Design of Multidimensional Classifiers using Fuzzy Brain Emotional Learning Model and Particle Swarm Optimization Algorithm

Yuan Sun, Chih-Min Lin*

Department of Electrical Engineering, Yuan Ze University, Tao-Yuan 320, Taiwan,
s1058505@mail.yzu.edu.tw; cml@saturn.yzu.edu.tw* (corresponding author)

Abstract: This study presents a multidimensional classifier design using a fuzzy brain emotional learning model, combined with a particle swarm optimization (PSO) algorithm that allows a network to automatically determine the optimum values for the weights of the reward signal. The multidimensional fuzzy brain emotional learning classifier(MFBELC) is first described with corresponding fuzzy inference rules; then the PSO algorithm is applied for the optimum parameter choice. This PSO-MFBELC is evaluated for the Wine dataset and Iris dataset, which are publicly available from the UCI machine learning database. A comparison of simulations using the proposed PSO-MFBELC shows that this classifier is superior to other algorithms in the recognition accuracy aspect.

Keywords: brain emotional learning model; fuzzy inference system; particle swarm optimization algorithm; multidimensional classifier

1 Introduction

The inspiration for emotional learning in the brain comes from the anatomical discovery of LeDoux's emotional learning mechanism in the mammalian brain in 1991 [1]. In 2001, an algorithm based on the computational model of emotional processing - brain emotional learning (BEL) was initially developed by Moren, with the advantages of low computational complexity, fast convergence, and good stability [2].

In several studies, BEL model has been widely used for control [3] [6] [7] [8], prediction [9]-[10] [11] [12], identification [13], [14] and binary classification [15]-[16] [17] [18]. In recent years, this model has also been extended to overcome the multi-classification problem [19], [20]. However, in the previous application studies (including the authors' past papers [5], [17], [18], [20]), it is difficult to determine the appropriate parameters of the BEL model, and the parameters that need to be set for different samples are different. Whether the setting of these parameters is appropriate or not has a great impact on the results. Most scholars usually use the trial-and-error method to set the parameters, but it is time-consuming and unstable. Therefore, in order to make the parameter setting more efficient and stable, several optimization algorithms have been proposed; some examples are given as follows. A gray wolf optimizer (GWO) algorithm has been proposed for tuning the parameters of Takagi-Sugeno proportional-integral fuzzy controllers (PI-FCs) [21]. Iterative feedback tuning (IFT) and iterative learning control (ILC) have been used to minimize the objective function [22]. A weighted interest pattern (WIP) mining method has been proposed to improve the performance of data mining [23].

This paper uses the particle swarm optimization (PSO) algorithm to search the appropriate parameters for achieving desired classification performance. The PSO algorithm is a simplified model based on swarm intelligence, which is inspired by the regularity of bird swarm activities [24]. Previously, PSO was applied to some artificial intelligent algorithms [25]-[26] [27] [28] [29], and until recently, some researchers have utilized PSO to find the most suitable parameters in the structure of the BEL model [30]-[33]. In spite of these applications, an appropriate choice of optimal parameters or fitness function for a PSO-BEL algorithm is necessary for different applications.

For intelligent systems, there were a lot of modeling techniques and they have been applied in various fields. The combination of fuzzy logic, neural network, genetic algorithm, and statistical analysis is analyzed in [34]. A new feature-based expert system modeling method is proposed in [35]. The modeling of a multi-relational classifier has been proposed based on canonical correlation analysis [36]. In [36], the method of model transformation based on tensor product model is applied to magnetic levitation systems

This paper aims to propose a more efficient multidimensional classifier. The brain emotion model, fuzzy inference system, and PSO algorithm are combined to form a new intelligent model. Then, a multidimensional fuzzy brain emotion learning classifier with reward signal optimization is developed. The main contributions of this paper are as follows. (1) A multidimensional classifier based on fuzzy inference system and BEL model (MFBELC) is proposed, (2) The PSO algorithm is successfully applied to search the optimal values of the two weight factors of reward signal in MFBELC, and then the classification performance is obviously improved. (3) The effectiveness of the proposed classifier has been verified by two multidimensional classification examples, and it can achieve better accuracy than most other classification models.

The rest of this study is organized as follows: Section 2 introduces the overall structure of the PSO multidimensional fuzzy brain emotional learning classifier, including the updating algorithm and the implementation process. Section 3 introduces the simulation results in detail and compares the performance of the proposed classifier with other models. Conclusions are detailed in Section 4.

2 The PSO - MFBELC Model

The proposed PSO-MFBELC model consists of two parts: the multidimensional fuzzy brain emotion learning classifier and PSO algorithm. The PSO algorithm searches the optimal parameters through iterations and then assigns these parameters to the MFBELC model. The details of the algorithm are described in Section 2.3.

2.1 Multidimensional Fuzzy Brain Emotional Learning Classifier

2.1.1 Fuzzy Inference Rules of MFBELC

In a traditional brain emotional learning model, sensory input is calculated in the sensory cortex and sent directly to the orbitofrontal cortex and amygdala, without any learning process. Different from the traditional BEL model, for the proposed MFBELC, the fuzzy inference rules are proposed and defined as:

$$\text{If } I_1 \text{ is } S_{1j} \text{ and } I_2 \text{ is } S_{2j}\,,\ \cdots,\ \text{ and } I_{n_i j} \text{ is } S_{n_i j}, \text{ then } A_l = V_{ijl} \tag{1}$$

$$\text{If } I_1 \text{ is } S_{1j} \text{ and } I_2 \text{ is } S_{2j}\,,\ \cdots,\ \text{ and } I_{n_i j} \text{ is } S_{n_i j}, \text{ then } O_l = W_{ijl} \tag{2}$$

$$\text{for } i = 1,2,\cdots,n_i\,,\quad j = 1,2,\cdots,n_j\,,\quad l = 1,2,\cdots,n_l$$

where n_i is the input dimension, n_l is the output dimension and n_j is the number of neurons. S_{ij} is the fuzzy set for the i-th input and j-th neuron. A_l is the l-th output of the amygdala, and O_l is the l-th output of the orbitofrontal cortex. V_{ijl} is the amygdala weight for the l-th output corresponding to the i-th input and j-th neuron in the consequent part. Likewise, W_{ijl} is the orbitofrontal cortex weight for the l-th output corresponding to the i-th input and j-th neuron in the consequent part.

