Computers in Biology and Medicine 36 (2006) 376–388
Computers in Biology and Medicine
www.intl.elsevierhealth.com/journals/cobm

A fuzzy clustering neural network architecture for classification of ECG arrhythmias

Yüksel Özbay[a,*], Rahime Ceylan[a], Bekir Karlik[b]

[a] *Department of Electrical & Electronics Engineering, Selcuk University, Konya, Turkey*
[b] *Department of Computer Engineering, Halic University, Istanbul, Turkey*

Received 6 August 2004; accepted 31 January 2005

Abstract

Accurate and computationally efficient means of classifying electrocardiography (ECG) arrhythmias has been the subject of considerable research effort in recent years. This study presents a comparative study of the classification accuracy of ECG signals using a well-known neural network architecture named multi-layered perceptron (MLP) with backpropagation training algorithm, and a new fuzzy clustering NN architecture (FCNN) for early diagnosis. The ECG signals are taken from MIT-BIH ECG database, which are used to classify 10 different arrhythmias for training. These are normal sinus rhythm, sinus bradycardia, ventricular tachycardia, sinus arrhythmia, atrial premature contraction, paced beat, right bundle branch block, left bundle branch block, atrial fibrillation and atrial flutter. For testing, the proposed structures were trained by backpropagation algorithm. Both of them tested using experimental ECG records of 92 patients (40 male and 52 female, average age is 39.75 ± 19.06). The test results suggest that a new proposed FCNN architecture can generalize better than ordinary MLP architecture and also learn better and faster. The advantage of proposed structure is a result of decreasing the number of segments by grouping similar segments in training data with fuzzy c-means clustering.

Keywords: Fuzzy clustering; Fuzzy c-means; ECG; Arrhythmia; Neural network; Pattern recognition; Multilayer perceptron

* Corresponding author. Tel.: +90 332 223 2081; fax: +90 332 241 0635.
E-mail addresses: yozbay@selcuk.edu.tr (Y. Özbay), rpektatli@selcuk.edu.tr (R. Ceylan), bkarlik@halic.edu.tr (B. Karlik).

0010-4825/$ - see front matter © 2005 Published by Elsevier Ltd.
doi:10.1016/j.compbiomed.2005.01.006

1. Introduction

Electrocardiography deals with the electrical activity of the heart. Monitored by placing sensors at the limb extremities of the subject, electrocardiogram (ECG) is a record of the origin and the propagation of the electrical potential through cardiac muscles. It is considered a representative signal of cardiac physiology, useful in diagnosing cardiac disorders [1,2].

The state of cardiac heart is generally reflected in the shape of ECG waveform and heart rate. It may contain important pointers to the nature of diseases afflicting the heart. However, bio-signals being non-stationary signals, the reflection may occur at random in the time-scale (that is, the disease symptoms may not show up all the time, but would manifest at certain irregular intervals during the day). Therefore, for effective diagnostics, ECG pattern and heart rate variability may have to be observed over several hours. Thus the volume of the data being enormous, the study is tedious and time consuming. Naturally, the possibility of the analyst missing (or misreading) vital information is high. Therefore, computer-based analysis and classification of diseases can be very helpful in diagnostics [1]. Several algorithms have been developed in the literature for detection and classification of ECG beats. Most of them use either time or frequency domain representation of the ECG waveforms, on the basis of which many specific features are defined, allowing the recognition between the beats belonging to different classes. The most difficult problem faced by today's automatic ECG analysis is the large variation in the morphologies of ECG waveforms, not only of different patients or patient groups but also within the same patient. The ECG waveforms may differ for the same patient to such an extent that they are dissimilar to each other and at the same time they are similar for different types of beats. This is main reason that the beat classifier, performing well on the training data, generalizes poorly when presented with different patients' ECG waveforms [2].

One of the methods of ECG beat recognition is neural network classification method [3–10]. multi-layer perceptron (MLP) [10–31], which can be called "conventional backpropagation neural networks (BPNN)", has been shown to be able to recognize and classify ECG signals more accurately. For example Ozbay et al. and Foo et al. have used neural networks [3–5]. However conventional BPNN suffers from slow convergence to local and global minima and from random settings of initial values of weights, which may make the neural networks have very poor mappings from inputs to output. So, researchers have started to use hybrid structure. For example Pilla and Lopes [6], Osowski and Linh [2] and Engin and Demirag [7] have used fuzzy hybrid neural network.

