Informatics in Medicine Unlocked 6 (2017) 1–11

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu

EEG signal classification using PSO trained RBF neural network for epilepsy identification

Sandeep Kumar Satapathy[a,*], Satchidananda Dehuri[b], Alok Kumar Jagadev[c]

[a] Department of Computer Science & Engineering, ITER, S'O'A University, Bhubaneswar, Odisha
[b] Department of Information & Communication Technology, Fakir Mohan University, Balasore, Odisha
[c] School of Computer Engineering, KIIT University, Bhubaneswar, Odisha

A R T I C L E I N F O

Keywords:
Electroencephalography
Radial basis function neural network
Particle swarm optimization
Discrete wavelet transform
Machine learning

A B S T R A C T

The electroencephalogram (EEG) is a low amplitude signal generated in the brain, as a result of information flow during the communication of several neurons. Hence, careful analysis of these signals could be useful in understanding many human brain disorder diseases. One such disease topic is epileptic seizure identification, which can be identified via a classification process of the EEG signal after preprocessing with the discrete wavelet transform (DWT). To classify the EEG signal, we used a radial basis function neural network (RBFNN). As shown herein, the network can be trained to optimize the mean square error (MSE) by using a modified particle swarm optimization (PSO) algorithm. The key idea behind the modification of PSO is to introduce a method to overcome the problem of slow searching in and around the global optimum solution. The effectiveness of this procedure was verified by an experimental analysis on a benchmark dataset which is publicly available. The result of our experimental analysis revealed that the improvement in the algorithm is significant with respect to RBF trained by gradient descent and canonical PSO. Here, two classes of EEG signals were considered: the first being an epileptic and the other being non-epileptic. The proposed method produced a maximum accuracy of 99% as compared to the other techniques.

1. Introduction

Electroencephalography [1] is the signal generated in the brain due to the communication of a large number of neurons among each other. This collision usually generates a very small quantity of electrical signal. Hence, Electroencephalography [1] measures this electrical activity to examine the human behavior. Careful analysis of these signals contribute to the detection of many disorders where approximately 1% of the entire world population are touched by this disease. Thus, it is necessary to identify and properly diagnose the disease. If a person has a seizure, it does not necessarily mean that the person is affected by epilepsy [2]. Hence, it is really difficult to detect and differentiate between epileptic seizures and others by manual visual inspection.

There are various methods available to record the EEG signal, such as, a 10–20 electrode placement scheme to measure the EEG. In this scheme there are several electrodes placed on the human scalp to record the EEG activity. The electrodes are placed in a 10–20 international standard, where these electrical activities in a human brain are recorded by the instrumentation connected to these electrodes via cabling.

Generally, doctors take a printed copy of this recorded signal and identify whether there is any sign of epilepsy or not. But this is quite difficult for differentiating between normal seizures and epileptic seizure through normal eyes. Hence, it is necessary to get such a system in which we can analyze the EEG signal [3] and properly differentiate between normal and epileptic seizure [4]. For our work, we have used MATLAB to analyze the EEG signals. According to the survey, it is clearly taken in that, the discrete wavelet transform (DWT) is the most effective method for analyzing the EEG signals. This method generally fits into the problem where the signals are very ephemeral in nature i.e. the frequency of signal changes rapidly with respect to time [5]. Subsequently, from the analysis of EEG signals by DWT [6,7] we can discover several statistical features that can be utilized for further processing.

After analysis of the signal, the most important phase is to classify the signals as epileptic seizures or normal. Classification is considered as a fundamental task in the field of data mining. In this method, identification of the specific data sample is made as to which pre-specified group it belongs to. Here, in this problem, we have specified two groups, one is normal and the other is an epileptic seizure group. Classification of seizures in EEG signal is usually considered as one of

* Corresponding author.
E-mail address: sandeepkumar04@gmail.com (S.K. Satapathy).

http://dx.doi.org/10.1016/j.imu.2016.12.001
Received 3 June 2016; Received in revised form 24 November 2016; Accepted 1 December 2016
Available online 08 December 2016
2352-9148/

the challenging tasks. In classification, we are given a set of instances consisting of several features or attributes called as training set. One of the attribute called the classifying attribute identifies the class to which each instance belongs. Some other mark of unknown instances called as testing set is used for evaluating the efficiency of classifier model.

Over many years NN have been very widely used in many biomedical signal analysis because they split the signals efficiently for decision making. Every classification system must be provided with a set of sample data that is represented by features extracted from a signal. Whereas, methods used for this, can be frequency domain features, wavelet transform, etc. Over the years there are several other architectures of NN model [8] that have been used such as Multilayer Perceptron Neural Network (MLPNN), Adaptive Neuro-Fuzzy Inference System (ANFIS), Radial Basis Function (RBF), Recurrent Neural Network (RNN), etc. Our classification method is based on RBFNNs which is itself a popular method practiced in many research fields because of its features, such as universal approximation, compact topology and faster learning speed.

The fundamental constraint in any classification method is its learning process. For any machine learning approach it is really important to choose the best learning method for classification. In our previous work [8], we had concluded that the RBFNN model required a better learning procedure for classification of EEG signals. Thus, in this paper, a modified version of the PSO algorithm is used to train the RBF network for classification of the EEG signal for epileptic seizure identification.

1.1. Related work

There are many researchers, who have proposed a number of methods to increase the performance of RBFNN in different applications. But in the application of EEG signal classification it is a whole new area. Several modified training methods have been suggested as provided below.

Vahid Fathi et al.[9] have proposed a novel PSO-OSD algorithm to improve the RBF learning algorithm in real time applications. Mazurowski et al.[10] have proposed a method for NN training and compared the same with back-propagation algorithm for medical decision making. Zhang et al.[11] have proposed a hybrid PSO-BP algorithm for training feed-forward NN. Ge et al.[12] suggested a modified PSO algorithm for training recurrent neural network.

Zhao .[13] have proposed a modified PSO algorithm called as CRPSO to train the NN for time series prediction. Guerra et al.[14] have proposed a novel method to train RBFNN using PSO and *k*-means clustering technique.

From these surveys, it is clearly understood that lot of research work have been done by researchers for performance enhancement of NN using PSO algorithm along with some variants of PSO. The remaining sections of this paper have been organized as follows. In Section 2, we have discussed the background details regarding the research work. Section 3, provides details about our proposed method for RBFNN training using IPSO. Section 4 describes the experimental detail used for the research work with the outcomes of experiments. At last, the paper concludes with its conclusion and future scopes.

2. Preliminaries

For our work, we have collected two different data samples of EEG. One is the EEG data for epileptic seizure identification from [15]. And, the other is an EEG data for eye state prediction. In the first phase of this problem, we can analyze the signal using DWT, which can provide several statistical features. These features can be used to construct a well defined dataset of samples and features.

2.1. Basics of discrete wavelet transform

Basically, all types of signals generated under medical diagnosis are analyzed in time domain with their amplitudes. Like EEG and ECG signals are generally collection of amplitudes with respect to time. If we plot this data it can give a shape from which the pathological condition of a patient can be observed. If there is any significant deviation in shape it can be shown and observed properly by visualizing the graph [16]. The same can be achieved by using any transformation technique such as Fourier Transform. But the major disadvantage of this is, it is not so effective for transient signals such as EEG. Hence, we need some other transform technique such as Wavelet Transformation for the analysis of EEG signal [16]. The basic idea behind this technique is to use a scale for analysis. This wavelet transform can be split up into two categories like Continuous Wavelet Transform (CWT), and DWT [17]. CWT was first made as an alternative to Short Time Fourier Transform (STFT). In this, the product of the signal with a role that is wavelet function is calculated [18]. This transformation is then calculated for different time domain. It is defined as given in Eq. (1).

$$\mathrm{CWT(a,b)}=\int_{-\infty}^{\infty} x(t).\varphi_{\mathrm{a,b}}^{\nabla}(t)\mathrm{dt} \tag{1}$$

where $x(t)$ represents the original signal. a, b represents the scaling factor and translation along the time axis respectively. The symbol ∇denotes the complex conjugation and $\varphi_{\mathrm{a,b}}^{\nabla}$is calculated by scaling the wavelet at time b and scale a.

$$\varphi_{\mathrm{a,b}}(t)=\frac{1}{\sqrt{|a|}}\varphi\left(\frac{t-b}{a}\right) \tag{2}$$

where $\varphi_{\mathrm{a,b}}(t)$represents the mother wavelet. In CWT, it is assumed that the scaling and translation parameter a and b change continuously. But the main disadvantage of CWT is the calculation of wavelet coefficients [18] for every possible scale can result in a large amount of data. It can overcome with the help of DWT. It analyzes the signal at a different frequency band by decomposing the signal into a set of high and low pass filters called as Approximation and Detailed coefficients. These coefficients can be calculated by using the wavelet toolbox available in MATLAB. Using the predefined functions available inside this toolbox, we can easily extract the features of EEG signal (as shown in Figs. 1–3). For these experimental work from the data available at [15], a rectangular window of length 256 discrete data were selected to form a single EEG segment. The wavelet coefficients have been computed using Daubechies of order four.

