Expert Systems with Applications 29 (2005) 506–514

Expert Systems with Applications

www.elsevier.com/locate/eswa

Recurrent neural networks employing Lyapunov exponents for EEG signals classification

Nihal Fatma Güler[a], Elif Derya Übeyli[b], İnan Güler[a,*]

[a]Department of Electronics and Computer Education, Faculty of Technical Education, Gazi University, Teknikokullar, 06500 Ankara, Turkey
[b]Department of Electrical and Electronics Engineering, Faculty of Engineering, TOBB Ekonomi ve Teknoloji Üniversitesi, Söğütözü, 06560 Ankara, Turkey

Abstract

There are a number of different quantitative models that can be used in a medical diagnostic decision support system including parametric methods, non-parametric methods and several neural network models. Unfortunately, there is no theory available to guide model selection. The aim of this study is to evaluate the diagnostic accuracy of the recurrent neural networks (RNNs) employing Lyapunov exponents trained with Levenberg–Marquardt algorithm on the electroencephalogram (EEG) signals. An approach based on the consideration that the EEG signals are chaotic signals was used in developing a reliable classification method for electroencephalographic changes. This consideration was tested successfully using the non-linear dynamics tools, like the computation of Lyapunov exponents. We explored the ability of designed and trained Elman RNNs, combined with the Lyapunov exponents, to discriminate the EEG signals (EEG signals recorded from healthy volunteers with eyes open, epilepsy patients in the epileptogenic zone during a seizure-free interval, and epilepsy patients during epileptic seizures). The RNNs achieved accuracy rates which were higher than that of the feedforward neural network models. The obtained results demonstrated that the proposed RNNs employing the Lyapunov exponents can be useful in analyzing long-term EEG signals for early detection of the electroencephalographic changes.

Keywords: Recurrent neural networks; Levenberg–Marquardt algorithm; Electroencephalogram (EEG) signals; Chaotic signal; Lyapunov exponents

1. Introduction

The electroencephalogram (EEG) signals reflect the electrical activity of the brain. The study of the brain electrical activity, through the electroencephalographic records, is one of the most important tools for the diagnosis of neurological diseases (Agarwal, Gotman, Flanagan, & Rosenblatt, 1998; Dingle, Jones, Carroll, & Fright, 1993). Large amounts of data are generated by EEG monitoring systems for electroencephalographic changes, and their complete visual analysis is not routinely possible. Computers have long been proposed to solve this problem and thus, automated systems to recognize electroencephalographic changes have been under study for several years. There is a strong demand for the development of such automated devices, due to the increased use of prolonged and long-term video EEG recordings for proper evaluation and treatment of neurological diseases and prevention of the possibility of the analyst missing (or misreading) information (Agarwal et al., 1998; Dingle et al., 1993; Finnerup, Fuglsang-Frederiksen, Rossel, & Jennum, 1999).

The idea of using artificial neural networks (ANNs) for pattern classification purposes has encountered, for a long time, the favour of many researchers (Güler & Übeyli, 2003; Miller, Blott, & Hames, 1992; Übeyli & Güler, 2004). ANNs are massively parallel, highly connected structures consisting of a number of simple, non-linear processing elements; because of their massively parallel structure, they can perform computations at a very high rate if implemented on a dedicated hardware; because of their adaptive nature, they can learn the characteristics of input signals and adapt to changes in the data; because of their non-linear nature they can perform functional approximation and signal filtering operations which are beyond optimal linear techniques. Feedforward neural networks are a basic type of neural networks capable of approximating generic classes

* Corresponding author. Tel.: +90 312 212 3976; fax: +90 312 212 0059.
E-mail address: iguler@gazi.edu.tr (İ. Güler).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.04.011

of functions, including continuous and integrable ones. One of the most frequently used feedforward neural network for pattern classification is the multilayer perceptron neural network (MLPNN) which is trained to produce a spatial output pattern in response to an input spatial pattern (Chaudhuri & Bhattacharya, 2000; Haykin, 1994). The mapping performed is static, therefore, the network is inherently not suitable for processing temporal patterns. Attempts have been made to use the MLPNN to classify temporal patterns by transforming the temporal domain into a spatial domain. An alternate neural network approach is to use recurrent neural networks (RNNs) which have memory to encode past history. Several forms of RNNs have been proposed and they may be classified as partially recurrent or fully recurrent networks (Gupta & McAvoy, 2000; Gupta, McAvoy, & Phegley, 2000; Saad, Prokhorov, & Wunsch, 1998). In partially recurrent networks, partial recurrence is created by feeding back delayed hidden unit outputs or the outputs of the network as additional input units. We used the partially recurrent networks, whose connections are mainly feedforward, but they include a carefully chosen set of feedback connections. One example of such a network is an Elman RNN which in principle is set up as a regular feedforward network (Elman, 1990).