2.1.2 Structure of a Multidimensional Fuzzy Brain Emotional Learning Classifier

Figure 1 shows a multidimensional brain emotional learning classifier with six layers: the sensory input, sensory cortex, orbitofrontal cortex, thalamus, amygdala, and the output space. The following details the data transmission of MFBELC and the basic functions of each layer.

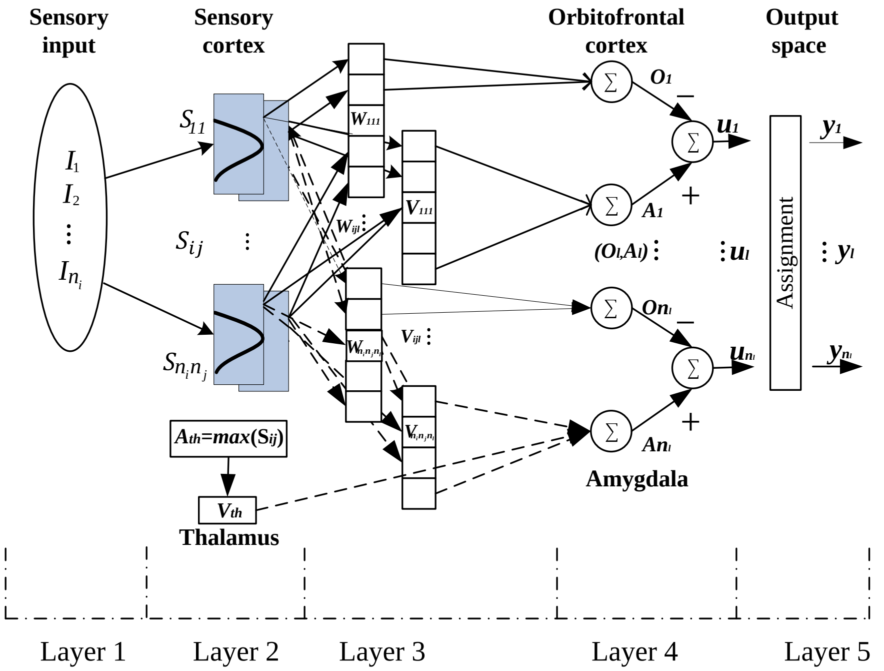


Figure 1
Structure of the multidimensional brain emotional learning classifier

a) Layer 1

Layer 1 is the sensory input space, where the input vector $I = [I_1, \cdots, I_i, \cdots, I_{n_i}]^T \in R^{n_i}$. In general, according to the given classification problem, the input dimension can also be regarded as the feature dimension.

b) Layer 2

Layer 2 is the sensory cortex space. Sensory input is transmitted to the orbitofrontal cortex and amygdala. In order to improve generalization ability and operation speed, The Gaussian function is used as a membership function as follows:

$$S_{ij} = exp\left[\frac{-(I_i - m_{ij})^2}{\sigma_{ij}^2}\right] \tag{3}$$

where m_{ij} and σ_{ij} correspond to the mean and variance of S_{ij}, respectively.

The thalamus receives the maximum signal from the sensory input layer, and it is known as the thalamic signal

$$A_{th} = max(S_{ij}) \tag{4}$$

c) Layer 3

Layer 3 is the weight space. Therein, a fuzzy output is represented by a block, which is the result of fuzzy inference rules.

For the amygdala system, this space is called the sensory weight space *V*, expressed in a vector form:

$$V = [V_{111}, \cdots, V_{ijl}, \cdots V_{n_i n_j n_l}]^T \in R^{n_i n_j n_l} \tag{5}$$

For the orbitofrontal cortex system, this space is called emotion weight space *W*, expressed in a vector form:

$$W = [W_{111}, \cdots, W_{ijl}, \cdots W_{n_i n_j n_l}]^T \in R^{n_i n_j n_l} \tag{6}$$

d) Layer 4

Layer 4 is the algebraic sum of input S_{ij} for the sensory cortex with activation weights.

For the orbitofrontal cortex, the corresponding nodes in the orbitofrontal cortex can receive the signals from the amygdala. The l-th output in the orbitofrontal cortex is

$$O_l = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} S_{ij} W_{ijl} \tag{7}$$

For the amygdala system, stimulation is received through the corresponding node in three parts: sensory input, reward signal, and thalamic signal. The l-th output in the amygdala is

$$A_l = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} S_{ij} V_{ijl} + A_{th} V_{th} \tag{8}$$

e) Layer 5

Layer 5 is the output space. It is the output of the brain emotional learning model, designed as

$$u_l = A_l - O_l \tag{9}$$

where u_l receives the l-th output from amygdala, and then subtracts the inhibitory outputs from the orbitofrontal cortex.

However, either in the training process or in the testing phase, the classification judgment result of each sample needs to be given, which means the output of the classifier should reflect the category label. For a binary classification problem, the sigmoid function is commonly adopted and a cut-off threshold is used to separate the two categories. For a multidimensional classifier, this method may produce some obstacles. Thus, the final output of this multidimensional brain emotional learning classifier is defined as

$$y_l = \begin{cases} 1, & u_l = max(u_1, u_2, \cdots, u_{n_l}) \\ 0, & else \end{cases} \tag{10}$$

From (10), obviously, the total output of the multidimensional classifier is presented as a multidimensional array and the index of the value 1 indicates the category label.