2.2. Feature extraction

From the data available at [15], a rectangular window of length 256 discrete data were selected to form a single EEG segment. The wavelet coefficients have been computed using Daubechies of order four. This technique was found to be more suitable because of its smoothing features which are more appropriate to detect changes in EEG signal. For our study, the original signal have been decomposed as four detailed coefficients (d1, d2, d3, d4) and four approximation coefficients (a1, a2, a3, a4). For simplicity, all the approximation coefficients are ignored except the one in the last step i.e. a_4. Hence, the signal is decomposed into five segments by using DWT. In this work, for four detailed coefficients we get 247 coefficients (129+66+34+18) and eighteen for approximation coefficients. Several statistical features have been extracted. But for this study, four important features were taken into considerations:

I. Maximum of wavelet coefficients in each sub-band.
II. Minimum of wavelet coefficients in each sub-band.
III. Mean of wavelet coefficients in each sub-band.
IV. Standard deviation of wavelet coefficients in each sub-band.

Therefore, for five coefficients all total twenty features have

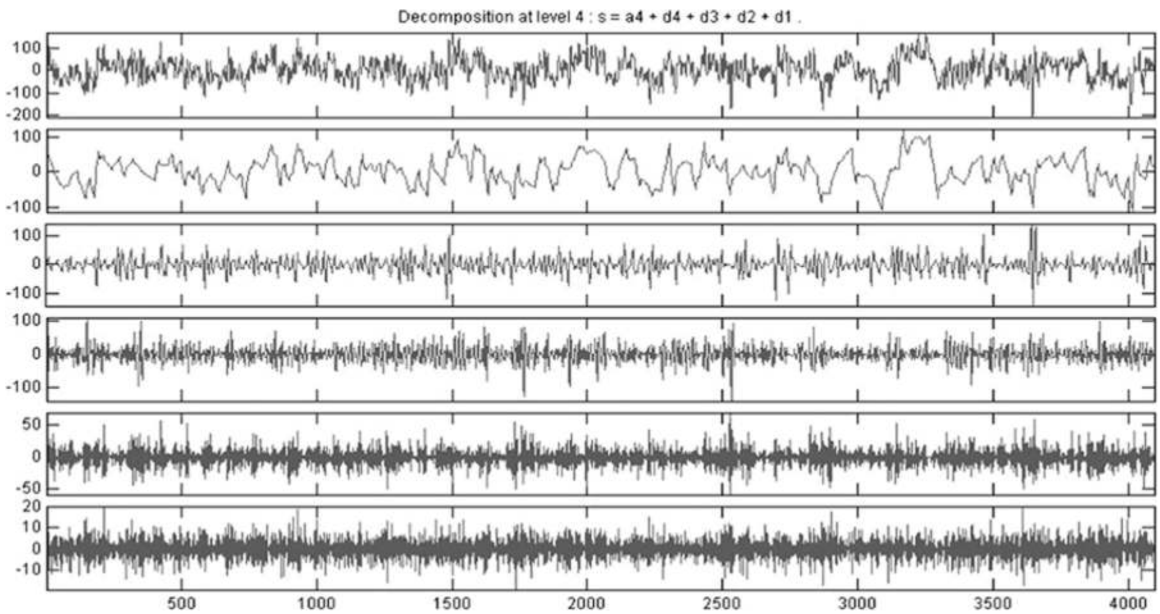

Fig. 1. : Single channel EEG signal decomposition of set A using db-2 up to level 4.

been extracted and the dataset have been constructed. Like DWT, there are many other techniques for extracting features from an EEG dataset. The features that can be extracted from the techniques include,

a. Fractal Dimension (Higuchi and Petrosian)
b. Hurst Exponent
c. Spectral, Approximation and SVD Entropy
d. Detrended Fluctuation Analysis
e. Hjorth Mobility and Complexity

These features extracted provides us a sector to explore the EEG dataset in a more detailed way for the purpose of classification. Hence, along with the features that have been extracted from DWT, we have also used these nine features for our experimental analysis process of classification. The techniques have been elaborated and stated as given below. For mathematical discussions, let us consider the signal for which features are extracted is $X = [x_1, x_2, x_3, \ldots, x_N]$

2.2.1. *Fractal dimension* [19,20]

It is one of the important features of a signal that may contain some information about the geometrical shape at different scales. These information's can be extracted using different methods such as proposed by Petrosian and Higuchi and named accordingly Petrosian Fractal Dimension (PFD) and Higuchi Fractal Dimension (HFD). Eq. (3) shows the formula for calculating PFD.

$$PFD = \frac{\log_{10}(S)}{\log_{10}(s) + \log_{10}\left(\frac{S}{S+0.4S_{\varphi}}\right)} \tag{3}$$

where S is the series length and $S_{\varphi}$is the number of sign changes in the signal.

Similarly, HFD is the slope of line that best fits the curve of $ln(Z(k))$ and $ln(1/k)$, $Z(k)$ is defined in Eq. (4).

$$Z(k) = \frac{\sum_{i=1}^{k} Z(i,\, k)}{k} where Z(m,\, k) = \frac{\sum_{i=2}^{\lfloor (N-m)/k \rfloor} |x_{m+ik} - x_{m+(i-1)k}|(N-1)}{\lfloor (N-m)/k \rfloor k} \tag{4}$$

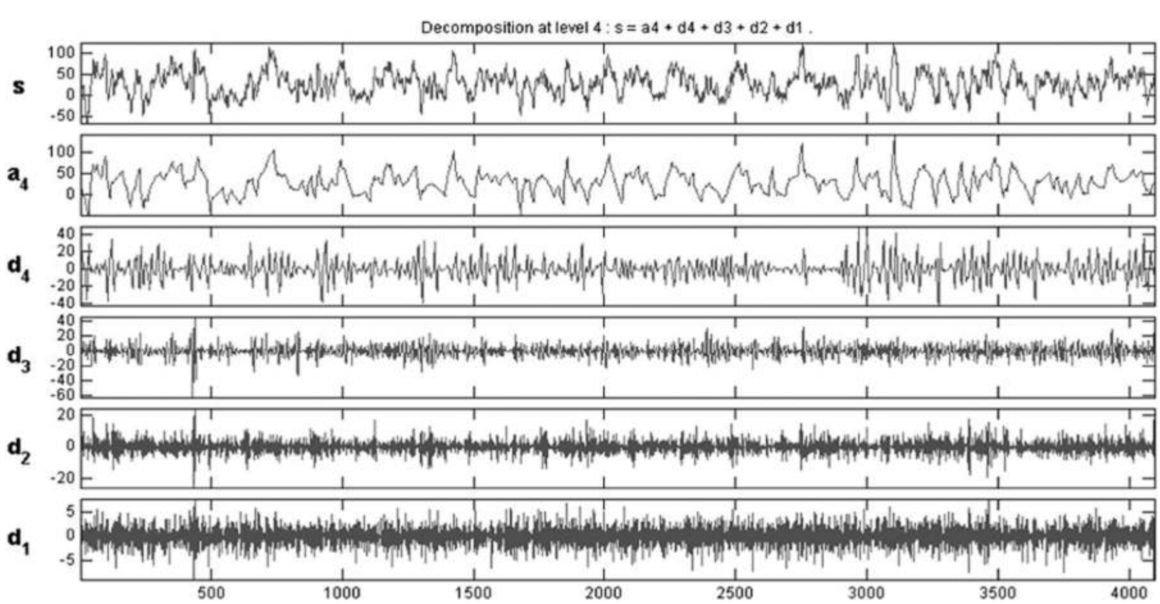

Fig. 2. : Single channel EEG signal decomposition of set D using db-2 up to level 4.