We evaluate the classification capabilities of the Elman RNNs trained with Levenberg–Marquardt algorithm on the EEG signals. Recent advances in the field of neural networks have made them attractive for analyzing signals. The application of neural networks has opened a new area for solving problems not resolvable by other signal processing techniques (Güler & Übeyli, 2003; Miller et al., 1992; Petrosian, Prokhorov, Homan, Dasheiff, & Wunsch, 2000; Petrosian, Prokhorov, Lajara-Nanson, & Schiffer, 2001; Shieh, Chou, Huang, & Kao, 2004; Übeyli & Güler, 2004). Up to now, there is no study in the literature relating to the assessment of accuracy of RNN employing Lyapunov exponents in analysis of the EEG signals. In this study, experimental results on the RNNs predictions for discrimination of the EEG signals (EEG signals recorded from healthy volunteers with eyes open, epilepsy patients in the epileptogenic zone during a seizure-free interval, and epilepsy patients during epileptic seizures) were presented. Dynamical measures especially Lyapunov exponents can serve as clinically useful parameters and contain a significant amount of information about the signal (Silipo, Deco, Vergassola, & Bartsch, 1998; Übeyli & Güler, 2004). Therefore, the Lyapunov exponents become a natural complement to our applications of the RNNs. The Lyapunov exponents of each EEG segment were computed, which were then input into the RNNs for training and testing purposes. The RNNs were chosen because they can implement extremely nonlinear decision boundaries and possess memory of the state, which is crucial for the considered task (Gupta & McAvoy, 2000; Gupta et al., 2000; Saad et al., 1998). We were able to achieve significant improvement in accuracy by using the RNNs compared to the feedforward neural network models.

The outline of this study is as follows. In Section 2, we briefly describe the sets of the EEG signals used in our study. In Section 3, we explain the Lyapunov exponents of the EEG signals in order to extract features characterizing the behavior of the signal under study. In Section 4, we present description of neural network models including RNNs used in this study. We also explain the Levenberg–Marquardt algorithm used for training the RNNs. In Section 5, we present the results of application of the RNNs to the EEG signals. Finally, in Section 6 we conclude the study.

2. Data selection

We used the data described in Andrzejak et al. (2001), which is publicly available. In this section, we restrict ourselves to only a short description and refer to Andrzejak et al. (2001) for further details. The complete dataset consists of five sets (denoted A–E), each containing 100 single-channel EEG signals of 23.6 s. Each signal has been selected after visual inspection for artifacts and has passed a weak stationarity criterion. Sets A and B have been taken from surface EEG recordings of five healthy volunteers with eyes open and closed, respectively. Signals in two sets have been measured in seizure-free intervals from five patients in the epileptogenic zone (D) and from the hippocampal formation of the opposite hemisphere of the brain (C). Set E contains seizure activity, selected from all recording sites exhibiting ictal activity. Sets A and B have been recorded extracranially, whereas sets C, D, and E have been recorded intracranially. In our applications, performance degraded for a more detailed classification which further dissociated between sets A (healthy volunteer, eyes open) and B (healthy volunteer, eyes closed), and sets D (epileptogenic zone) and C (hippocampal formation of opposite hemisphere). Therefore, in the present study we classified three dataset (A, D, E) of the complete dataset.

3. Lyapunov exponents

Lyapunov exponents are a quantitative measure for distinguishing among the various types of orbits based upon their sensitive dependence on the initial conditions, and are used to determine the stability of any steady-state behavior, including chaotic solutions. The reason why chaotic systems show aperiodic dynamics is that phase space trajectories that have nearly identical initial states will separate from each other at an exponentially increasing rate captured by the so-called Lyapunov exponent (Abarbanel, Brown, & Kennel, 1991; Haykin & Li, 1995). This is defined as follows. Consider two (usually the nearest) neighboring points in phase space at time 0 and at time t, distances of the points in the ith direction being $\|\delta x_i(0)\|$ and $\|\delta x_i(t)\|$, respectively.

The Lyapunov exponent is then defined by the average growth rate λ_i of the initial distance

$$\frac{\|\delta x_i(t)\|}{\|\delta x_i(0)\|} = 2^{\lambda_i t}(t \to \infty) \quad \text{or} \quad \lambda_i = \lim_{t\to\infty} \frac{1}{t} \log_2 \frac{\|\delta x_i(t)\|}{\|\delta x_i(0)\|}. \tag{1}$$

The existence of a positive Lyapunov exponent indicates chaos (Abarbanel et al., 1991; Haykin & Li, 1995). This shows that any neighboring points with infinitesimal differences at the initial state abruptly separate from each other in the ith direction. In other words, even if the initial states are close, the final states are much different. This phenomenon is sometimes called sensitive dependence on initial conditions. Numerous methods for calculating the Lyapunov exponents have been developed during the past decade (Abarbanel, Brown, Sidorowich, & Tsimring, 1993). Generally, the Lyapunov exponents can be estimated either from the equations of motion of the dynamic system (if it is known) (Benettin, Galgani, Giorgilli, & Strelcyn, 1980), or from the observed time series (Packard, Crutchfield, Farmer, & Shaw, 1980). The latter is what is of interest due to its direct relation to the work in this paper. The idea is based on the well-known technique of state space reconstruction with delay coordinates to build a system with Lyapunov exponents identical to that of the original system from which our measurements have been observed. Generally, Lyapunov exponents can be extracted from observed signals in two different ways. The first is based on the idea of following the time-evolution of nearby points in the state space (Wolf, Swift, Swinney, & Vastano, 1985). This method provides an estimation of the largest Lyapunov exponent only. The second method is based on the estimation of local Jacobi matrices (Sano & Sawada, 1985) and is capable of estimating all the Lyapunov exponents. Vectors of all the Lyapunov exponents for particular systems are often called their Lyapunov spectra.