This paper was proposed using fuzzy c-means (FCM) clustering algorithm to make a neural network system more effective. The structure proposed in this paper is composed of two subnetworks: fuzzy classifier and neural network. The fuzzy self-organizing layer performs the preclassification task and the following multilayer perceptron works as a final classifier. The fuzzy stage is responsible for the analysis of the distribution of data and grouping them into clusters with different membership values. On the basis of these membership values, the MLP network classifies the applied input vector, representing the heartbeat to the appropriate class. On the other hand, a number of segments in training patterns are reduced using FCM clustering in fuzzy self-organizing layer before inputs are presented to the MLP. The obtained new training data whose number of segments is decreased using fuzzy clustering are presented to the MLP. Therefore, training period of the neural network is decreased. This method proposed in this paper was used for electrocardiographic beat recognition and classification.

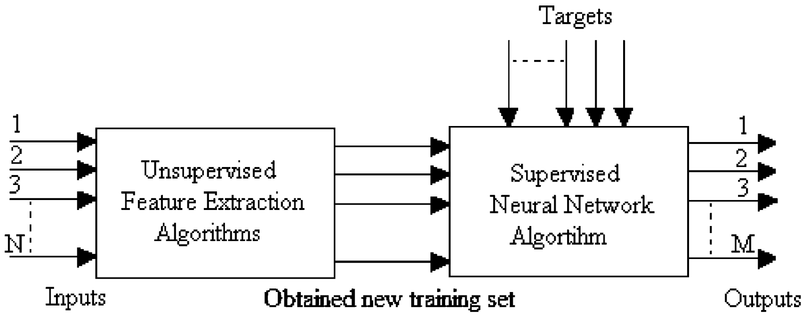

Fig. 1. FCNN architecture.

2. Fuzzy clustering neural networks

Different solutions for pattern recognition of the type of ECG waveform are presented in the literature, such as the MLP approach, self-organizing map and the LVQ. We will present the combination of the fuzzy self-organizing layer and the MLP connected in cascade, named the fuzzy clustering neural network (FCNN). The structure of such a network is presented in Fig. 1.

The self-organizing layer is responsible for the clustering of the input data. However, it is fuzzy clustering, in which the input vector x is preclassified to all sets with different membership values. The penetration of the data space is better and the localization of the input vector x in the data space is more precise. The outputs of all self-organizing neurons (the cluster centers) form the input vector to the second subnetwork (MLP). MLP subnetwork is responsible for the final classification of the ECG beat.

2.1. FCM clustering layer

Structure identification of fuzzy systems is possible by constructing enough rules with appropriate input and output membership functions. The identified model can then be used to describe the behavior of the target system as well as for prediction purposes. In this paper, we have trained the fuzzy layer by using the FCM clustering algorithm.

The idea of fuzzy clustering is to divide the data into fuzzy partitions that overlap with one another. Therefore, the inclusion of data in a cluster is defined by a membership grade in [0,1]. Formally, clustering an unlabeled data $X = \{x_1, x_2, \ldots, x_N\} \subset R^h$, where N represents the number of data vectors and h the dimension of each data vector, is the assignment of c partition labels to the vectors in X. c-partition of X constitutes sets of $(cN)\{u_{ik}\}$ membership values that can be conveniently arranged as a $(c \times N)$ matrix $U = [u_{ik}]$. The problem of fuzzy clustering is to find the optimum membership matrix U. The most widely used objective function for fuzzy clustering is the weighted within-groups sum of squared errors J_m, which is used to define the following constrained optimization problem [10–21]:

$$\min \left\{ J_m(U, V, X) = \sum_{k=1}^{N} \sum_{i=1}^{c} (u_{ik})^m \|x_k - v_i\|_A^2 \right\}, \tag{1}$$

where

$$U \in M_{fcn} = \left\{ U \in \Re^{cN} \left| \begin{matrix} 0 \leqslant u_{ik} \leqslant 1 \ \forall ik \ \& \ \forall k, u_{ik} > 0 \, \exists i \\ 0 < \sum_{k=1}^{N} u_{ik} > \eta \ \forall i \ \& \ \sum_{i=1}^{c} u_{ik} = 1 \ \forall k \end{matrix} \right. \right\}.$$

$V = \{v_1, v_2, \ldots, v_c\}$ is the vector of (unknown) cluster centers, and $\|x\|_A = (x^T A x)^{1/2}$ an inner product norm. A is an $h \times h$ positive definite matrix, which specifies the shape of the clusters. The matrix A is commonly selected as the identity matrix, leading to Euclidean distance and, consequently, to spherical clusters.