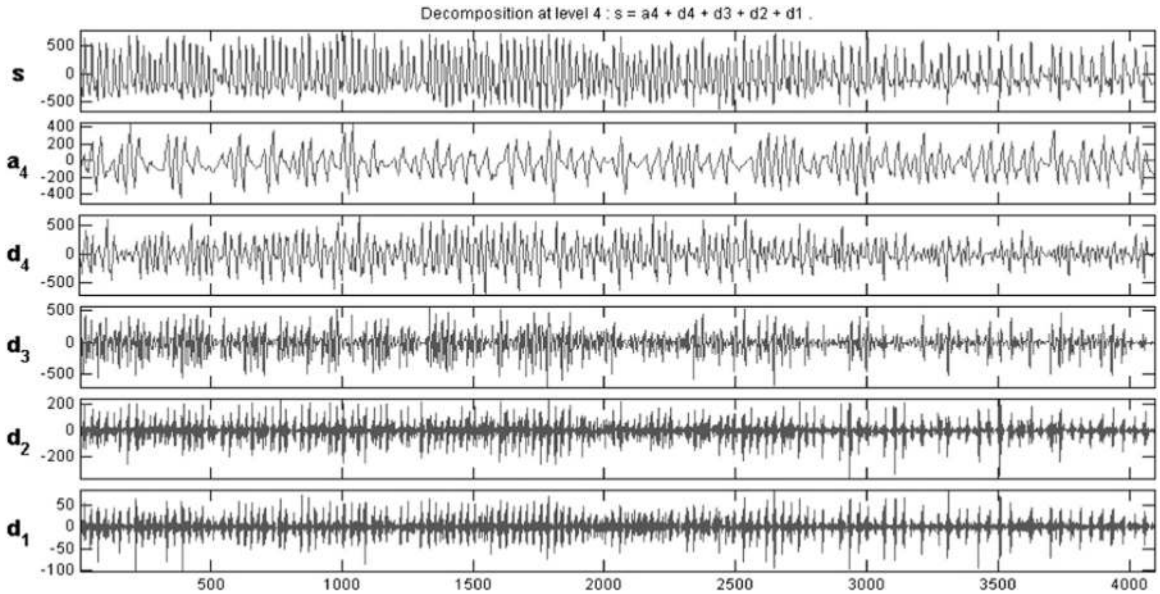

Fig. 3. : Single channel EEG signal decomposition of set E using db-2 up to level 4.

This algorithm constructs k new series from the original series as shown below:

$x_m, x_{m+k}, x_{m+2k},\ldots,x_{m+[(N-m)/k]k}$ here $m = 1,2, \ldots,k$

2.2.2. Hurst exponent [21]

It is generally used as a measure of long term memory of time series data. It can be calculated by first calculating deviation from mean of time series and then by calculating the rescaled range statistics (R/S). First, we have to calculate the accumulated deviation from mean of time series within the range T as shown in Eq. (5).

$$X(t, P)= \sum_{i=1}^{t} x_i-\bar{x}\ Where\ \bar{x}=\frac{1}{P} \sum_{i=1}^{P} x_i,\ t\in[1. .N] \tag{5}$$

Then $R(P)/S(P)$ is calculated as per the formula shown in Eq. (6).

$$\frac{R(P)}{S(P)}=\frac{max(X(t, P)) - min(X(t, P))}{\sqrt{(1/P) \sum_{t=1}^{P} [x_t - \bar{x}]^2}} \tag{6}$$

The Hurst Exponent is calculated as the slope of line produced by $ln(R(P)/S(P))$ versus $ln(P)$.

2.2.3. Spectral, approximation and SVD entropy [22–25]

Entropy can be measured as the spread of data and data with broad or flat probability distribution usually have a high entropy and vice versa. This is one of the statistical descriptor of variations in EEG signal. Spectral entropy can be specified in terms of Power Spectral intensity (PSI) and Relative Intensity Ratio (RIR) as indicated in Eq. (7).

$$SI=\frac{-1}{log(K)} \sum_{i=1}^{K} RIR_i log RIR_i \tag{7}$$

Where, $RIR_j=\frac{PSI_j}{\sum_{k=1}^{K-1} PSI_k}$ and $PSI_k= \sum_{i=\lfloor N(f_k/f_s)\rfloor}^{\lfloor N(f_k+1/f_s)\rfloor} |X_i|,\ k=1, 2, \ldots,K-1$

f_s is the sampling rate, X_i denotes FFT of time series x_i. f_1 to f_K represents K slices of the frequency band of equal or unequal widths. Similarly, *approximation entropy* is a statistical parameter computed for a time series. SVD entropy defines an entropy measure by the help of Singular Value Decomposition.

2.2.4. Detrended fluctuation analysis [26]

It is another important feature extracted for analysis of signals with scale invariant structure. It is a method for determining statistical self-affinity of a signal. The exponents obtained are almost similar to Hurst exponent.

2.2.5. Hjorth mobility and complexity [27]

Hjorth parameters generally describe the statistical properties of a signal. This is a very popular signal analysis method proposed by Hjorth in 1970, used for analyzing electroencephalogram signals. It has mainly three kinds of parameters such as activity, mobility and complexity. In this paper, we have used the last two for analysis of EEG signal, which uses the activity parameter. Mathematically, it can be defined as shown in Eq. (8).

$$Mobility=\sqrt{B2/AVG}\ Complexity=\sqrt{(B4*AVG)/(B2*B2)} \tag{8}$$

where $AVG = \sum x_i \sum x_i/N$, $B2 = \sum d_i/N$, $B4 = \sum ((d_i - d_i - 1)^2/N)$, $d_i = x_i - x_i - 1$

After the features have been extracted, the next important task is to design a classifier model for classifying the seizure and non-seizure signals. For this task, the radial basis function neural network (RBFNN) was considered for its architectural simplicity and less number of parameters required for adjustment. This technique has been briefly described in the next section.

2.3. Computational model of a radial basis function neural network (RBFNN)

RBFNN is one of the simplest form of Neural Network consisting of exactly three layers, namely input, hidden, and output layer. The limitation of only three layers makes it simpler and somehow the efficient neural network architecture (as shown in Fig. 4). The idea of RBFNN has been derived from function approximation. An RBF network positions one or more RBF neurons in the space described by the predictor variables [28]. This space has as many dimensions as there are predictor variables. The Euclidean distance is computed from the point being evaluated to the center of each neuron. The RBF is so named because the radius distance is the argument to the function. The output of RBFNN depends on the distance of the input from a given stored vector. For this research work, we have taken N number of input neurons, m number of hidden neurons and one output neuron. There are several kernel functions used in RBFNN, such as Gaussian, Multi-quadric, Inverse Multi-quadric, Mexicanhat, etc. Each of the functions has its own benefits depending on the data domain they are used. Based on the recommendation of our previous research, we use to verify the performance of Gaussian, Multi-quadric, Inverse Multi-quadric basis function in RBFNNs for identification of epileptic seizure.

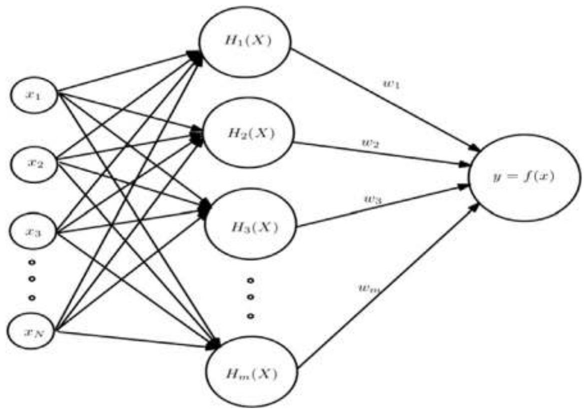

Fig. 4. : Radial Basis Function Neural Network Architecture.

2.3.1. Learning of RBFNN

Learning or training of a network is a process by which it conforms to the environment by adjusting a few parameters. For RBFNN, to generate the desired output for a given input there are mainly three adjustable parameters, such as Center, Spread, and Weight. Other than this, out of several learning algorithms gradient descent approach is most widely practiced. This is a first order derivative based optimization algorithm [29,30] for finding local minimum of a subroutine. Conceding to the Eq. (9), the error can be estimated by determining the difference between desired and real output. Here, the MSE is taken as a function [31,32] with parameters as center (c), spread (σ) and weight (w).

$$MSE(c, \sigma, w) = \frac{1}{n}\left(\sum_{i=1}^{n} d_i - \sum_{j=1}^{m} w_j * H_j(x)\right)^2 \tag{9}$$

where d_i is the desired output, $H_j(x)$ is the basis function used for RBFNN and n is the total number of samples.

Then the partial derivative of this error with respect to weight and center can be calculated to adjust the parameter with minimizing the error. The formula of gradient descent [33,34] is stated in Eq. (10).

$$w_i = w_i - \eta\frac{\partial E}{\partial w_i},\ c_{ij} = c_{ij} - \eta\frac{\partial E}{\partial c_{ij}} \tag{10}$$

where η is the learning parameter or step size. We have performed several experimental evaluations by considering different η values between 0.5 and 1.0. The detailed result is given in the next section. There are as well various other learning techniques like Particle Swarm Optimization [35], Genetic Algorithm [36], and Differential evolution [37] etc. Basically, RBF networks are used in many applications because of its architectural simplicity and requirement for less number of adjustable parameters. As a result, to employ the RBFNN in the relevance of EEG classification, we require some additional techniques for improving its performance. This can be done by integrating the optimization techniques with training methods. There are several optimization techniques available such as PSO, Artificial Bee Colony (ABC), Genetic Algorithm (GA) etc. Yet again, we opt for PSO optimization technique owing to its requirement for less number of adjustable parameters and its capability to produce global optimal solutions.