4. Description of neural network models

4.1. Recurrent neural networks

ANNs can be trained to recognize patterns and the non-linear models developed during training allow neural networks to generalize their conclusions and to make application to patterns not previously encountered. A particular architecture of the neural models is the multi-layered architecture. Multilayered networks can be classified as feedforward and feedback networks, with respect to the direction of their connections (Chaudhuri & Bhattacharya, 2000; Haykin, 1994). RNNs can perform highly non-linear dynamic mappings and thus have temporally extended applications, whereas multilayer feedforward networks are confined to performing static mappings (Elman, 1990; Gupta & McAvoy, 2000; Gupta et al., 2000; Saad et al., 1998). RNNs have been used in a number of interesting applications including associative memories, spatiotemporal pattern classification, control, optimization, forecasting and generalization of pattern sequences (Petrosian et al., 2000, 2001; Shieh et al., 2004).

Fully recurrent networks use unconstrained fully interconnected architectures and learning algorithms that can deal with time-varying input and/or output in non-trivial ways. In spite of several modifications of learning algorithms to reduce the computational expense, fully recurrent networks are still complicated when dealing with complex problems. Therefore, we used the partially recurrent networks, whose connections are mainly feedforward, but they include a carefully chosen set of feedback connections. The recurrence allows the network to remember cues from the past without complicating the learning excessively. The structure proposed by Elman (1990) is an illustration of this kind of architecture. We used Elman RNN in our applications and therefore in the following the Elman RNN is presented.

An Elman RNN is a network which in principle is set up as a regular feedforward network. This means that all neurons in one layer are connected with all neurons in the next layer. An exception is the so-called context layer which is a special case of a hidden layer. Fig. 1 shows the architecture of an Elman RNN. The neurons in the context layer (context neurons) hold a copy of the output of the hidden neurons. The output of each hidden neuron is copied into a specific neuron in the context layer. The value of the context neuron is used as an extra input signal for all the neurons in the hidden layer one time step later. Therefore, the Elman network has an explicit memory of one time lag (Elman, 1990).

Similar to a regular feedforward neural network, the strength of all connections between neurons are indicated with a weight. Initially, all weight values are chosen randomly and are optimized during the stage of training.

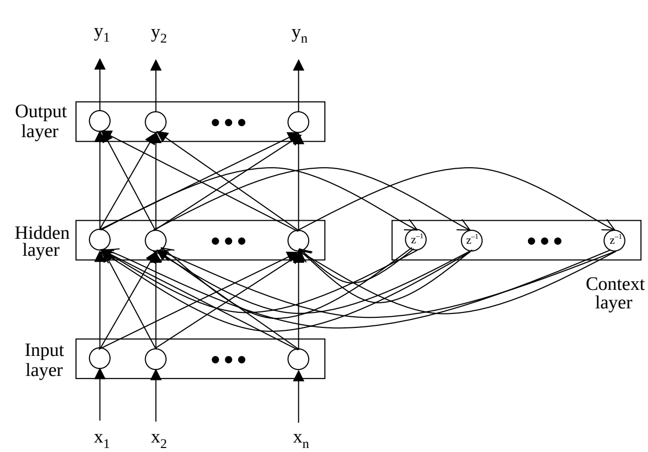

Fig. 1. A schematic representation of an Elman recurrent neural network. z^{-1} represents a one time step delay unit.

In an Elman network, the weights from the hidden layer to the context layer are set to one and are fixed because the values of the context neurons have to be copied exactly. Furthermore, the initial output weights of the context neurons are equal to half the output range of the other neurons in the network. The Elman network can be trained with gradient descent backpropagation and optimization methods, similar to regular feedforward neural networks (Pineda, 1987). The backpropagation has some problems for many applications. The algorithm is not guaranteed to find the global minimum of the error function since gradient descent may get stuck in local minima, where it may remain indefinitely. In addition to this, long training sessions are often required in order to find an acceptable weight solution because of the well-known difficulties inherent in gradient descent optimization (Chaudhuri & Bhattacharya, 2000; Haykin, 1994). Therefore, a lot of variations to improve the convergence of the backpropagation were proposed. Optimization methods such as second-order methods (conjugate gradient, quasi-Newton, Levenberg–Marquardt) have also been used for neural networks training in recent years. The Levenberg–Marquardt algorithm combines the best features of the Gauss–Newton technique and the steepest-descent algorithm, but avoids many of their limitations. In particular, it generally does not suffer from the problem of slow convergence (Battiti, 1992; Hagan & Menhaj, 1994) and can yield a good cost function compared with the other training algorithms.

4.2. Levenberg–Marquardt algorithm

Essentially, the Levenberg–Marquardt algorithm is a least-squares estimation algorithm based on the maximum neighborhood idea. Let $E(\mathbf{w})$ be an objective error function made up of m individual error terms $e_i^2(\mathbf{w})$ as follows

$$E(\mathbf{w}) = \sum_{i=1}^{m} e_i^2(\mathbf{w}) = \|f(\mathbf{w})\|^2, \tag{2}$$

where $e_i^2(\mathbf{w}) = (\mathbf{y}_{di} - \mathbf{y}_i)^2$ and $\mathbf{y}_{di}$ is the desired value of output neuron i, $\mathbf{y}_i$ is the actual output of that neuron.