2.1.3 Learning Algorithm for MFBELC

Each emotional learning process in the amygdala and orbitofrontal cortex is a process of dynamic weight adjustment. According to an associative learning method [38], the l-th weight updating formulas of the amygdala and orbitofrontal cortex are respectively applied as

$$\Delta V_{ijl} = \lambda_v \ \ (S_{ij} max(0, REW_l - A_l)) \tag{11}$$

$$\Delta W_{ijl} = \lambda_w(S_{ij}(A_l - O_l - A_{th}V_{th} - REW_l)) \tag{12}$$

where λ_v and λ_w are the learning rates respectively for the amygdala and orbitofrontal cortex, which are the key elements that bear the influence on the learning speed. REW_l is the reward signal for the l-th output. Define the l-th output error as

$$e_l = t_l - y_l \tag{13}$$

where t_l and y_l are the l-th expected target and assignment output, respective.

Then, in this study, the reward signal can be a function of the error signal and the output of the model; it is selected as:

$$REW_l = k_1 e_l + k_2 u_l \tag{14}$$

where k_1 and k_2 are both weight factors, which are adjusted respectively for the expectation of error reduction and output. In general, the value of k_1 should be larger than that of k_2, because the error of model in learning process is always smaller than the output, and these two weight factors will be automatic searched by the PSO algorithm in this design.

Define the cost function

$$E_l = \frac{1}{2} e_l^2 \tag{15}$$

Because the gradient descent method can reduce the error as quickly as possible, the adjustment of the mean and variance of Gaussian function is generated by the gradient descent algorithm for minimizing the cost function, as

$$\begin{aligned} \Delta m_{ij} &= -\lambda_m \frac{\partial E_l}{\partial m_{ij}} = -\lambda_m \frac{\partial E_l}{\partial e_l}\frac{\partial e_l}{\partial y_l}\frac{\partial y_l}{\partial u_l}\frac{\partial u_l}{\partial s_{ij}}\frac{\partial s_{ij}}{\partial m_{ij}} \\ &= \lambda_m e_l \cdot y_l \cdot (1 - y_l) \cdot \left(V_{ijl} - W_{ijl}\right) \cdot S_{ij} \cdot \frac{2(I_i - m_{ij})}{\sigma_{ij}^2} \end{aligned} \tag{16}$$

$$\begin{aligned} \Delta \sigma_{ij} &= -\lambda_\sigma \frac{\partial E_l}{\partial \sigma_{ij}} = -\lambda_\sigma \frac{\partial E_l}{\partial e_l}\frac{\partial e_l}{\partial y_l}\frac{\partial y_l}{\partial u_l}\frac{\partial u_l}{\partial s_{ij}}\frac{\partial s_{ij}}{\partial \sigma_{ij}} \\ &= \lambda_\sigma e_l \cdot y_l \cdot (1 - y_l) \cdot \left(V_{ijl} - W_{ijl}\right) \cdot S_{ij} \cdot \frac{2(I_i - m_{ij})}{\sigma_{ij}^3} \end{aligned} \tag{17}$$

where λ_m and λ_σ are the learning rates.

2.2 Particle Swarm Optimization Algorithm

2.2.1 PSO Description

Particle Swarm Optimization (PSO) was inspired by the research of birds' foraging behavior in nature: a bevy of birds look for food in a random location in an area, and each bird can know what's the distance between their current position is and the location of the food. Therefore, as long as the bird closest to the food is found, searching around it is an effective way. Consequently, the heart of the matter of finding the optimal parameters is usually solved by using the PSO algorithm. In PSO algorithm, birds in search space are replaced by particles. Each particle has its direction and speed when searching. Then, the particle adjusts its position and direction according to the current best particle position and direction and searches the solution space [39].

In the process of each iteration, the particle swarm constantly adjusts and updates its position and speed by the following formula [40]:

$$V_{od}^{k+1} = \omega V_{od}^{k} + c_1 r_1 (p_{od}^{k} - L_{od}^{k}) + c_2 r_2 (G_{od}^{k} - L_{od}^{k}) \tag{18}$$

$$L_{od}^{k+1} = L_{od}^{k} + V_{od}^{k+1} \tag{19}$$

where $d = 1,2,\cdots,D$; $o = 1,2,\cdots,N$. V_{od}^{k} is the velocity of the o-th particle in the d-th dimensional space, L_{od}^{k} is the position of the o-th particle in the d-th dimensional space; L_{od}^{k+1} and V_{od}^{k+1} represent the updated values of L_{od}^{k} and V_{od}^{k}, respectively; k represents the number of iterations; c_1 and c_2 are acceleration factors, both of which are non-negative constants. They play a role in adjusting each particle to obtain the optimal individual step size and the optimal group step size, respectively. r_1 and r_2 are random numbers in the interval [0,1]; ω is the inertia weight, which means that the particle inherits the proportion of the previous speed. In PSO algorithm, the velocity and position of particles are often restricted to the region $[V_{min}, V_{max}]$ and $[L_{min}, L_{max}]$, so as to avoid the blind search of particles in the space of feasible solutions, which results in the loss of the superiority of the algorithm itself. The region of the velocity is often adjusted according to the range of position.

2.2.2 Fitness Function

The fitness function is applied to choose the best particle, and for the classification problem, it can be evaluated by the accuracy of the classifier. The following fitness function is applied for the whole particle search

$$fitness = 1 - ACC_{train} \tag{20}$$

where ACC_{train} is the training accuracy rate in the MFBELC. That is to say, the fitness function of PSO is equal to the error of the training results in MFBLEC, PSO finds the optimal parameters k_1 and k_2 according to minimizing the fitness value. If the error is small, it means that k_1 and k_2 can achieve good training results.

2.3 PSO-MFBELC

The flowchart of PSO-MFBELC is shown in Figure 2. The specific steps to achieve this algorithm are depicted as follows.

