonic tissue characterization by texture analysis. Elimination of tissue-independent factors, in *Proc. SPIE 768*, pp. 128–134 (1987).

[5] Lizzi, F.L., King, D.L., Rorke, M.C. et al., Comparison of theoretical scattering results and ultrasonic data from clinical liver examinations, *Ultrasound Med. Biol. 14*, 377–386 (1988).

[6] Garra, B.S., Insana, M.F., Shawker, T.H. et al., Quantitative ultrasonic detection and classification of diffuse liver disease: comparison with human observer performance, *Inv. Radiol. 24*, 196–203 (1989).

[7] Oosterveld, B.J., *On the Quantitative Analysis of Ultrasound Signals with Applications to Diffuse Liver Disease*, Ph.D. Thesis University of Nijmegen (1990).

[8] Oosterveld, B.J., Thijssen, J.M., Hartman, P.C., Romijn, R.L., and Rosenbusch, G.J.E., Ultrasound attenuation and texture analysis of diffuse liver disease: methods and preliminary results, *Phys. Med. Biol. 36*, 1013–1064 (1991).

[9] Oosterveld, B.J., Thijssen, J.M., Hartman, P.C., and Rosenbusch, G.J.E, Detection of diffuse liver disease by quantitative echography: dependence on a priori choice of parameters, *Ultrasound Med. Biol. 19*, 21–25 (1993).

[10] Whittaker, A.D., Park, B.S., McCauley, J.D., and Huang, Y., Ultrasonic Signal Classification for Beef Quality Grading through Neural Networks, in *Proceedings of the 1991 Symposium on Automated Agriculture for the 21st Century*, pp. 116–125.

[11] Thijssen, J.M., Oosterveld, B.J., Hartman, P.C., and Rosenbusch, G.J.E., Correlations between acoustic and texture parameters from RF- and B-mode liver echograms, *Ultrasound Med. Biol. 19*, 13–20 (1993).

[12] Ostrem, J.S., Valdes, A.D., and Edmonds, P.D., Application of Neural Nets to Ultrasound Tissue Characterization, *Ultrasonic Imaging 13*, 298–299 (1991).

[13] Dayhoff, J.E., *Neural Network Architectures. An Introduction* (Van Nostrand Reinhold, New York, 1990).

[14] Hertz, J., Krogh, A., and Palmer, R.G., *Introduction to the Theory of Neural Computation* (Addison-Wesley Publishing Company, Redwood City, 1991).

[15] Albert, A., and Harris, E.K., *Multivariate interpretation of clinical laboratory data* (Marcel Dekker, New York, 1987).

[16] Gebbinck, M.S. klein, *Ultrasonic Tissue Characterization Using Neural Networks*, M.Sc. Thesis University of Nijmegen (1992).

[17] Hartman, P.C., Oosterveld, B.J., Thijssen, J.M., and Rosenbusch, G.E.J., Variability of quantitative echographic parameters of the liver: intra- and inter-individual spread, temporal- and age-related effects, *Ultrasound Med. Biol. 17*, 857–867 (1991).

[18] Holmström, L., and Koistinen, P., Using additive noise in back-propagation training, *IEEE Trans. Neural Networks 3*, 24–38 (1992).

[19] SAS Institute Inc., *SAS/STAT User's Guide Version 6, Volume 1* (SAS Institute Inc., Cary (NC), 1989).

[20] DeSieno, D., Adding a conscience to competitive learning, in *Proc. IEEE Int. Conf. Neural Networks 1988*, pp. 117–124 (1988).

[21] Kohonen, T., *Self-organization and Associative Memory* (Springer-Verlag, Berlin, 1984).

[22] Rumelhart, D.E., Hinton, G., Williams, R., Learning representations by backpropagating errors, *Nature 323*, 533–536 (1986).

[23] Hartman, P.C., Oosterveld, B.J., Thijssen, J.M., Rosenbusch, G.J.E., Berg, J. van den, Detection and differentiation of diffuse liver disease by quantitative echography: a retrospective assessment, *Inv. Radiol. 28*, 1–6 (1993).