It is assumed that function $f(\cdot)$ and its Jacobian J are known at point $\mathbf{w}$. The aim of the Levenberg–Marquardt algorithm is to compute the weight vector $\mathbf{w}$ such that $E(\mathbf{w})$ is minimum. Using the Levenberg–Marquardt algorithm, a new weight vector $\mathbf{w}_{k+1}$ can be obtained from the previous weight vector $\mathbf{w}_k$ as follows

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \delta\mathbf{w}_k, \tag{3}$$

where $\delta\mathbf{w}_k$ is defined as

$$\delta\mathbf{w}_k = -(J_k^{\mathrm{T}} f(\mathbf{w}_k))(J_k^{\mathrm{T}} J_k + \lambda\mathbf{I})^{-1}. \tag{4}$$

In Eq. (4), J_k is the Jacobian of f evaluated at $\mathbf{w}_k$, λ is the Marquardt parameter, $\mathbf{I}$ is the identity matrix (Battiti, 1992; Hagan & Menhaj, 1994). The Levenberg–Marquardt algorithm may be summarized as follows:

(i) compute $E(\mathbf{w}_k)$,
(ii) start with a small value of λ ($\lambda = 0.01$),
(iii) solve Eq. (4) for $\delta\mathbf{w}_k$ and compute $E(\mathbf{w}_k + \delta\mathbf{w}_k)$,
(iv) if $E(\mathbf{w}_k + \delta\mathbf{w}_k) \geq E(\mathbf{w}_k)$, increase λ by a factor of 10 and go to (iii),
(v) if $E(\mathbf{w}_k + \delta\mathbf{w}_k) < E(\mathbf{w}_k)$, decrease λ by a factor of 10, update $\mathbf{w}_k : \mathbf{w}_k \leftarrow \mathbf{w}_k + \delta\mathbf{w}_k$ and go to (iii).

The Elman RNNs proposed for classification of the EEG signals were implemented by using the MATLAB software package (MATLAB version 7.0 with neural networks toolbox).

5. Experimental results

5.1. Feature extraction by computing Lyapunov exponents

A rectangular window, which was formed by 256 discrete data, was selected so that it contained a single EEG segment. The waveforms of three different EEG segments classified in the present study are shown in Fig. 2(a)–(c). In the present study, the technique used in the computation of Lyapunov exponents was related with the Jacobi-based algorithms. The Lyapunov exponents of the typical segment of EEG signals (set A—EEG signals recorded from healthy volunteers with eyes open, set D—EEG signals recorded from epilepsy patients in the epileptogenic zone during a seizure-free interval, and set E—EEG signals recorded from epilepsy patients during epileptic seizures) are given in Fig. 3(a)–(c), respectively. It can be noted that the Lyapunov exponents of the three types of EEG signals are different from each other and therefore, they can be used in discrimination of the EEG signals. From Fig. 3(a) one can see that all the Lyapunov exponents are positive, which confirm the chaotic nature of the EEG signals recorded from healthy volunteers with eyes open. As it is seen from Fig. 3(b) and (c) there are positive Lyapunov exponents, which confirm the chaotic nature of the EEG signals recorded from epilepsy patients in the epileptogenic zone during a seizure-free interval, and epilepsy patients during epileptic seizures. The Lyapunov exponents were computed using the MATLAB software package.

Feature selection provides a means for choosing the features which are best for classification, based on various criteria. The feature selection process performed on a set of predetermined features. Features are selected based on either (1) best representation of a given class of signals, or (2) best distinction between classes. Therefore, feature selection plays an important role in classifying systems such as neural networks. For each EEG segment, 128 Lyapunov exponents were computed. High-dimension of feature

