Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 7, No. 3, August 2016), Pages: 1-14
www.jacr.iausari.ac.ir

Multi-layer Perceptron Neural Network Training Based on Improved of Stud GA

Firozeh Razavi[1], Faramarz Zabihi[✉2], Mirsaeid Hosseini Shirvani[2]

1) Department of Management and Economics, Science and Research Branch, Islamic Azad University, Tehran, Iran

2) Department of Computer Engineering, Sari branch, Islamic Azad University, Sari, Iran

firozeh.razavi@srbiau.ac.ir; faramarz.zabihi@qiau.ac.ir; mirsaeid_hosseini@iausari.ac.ir

Received: 2014/05/29; Accepted: 2015/08/31

Abstract

Neural network is one of the most widely used algorithms in the field of machine learning, On the other hand, neural network training is a complicated and important process. Supervised learning needs to be organized to reach the goal as soon as possible. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. Hence, in this paper, it is attempted to use improve Stud GA to find optimal weights for multi-layer Perceptron neural network. Stud GA is improved from genetic algorithms that perform information sharing in a particular way. In this study, chaotic system will be used to improve Stud GA. The comparison of proposed method with Imperialist Competitive Algorithm, Quad Countries Algorithm, Stud GA, Cuckoo Optimization Algorithm and Chaotic Cuckoo Optimization Algorithm on tested data set (Wine, Abalone, Iris, WDBC, PIMA and Glass) with determined parameters, as mentioned the proposed method has a better performance.

Keywords: *Genetic Algorithms, Chaos Theory, Artificial Neural Networks, Evolutionary–Neural Network, Stud GA*

1. Introduction

Due to learning capability, generalization ability and robustness, neural networks are widely used in the field of machine learning. Neural networks have expanded considerably in fields such as classification and diagnosis of patterns, signal processing, time series forecasting, modeling, control, optimization of expert and fuzzy systems, financing, insurance, construction of industrial facilities and transportation[1]. Adjusting the weights among neurons in different layers is known as training of neural network and is one of the most important issues in neural network. To train the neural network, a set called training collection is given to it by witch it can achieve the desired output. When the neural network architecture is determined, the stage of adjusting neural network weights is continued till neural network output is close to the desired output. Then the test set is used to determine the level of generalization of neural network[2]. Since adjusting the weights and the structure of the neural network was of great importance, the neural-evolutionary network emerged [3] and plenty of evolutionary algorithms have been used for this. Some of the latest used algorithms include genetic algorithms [4], particle swarm [5], ant colonies[6], honey bee colonies [7], imperialist competitive algorithm [8] and harmony search algorithm [9]. Neural network weight determination by most of evolutionary algorithms requires more time than gradient-based training .In the second part of this article, an overview of related work in this area is discussed. In section 3 we will introduce the Stud GA. In Section 4, the proposed training algorithm for finding the optimal weights of multi-layer perceptron is presented. In Section 5, the experimental results of the proposed methods on public datasets (Wine, Abalone Iris, WDBC, PIMA and Glass) are presented and they will be compared with the results of the imperialist competitive algorithm, quad countries algorithm, Stud GA, Cuckoo optimization algorithm and Chaotic Cuckoo optimization algorithm. And finally section 6 deals with conclusion and future works.