Fuzzy partitions are carried out using the FCM algorithm through an iterative optimization of [10–16] according to the following steps [10]:

Step 1: Choose the number of clusters (c), weighting exponent (m), iteration limit (iter), termination criterion ($\varepsilon > 0$), and norm for error $\|V_t - V_{t-1}\|$.

Step 2: Guess initial position of cluster centers:

$$V_0 = \{v_{1,0}, v_{2,0}, \ldots, v_{c,0}\} \subset \Re^{ch}.$$

Step 3: Iterate for $t = 1$ iter, calculate

$$u_{ik,t} = \left[\sum_{j=1}^{c} \left(\frac{\|X_k - V_{i,t-1}\|_A}{\|X_k - V_{j,t-1}\|_A} \right)^{2/m-1} \right]^{-1} \tag{2}$$

and

$$V_{i,t} = \frac{\sum_{k=1}^{N} (u_{ik,t})^m x_k}{\sum_{k=1}^{N} (u_{ik,t})^m}. \tag{3}$$

IF error $= \|V_t - V_{t-1}\| \leqslant \varepsilon$, THEN stop, and put $(U_f, V_f) = (U_t, V_t)$ for NEXT t.

2.2. Multilayer perceptron

In this study, a three-layered feed-forward NN is used and trained with the error backpropagation. The input signals of NN are formed by cluster centers that generalized by FCM clustering. Fig. 2 shows a general structure of the NN. The backpropagation training with generalized delta learning rule is an iterative gradient algorithm designed to minimize the root mean square error between the actual output of a multilayered feed-forward NN and a desired output. Each layer is fully connected to the previous layer, and has no other connection [8,9].

2.2.1. Backpropagation algorithm

1. *Initialization*: Set all the weights and biases to small real random values.
2. *Presentation of input and desired output*: Present the input vector $x(1), x(2), \ldots, x(N)$ and corresponding desired response $d(1), d(2), \ldots, d(N)$, one pair at a time, where N is the number of training patterns.

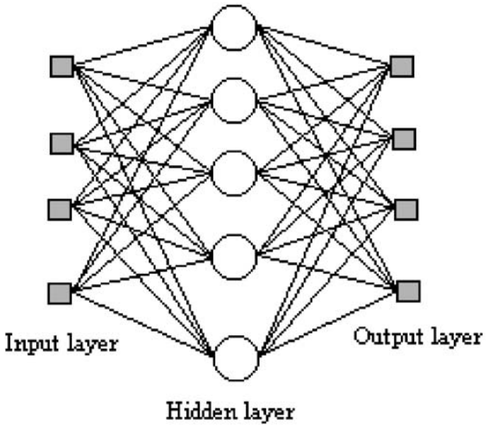

Fig. 2. The general structure of MLP NNs.

3. *Calculation of actual outputs*: Use Eq. (4) to calculate the output signals $y_1, y_2, \ldots, y_{N_M}$

$$y_i = \varphi\left(\sum_{j=1}^{N_{M-1}} w_{ij}^{(M-1)} x_j^{(M-1)} + b_i^{(M-1)}\right), \quad i = 1, \ldots, N_{M-1}. \tag{4}$$

4. Adaptation of weights (w_{ij}) and biases (b_i):

$$\Delta w_{ij}^{(l-1)}(n) = \mu x_j(n)\delta_i^{(l-1)}(n), \tag{5}$$

$$\Delta b_i^{(l-1)}(n) = \mu\delta_i^{(l-1)}(n), \tag{6}$$

where

$$\delta_i^{(l-1)}(n) = \begin{cases} \varphi'(\mathrm{net}_i^{(l-1)})[d_i - y_i(n)], & l = M, \\ \varphi'(\mathrm{net}_i^{(l-1)}) \sum_k w_{ki}\delta_k^{(l)}(n), & 1 \leqslant l \leqslant M, \end{cases} \tag{7}$$

in which $x_j(n)$ = output of node j at iteration n, l is layer, k is the number of output nodes of neural network, M is output layer, φ is activation function. The learning rate is represented by μ. It may be noted here that a large value of the learning rate may lead to faster convergence but may also result in oscillation. In order to achieve faster convergence with minimum oscillation, a momentum term may be added to the basic weight updating equation. In this study, the learning rate parameter was chosen as 1.0 via experimentation.

After completing the training procedure of the neural network, the weights of MLP are frozen and ready for use in the testing mode.