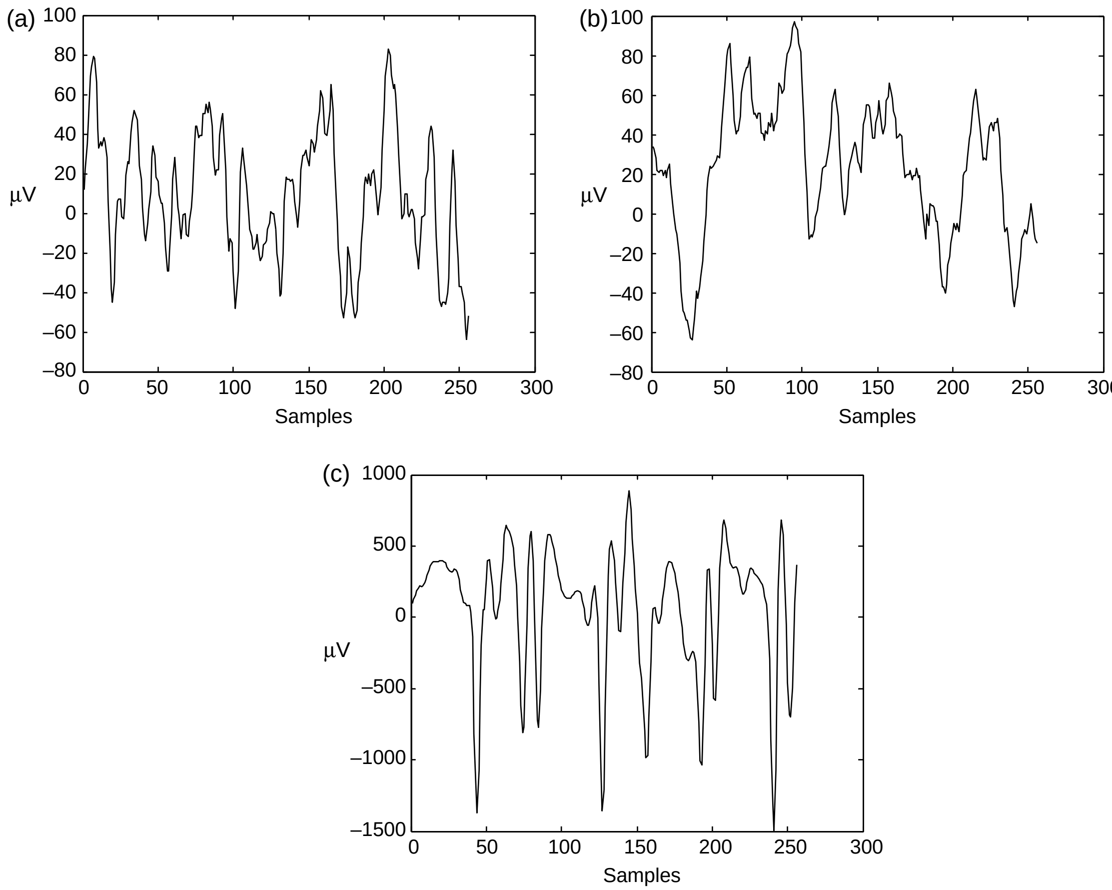

Fig. 2. Waveforms of the EEG segments (a) set A (EEG signals recorded from healthy volunteers with eyes open), (b) set D (EEG signals recorded from epilepsy patients in the epileptogenic zone during a seizure-free interval), (c) set E (EEG signals recorded from epilepsy patients during epileptic seizures).

vectors increased computational complexity and the neural networks trained on these feature vectors produced lower accuracy. Therefore, we used the following statistical features to reduce the dimensionality of the Lyapunov exponents:

1. Mean of the absolute values of the Lyapunov exponents in each segment.
2. Maximum of the absolute values of the Lyapunov exponents in each segment.
3. Average power of the Lyapunov exponents in each segment.
4. Standard deviation of the Lyapunov exponents in each segment.

Features 1–4 represent the Lyapunov exponents distribution of the EEG signals. These features (four Lyapunov exponents) computed for each segment were used for classification of the EEG signals. We input 128 Lyapunov exponents (for each EEG segment) and the selected Lyapunov exponents (four Lyapunov exponents for each EEG segment) into RNNs for training and testing. The results showed an apparent improvement of network behavior when trained on four selected Lyapunov exponents as opposed to 128 Lyapunov exponents.

5.2. Application of recurrent neural networks to EEG signals

The RNN architecture used for classification of the EEG signals is shown in Fig. 1. The key design decisions for the neural networks used in classification are the architecture and the training process. ANN architectures are derived by trial and error and the complexity of the neural network is characterized by the number of hidden layers. There is no general rule for selection of appropriate number of hidden layers. A neural network with a small number of neurons may not be sufficiently powerful to model a complex function. On the other hand, a neural network with too many neurons may lead to overfitting the training sets and lose its ability to generalize which is the main desired characteristic of a neural network. The most popular approach to finding the optimal number of hidden layers is by trial and error. We tested different network architectures and our architecture studies confirmed that for the EEG signals, networks with one hidden layer consisting of 15 recurrent neurons results in higher classification accuracy. Apart from one hidden layered network architecture, although we experimented extensively with larger networks with two hidden layers (the largest RNN used had the $4\times25r\times15r\times3$ architecture,