In this paper, we have proposed an improved training algorithm for RBFNN based on Particle Swarm Optimization (PSO) algorithm. The different parameters such as center, spread and weight are trained by using swarm optimization algorithm. This is a nature inspired algorithm from the behavior of bird flocks. It has been explained in the next section.

2.4. Basics of particle swarm optimization

PSO algorithm was developed initially by Kennedy and Eberhart in 1995 [38]. This algorithm is a nature inspired algorithm, that is inspired from the behavior of bird flocks called as a swarm. In this algorithm, each solution is represented as a vector called as a particle (bird). Here, the population (swarm) may contain any random number of initial solutions (particles). Each particle starts with its initial position and velocity, then moves in the solution space to achieve the optimum result. The main computational steps of PSO include generating initial position & velocity of each particle in population, updating position and velocity for a certain number of generations to get the optimal solution.

Let us discuss about the mathematical computation of PSO algorithm. Let any particle $\vec{x_k}$(solution) in n-dimensional space is represented in Eq. (11).

$$\vec{x_k} = \{x_{k1}, x_{k2}, x_{k3}, \ldots, x_{kn}\} \tag{11}$$

Where, $k=1, 2, 3, \ldots, d$ and d is the number of particles in the swarm. Each particle maintains its own velocity, let represented as given in Eq. (12).

$$\vec{v_k} = \{v_{k1}, v_{k2}, v_{k3}, \ldots, v_{kn}\} \tag{12}$$

Also, in this algorithm each particle maintains its personal best position called as p_{best} and a best solution among all the particles called as g_{best}. In each iteration or generation, the particles move towards optimal solution by updating their velocity and position according to the formula given in Eqs. (13) and (14).

$$\vec{v_k}(t+1) = \lambda * \vec{v_k}(t) + c1 * r1 * (\vec{p_k}(t) - x_k(t)) + c2 * r2 * (\vec{g_k}(t) - \vec{x_k}(t)) \tag{13}$$

$$\vec{x_k}(t+1) = \vec{x_k}(t) + \vec{v_k}(t+1) \tag{14}$$

where $\vec{v_k}(t+1)$represents the velocity of k^{th}particle at t+1 iteration. λ is the inertia weight, $\vec{v_k}(t)$ represents velocity of k^{th}particle at t iteration. $\vec{p_k}(t)$, $\vec{g_k}(t)$represents the personal best of the particle and global best of swarm at t iteration respectively. $\vec{x_k}(t)$, $\vec{x_k}(t+1)$are the previous and present solutions respectively. $c1$ and $c2$ are two positive real constants known as self confidence factor and swarm confidence factor respectively. $r1$ and $r2$ are any random number generated in between [0,1]. From the survey, it has been proved that larger inertia weight performs more efficient global search and smaller inertia weight performs efficient local search [39]. Hence, this inertia weight can be considered as an important parameter to tune the performance of PSO algorithm. This paper proposes a novel strategy to vary the inertia weight in each iteration to perform the efficient global search [40].

3. Improved PSO model for classification using RBFNN

This section will describe the new proposed learning method for RBFNN known as Improved Particle Swarm Optimization (IPSO) algorithm for classification task. This section has different parts which describe the IPSO details, then how it is used for learning RBFNN and finally an algorithmic description about the proposed model.

3.1. Improved PSO model

One of the important drawbacks of PSO algorithm is, it's quite slow searching around the global optimum. The improved PSO algorithm is based on a general PSO algorithm. The main idea of improving the base algorithm is to do a faster search around the global optimum [35]. Hence, the basic PSO algorithm has been modified as follows.

In Eq. (13), the inertia weight (λ) is generally taken as a constant value for the total number of generations. This can be modified by decreasing λ gradually as the number of generations (or iteration) increases. Thus, we can reduce the search space for global optimum by reducing the value as the number of generation increases. After each generation the best particle in the previous generation will replace the worst particle in the current generation. Several selection strategies have been suggested by researchers. In this research work, we have applied two types of selection strategy sequentially for inertia weight, one is linear selection and the other is the non-linear selection. In linear selection, λ should reduce rapidly, while around the optimum λ will reduce slowly. Mathematically, it can be described as follows.

Let λ_0 is the initial value of inertia weight, λ_1 is the end point of linear selection, $g1$ is the number of generations for linear selection and $g2$ is the number of generations of non-linear selection. Then, according to the proposed algorithm for 1 to $g1$ no. of generations the inertia weight for PSO will be calculated as given in Eq. (15).

$$\lambda_1 = \lambda_0 - ((\lambda_1/g1)*i), \text{where} i=1,2,3,\ldots,g1 \tag{15}$$

For $g1$ to $g2$ no. of generations the inertia weight for PSO will be calculated according to Eq. (16).

$$\lambda_1 = (\lambda_0 - \lambda_1)*\exp(((g1+1)-i)/i), \text{where} i=g1\ldots g2 \tag{16}$$

Generally, the value of $g1$ and $g2$ is selected for empirical study. In this research work, we have considered the total number of generations as 100. Linear and non-linear selection of the inertia weight takes place for 50% of the total number of generations. The detailed experimental evaluation of this operation has been explicated in the following part.

3.2. Improved PSO-RBFNN method for classification of EEG signal

This section describes the detailed procedure for the proposed model (as given in Fig. 5). The classifier model consists of mainly three phases. In the first phase, the data preprocessing is done. Since, we are considering EEG signal classification for epilepsy as our problem area, hence the preprocessing of data is necessary. But EEG data for eye state prediction, preprocessing is not required. The signal analysis and feature extraction are performed by using DWT. In the second stage, some parts of these datasets are provided for training RBFNN using our proposed improved PSO algorithm. The detailed algorithmic procedures are described in the next section. In the final and third phase, the network model is examined by using remaining portions of dataset. This testing of classifier model also includes the validation procedure. The different measures to calculate efficiency of model has been described in Section 4.

3.3. Algorithmic description of the proposed model

The following algorithm/pseudo code describes the detailed structure of our proposed model:

Algorithm:

For each particle do
Initialize particle position and velocity
End For
While stopping criteria is not fulfilled do
Calculate the inertia weight using Eqs.(15) or (16)depending on generation number
For each particle do
Calculate fitness value (Using MSE of RBFNN)
If fitness value is better than best fitness value in particle history (pBest) then
Set current position as pBest
End If
End For
Choose the global best (gBest) as the particle with best fitness value among all the particles
For each particle do
Calculate particle velocity using Eq.(13)
Update particle position (Center, Spread & Weight) using Eq. (14)
End For
End While

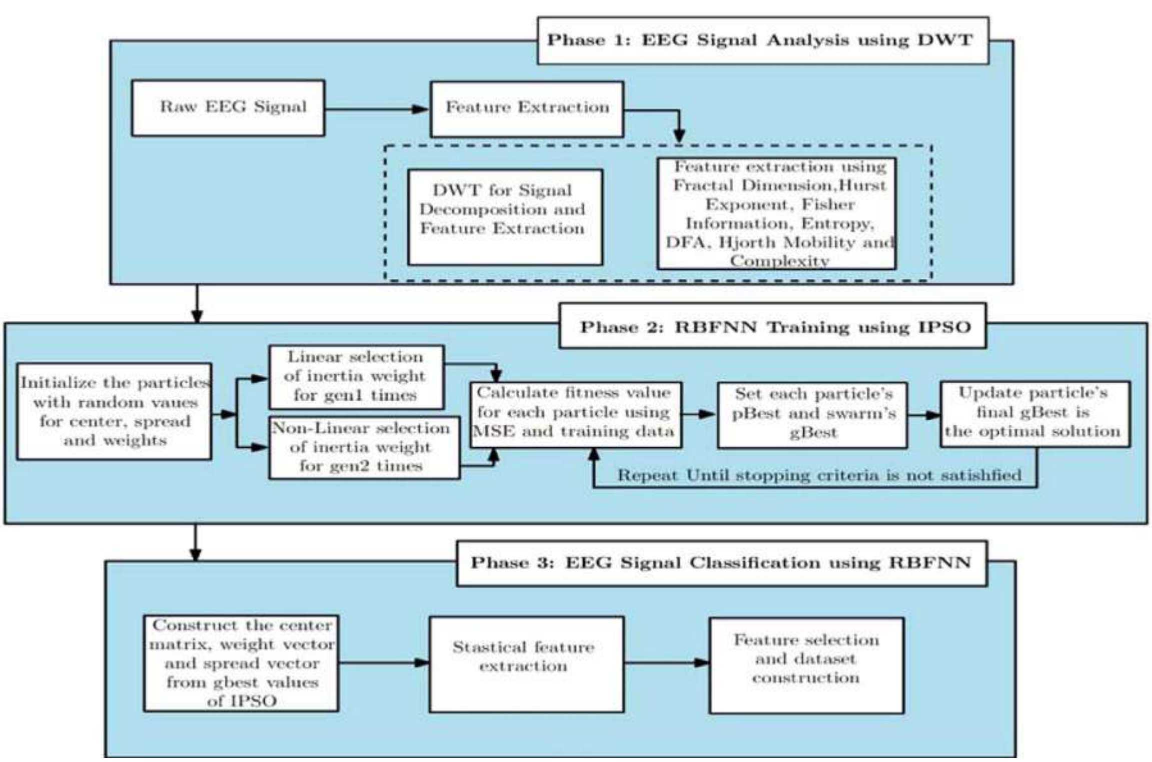

Fig. 5. : Proposed model architecture for EEG signal classification using RBFN with IPSO.