Table 1
The number of cluster centers for each arrhythmia

Arrhythmia type	For 106 sets	For 67 sets	For 212 sets
N	15	9	30
Br	15	9	30
VT	6	5	12
SA	15	9	30
APC	6	5	12
P	10	6	20
R	10	6	20
L	10	6	20
A.Fib.	10	6	20
A.Fl.	9	6	18
Total	106	67	212

3. Numerical experiments

In this paper, a clustering-based approach is adopted for ECG signal, which belongs to ten different arrhythmias used as input to the NN (see Fig. 1).

3.1. Structure and training data

Training data of ECG arrhythmias used in this study were taken from MIT-BIH ECG Arrhythmias Database. Selected types of arrhythmias were normal sinus rhythm (N), sinus bradycardia (Br), ventricular tachycardia (VT), sinus arrhythmia (SA), atrial premature contraction (APC), paced beat (P), right bundle branch block (R), left bundle branch block (L), atrial fibrillation (A.Fib.) and atrial flutter (A.Fl.). Training patterns had been sampled at 360 Hz, so we arranged them as 200 samples in the intervals of R–R for all arrhythmias, which are called segments. Training patterns were formed by mixing from the arrhythmias preprocessed by the order given above. The size of the training patterns was 106 segments $\times$ 200 samples. Combined these training patterns were called an original training set. Training patterns were clustered using FCM clustering algorithm before the NN training. Two processes were implemented in this part of study. Firstly, a number of segments in each type of arrhythmia were reduced by using FCM clustering, causing the new training set, whose size was 67 segments $\times$ 200 samples. However, secondly, we proposed an approach by which a number of segments are increased using FCM clustering, so we obtained a second new training set with the size of 212 segments $\times$ 200 samples. Here, processes of reducing and/or increasing segments were made separately for each type of arrhythmia. The number of segments for each type of arrhythmias in new training sets and original training set is given in Table 1. Finally, the obtained two new training sets and original training set were classified using the NN. The backpropagation algorithm was used for training of NN.

The optimum number of hidden nodes was determined via experimentation. The experimental results, show that the optimum number of hidden nodes was 48 with the highest classification accuracy of 99.9% for 10,000 iterations, can be seen in Fig. 3. The designed optimum neural network architecture is shown in Fig. 4. Here nodes of input layer, hidden layer and output layer were used as 200, 48 and 10, respectively. Learning rate was chosen as 1.0 in training via experimentation.

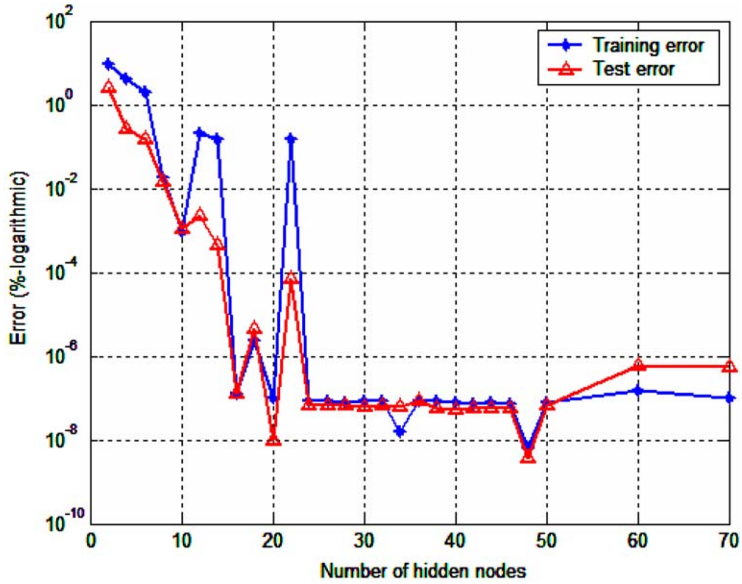

Fig. 3. The experimental results show that optimum number of hidden nodes was 48.

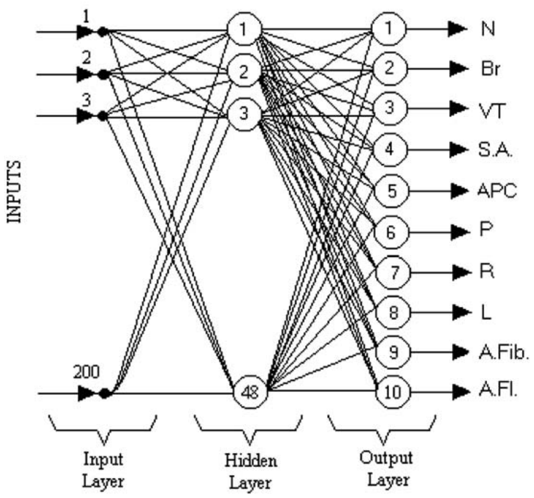

Fig. 4. Optimum neural network architecture used as final classifier.