2. Related Work

In this section, some evolutionary algorithms which are used for weight determination application and determination of neural network topology are reviewed. In [10] honey bee colonies were used to find optimized weights of back propagation neural network and the proposed method was tested on several public data sets. In [11] the back-propagation neural network was used to identify the characters of license plate of automobiles and genetic algorithm was used to determining the optimal weights of the neural network. Yaghini et al [12] used a combination of three methods and the development of particle swarm algorithm for training of neural network and the results of experiments were compared with public data sets. Sheikhan et al [13] used the combination of genetic algorithm and ant colonies algorithm to extract the characteristics of Perceptron neural network in forecasting the load of power grid (which is a time series issue). In [14] a developed imperialist competitive algorithm called chaotic imperialist competitive algorithm was used to determine the weights of neural network for Tehran Stock Exchange data set. And better results were obtained comparing with genetic algorithm, particle swarm and imperialist competitive algorithm. In [8] imperialist competitive algorithm was used to determine the weights of neural network weights and the results for some public data sets were compared with genetic algorithm. ICA starts with an initial random population which is called country .some of the best countries in the population selected to be the imperialists and the rest form the colonies of these imperialists. Imperialistic competition among these empires forms the proposed evolutionary algorithm. During this competition, weak empires collapse and powerful empires take possession of their colonies. Imperialistic competition converges to a state in which there exists only one empire and colonies have the same cost function value as the imperialist. And also, Quad Countries Algorithm, QCA, that is improvement of ICA for neural network training has been applied in [15]. In quad countries algorithm, independence and seeking independence countries are added into existing country set of imperialist competitive algorithm. Adding these countries makes it have extensible area to search and subsequently to attain appropriate responses for training of neural network. And published paper in [16] uses Cuckoo optimization algorithm to improve multi-layer Perceptron neural network training and applies UCI dataset for test improved Perceptron neural network. Also, this algorithm launches with initial population of cuckoos like other evolutionary algorithms. Initial cuckoos lay eggs in some host bird holes. The eggs with the most similarities with host eggs benefit most chance to survive and to rise and the eggs with low similarities with host eggs will be recognized and subsequently ruined. Much rising eggs in specific habitat is then much utilization of that habitat is and COA's objective is to maximize the habitat utilization. Cuckoos quest for utilized habitat with the most survival rate to lay eggs. After the eggs rise and turn to mature cuckoos, they construct societies and migrate into the best habitat for laying eggs and then each cuckoo lays egg in random holes. This process will be repeated until it reaches the maximum utilization and the most number of cuckoo populations are gathered in specific area. F. razavi et al. in [17]have suggested chaotic cuckoo optimization algorithm in which ameliorates existing cuckoo optimization algorithm. In this approach in main body of algorithm after each iteration, cuckoo optimizer selects the best found positions and by applying chaotic optimizing along with chaotic coefficient breeds some re-produce from optimal solution. Therefore, new responses that are similar to the best found solution will be replaced with the worst fitness between individual populations. So, exact local search will be gained circa the optimal solution. Stud GA is a method for sharing information in genetic algorithm. This method by crossover operation between the best individual and everyone else cause individuals in each generation benefit the information of the best individual of the previous generation. So this algorithm was improved in this article to be able to use it for neural network training.

3. Basic Algorithms

3.1. Standard Genetic Algorithm

Genetic Algorithm is a basic algorithm in the area of optimizing algorithm, which was presented by Mr. Holland [18]. In this algorithm, first an initial population is randomly generated in the atmosphere of the problem. Then using Darwin`s principal of survival of the fittest, individual who have less propriety are removed. Repeating of this step causes that responses with best fitness remain. The main three operators here are crossover, selection and mutation [19]. After encoding of chromosome (depending on the problem), genetic algorithm generate the initial population of individuals (responses). The fitness of each individual is calculated and a number of individuals are selected for reproduction. Individuals with higher fitness will have more chance to reproduce. After selecting parents, the operation of intersection will be performed among them and children will be produced. After intersection operation, mutation operation will be performed on children and produced offspring are added to the current population. But due to the limited number of individuals in each generation, people who have less fitness are eliminated from the population. An example of intersection and mutation operations can be seen in Figure 1. Intersection, mutation and selection operations will be repeated until converging to an optimal solution.

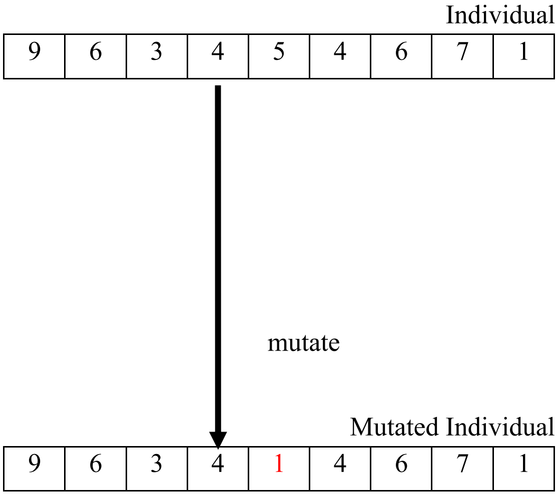

Figure 1. illustration of mutation

Figure 2. illustration of crossover

3.2. Stud GA

Using the best individual as one of the parents for reproduction in each generation of genetic algorithm is the main idea of Stud GA. In this method, parents are not chosen randomly and crossover operator is performed between the best individual and all the other individuals. In fact the intersection operator is the key operator in this method [20]. In each generation the best individual is chosen from the population and does the crossover operator with everyone else (crossover operator is done when the similarity of the two selected individuals is not less than a certain degree). In this way, the information in genetic algorithm is shared [21]. If the similarity of two selected individual is higher than a certain degree, the individual selected for crossover operator by stud will change (mutate). Stud GA algorithm flowchart is shown in Figure 3.