Table 1
Description of benchmark EEG dataset for Epilepsy identification and Eye state prediction.

Datasets		No. of features	No. of classes	No. of patterns
EEG dataset for epilepsy identification	Set (A & E)	20	2	200
	Set (D & E)	20	2	200
	Set (A +D & E)	20	2	300
EEG dataset for Eye state prediction		14	2	14,980

Table 2
Description of parameters used for RBFNN.

Symbols used	Description	Considered value/Size
N	Number of input vectors	200 or 300
D	Desired output vector	200×1 or 300×1
M	Number of hidden neurons	40
W	Weight vector	40×1
N	Number of input neurons	20
X	Input vector	1×20
C	Center matrix	40×20

Table 3
Description of parameters used for IPSO.

Symbols used	Description	Considered value/ Size
λ_0	Initial inertia weight	0.8
λ_1	Final inertia weight for linear selection	0.5
c1	Local search coefficient	0.9
c2	Global search coefficient	0.9
P	Population size	25
g1	Number of generations for linear increment	50
g2	Number of generations for non-linear increment	50

Table 4
Confusion matrix.

	Predicted class	
Actual Class	TP	FN
	FP	TN

Table 5
Value of Z_{CL} on different confidence intervals.

Confidence Level	90%	95%	98%	99%
Value of Z_{CL}	1.64	1.96	2.33	2.58

4. Experimental work

This section describes the detailed analysis of experimental works carried out for our proposed model. The computational complexity of the proposed algorithm may change for different datasets depending on its size. The parametric values may vary accordingly.

4.1. Datasets description

In this research study, we have conducted several experiments on two dissimilar types of datasets. One of them is an EEG dataset for epileptic seizure identification and the other one is an EEG dataset for Eye state prediction. These are openly available source of data for EEG used by many researchers for their research study. EEG data for epilepsy is primarily categorized into five types, set A, B, C, D and E (as shown in Table 1). Each set contains 100 single channel EEG segment. Each segment is of 23.6 s duration. Altogether, these data have been fixed by removing artifacts due to eye or muscle movements. Set A and B have been collected from healthy patients having eyes open and closed respectively. Set C, D and E have been collected from epileptic patients, but C and D recorded in seizure-free activity, where set E contains seizure activity. EEG data for eye state prediction is already in a sample - feature format for classification problem [41].

4.2. Environment

This paper has been supported by a lot of experimental evaluations. These results of the evaluation have been used for providing correctness of our proposed model and all the experiments have been carried out in the Java platform. The latest version of Java is used that is JDK 1.8 with Eclipse Mars as IDE. The operating system used is Linux Mint 17.2 with hardware configuration as RAM of 2 GB and Intel processor. There are several tools of Java that have been used such as classes, packages, enumerations, interfaces. Several frameworks have been designed for machine learning techniques, optimization techniques, graph drawing and designing.

4.3. Parameter details

The different significant parameters used for RBFNN are center, spread and weight. The different symbols used for RBFNN, PSO and IPSO are described in Tables 2, 3.

4.4. Evaluation metrics

Generally, the evaluation of a classification problem is based on a matrix called as a confusion matrix [42] with the number of testing samples correctly classified and incorrectly classified represented as follows (as shown in Table 4).

So, the accuracy can be measured according to Eq. (17).

$$\text{Accuracy}=\frac{TN+TP}{TN+TP+FN+FP} \tag{17}$$

For a binary classification problem, the other measures include Precision, Sensitivity or Recall and Specificity. The formula to derive these measures are given in Eqs. (18), (19) and (20).

$$\text{Precision}=\frac{TP}{TP+FP} \tag{18}$$

$$\text{Recall}=\frac{TP}{TP+FN} \tag{19}$$

$$\text{Specificity}=\frac{TN}{FP+TN} \tag{20}$$

The precision and recall can be combined together to calculate *an F - measure* [43]. A constant β controls the tradeoff between precision and recall. Formula for calculating *F*-measure is given in Eq. (21).

$$F-\text{measure}=\frac{(\beta^2+1)*\text{Precision}*\text{Recall}}{\beta^2*\text{Precision}+\text{Recall}} \tag{21}$$

In general, β value is taken as 0.9 for better analysis of *F*-measure.

The most important question in any classification task is that "how accurate is the accuracy rate estimate". The accuracy rate estimate is more accurate based on the larger size of the test set. This can be estimated by the confidence interval for a gifted degree of statistical significance. When we value the accuracy on a test set, we are actually

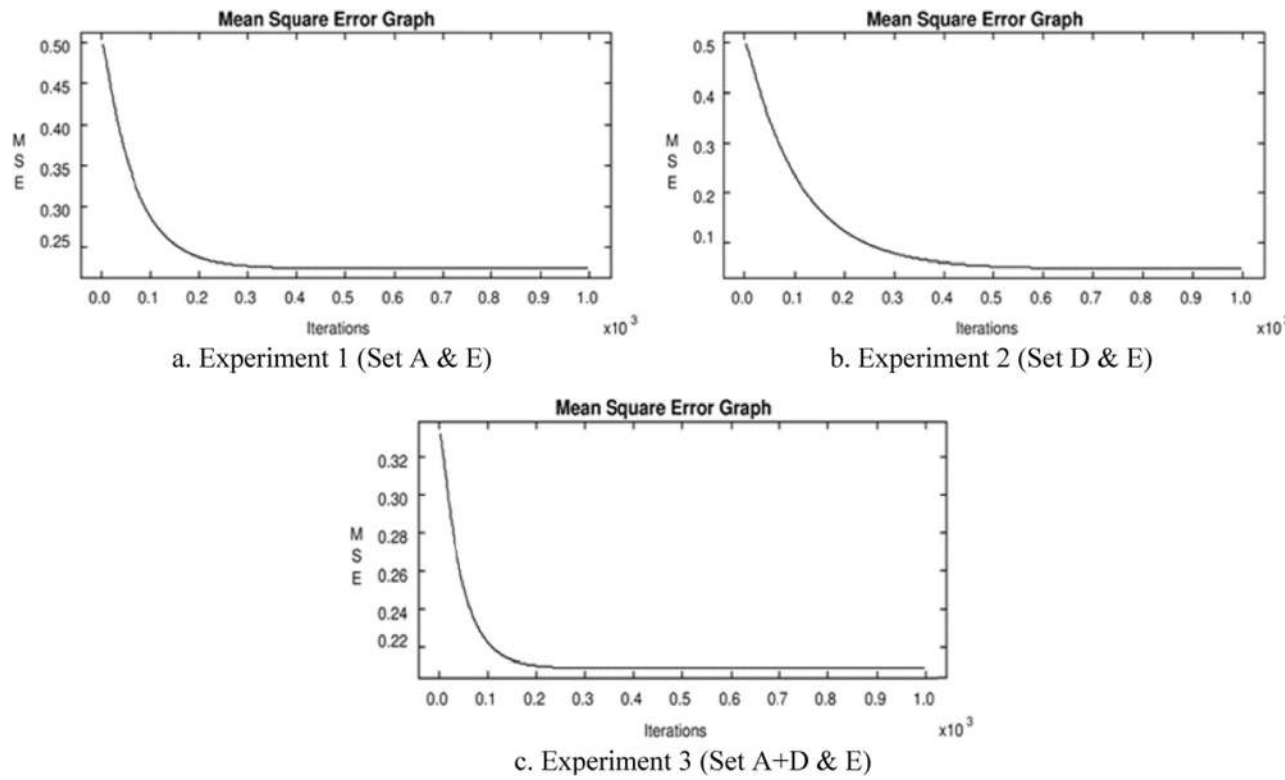

Fig. 6. : Mean square error graph of RBFNN with GD for experiment 1, 2, 3.