3.2. Test results

Table 2 describes the training errors and test errors for each arrhythmia obtained with three training sets whose lengths are 106 segments, 67 segments and 212 segments, respectively, after 2000 iterations. In Table 2, MCN and RMC are described as misclassification number and rate of misclassification, respectively. As noted, recognition rates vary between 98% and 99.9% with average recognition accuracy

Table 2
Classification results for each arrhythmia in training

Arrhythmia type of test pattern	Number of rhythm	For 106 sets		For 67 sets		For 212 sets	
		MCN	RMC (%)	MCN	RMC (%)	MCN	RMC (%)
N	15	0	0	0	0	0	0
Br	15	0	0	1	6.6	0	0
VT	6	0	0	0	0	0	0
SA	15	2	13.3	1	6.6	2	13.3
APC	6	1	16.6	1	16.6	1	16.6
P	10	0	0	0	0	0	0
R	10	0	0	0	0	0	0
L	10	0	0	0	0	0	0
A.Fib.	10	0	0	0	0	0	0
A.Fl.	9	0	0	0	0	0	0
Total	106	3	2.83	3	2.83	3	2.83
Average training error (%)		0.28		6.17×10^{-5}		0.15	
Training time (s)		196.9		97.8		197.6	

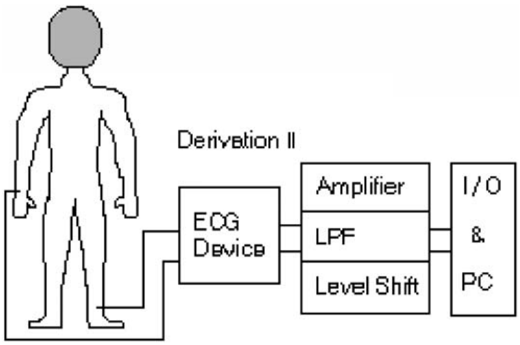

Fig. 5. ECG recording system.

99%. It follows from these results that the recognition rate, obtained in ANN trained with 67 segments and 212 segments, is better than results obtained in ANN trained with 106 segments.

Furthermore, NN was trained with backpropagation algorithm for 10,000 iterations by using three different training sets (feature sets), seperately. The structure trained by using three different training sets was tested using the testing ECG recorded from 92 patients (40 male and 52 female, average age is 39.75 ± 19.06). The ECG records were made in the Cardiology Department, Faculty of Medicine at Selcuk University in Konya, Turkey. Fig. 5 shows the system used in this study for recordings of ECG signals. Derivation II was chosen and Ag–AgCl was used as surface electrodes. Outside ECG device, measured ECG signal was amplified and filtered by low pass filter that was second-order Butterworth and its cut-off frequency was 28 Hz. Then, a dc level was added to the filtered signal in order to have accord with I/O card. Finally, we recorded the ECG signals of patients on PC, sampling frequency was 360 Hz. These recorded patterns were normalized between 0 and 1 and were arranged as 200 samples in the intervals of R–R.

Table 3
Classification results for 92 patients in test

No.	Architecture of NN	Number of segments of training set	Average training error (%)	Average test error (%)
1	200:48:10	106	0.28	0.22
2	200:48:10	67	9.92×10^{-10}	0.19
3	200:48:10	212	0.14	0.44

Table 4
The results from patient test data classified by neural networks trained with 106 segments training set

No.	M/F	Age	Samples	Number of segments	Q_1 N	Q_2 Br	Q_3 T	Q_4 S	Q_5 Apc	Q_6 P	Q_7 R	Q_8 L	Q_9 Afib	Q_{10} Aflt	?	Error %
1	M	23	11,200	56	25	0	0	0	0	0	0	0	3	0	28	0.0556
2	M	23	3600	18	0	0	0	0	0	0	0	0	0	0	18	4.82×10^{-5}
3	M	23	18,600	93	28	0	0	0	0	0	0	0	3	0	62	0.0555
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
90	F	16	13,800	69	0	0	0	0	0	0	0	0	0	0	69	8.3×10^{-4}
91	F	16	14,400	72	0	0	0	0	0	0	0	0	0	0	72	1.63×10^{-4}
92	F	16	17,400	87	0	0	0	0	0	0	80	0	0	0	7	0.6530
				5342	945	492	359	290	68	50	808	55	585	94	1596	0.22

Table 5
The results from patient test data classified by neural networks trained with 67 segments training set