4. The Proposed Method

4.1. The proposed chaotic Stud GA

To improve efficiency of Stud GA, local chaotic search can be used. Logistic map is one of the well-known chaotic maps; this chaotic map shows that complex behaviors can use simple deterministic dynamical systems [22-24]. In this article we will use chaos map of Logistic map, whose relationship is as follows:

$$x_{k+1} = \alpha \times x_k (1 - x_k) \tag{1}$$

In the above equation, x_k is k_{th} chaotic number and k denoting the iteration number. Obviously, if $x_0 \in (0,\ 1)$ and $x_0 \notin \{0,\ 0.25,\ 0.75,\ 1\}$, then $x \in (0,\ 1)$.

In the application of neural network training α is considered 3.8. A sample of logistic map output with a parameter adjusted for the article is shown in figure 4.

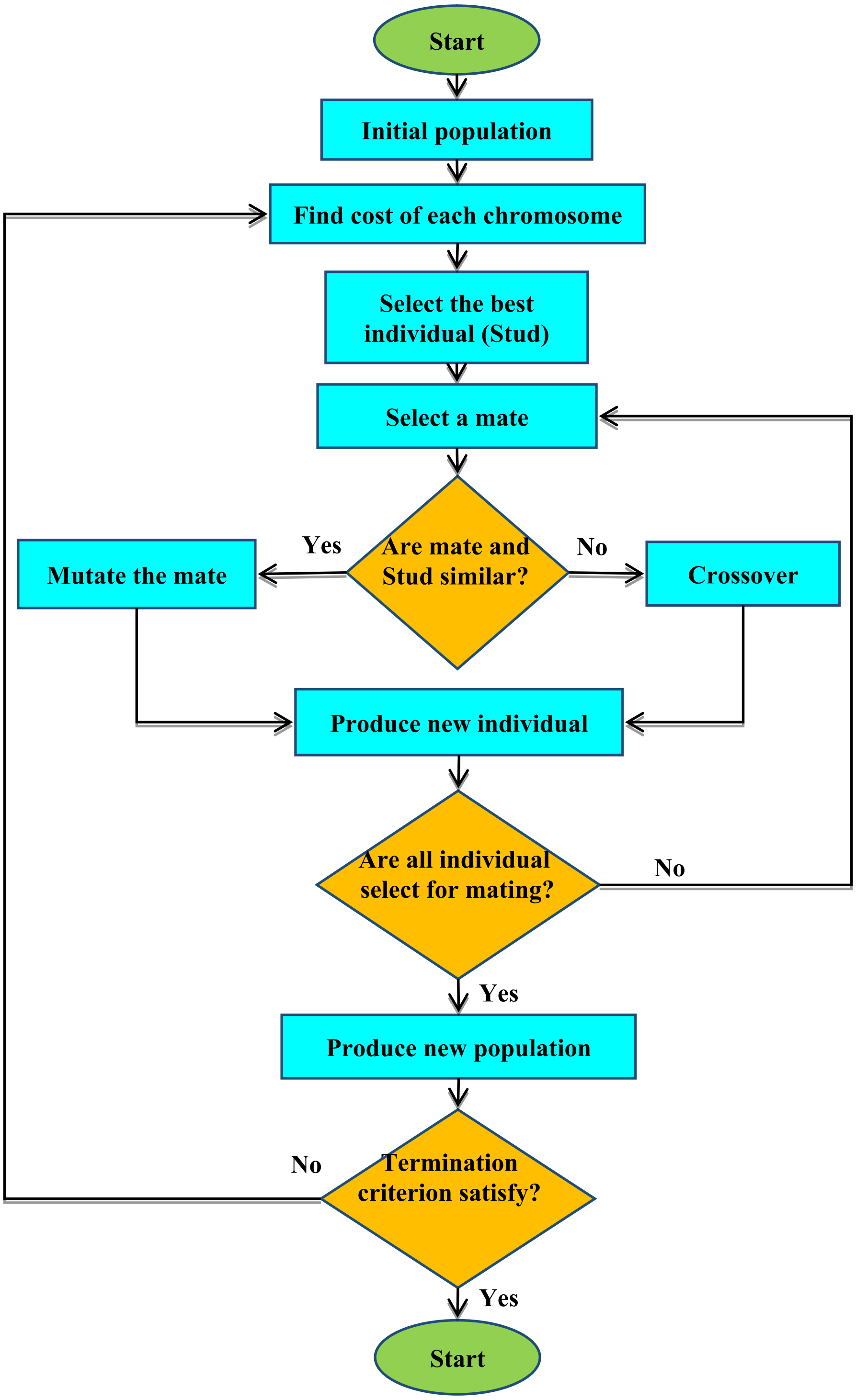

Figure 3. Flowchart of the Stud GA algorithm

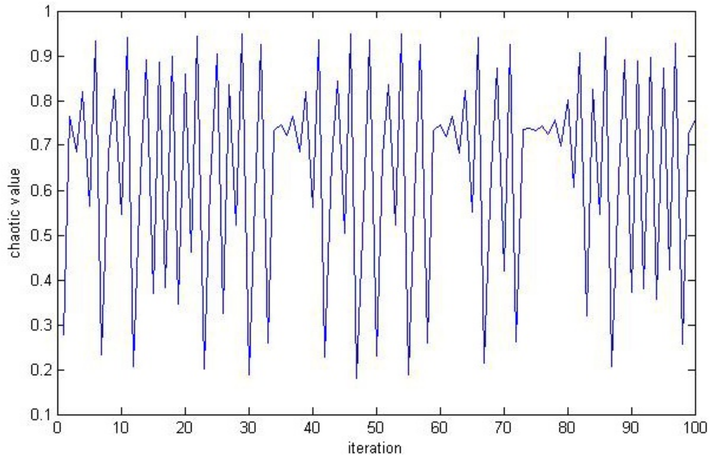

Figure 4: A sample of Logistic map output

To perform a local search around the best response, first the best individual based on the usefulness will be selected. Then individuals are generated by chaotic system surrounding the best individual and they replace 10% of the existing individuals (individuals with little benefit will be removed.)

The proposed algorithm flowchart is shown in Figure 5.

4.2. The proposed evolutionary–neural network

In this section we will explain the proposed method for determination of the most appropriate weights of neural network. To get the best weight of neural network, first the best neural network structure can be determined experimentally. After determining the structure of the neural network, genetic algorithm parameters are quantified. Bias and the number of connections between the layers of the neural network are established, the number of bias and connections between the layers of the neural network are established as the length of each individual chromosome. Random quantification is done for each individual so that the initial population of individuals is formed. In the current stage the best individual in the population as one of the parents will be taken into account to produce for all individual in next generation and new population is produced by cross over operation on all of individuals in population to produce the best individual for current generation and then mutation operation will be applied. Fitness function will be done on produced individuals and then the bests will be survived. The best individual is identified and new individuals are generated by chaotic system and will replace with %10 of worst individuals. Algorithm execution and producing individuals continues several generations by chaotic system. In the end, the best one is selected as the best weight for neural networks. Bias and the weights of neural network layers are quantified with values of the superior individual then network is evaluated by test data. Cost function calculates the mean square error as the utility of each individual in the genetic algorithm.