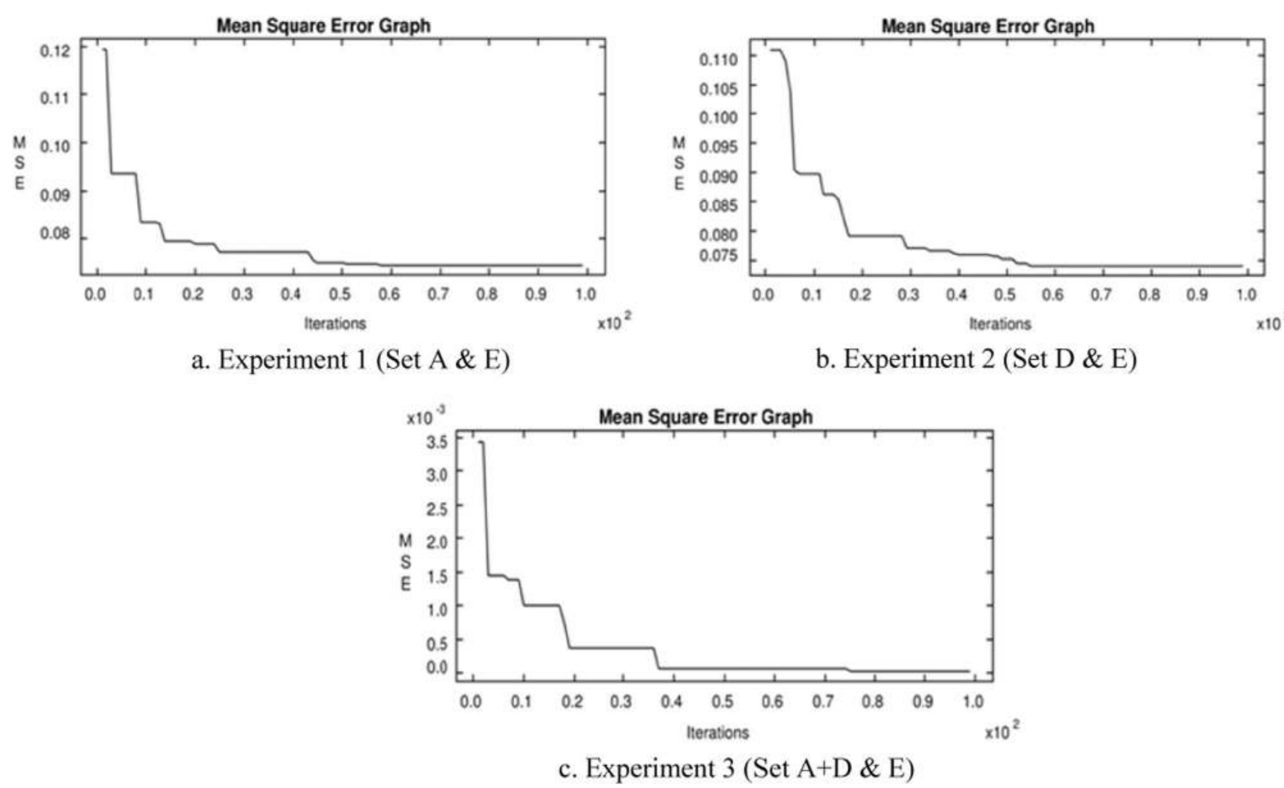

Fig. 7. : Mean square error graph of RBFNN with PSO for experiment 1, 2, 3.

performing random experiments on different independent test sets. Let *TS* is the total test set and *ITS* is any independent test set from the total test set. Let Acc_{TS} is the accuracy of the total test set and Acc_{ITS} is the accuracy of independent test set. Then, the accuracy of classifier on total test set can be represented according to Eq. (22).

$$\mathrm{Acc}_{\mathrm{TS}}=\mathrm{Acc}_{\mathrm{ITS}}\pm Z_{\mathrm{CL}}*\mathrm{SD}_{\mathrm{ITS}} \tag{22}$$

Where, Z_{CL} is the value of a standard normal random variable associated with a desired confidence level, *CL*. SD_{ITS} is the standard deviation of accuracy estimate Acc_{ITS}. Value of Z_{CL} for confidence level 90%, 95%, 98%, 99% are given in Table 5 assuming two sided

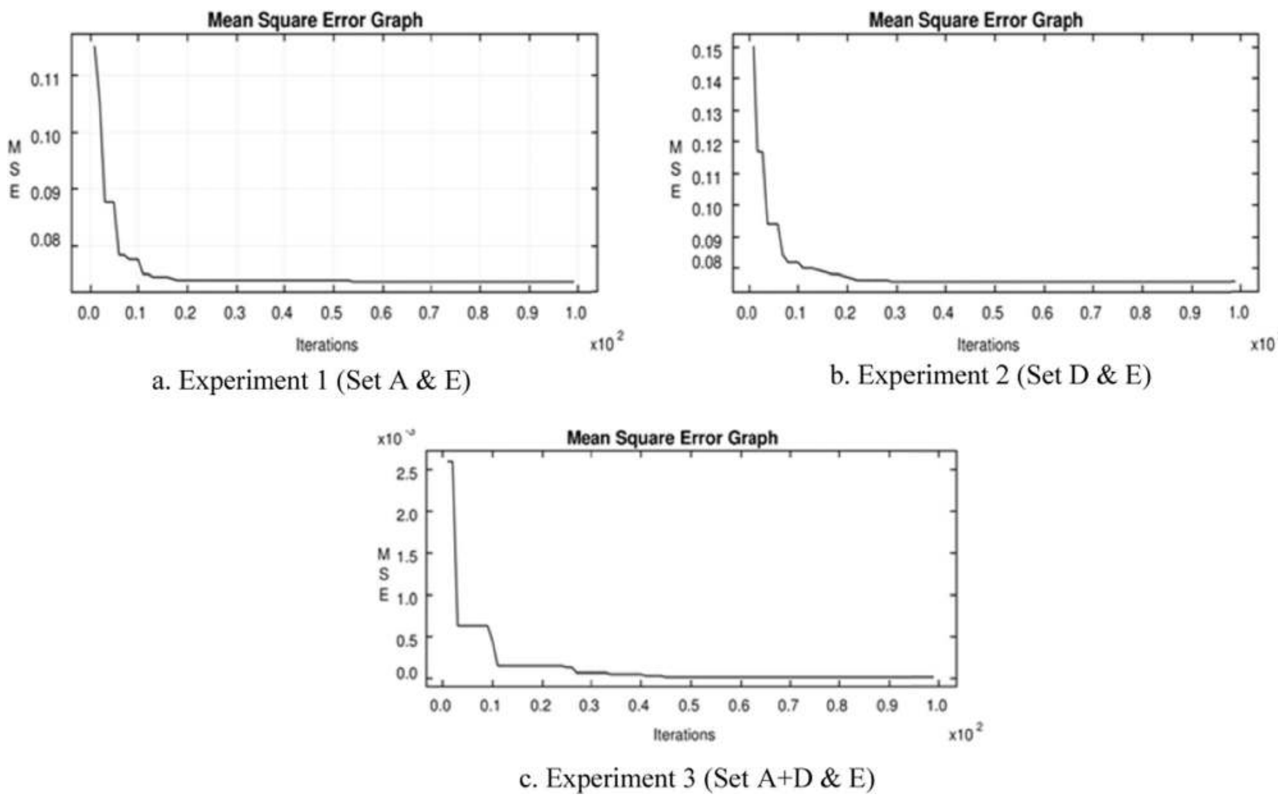

Fig. 8. : Mean square error graph of RBFNN with Improved PSO for experiment 1, 2, 3.

Table 6
Training and Testing accuracy Comparison of RBF Network (Inverse Multi-quadric) trained with GD, PSO and IPSO (Confidence level 98%).

Datasets used in experiment	RBF trained with GD		RBF trained with general PSO		**RBF Trained with improved PSO**	
	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy	**Training accuracy**	**Testing accuracy**
EEG for Epilepsy (Set A & E)	97.0 ± 0.033	70.0 ± 0.089	97.0 ± 0.033	96.00 ± 0.038	**99.0 ± 0.019**	**99.0 ± 0.019**
EEG for Epilepsy (Set D & E)	89.2 ± 0.060	84.0 ± 0.071	98.0 ± 0.027	96.00 ± 0.038	**99.0 ± 0.019**	**97.0 ± 0.033**
EEG for Epilepsy (Set A+D & E)	80.4 ± 0.063	75.3 ± 0.069	85.7 ± 0.056	78.6 ± 0.065	**90.6 ± 0.046**	**84.6 ± 0.057**
EEG for Eye State Prediction	90.6 ± 0.006	86.4 ± 0.007	93.4 ± 0.005	87.54 ± 0.0074	**98.3 ± 0.0029**	**95.19 ± 0.0048**

Table 7
Comparison of performance of RBF Network (Gaussian) trained with GD, PSO and IPSO (Confidence level 98%).