No.	M/F	Age	Samples	Number of segments	Q_1 N	Q_2 Br	Q_3 T	Q_4 S	Q_5 Apc	Q_6 P	Q_7 R	Q_8 L	Q_9 Afib	Q_{10} Aflt	?	Error %
1	M	23	11,200	56	41	0	0	0	0	0	0	0	0	0	15	0.1535
2	M	23	3600	18	0	2	0	0	0	0	0	0	0	0	16	0.9822
3	M	23	18,600	93	67	0	0	0	0	0	0	0	0	0	26	0.3293
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
90	F	16	13,800	69	0	0	0	0	0	0	0	0	0	0	69	1.36×10^{-6}
91	F	16	14,400	72	0	0	0	0	0	0	0	0	0	0	72	0.0635
92	F	16	17,400	87	0	0	0	0	0	0	47	0	0	0	40	1.4100
				5342	1093	316	194	559	68	45	844	54	556	94	1519	0.19

Table 3 shows the classification results for test data recorded from patients. It can be seen in Table 3 that recognition rates are approximately 99% for three architectures trained with three different training sets. Detailed classification results for test data that are recorded from 92 patients can be seen in Tables 4–6. Our test data contains 5342 segments from 92 patients. Table 4, in the bottom row, gives the total number of segments for each arrhythmia for 92 patients. For example, 945 segments are classified

Table 6
The results from patient test data classified by neural network trained with 212 segments training set

No.	M/F	Age	Samples	Number of segments	Q_1 N	Q_2 Br	Q_3 T	Q_4 S	Q_5 Apc	Q_6 P	Q_7 R	Q_8 L	Q_9 Afib	Q_{10} Aflt	?	Error %
1	M	23	11,200	56	23	0	0	3	0	0	0	0	0	0	30	0.1159
2	M	23	3600	18	0	18	0	0	0	0	0	0	0	0	0	1.1126
3	M	23	18,600	93	28	17	0	0	0	0	0	0	0	0	48	0.2783
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
..	..	..	.....	..	..	..	..	..	..	..	..	..	..	..	..	..
90	F	16	13,800	69	0	0	0	0	0	0	0	0	0	0	69	2.23×10^{-5}
91	F	16	14,400	72	0	0	0	0	0	0	0	0	0	0	72	7.26×10^{-4}
92	F	16	17,400	87	0	0	0	0	0	0	0	0	0	0	87	2.37×10^{-6}
				5342	985	897	202	266	247	47	787	56	503	96	1242	0.44

as normal sinus rhythm from our test data. The right-most column gives the classification error for each patient 0.22 is the mean error.

3.3. Calculation of training and test errors

We found training errors given in tables and figures according to

$$\text{Training error}(\%) = \left(\frac{\sum_{i=1}^{k} |t(i) - a(i)|}{mn} \right) \times 100 \tag{8}$$

where $t(i)$ is the desired output, $a(i)$ is output of neural network, k is the number of samples in training data (for example, in this study, for original training set, k is 21,200 samples, 106 segments $\times$ 200 samples), m is the number of segments in training data and n is the number of output of neural network.

We developed an algorithm for evaluation of test results. Desired values of node output of the output layer were logic-1 or logic-0 in the training pattern. Node output were changing between 0 and 1. $y(i)$ is the node output from a node in the output layer. If $y(i) \geqslant 0.5$, we used $h(i) = |1 - y(i)|$ in the error calculation according to Eq. (9). Furthermore, if $y(i) \geqslant 0.5$ and $y(i) > (\textit{other node outputs})$, we interpreted this as arrhythmia for the corresponding node. If $y(i) < 0.5$, we interpreted that there is no arrhythmia, and we used $h(i) = |0 - y(i)|$ according to Eq. (10). If all of node output were $y(i) < 0.5$, then an unknown state occurs. ANN did not classify this test pattern since similar pattern was not taught to the ANN beforehand. The unknown state is represented as "?" in Tables 4–6.

$$y(i) \geqslant 0.5 \rightarrow h = \sum_{i=1}^{k} |1 - y(i)|, \tag{9}$$

$$y(i) < 0.5 \rightarrow h = \sum_{i=1}^{k} |0 - y(i)|, \tag{10}$$

where k is a number of samples in test data. Then, average test errors given in tables and figures are found according to Eq. (11) by using calculated errors (h)

$$\text{Test error}(\%) = \left(\frac{\sum_{j=1}^{n} h(j)}{mn}\right) \times 100, \tag{11}$$

where m is the number of segments in test data and n is the number of outputs of neural network.