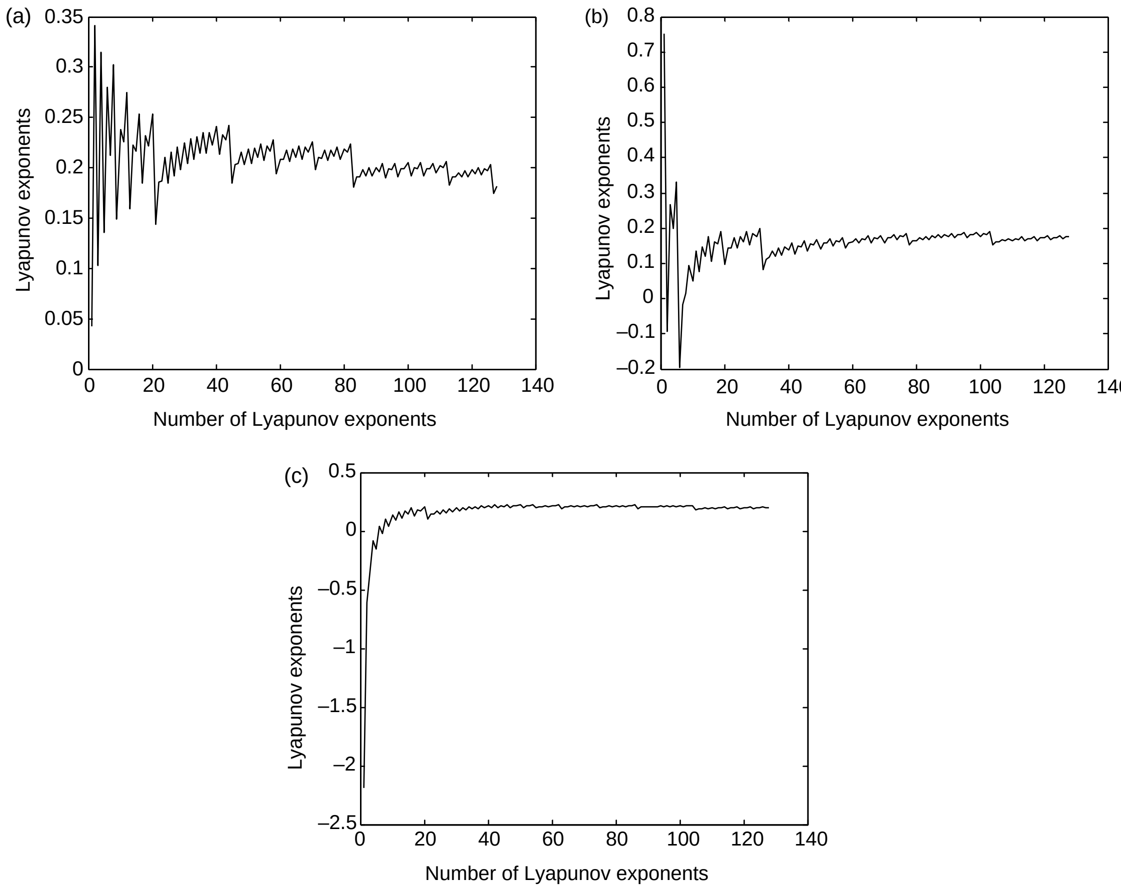

Fig. 3. Lyapunov exponents of the EEG segments (a) set A (EEG signals recorded from healthy volunteers with eyes open), (b) set D (EEG signals recorded from epilepsy patients in the epileptogenic zone during a seizure-free interval), (c) set E (EEG signals recorded from epilepsy patients during epileptic seizures).

where *r* stands for a recurrent neuron), we did not observe a noticeable improvement in training and testing performance. The number of output was 3 and samples with target outputs healthy segments, seizure free epileptogenic zone segments, epileptic seizure segments were given the binary target values of (0,0,1), (0,1,0), (1,0,0), respectively. In this application, in the hidden layers and the output layer sigmoidal function was used as activation function. The sigmoidal function with the range between zero and one introduces two important properties. First, the sigmoid is non-linear, allowing the network to perform complex mappings of input to output vector spaces, and secondly it is continuous and differentiable, which allows the gradient of the error to be used in updating the weights.

The adequate functioning of neural networks depends on the sizes of the training set and test set. In the RNNs, the 2400 vectors (800 vectors from each class) were used for training and the 2400 vectors (800 vectors from each class) were used for testing. A practical way to find a point of better generalization is to use a small percentage (around 20%) of the training set for cross-validation. For obtaining a better network generalization 480 vectors (160 vectors from each class) of training set, which were selected randomly, were used as cross-validation set. Beside this, in order to enhance the generalization capability of the RNN, the training and the test sets were formed by data obtained from different subjects. For all of the segments, waveform variations were observed among the vectors belonging to the same class.

For the three diagnostic classes (sets A, D, and E) training and test sets were formed by 4800 vectors (1600 vectors from each class) of four dimensions (selected Lyapunov exponents). The training holds the key to an accurate solution, so the criterion to stop training must be very well described. When the network is trained too much, the network memorizes the training patterns and does not generalize well. Cross-validation is a highly recommended criterion for stopping the training of a network. When the error in the cross-validation increases, the training should be stopped because the point of best generalization has been reached. Training of the RNN was done in 300 epochs since the cross-validation errors began to rise at 300 epochs (Fig. 4). From Fig. 4, one can see that the values of mean square error (MSE) converged to small constants approximately zero in 300 epochs and therefore training of the RNN was determined to be successful. However, the MLPNN

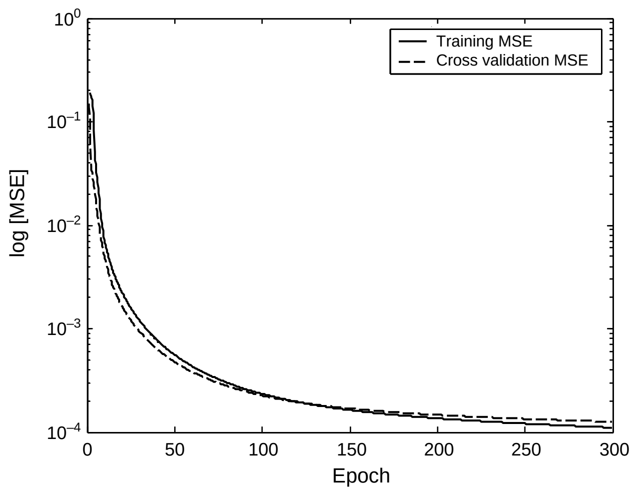

Fig. 4. Training and cross-validation MSE curves of the RNN.

trained with the Levenberg–Marquardt algorithm had a slow convergence and MSE converged to a small constant of approximately zero in 1500 epochs. Thus, the convergence rate of the RNN presented in this study was found to be higher than that of the MLPNN.