Datasets used in experiment	RBF trained with GD		RBF Trained with general PSO		**RBF trained with improved PSO**	
	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy	**Training accuracy**	**Testing accuracy**
EEG for Epilepsy (Set A & E)	97.0 ± 0.039	70.0 ± 0.106	97.0 ± 0.039	96.00 ± 0.0456	**99.0 ± 0.023**	**99.0 ± 0.023**
EEG for Epilepsy (Set D & E)	89.2 ± 0.072	84.0 ± 0.085	98.0 ± 0.032	96.00 ± 0.0456	**99.0 ± 0.023**	**97.0 ± 0.039**
EEG for Epilepsy (Set A+D & E)	80.4 ± 0.075	75.3 ± 0.100	85.7 ± 0.066	78.6 ± 0.095	**90.6 ± 0.055**	**84.6 ± 0.084**
EEG for Eye State Prediction	90.6 ± 0.0078	86.4 ± 0.009	93.4 ± 0.006	87.54 ± 0.007	**98.3 ± 0.003**	**95.19 ± 0.0057**

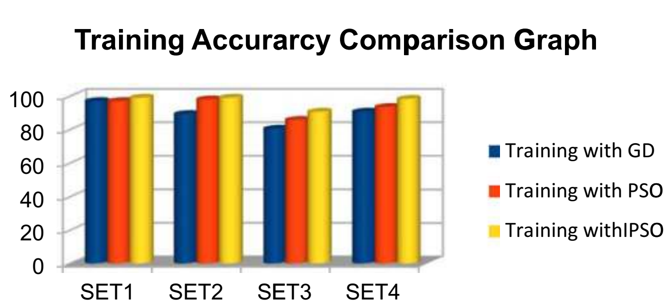

Fig. 9. : Comparison of training accuracy for different datasets using different training techniques.

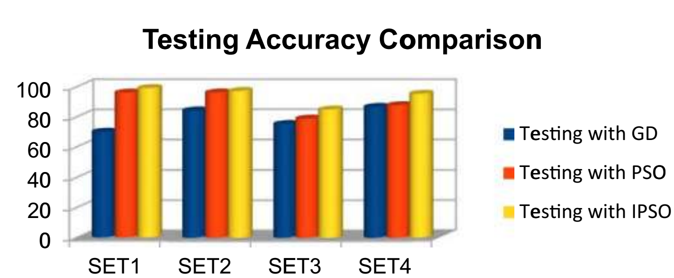

Fig. 10. : Comparison of testing accuracy for different datasets using different training techniques.

Table 8
Other performance measures for RBFNN trained with GD approach.

Datasets used in Experiment	Precision	Recall	Specificity	F-measure
EEG for Epilepsy (Set A & E)	0.9	0.642	0.833	0.763
EEG for Epilepsy (Set D & E)	0.98	0.845	0.976	0.914
EEG for Epilepsy (Set A+D & E)	0.26	1.0	0.729	0.388
EEG for Eye State Prediction	1.0	0.782	1.0	0.889

Table 9
Other performance measures for RBFNN trained with PSO approach.

Datasets used in Experiment	Precision	Recall	Specificity	F-measure
EEG for Epilepsy (Set A & E)	0.92	1.0	0.926	0.954
EEG for Epilepsy (Set D & E)	0.92	1.0	0.926	0.954
EEG for Epilepsy (Set A+D & E)	0.7	1.0	0.869	0.808
EEG for Eye State Prediction	1.0	0.782	1.0	0.889

Table 10
Other performance measures for RBFNN trained with Improved PSO approach.

Datasets used in experiment	Precision	Recall	Specificity	F-measure
EEG for Epilepsy (Set A & E)	**0.98**	**1.0**	**0.98**	**0.988**
EEG for Epilepsy (Set D & E)	**0.94**	**1.0**	**0.943**	**0.966**
EEG for Epilepsy (Set A+D & E)	**0.78**	**1.0**	**0.9**	**0.865**
EEG for Eye State Prediction	**0.892**	**1.0**	**0.92**	**0.937**

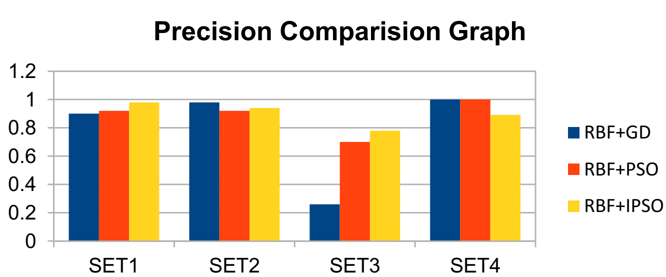

Fig. 11. : Comparison of Precision for different datasets using different training techniques.

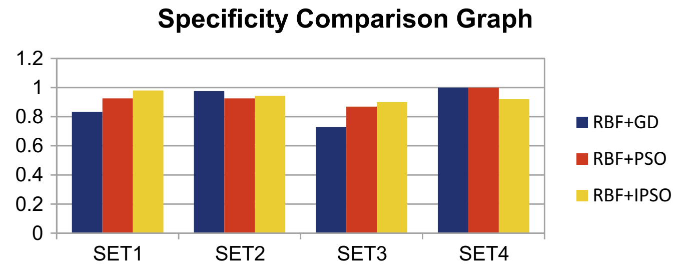

Fig. 13. : Comparison of Specificity for different datasets using different training techniques.

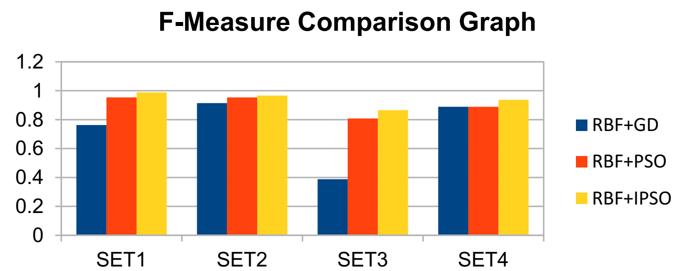

Fig. 14. : Comparison of F-measure for different datasets using different training techniques.

confidence intervals.

Standard deviation can be calculated as given in Eq. (23).

$$SD_{ITS}=\sqrt{(Acc_{ITS}*(1-Acc_{ITS}))/n} \quad (23)$$

Where, n is the number of data instances in any independent test set. The validation of the results has been performed by k-fold cross validation. Here, k value is chosen as 10. Hence, the whole dataset is divided into 10 exceptional subsets. In each cycle of classification process, one set is in use for testing purpose and rest of the sets are taken for training purpose. Thus, total 10 cycles for classification task have been performed and the performance metrics are calculated. As a result, the average of these metrics is taken as the final performance results. There is a very minute difference between the best performance results and average performance results through cross validation.

4.5. Results and analysis

This section provides the detailed experimental evaluation results performed according to above said procedures. Different experiments have been performed to analyze the efficiency of classification process trained with improved particle swarm optimization for classifying epileptic seizure from non-epileptic ones. To prove this, it has been compared with other existing techniques such as RBFNN trained with gradient descent approach and RBFNN trained with the conventional PSO algorithm.

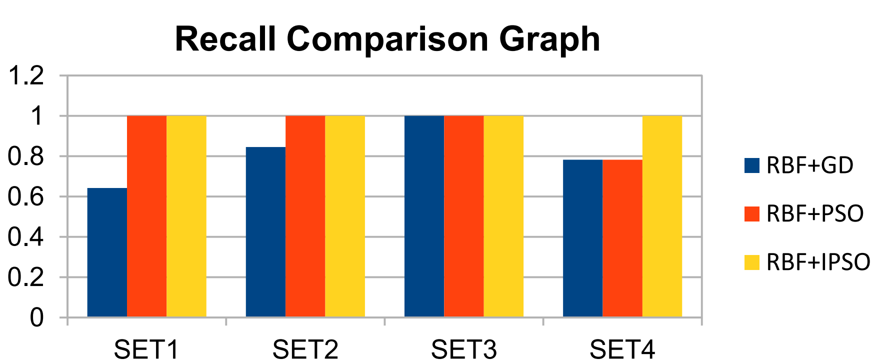

Fig. 12. : Comparison of Recall for different datasets using different training techniques.

Figs. 6, 7 and 8 shows the graph for the mean square error with respect to the number of iterations for different datasets with different training algorithms (Fig. 6 for RBFNN with GD approach, Fig. 7 for RBFNN with conventional PSO and Fig. 8 for RBFNN with improved PSO). As compared to Fathi et al. [9] where the PSO algorithm has been applied to optimize Optimum Steepest Decent (OSD) algorithm, here we have tried to optimize the parameters directly used in RBF network such as center, weight and spread parameter values.

These graphs show the comparison of different methods. The following tables (Tables 6, 7) show the training and testing accuracy of different training methods. Table 6 shows different accuracies validated at a confidence level 95% and Table 7 shows accuracies validated at confidence level 98%.

Figs. 9 and 10 shows the graph representation for comparison of different training algorithms on 4 different datasets. Here, SET1 represents EEG data for epilepsy with set A and E. SET2 represents EEG data for epilepsy with set D and E. SET3 represents EEG data for epilepsy with set A, D and E. SET4 represents EEG data for eye state prediction.