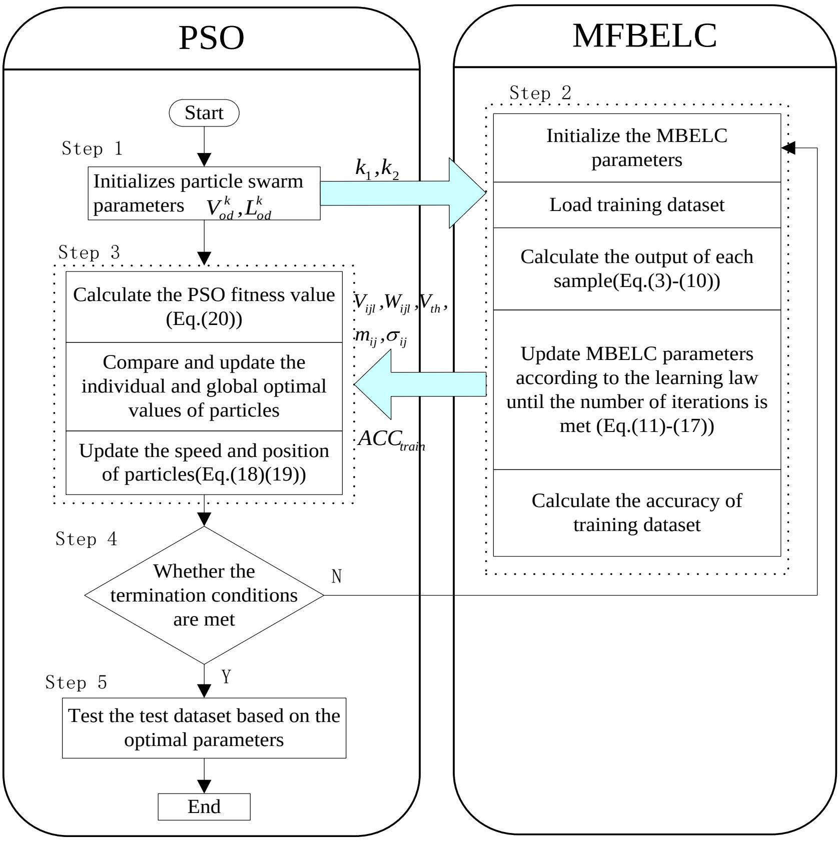

PSO
MFBELC
Start
Step 1
Initializes particle swarm parameters V_{od}^{k}, L_{od}^{k}
k_1, k_2
Step 2
Initialize the MBELC parameters
Load training dataset
Calculate the output of each sample(Eq.(3)-(10))
Update MBELC parameters according to the learning law until the number of iterations is met (Eq.(11)-(17))
Calculate the accuracy of training dataset
Step 3
Calculate the PSO fitness value (Eq.(20))
Compare and update the individual and global optimal values of particles
Update the speed and position of particles(Eq.(18)(19))
$V_{ijl}, W_{ijl}, V_{th}, m_{ij}, \sigma_{ij}$
ACC_{train}
Step 4
Whether the termination conditions are met
N
Y
Step 5
Test the test dataset based on the optimal parameters
End

Figure 2
Flowchart for a PSO-MFBELC

- Step 1. Generate the particle population for PSO, including particle swarm size, velocity vector, position, and other parameters. In this step, the initial values of k_1 and k_2, which are the parameters to be optimized, are transmitted to the MFBELC training stage.

- Step 2. Initialize the parameters of MFBELC and the training phase will be operated by the specified learning algorithms. Set the initial conditions and inputs. In general, the initial values of the parameters of MFBELC, such as $V_{ij}, W_{ij}, V_{th}, m_{ij}, \sigma_{ij}$ are chosen as random values. The features of training samples are put as the input of MFBELC, and the output are obtained by equations (3)-(10). Then, the parameters of MFBELC are modified according to the learning algorithm, by equations (11), (12), and (16), (17). When the training phase is finished, the accuracy of the training dataset is returned to PSO.

- Step 3. The fitness function is calculated by equation (20) and is assigned to each particle. The individual extreme values, global extreme values, and the speed and position of the particles are compared and updated in this step, by equations (18)-(19).

- Step 4. Determine whether the condition meets the number of iterations and the desired precision requirements. That is, the error value ≦ 0.2% or the set number of iterations 500 is achieved. If the stop condition is reached, jump out of the iteration; if not, move back to step 2.

- Step 5. Obtain the optimized parameters. The performance of the test is calculated and the effectiveness of the model is evaluated.

3 Simulation Results

In this section, to verify the performance of the proposed model, the proposed model is evaluated for two standard multiclass datasets: 1) Wine; 2) Iris, which are shared in the UCI machine learning database [40].

3.1 Description of Dataset

The description of the two datasets and training-testing proportion are shown in Table 1. There are 178 samples in the wine dataset, of which 100 samples are used for training and the other 78 samples are used for testing. Iris data set has 150 samples, of which 75 samples are used for training and the other 75 samples are used for testing.

Table 1

Description of datasets

Dataset	**No. of Sample**	**No. of training sample**	**No. of testing sample**	**No. of Cluster**	**No. of Attribute**
Wine	178	100	78	3	13
Iris	150	75	75	3	4

Before constructing the model, if the data has a high diversity among different attributes, it should be preprocessed. Such as the Wine dataset, the values of input features are normalized between the range [0, 1] first.

3.2 Experimental Results

From Figure 2, the proposed method begins with the initialization of parameters for PSO. The parameters applied for the two datasets are tabulated in Table 2, which shows the parameters and their numbers and ranges for the PSO to search k_1 and k_2. In general, the most important consideration of parameter setting is whether it can converge during the iteration. In most situations, the size of swarms, or the maximal iteration number may affect the speed of training. Besides, the parameters for the MFBELC, including m_{ij}, σ_{ij}, V_{ijl}, W_{ijl} are randomly initialized.

Table 3 shows the parameter setting for the MFBELC and the search results of PSO-MFBELC. It shows that due to the use of PSO for optimization, appropriate values of k_1 and k_2 are determined and the training epochs are much reduced. It has been demonstrated that the process of optimization is also beneficial to the learning of the MFBELC model and the burden of MFBELC could be reduced.

Furthermore, the optimal values of k_1 and k_2 for the two datasets are different. This also explains the difficulty of the selection of the two parameters and it is necessary to introduce PSO algorithm for parameter optimization.