In classification, the aim is to assign the input patterns to one of several classes, usually represented by outputs restricted to lie in the range from 0 to 1, so that they represent the probability of class membership. While the classification is carried out, a specific pattern is assigned to a specific class according to the characteristic features selected for it. In this study, there were three classes: set A (EEG signals recorded from healthy volunteers with eyes open), set D (EEG signals recorded from epilepsy patients in the epileptogenic zone during a seizure-free interval), and set E (EEG signals recorded from epilepsy patients during epileptic seizures). Classification results of the RNN were displayed by a confusion matrix. The confusion matrix showing the classification results of the RNN is given below.

Confusion matrix

Output/desired	Result (healthy segment)	Result (seizure free epileptogenic zone segment)	Result (epileptic seizure segment)
Result (healthy segment)	779	11	12
Result (seizure free epileptogenic zone segment)	13	775	19
Result (epileptic seizure segment)	8	14	769

According to the confusion matrix, 13 healthy segments were classified incorrectly by the RNN as seizure free epileptogenic zone segments, eight healthy segments were classified as epileptic seizure segments, 11 seizure free epileptogenic zone segments were classified as healthy segments, 14 seizure free epileptogenic zone segments were classified as epileptic seizure segments, 12 epileptic seizure segments were classified as healthy segments and 19 epileptic seizure segments were classified as seizure free epileptogenic zone segments.

The test performance of the RNN was determined by the computation of the following statistical parameters:

Specificity: Number of correct classified healthy segments/number of total healthy segments;
Sensitivity (seizure free epileptogenic zone segments): Number of correct classified seizure free epileptogenic zone segments/number of total seizure free epileptogenic zone segments;
Sensitivity (epileptic seizure segments): Number of correct classified epileptic seizure segments/number of total epileptic seizure segments;
Total classification accuracy: Number of correct classified segments/number of total segments.

The values of these statistical parameters are given in Table 1. As it is seen from Table 1, the RNN classified healthy segments, seizure free epileptogenic zone segments and epileptic seizure segments with the accuracy of 97.38, 96.88 and 96.13%, respectively. The healthy segments, seizure free epileptogenic zone segments and epileptic seizure segments were classified with the accuracy of 96.79%. The correct classification rates of the MLPNN were 92.25% for healthy segments, 91.13% for seizure free epileptogenic zone segments, and 90.63% for epileptic seizure segments. Thus, the accuracy rates of the RNN presented for this application were found to be higher than that of the MLPNN.

The difference between the output of the network and the desired response is referred to as the error and can be measured in different ways. In this study, MSE, mean absolute error (MAE), minimum absolute error, maximum absolute error, and correlation coefficient (r) were used for measuring error of the proposed RNN. The sizes of MSE and MAE can be used to determine how well the network output fits the desired output, but they may not reflect whether the two sets of data move in the same direction. The correlation coefficient solves this problem. The correlation coefficient is limited within the range $[-1,1]$. When $r=1$ there is a perfect positive linear correlation between network output and desired output, which means that they vary by the same amount. When $r=-1$, there is a perfectly linear negative correlation between network output

Table 1
The values of statistical parameters

Statistical parameters	Values (%)
Specificity	97.38
Sensitivity (seizure free epileptogenic zone segments)	96.88
Sensitivity (epileptic seizure segments)	96.13
Total classification accuracy	96.79

Table 2
The values of performance evaluation parameters during test process of the RNN

Performance	Result (healthy segment)	Result (seizure free epileptogenic zone segment)	Result (epileptic seizure segment)
MSE	0.000195	0.000254	0.000317
MAE	0.009762	0.011258	0.012540
Minimum absolute error	0.002371	0.003253	0.003564
Maximum absolute error	0.083523	0.096725	0.101572
r	0.986211	0.975106	0.976488

and desired output, that means they vary in opposite ways. When $r=0$ there is no correlation between network output and desired output. Intermediate values describe partial correlations. The values of performance evaluation parameters during test process of the presented RNN for healthy segments, seizure free epileptogenic zone segments and epileptic seizure segments are given in Table 2. The classification results, the values of statistical parameters, and performance evaluation parameters indicated that the RNN trained with the Levenberg–Marquardt algorithm had considerable success in discriminating the EEG signals.