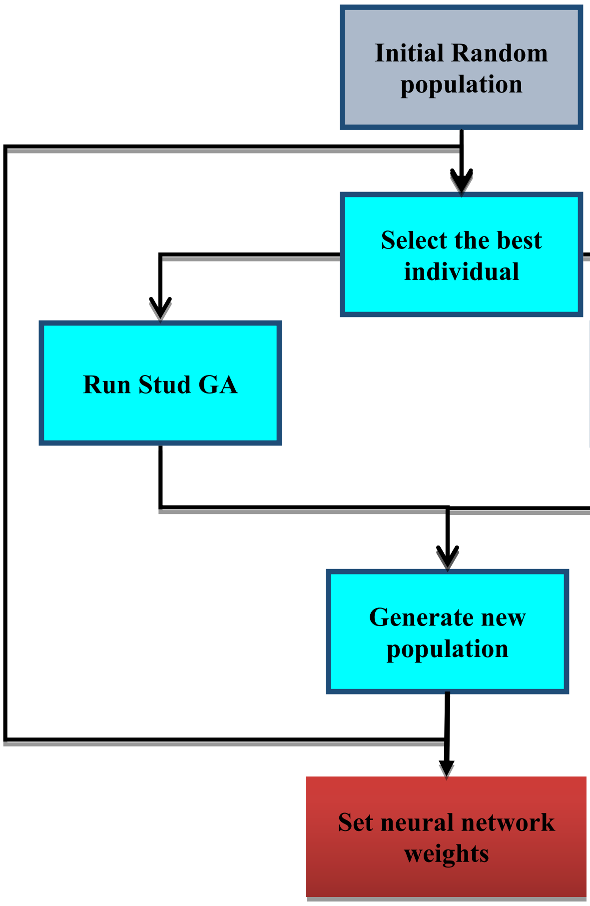

Figure 5. Proposed algorithm flowchart

The more appropriate individuals have less mean square errors. Formula 2 states how to calculate the mean square error for each individual.

$$\text{Fitness}(\text{Individual}) = \text{MSE} = [\frac{1}{N} \times \sum_{i=1}^{N} (y_{real} - y_{net})^2] \qquad (2)$$

y_{real} is the expected output from neural network, y_{net} is the actual output of neural network, N is the number of training samples and individual denote the solution which its mean square error should be computed. In Stud GA-ANN and our proposed algorithm, standard 2-point cross over operation and uniform mutation operation were applied. Since one of the parents is definitely the best in the generation, there is no selection operation and the best of each generation has natality with other population.As can be seen in figure 6, for the sake of next step population production each individual must be considered with the best individual for crossover operation till making new population and then mutation operation will be done on new population and generation chromosomes will be modified afterward fitness function will be calculated for new generation and finally several individuals with better fitness will be survived and individuals with the worst fitness will be omitted from population. After producing next generation, number of individuals will be produced by crossover between the best individual and generated population with chaotic seri and will be replaced with 10% of the worst individuals.