Table 2
Parameters used for PSO

Parameter	Value	
	Wine dataset	Iris dataset
Swarm Size	20	20
Max of Generations	10	10
ω	[0.2-0.8]	[0.2-0.8]
c_1	2	2
c_2	2	2
$[V_{min}, V_{max}]$ for searching k_1	[-10,10]	[-10,10]
$[V_{min}, V_{max}]$ for searching k_2	[-1,1]	[-1,1]
$[L_{min}, L_{max}]$ for searching k_1	[0,200]	[0,1000]
$[L_{min}, L_{max}]$ for searching k_2	[-5,5]	[-5,5]

Table 3
Comparison of the parameters for different MFBELC models

Dataset	**Model**	k_1	k_2
Wine	MFBELC	150	1
	PSO-MFBELC	172.27*	0.64*
Iris	MFBELC	20	1
	PSO-MFBELC	615.68*	1.09*

"'' refers to the parameter value after PSO optimization*

The classification results of PSO-MFBELC are represented by the confusion matrix as shown in Figure 3. The green squares in the confusion matrix represent the number of correctly identified samples. For example, the number in the first row and the first column represents the sample target is class 1, and the prediction result is also class 1. The red square represents the number of samples with incorrect predictions. For example, the number in the second column of the third row indicates that the sample target is class 2, and the prediction result is that there is in class 3. In the multi-classification confusion matrix, the precision is divided by the value on the main diagonal by the sum of the row in which the value is located, and the sensitivity is equal to the value on the main diagonal divided by the sum of the column in which the value is located. Finally, the accuracy is the sum of the diagonal values.

The classification result for the PSO-MFBELC is shown in Figure 4 and Figure 5, respectively, for the two datasets, and the percentage below the number in the box represents the number divided by the total number of samples. There are two parts: one for training samples and the other for testing samples; both displayed in their respective confusion matrix. From Figure 4, on Wine dataset, in the training phase, there is one sample misclassified. Meanwhile, there are two samples misclassified in the testing phase. Therefore, the accuracy values for the training and testing dataset are 99.0% and 97.4%, respectively. Likewise, from Figure 5, on the Iris dataset, the accuracy values for training and testing are 93.3% and 98.7%, respectively.

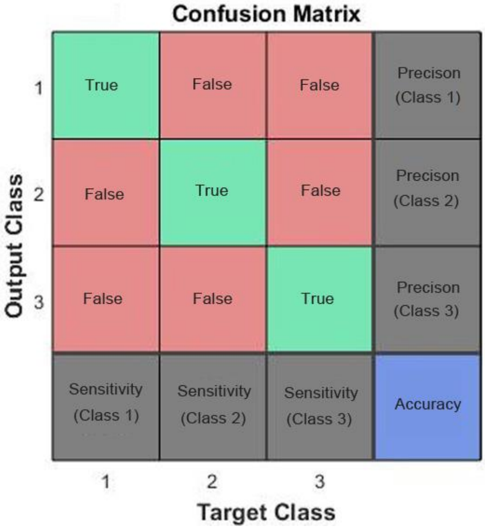
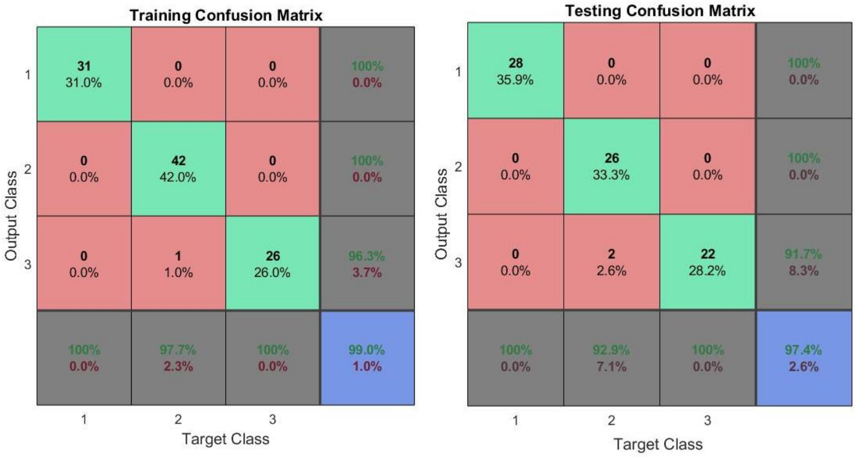
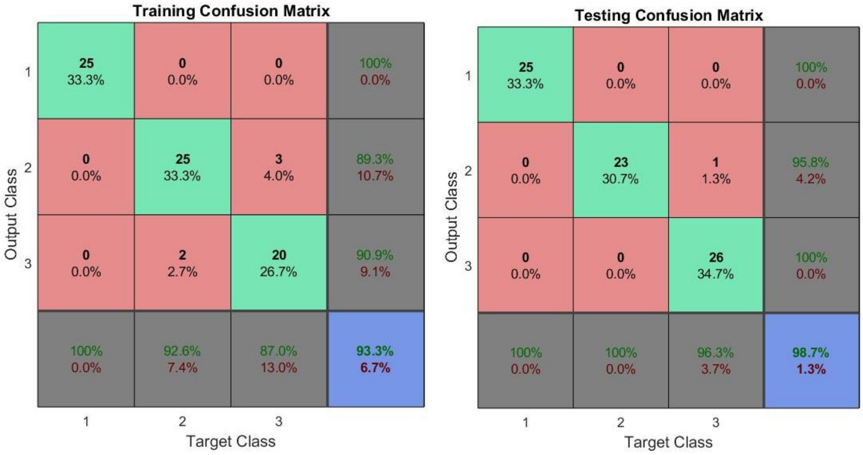


Figure 3
Confusion matrix description



(a) (b)

Figure 4
Classification result for the PSO-MFBELC on Wine dataset, (a) training phase, (b) testing phase



(a) (b)

Figure 5

Classification result for the PSO-MFBELC on Iris dataset, (a) training phase, (b) testing phase

3.3 Performance Evaluation

In order to illustrate the effectiveness of the new model, the PSO-MFBELC is compared with MFBELC without PSO. For the purpose to ensure a fair comparison, the simulations are repeated for 10 runs for both algorithms. The comparisons of testing accuracies are displayed in Figure 6 and Figure 7. Apparently, the accuracy of PSO-MFBELC is higher than that of MFBELC. Besides, the values of accuracies obtained by PSO-MFBELC are relatively more stable than that of MFBELC. Table 4 lists the training accuracy and testing accuracy of two datasets with 10 times running for the PSO-MFBELC.