4. Results

Three different training sets were used in this study. First, all training sets were used for training and testing using NN and FCNN. The corresponding results for the generalized case in terms of different convergence rates 98.9% for NN and 99.9% for FCNN after 2000 iterations.

Fig. 6 shows variation of training errors in respect of iteration obtained with three different training sets. As noted, with all the three different training sets, the classification rate decreases as the number of iterations increases. It is noted in the results above that the FCNN converges to a determined error goal faster than NN. Moreover, in most cases, compared to other NNs a more accurate recognition rate (99.9%) is obtained with the FCNN. These results are significantly better than those previously reported; Özbay et al. found 97.8% classification rate [3], and Osowski et al. found 96% classification rate [2].

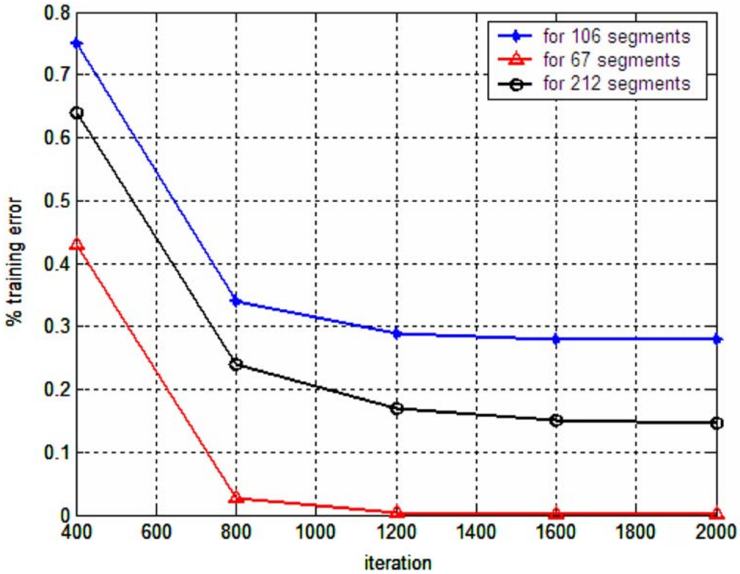

Fig. 6. Classification results obtained with three different training patterns.

5. Conclusion

In this paper, the new FCNN has been developed and presented to classify electrocardiography signals by using three different training sets. For the training and testing, these training sets (or feature sets) are used by FCNN.

A comparative assessment of the performance of FCNN with MLP NN show that more reliable results are obtained with the FCNN for the classification of ECG signals. MLP NNs are still able to generalize with good recognition accuracy. However, they take longer to train. The aim in developing FCNN was to achieve more optimum results with relatively few signal features. It has been demonstrated that the training time of the FCNN was 60% of the time required by the MLP NN. We hope that the performance of the method will be better, if the number of the beats is increased for the training. Furthermore, this technique, which incorporates the techniques of fuzzy rule-based models, backpropagation learning and the FCM clustering methods and combines their advantages, can be said to be more capable of recognizing other biological signals than conventional BPNN [1–7,9,12].

References

[1] R. Acharya, P.S. Bhat, S.S. Iyengar, A. Roo, S. Dua, Classification of heart rate data using artificial neural network and fuzzy equivalence relation, J. Pattern Recognition Soc. (2002).

[2] S. Osowski, T.H. Linh, ECG beat recognition using fuzzy hybrid neural network, IEEE Trans. Biomed. Eng. 48 (11) (2001) 1265–1271.

[3] Y. Ozbay, B. Karlik, A recognition of ECG arrhythmias using artificial neural network, Proceedings of the 23rd Annual Conference, IEEE/EMBS, Istanbul, Turkey, 2001.

[4] Y. Ozbay, Fast recognition of ECG arrhythmias, Ph.D. Thesis, Institute of Natural and Applied Science, Selcuk University, 1999.

[5] S.Y. Foo, G. Stuart, B. Harvey, A. Meyer-Baese, Neural network-based ECG pattern recognition, Eng. Appl. Artif. Intell. 15 (2002) 253–260.

[6] V. Pilla, H.S. Lopes, Evolutionary training of a neuro-fuzzy network for detection of P wave of the ECG, Proceedings of the Third International Conference on Computational Intelligence and Multimedia Applications, New Delhi, India, 1999, pp. 102–106.

[7] M. Engin, S. Demirağ, Fuzzy-hybrid neural network based ECG beat recognition using three different types of feature set, Cardiovasc. Eng. Int. J. 3 (2) (2003) 71–80.

[8] S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan, New York, 1994.