As per descriptions given in Section 4.4 other than accuracy, several different measures have been considered in comparing different techniques. Those include precision, recall, specificity and F-measure. Table 8 shows the values of these measures for RBFNN trained with GD approach. Table 9 shows the values of these measures for RBFNN trained with conventional PSO algorithm. Table 10 shows the values of these measures for RBFNN trained with the improved PSO algorithm.

Similarly, the graph representations for comparing different techniques have been provided in Figs. 11–14 respectively. From these detailed experimental evaluation, it can be strongly proved that RBFNN trained with improved PSO algorithm outperforms other techniques for the classification of EEG signal in epileptic seizure identification. The main advantages of this proposed method is that it is able to classify epileptic seizures and non-epileptic seizures with maximum accuracy for different cases such as Set A-B, D-E, AD-E and eye state EEG. One of the disadvantage we can say there is a very small increase in time required for classification as compared to others (PSO and GD) which can be considered negligible as compared to increase the accuracy.

5. Conclusion and future work

In this research work, a new modified PSO algorithm have been proposed to train the RBFNN more efficiently to classify the epileptic seizures. Also, this technique was examined on a different dataset, i.e. Eye state prediction. This proposed technique was compared with few other available techniques (Gradient decent, convention PSO) by rigorous and thorough practical implementations and experimental results. Thus, it was proved that the proposed technique outperformed the other existing techniques. In this research work, DWT was only utilized for analysis and statistical feature extraction from EEG datasets for epilepsy. For eye state prediction, the dataset was already in the format for classification. Our future work would focus, to broaden this work for handling imbalanced datasets for classification task.

References

[1] Niedermeyer E, Lopesda Silva F. Electroencephalography: basic principles, clinical applications, and related fields, 5th ed.. London: Lippincott Williams and Wilkins; 2005.

[2] Witte H, Iasemidis LD, Litt B. Special issue on epileptic seizure prediction May. IEEE Trans Biomed Eng 2003;50(5):537–8.

[3] Sanei S, Chambers JA. EEG signal processing. New York: Wiley; 2007.

[4] Lehnertz K. Non-linear time series analysis of intracranial EEG recordings in patients with epilepsy–an overview. Int J Psychophysiol 1999;34(1):45–52.

[5] Alicata FM, Stefanini C, Elia M, Ferri R, Del Gracco S, Musumeci SA. Chaotic behavior of EEG slow-wave activity during sleep. Electroencephalogr Clin Neurophysiol 1996;99(6):539–43.

[6] Ocak H. Automatic detection of epileptic seizures in EEG using discrete wavelet transform and approximate entropy. Expert Syst Appl 2009;36(2):2027–36.

[7] Acharya UR, Sree SV, Swapna G, Joy Martis R, Suri JS. Automated EEG analysis of epilepsy: a review. Knowl Based Syst 2013;45:147–65.

[8] Satapathy SK, Jagadev AK, Dehuri S. An empirical analysis of training algorithms of neural networks: a case study of EEG signal classification using java framework. Adv Intell Syst Comput 2015;309:151–60.

[9] Fathi V, Montazer GA. An improvement in RBF learning algorithm based on PSO for real time application. Neurocomputing 2013;111:169–76.

[10] Mazurowski MA, Habas PA, Zurada JM, Lo JY, Baker JA, Tourassi GD. Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance. Neural Netw 2008;21(2–3):427–36.

[11] Zhang J, Lok T, Lyu M. A hybrid particle swarm optimization-back-propagation algorithm for feed forward neural network training. Appl Math Comput 2007;185(2):1026–37.

[12] Ge HW, Qian F, Liang YC, Du WL, Wang L. Identification and control of nonlinear systems by a dissimilation particle swarm optimization-based elman neural network. Nonlinear Anal: Real World Appl 2008;9(4):1345–60.

[13] Zhao L, Yang Y. PSO-based single multiplicative neuron model for time series prediction. Expert Syst Appl 2009;36:2805–12.

[14] Guerra FA, Coelho LDS. Multi-step ahead nonlinear identification of lorenz's chaotic system using radial basis neural network with learning by clustering and particle swarm optimization. Chaos Solitons Fractals 2008;35(5):967–79.

[15] Andrzejak RG, Lehnertz K, Rieke C, Mormann F, David P, Elger CE. Indications of nonlinear deterministic and finite dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. Phys Rev 2001(64), [0619071-8].

[16] Gotman J. Automatic recognition of epileptic seizure in the EEG. Electroencephalogr Clin Neurophysiol 1982;54(5):530–40.

[17] Subasi A, Alkan A, Kolukaya E, Kiymik MK. Wavelet neural network classification of EEG signals by using AR model with MLE preprocessing. Neural Netw 2005;18(7):985–97.

[18] Subasi A, Gursoy MI. EEG signal classification using PCA, ICA, LDA and support vector machines. Expert Syst Appl 2010;37(12):8659–66.

[19] Petrosian A. Kolmogorov complexity of finite sequences and recognition of different preictal EEG patterns. In: Proceedings of the 8th IEEE symposium on computer-based medical systems; 212–217; 1995.

[20] Higuchi T. Approach to an irregular time series on the basis of the fractal theory. Physica D 1988;31(2):277–83.

[21] Balli T, Palaniappan R. A combined linear & nonlinear approach for classification of epileptic EEG signals. In: Proceedings of the 4th international IEEE/EMBS conference on neural engineering (NER '09); 714–717; 2009.

[22] James CJ, Lowe D. Extracting multisource brain activity from a single electromagnetic channel. Artif Intell Med 2003;28(1):89–104.

[23] Inouye T, Shinosaki K, Sakamoto H. Quantification of EEG irregularity by use of the entropy of the power spectrum. Electroencephalogr Clin Neurophysiol 1991;79(3):204–10.

[24] Pincus SM, Gladstone IM, Ehrenkranz RA. A regularity statistic for medical data analysis. J Clin Monit Comput 1991;7(4):335–45.

[25] Roberts SJ, Penny W, Rezek I. Temporal and spatial complexity measures for electroencephalogram based brain-computer interfacing. Med Biol Eng Comput 1999;37(1):93–8.

[26] Peng CK, Havlin S, Stanley HE, Goldberger AL. Quantification of scaling exponents and crossover phenomena in non-stationary heartbeat time series. Chaos 1995;5(1):82–7.

[27] Hjorth B. EEG analysis based on time domain properties. Electroencephalogr Clin Neurophysiol 1970;29(3):306–10.

[28] Aslan K, Bozdemir H, Sahin C, Ogulata SN, Erol R. A radial basis function neural network model for classification of epilepsy using EEG signals. J Med Syst 2008;32(5):403–8.

[29] Gabor AJ, Seyal M. Automated interictal EEG spike detection using artificial neural networks. Electroencephalogr Clin Neurophysiol 1992;83(5):271–80.

[30] Nigam VP, Graupe D. A neural-network-based detection of epilepsy. Neurol Res 2004;26(1):55–60.

[31] Majumdar K. Human scalp EEG processing: various soft computing approaches. Appl Soft Comput 2011;11(8):4433–47.

[32] Witte H, Iasemidis LD, Litt B. Special issue on epileptic seizure prediction. IEEE Trans Biomed Eng 2003;50:537–8.

[33] Robert C, Gaudy JF, Limoge A. Electroencephalogram processing using neural networks. Clin Neurophysiol 2000;113(5):694–701.

[34] Elo P, Saarinen J, Varri A, Nieminen H, Kaski K. Classification of epileptic EEG by using self-organizing maps. In: Aleksander I, Taylor J. (Eds.), Artificial neural networks, 2; 1992. p. 1147– 1150.

[35] Dehuri S, Roy R, Cho SB, Ghosh A. An improved swarm optimized functional link artificial neural network (ISO-FLANN) for classification. J Syst Softw 2012;85(6):1333–45.

[36] Dehuri S, Cho SB. Evolutionary optimized features in functional link neural network for classification. Expert Syst Appl 2010;37(6):4379–91.

[37] Dash ChSK, Dash AP, Dehuri S, Cho SB, Wang GN. DE+RBFNs based classification: a special attention to removal of inconsistency and irrelevant features. Eng Appl AI 2013;26(10):2315–26.

[38] Kennedy J, Eberhart RC. Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks IV:1995; 1942–1948.

[39] Qasem SN, Shamsuddin SM. Hybrid learning enhancement of RBF network based on particle swarm optimization; 5553; 2009. p. 19–29.

[40] Shi Y, Eberhart RC. A modified particle swarm. In: Proceedings of the IEEE international conference on evolutionary computation: 1998.

[41] Wang T, Guan S, Man KL, Ting TO. EEG Eye state identification using incremental attribute learning with time series classification. Math Probl Eng 2014;2014:1–10.

[42] Tom F. An introduction to ROC analysis. Pattern Recognit Lett 2006;27:861–74.

[43] John M, Francis K, Richard S, Ralph W. Performance measures for information extraction, In: Proceedings of DARPA Broadcast News Workshop, Herndon, VA, February; 1999.