Table 4

PSO-MFBELC running results (%)

Dataset	**Wine**		**Iris**	
Time	**Train Accuracy**	**Test Accuracy**	**Train**	**Test Accuracy**
1	99.00	97.44	93.33	98.67
2	98.00	98.72	93.33	98.67
3	98.00	97.44	94.67	98.67
4	99.00	97.44	93.33	97.33
5	98.00	97.44	93.33	98.67
6	98.00	96.15	93.33	98.67
7	99.00	97.44	93.33	98.67
8	98.00	97.44	94.67	98.67

9	98.00	97.44	93.33	98.67
10	98.00	96.15	93.33	98.67

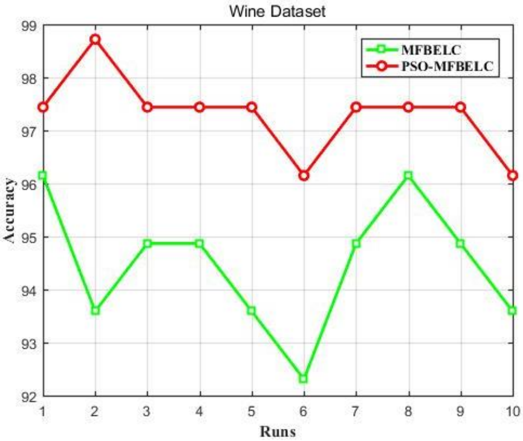


Figure 6
Comparison of testing accuracy values for different models on Wine dataset

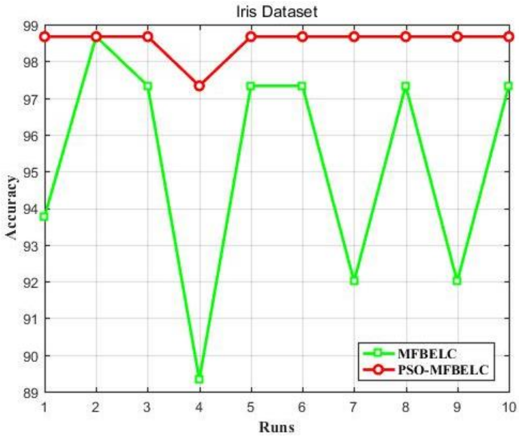


Figure 7
Comparison of testing accuracy values for different models on Iris dataset

Table 5 summarizes the results of accuracy for the two classifiers when operating for 10 times on different datasets. On the Wine dataset, the maximum training accuracy of PSO-MFBELC and MFBELC are100% and 99%, respectively; while the average values are 100% and 98.3%, respectively. The maximum testing accuracy of PSO-MFBELC and MFBELC are 98.72% and 96.15%, respectively; while the average values are 97.31% and 94.49%, respectively. On the Iris dataset, for PSO-MFBELC and MFBELC, although the maximum testing accuracy of the two algorithms is both 98.67%; however, the average values are 98.54% and 94.86%, respectively. Therefore, to summarize, the PSO-MFBELC model has better performance than the MFBELC model, which clearly indicates that taking advantage of the PSO algorithm to train the MFBELC model could enhance the performance of the brain emotional learning classifier.

Table 5

Performance comparisons (%)

Dataset	**Model**	**Train Accuracy (ACC)**			**Test Accuracy (ACC)**		
		Highest	**Lowest**	**Avg**	**Highest**	**Lowest**	**Avg**
Wine	MFBELC	100	100	100	96.15	92.23	94.49
	PSO-MFBELC	99	98	98.3	98.72	96.15	97.31
Iris	MFBELC	93.33	90.67	92.66	98.67	89.21	94.86
	PSO-MFBELC	94.67	93.33	93.6	98.67	97.33	98.54

In order to verify the proposed method, the comparison with other methods, shown in published literatures [42]-[51] which used the same data set, is summarized in Table 6. The classification accuracy of the proposed PSO-MFBELC is satisfactory compared to the other classifiers.

Table 6

Reported results in literatures

Dataset	**Author (Year)**	**Method**	**Test Accuracy (ACC)**
Wine	Hidayat et al.[42](2016)	ACO	89.90
	Chen et al. [43](2016)	DCQGA-SVM	90.4109
	Chakravarty et al. [44](2015)	CSFLNFN	93.5
	Guerrero-Enamorado et al. [45] (2016)	LOGIT-BOOST	97.07
	Xu et al. [46](2015)	FKNN+FLMDA	98.10
	Wongthanavasu et al. [47](2016)	CAC	98.18
	Xu et al. [48](2016)	Bayes net	98.59
Iris	Hidayat et al. [42](2016)	PREACO	90
	Zhang et al. [49](2015)	W-KNN	95.83
	Guerrero-Enamorado et al. [45] (2016)	MCGEP	96.53
	Xu et al. [48](2016)	DC-Core samples	96.67
	Yu et al. [50](2015)	SI-INNO + LGC	98
	Wongthanavasu et al. [47](2016)	Linear SVM	98.01
	Chakravarty et al. [44](2015)	CSMLP	98.1
	Chen et al. [43](2016)	DCQGA-SVM	98.3
	Syaliman et al. [51](2018)	LMKNN+DWKNN	98.33

Conclusion

Based on the fuzzy brain emotional learning model and PSO algorithm, this paper constructs a PSO-MFBELC for multiple classifications. PSO is used to automatically search for appropriate values of the weights of the reward signal of MFBELC, which affects the training speed and predictive accuracy of the classification model. Then, the proposed PSO-MFBELC is applied to two datasets. Numerical simulations show that the algorithm has high generalization ability and accuracy, not only the model structure is simple but also easy to implement. It is clear that this method can also be applied to other multidimensional classification problems.