[9] B. Karlık, M.O. Tokhi, M. Alcı, A fuzzy clustering neural network architecture for multifunction upper-limb prosthesis, IEEE Trans. Biomed. Eng. 50 (11) (2003) 1255–1261.

[10] J.S.R. Jang, C.T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall, Englewood Cliffs, NJ, USA, 1997.

[11] T.W. Liao, A.K. Celmins, R.J. Hammell II, A fuzzy c-means variant for the generation of fuzzy term sets, Fuzzy Sets Syst. 135 (2003) 241–257.

[12] H. Seker, D.H. Evans, N. Aydin, E. Yazgan, Compensatory fuzzy neural network-based intelligent detection of abnormal neonatal cerebral doppler ultrasound waveforms, IEEE Trans. Inform. Technol. Biomed. 5 (3) (2001) 187–194.

[13] Y. Zhang, A. Kandel, Compensatory neuro-fuzzy systems with fast learning algorithms, IEEE Trans. Neural Networks 9 (1) (1998) 83–105.

[14] J. Fan, W. Zhen, W. Xie, Supervised fuzzy c-means clustering algorithm, Pattern Recognition Lett. 24 (2003) 1607–1612.

[15] R. Li, M. Mukaidono, I.B. Turksen, A fuzzy neural network for pattern classification and feature selection, Fuzzy Sets Syst. 130 (2002) 101–108.

[16] R. Pektatli, Y. Ozbay, M. Ceylan, B. Karlik, Classification of ECG signals using fuzzy clustering neural networks (FCNN), Proceedings of the International XII, TAINN'03, vol. 1(1), Çanakkale, Turkey, 2003, pp. 105–108.

[17] G. Castellano, A.M. Fanelli, A self-organizing neural fuzzy inference network, Proceedings of the IEEE International Joint Conference on Neural Networks, vol. 5, Italy, 2000, pp. 14–19.

[18] G. Castellano, A.M. Fanelli, Fuzzy inference and rule extraction using a neural network, Neural Network World J. 3 (2000) 361–371.
[19] D. Dazzi, F. Taddei, A. Gavarini, E. Uggeri, R. Negra, A. Pezzarossa, The control of blood glucose in the critical diabetic patient: A neuro-fuzzy method, J. Diabetes Complications 15 (2001) 80–87.
[20] R.K. De, J. Basak, S.K. Pal, Unsupervised feature extraction using neuro-fuzzy approach, Fuzzy Sets Syst. 126 (2002) 277–291.
[21] P. Meesad, G.G. Yen, Pattern classification by a neuro fuzzy network application to vibration monitoring, ISA Trans. 39 (2000) 293–308.
[22] N.R. Pal, J.C. Bezdek, E.C. Tsao, Generalized clustering networks and Kohonen's self-organizing scheme, IEEE Trans. Neural Networks 4 (4) (1993) 549–557.
[23] W. Pedrycz, Conditional fuzzy clustering in the design of radial basis function neural networks, IEEE Trans. Neural Networks 9 (4) (1998) 601–612.
[24] M. Haseyama, H. Kitajima, An ARMA order selection method with fuzzy reasoning, Signal Process. 81 (2001) 1331–1335.
[25] M. Ronen, Y. Shabtai, H. Guterman, Rapid process modelling-model building methodology combining unsupervised fuzzy clustering and supervised neural networks, Comput. Eng. 22 (1998) 1005–1008.
[26] Y.M. Sebzalli, X.Z. Wang, Knowledge discovery from process operational data using PCA and fuzzy clustering, Eng. Appl. Artif. Intell. 14 (2001) 607–616.
[27] A. Shaout, J. Scharbonea, Fuzzy logic based modification system for the learning rate in back-propagation, Comput. Electr. Eng. 26 (2000) 125–139.
[28] S. Stoeva, A. Nikov, A fuzzy back-propagation algorithm, Fuzzy Sets Syst. 112 (2000) 27–39.
[29] M. Yang, H. Liu, Fuzzy least-squares algorithms for interactive fuzzy linear regression models, Fuzzy Sets Syst. 135 (2003) 305–316.
[30] L.O. Hall, A.M. Bensaid, L.P. Clarke, R.P. Velthuizen, M.S. Silbiger, J.C. Bezdek, A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images on the brain, IEEE Trans. Neural Networks 3 (5) (1992) 672–682.
[31] J. Leski, E. Czogala, A new artificial neural network based fuzzy inference system with moving consequents in if–then rules and selected applications, Fuzzy Sets Syst. 108 (1999) 289–297.