6. Conclusion

This study presents an attempt to develop a reliable computerized algorithm, which can detect electroencephalographic changes. We applied the RNNs employing Lyapunov exponents to the EEG signals classification problem. Assuming the Lyapunov exponents of the EEG segments as inputs of the RNNs, the performance of the RNNs was demonstrated in a particular clinical situation. The Levenberg–Marquardt algorithm was used for training the RNNs so that it generally does not suffer from the problem of slow convergence and can yield a good cost function compared with the other training algorithms. The classification results, the values of statistical parameters, and performance evaluation parameters were used for evaluating performances of the classifiers. We have presented encouraging results in detecting electroencephalographic changes obtained with the RNNs and Lyapunov exponents. Our results led to three major conclusions: (1) The overall results of the RNNs were better when they were trained on four selected Lyapunov exponents (for each EEG segment) rather than on 128 Lyapunov exponents (for each EEG segment); (2) Our architecture studies confirmed that for the EEG signals, the RNN with one hidden layer has better generalization properties and results in higher classification accuracy; (3) The accuracy rates achieved by the RNNs were found to be higher than that of the MLPNN.

Acknowledgements

This study has been supported by the State Planning Organization of Turkey (Project no: 2003K 120470-20, Project name: Biomedical signal acquisition, processing and imaging).

References

Abarbanel, H. D. I., Brown, R., & Kennel, M. B. (1991). Lyapunov exponents in chaotic systems: Their importance and their evaluation using observed data. *International Journal of Modern Physics B*, *5*(9), 1347–1375.

Abarbanel, H. D. I., Brown, R., Sidorowich, J. J., & Tsimring, L. S. (1993). The analysis of observed chaotic data in physical systems. *Reviews of Modern Physics*, *65*(4), 1331–1392.

Agarwal, R., Gotman, J., Flanagan, D., & Rosenblatt, B. (1998). Automatic EEG analysis during long-term monitoring in the ICU. *Electroencephalography and Clinical Neurophysiology*, *107*(1), 44–58.

Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P., & Elger, C. E. (2001). Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E*, *64*, 061907.

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

Benettin, G., Galgani, L., Giorgilli, A., & Strelcyn, J.-M. (1980). Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. *Meccanica*, *15*(1), 9–30.

Chaudhuri, B. B., & Bhattacharya, U. (2000). Efficient training and improved performance of multilayer perceptron in pattern classification. *Neurocomputing*, *34*, 11–27.

Dingle, A. A., Jones, R. D., Carroll, G. J., & Fright, W. R. (1993). A multistage system to detect epileptiform activity in the EEG. *IEEE Transactions on Biomedical Engineering*, *40*(12), 1260–1268.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Finnerup, N. B., Fuglsang-Frederiksen, A., Rossel, P., & Jennum, P. (1999). A computer-based information system for epilepsy and electroencephalography. *International Journal of Medical Informatics*, *55*(2), 127–134.

Gupta, L., & McAvoy, M. (2000). Investigating the prediction capabilities of the simple recurrent neural network on real temporal sequences. *Pattern Recognition*, *33*(12), 2075–2081.

Gupta, L., McAvoy, M., & Phegley, J. (2000). Classification of temporal sequences via prediction using the simple recurrent neural network. *Pattern Recognition*, *33*(10), 1759–1770.

Güler, İ. , & Übeyli, E. D. (2003). Detection of ophthalmic artery stenosis by least-mean squares backpropagation neural network. *Computers in Biology and Medicine*, *33*(4), 333–343.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, *5*(6), 989–993.

Haykin, S. (1994). *Neural networks: A comprehensive foundation*. New York: Macmillan.

Haykin, S., & Li, X. B. (1995). Detection of signals in chaos. *Proceedings of the IEEE*, *83*(1), 95–122.

Miller, A. S., Blott, B. H., & Hames, T. K. (1992). Review of neural network applications in medical imaging and signal processing. *Medical and Biological Engineering and Computing*, *30*, 449–464.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., & Shaw, R. S. (1980). Geometry from a time series. *Physical Review Letters*, *45*(9), 712–716.

Petrosian, A., Prokhorov, D., Homan, R., Dasheiff, R., & Wunsch, D. (2000). Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG. *Neurocomputing*, *30*, 201–218.

Petrosian, A. A., Prokhorov, D. V., Lajara-Nanson, W., & Schiffer, R. B. (2001). Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG. *Clinical Neurophysiology*, *112*(8), 1378–1387.

Pineda, F. J. (1987). Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, *59*(19), 2229–2232.

Saad, E. W., Prokhorov, D. V., & Wunsch, D. C. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions on Neural Networks*, *9*(6), 1456–1470.

Sano, M., & Sawada, Y. (1985). Measurement of the Lyapunov spectrum from a chaotic time series. *Physical Review Letters*, *55*(10), 1082–1085.

Shieh, J.-S., Chou, C.-F., Huang, S.-J., & Kao, M.-C. (2004). Intracranial pressure model in intensive care unit using a simple recurrent neural network through time. *Neurocomputing*, *57*, 239–256.

Silipo, R., Deco, G., Vergassola, R., & Bartsch, H. (1998). Dynamics extraction in multivariate biomedical time series. *Biological Cybernetics*, *79*(1), 15–27.

Übeyli, E. D., & Güler, İ. (2004). Detection of electrocardiographic changes in partial epileptic patients using Lyapunov exponents with multilayer perceptron neural networks. *Engineering Applications of Artificial Intelligence*, *17*(6), 567–576.

Wolf, A., Swift, J. B., Swinney, H. L., & Vastano, J. A. (1985). Determining Lyapunov exponents from a time series. *Physica D*, *16*(3), 285–317.