Acknowledgement

This paper was supported in part by the Ministry of Science and Technology of the Republic of China under Grant MOST 106-2221-E-155-MY3.

References

[1] J. E. LeDoux, "Emotion and the limbic system concept," Concepts Neurosci, Vol. 2, pp. 169-199, 1991

[2] C. B. J. Moren, C. Balkenius, "Emotional learning:a computational model of the amygdala," Cybernetics & Systems, Vol. 32, No. 6, pp. 611-636, 2001

[3] M. A. Sharbafi, C. Lucas, R. Daneshvar, "Motion control of omni-directional three-wheel robots by brain-emotional-learning-based intelligent controller," IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), Vol. 40, No. 6, pp. 630-638, 2010

[4] P. K. Muthusamy, M. Garratt, H. Pota, et al. "Bidirectional fuzzy brain emotional learning control for aerial robots," 2018 IEEE Symposium Series on Computational Intelligence (SSCI), 2018, pp. 146-153

[5] Q. Wu, C. M. Lin, W. Fang, et al. "Self-organizing brain emotional learning controller network for intelligent control system of mobile robots," IEEE Access, pp. 59096-59108, 2018

[6] M. Jafari, R. Fehr, L. R. G. Carrillo, H. Xu, "Brain emotional learning-based intelligent tracking control for unmanned aircraft systems with uncertain system dynamics and disturbance," 2017 International conference on unmanned aircraft systems (ICUAS), IEEE, 2017, pp. 1470-1475

[7] M. Jafari, A. M. Shahri, S. B. Shuraki, "Speed control of a digital servo system using brain emotional learning based intelligent controller," Power Electronics, Drive Systems and Technologies Conference (PEDSTC), 2013 4th, IEEE, 2013, pp. 311-314

[8] C. F. Hsu, C. T. Su, T. T. Lee, "Chaos synchronization using brain-emotional-learning-based fuzzy control," Joint International Conference on

Soft Computing & Intelligent Systems, IEEE, 2016, pp. 811-816

[9] M. Parsapoor, U. Bilstrup, "Chaotic time series prediction using brain emotional learning based recurrent fuzzy system (BELRFS)," International Journal of Reasoning-Based Intelligent Systems, Vol. 5, No. 2, pp. 113-126, 2013

[10] E. Lotfi, M. R. Akbarzadeht, "Adaptive brain emotional decayed learning for online prediction of geomagnetic activity indices," Neurocomputing, Vol. 126, No. 3, pp. 188-196, 2014

[11] H. S. A. Milad, U. Farooq, M. E. El-Hawary, et al. "Fuzzy logic based parameter adjustment model for adaptive decayed brain emotional learning network with application to online time series prediction," 2017 IEEE Electrical Power and Energy Conference (EPEC), IEEE, 2017, pp. 1-6

[12] J. Ayubi, A. Omidi, S. M. Barakati, P. Ayubi, "Short term load forecasting based on brain emotional predictor," 2015 20th Conference on Electrical Power Distribution Networks Conference (EPDC), IEEE, 2015, pp. 37-41

[13] E. Lotfi, S. Setayeshi, S. Taimory, "A neural basis computational model of emotional brain for online visual object recognition," Applied Artificial Intelligent, Vol. 28, No. 8, pp. 814-834, 2014

[14] S. Motamed, S. Setayeshi, A. Rabiee, "Speech emotion recognition based on a modified brain emotional learning model," Biologically Inspired Cognitive Architectures, Vol. 19, pp. 32-38, 2017

[15] E. Lotfi, "Mathematical modeling of emotional brain for classification problems," Proceedings of IAM, Vol. 2, No. 1, pp. 60-71, 2013

[16] M. Parsapoor, U. Bilstrup, "Brain emotional learning based fuzzy inference system (Modified using radial basis function)," Eighth International Conference on Digital Information Management, IEEE, 2014, pp. 206-211

[17] Q. Q. Zhou, F. Chao, C. M. Lin, "A functional-link-based fuzzy brain emotional learning network for breast tumor classification and chaotic system synchronization," International Journal of Fuzzy Systems,Vol. 20, No. 2, pp. 349-365, 2018

[18] Y. Sun, C. M. Lin, "A fuzzy brain emotional learning classifier design and application in medical diagnosis," Acta Polytechnica Hungarica, Vol. 16, No. 4, pp. 27-43, 2019

[19] M. Asad, U. Farooq, J. Gu, et al. "Neo-fuzzy supported brain emotional learning based pattern recognizer for classification problems," IEEE Access, 2017, pp. 6951-6968

[20] J. Zhao, C. M. Lin, "Multidimensional classifier design using wavelet fuzzy brain emotional learning neural networks," Journal of Intelligent & Fuzzy Systems, Vol. 36, No. 2, pp. 1099-1107, 2019

[21] R. E. Precup, R. C. David, E. M. Petriu, et al. "Grey wolf optimizer-based approach to the tuning of PI-fuzzy controllers with a reduced process parametric sensitivity," IFAC Papers OnLine, Vol. 49, No. 5, pp. 55-60, 2016

[22] S. Preitl, R. E. Precup, Z. Preitl, et al. "Iterative feedback and learning control. Servo systems applications," IFAC Proceedings, Vol. 40, No. 8, pp. 16-27, 2007

[23] U. Yun. "Efficient mining of weighted interesting patterns with a strong weight and/or support affinity," Information Sciences, Vol. 177, No. 17, pp. 3477-3499, 2007

[24] J. Kennedy, R. C. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95-International Conference on Neural Networks. IEEE, 1995, pp. 1942-1948

[25] D. R. Nayak, R. Dash, B. Majhi, "Discrete ripplet-II transform and modified PSO based improved evolutionary extreme learning machine for pathological brain detection," Neurocomputing, Vol. 282, pp. 232-247, 2018

[26] S. K. Satapathy, S. Dehuri, A. K. Jagadev, "EEG signal classification using PSO trained RBF neural network for epilepsy identification," Informatics in Medicine Unlocked, Vol. 6, pp. 1-11, 2017

[27] S. Saraswathi, S. Sundaram, N. Sundararajan, et al. "ICGA-PSO-ELM approach for accurate multiclass cancer classification resulting in reduced gene sets in which genes encoding secreted proteins are highly represented," IEEE/ACM Transactions on Computational Biology and Bioinformatics, Vol. 8, No. 2, pp. 452-463, 2010

[28] C. M. Lin, T. L. Le, "WCMAC-based control system design for nonlinear systemsusing PSO," Journal of Intelligent & Fuzzy Systems, Vol. 33, No. 2, pp. 807-818, 2017

[29] R. J. Wai, J. D. Lee, K. L. Chuang, "Real-time PID control strategy for maglev transportation system via particle swarm optimization," IEEE Transactions on Industrial Electronics, Vol. 58, No. 2, pp. 629-646, 2010

[30] H. T. Dorrah, A. M. El-Garhy, M. E. El-Shimy, "PSO-BELBIC scheme for two-coupled distillation column process," Journal of Advanced Research, Vol. 2, No. 1, pp. 73-83, 2011

[31] T. L. Le, C. M. Lin, T. T. Huynh, "Self-evolving type-2 fuzzy brain emotional learning control design for chaotic systems using PSO," Applied Soft Computing Journal, Vol. 73, pp. 418-433, 2018

[32] Y. Mei, X. Yang, Z. Liu, et al. "Real-time facial expression recognition based on the improved brain emotional learning model," 2018 37th Chinese Control Conference (CCC), IEEE, 2018, pp. 3924-3928

[33] S. H. Fakhrmoosavy, S. Setayeshi, A. Sharifi, "An intelligent method for generating artificial earthquake records based on hybrid PSO–parallel brain emotional learning inspired model," Engineering with Computers, Vol. 34, No. 3, pp. 449-463, 2018

[34] C. A. Laurentys, C. H. M. Bomfim, B. R. Menezes, et al. "Design of a pipeline leakage detection using expert system: A novel approach," Applied Soft Computing, Vol. 11, No. 1, pp. 1057-1066, 2011

[35] C. Pozna, R. E. Precup. "Applications of signatures to expert systems modelling," Acta Polytechnica Hungarica, Vol. 11, No. 2, pp. 21-39, 2014

[36] Zall R, Mohammad Reza K, "On the construction of multi-relational classifier based on canonical correlation analysis," International Journal on Artificial Intelligence Tools, Vol. 17, No. 2, pp. 23-43, 2019

[37] E. L. Hedrea, R. E. Precup, C. A. Bojan-Dragos. "Results on tensor product-based model transformation of magnetic levitation systems," Acta Polytechnica Hungarica, Vol. 16, No. 9, pp. 93-111, 2019

[38] C. Lucas, D. Shahmirzadi, N. Sheikholeslami, "Introducing BELBIC: brain emotional learning based intelligent controller," Intelligent Automation & Soft Computing, Vol. 10, No. 1, pp. 11-21, 2004

[39] E. A. Grimaldi, F. Grimaccia, M. Mussetta, et al. "PSO as an effective learning algorithm for neural network applications," 2004 3rd International Conference on Computational Electromagnetics and Its Applications, IEEE, 2004, pp. 557-560

[40] Y. Shi, R. Eberhart, "A modified particle swarm optimizer," 1998 IEEE international conference on evolutionary computation proceedings, IEEE world congress on computational intelligence, IEEE, 1998, pp. 69-73

[41] UCI repository of machine learning databases. Available at: http://archive.ics.uci.edu/ml/datasets/

[42] D. T. Hidayat, C. Fatichah, V. Raden, "Pattern reduction enhanced ant colony optimization clustering algorithm," International Seminar on Application for Technology of Information & Communication, IEEE, 2016, pp. 317-322

[43] P. Chen, L. Yuan, Y. He, et al. "An improved SVM classifier based on double chains quantum genetic algorithm and its application in analogue circuit diagnosis," Neurocomputing, Vol. 211, No. 26, pp. 202-211, 2016

[44] S. Chakravarty, P. Mohapatra, "Multi-class classification using cuckoo search based hybrid network," Power, Communication & Information Technology Conference, IEEE, 2015, pp. 1-8

[45] A. Guerrero-Enamorado, C. Morell, A. Y. Noaman, et al. "An algorithm evaluation for discovering classification rules with gene expression programming," International Journal of Computational Intelligence

Systems, Vol. 9, No. 2, pp. 263-280, 2016

[46] J. Xu, Z. Gu, K. Xie, "Fuzzy local mean discriminant analysis for dimensionality reduction," Neural Processing Letters, Vol. 44, No. 3, pp. 1-18, 2015

[47] S. Wongthanavasu, J. Ponkaew, "A cellular automata-based learning method for classification," Expert Systems with Applications, Vol. 49, pp. 99-111, 2016

[48] X. Xu, J. Zheng, J. Yang, et al. "Data classification using evidence reasoning rule," Knowledge Based Systems, Vol. 21, No. 8, pp. 1-8, 2016

[49] L. Zhang, C. Zhang, Q. Xu, et al. "Weigted-KNN and its application on UCI," IEEE International Conference on Information & Automation, IEEE, 2015, pp. 1748-1750

[50] C. Yu, F. Li, G. Li, et al. "Multi-classes imbalanced dataset classification based on sample information," 2015 IEEE 17th International Conference on High Performance Computing and Communications (HPCC), IEEE, 2015, pp. 1768-1773

[51] K. U. Syaliman, E. B. Nababan, O. S. Sitompul, "Improving the accuracy of k-nearest neighbor using local mean based and distance weight," Journal of Physics: Conference Series, 2018, pp. 1-6