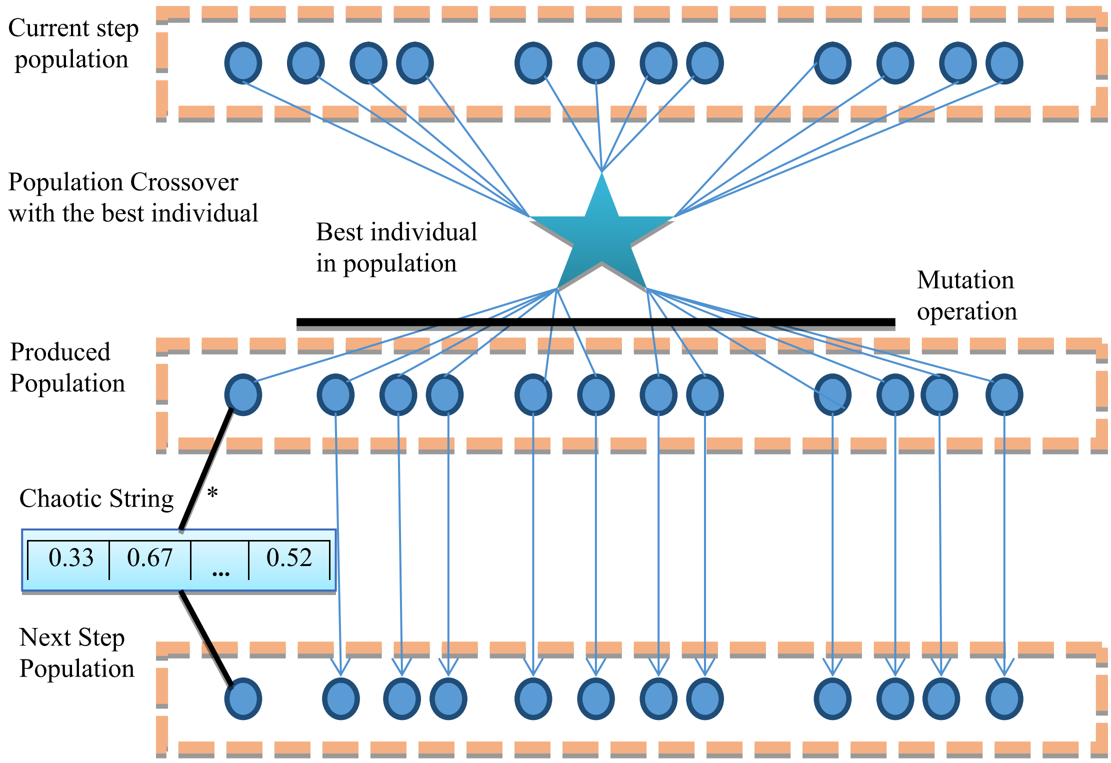

Figure6: Presentation of next step population production

5. Experimental Results

To perform the test, values of implementation methods parameters were experimentally determined and shown in tables 1, 2, 3 and 4. Topology of hidden layer neural network's is experimentally determined and displayed in Table 5. Algorithm Convergence Graph for each data set in 100 iterations will be presented next along with an explanation for each set of data. The procedure of the algorithm simulation is implemented in Matlab 2013, and is executed by the computer with 3.2 GHZ CPU, 4 GB memory, and operation system of Windows 7.

Table1. Values of cuckoo optimization algorithm parameter and chaotic cuckoo optimization algorithm

Algorithm	Maximum Number of cuckoos	KNN cluster Number	Maximum Number of Eggs	Minimum Number of Eggs	Number of Cuckoos
COA	50	2	6	2	30
CCOA	50	2	6	2	30

Table 2. values of quad countries algorithm parameters

Algorithm	δ	eta	ieta	No of seeking independence Countries	No of independence Countries	No of Colony	No of Initial Imperialists
QCA	0.01	0.01	0.05	10	10	100	10

Table 3. values of imperialist competition algorithm parameter

Algorithm	Assimilation Angle Coefficient (γ)	Assimilation Coefficient (β)	No of Decades	Zeta	Revolution Rate	No of Countries	No of Initial Imperialists
ICA	0.5	2	100	0.02	0.2	200	20

Table 4. values of Genetic algorithm parameter

Algorithm	Population	Generation	Mutation rate	crossover rate	minimum diversity
Stud GA	150	100	0.001	0.9	10%
Proposed GA	150	100	0.001	0.9	10%

Table 5. Neural network hidden layer typology

Data set	Abalone	Wine	Glass	Iris
Best Topology	8-2-2	7-4-3	12-5-6	7-4-3

5.1. An Introduction of Datasets

The data set used to test the proposed method is as follows:

Iris Dataset: this data set contains 3classes and 150 samples and each sample has four features. In Figure 6 the convergence trend for the proposed method is shown on the Iris data set.

Wine Dataset: this data set contains 3classes and 178 samples and each sample has 13 features. In Figure 7 the convergence trend for the proposed method is shown on the Wine data set.

Glass Dataset: this data set contains 6classes and 214samples and each sample has 10 features. In Figure 8 the convergence trend for the proposed method is shown on the Glass data set.

Data set Abalone: this data set consists of 4 classes, 4177 samples and 8 characteristics for each sample. Figure 9 shows the divergence trend of the proposed method on data set Abalone.

The data sets used to test the proposed method in reference [25] are available. In implementations, 60% of samples are used for training and 40% of them are used to test the neural network.

5.2. Test results and comparisons

In this section, the proposed method`s convergence graph for each of the three data sets and the results of implementing of proposed method and four other algorithms can be seen. Test results of the proposed method, the imperialist competitive algorithm, the quad countries algorithm, Stud GA and Cuckoo optimization algorithm for the three introduced data sets are shown which shows that the proposed method has better results than the other four algorithms. The results in table 3 are the mean of 10 times of performing the experiments.

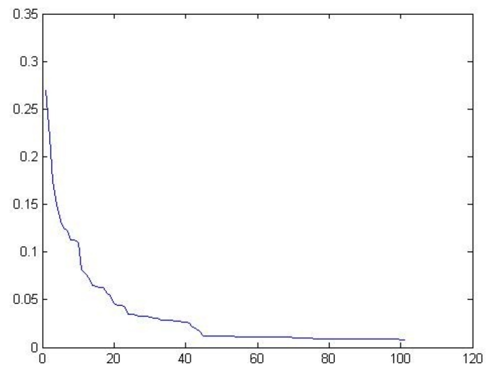

Figure 8. A sample convergence trend for Wine data sets for propose method

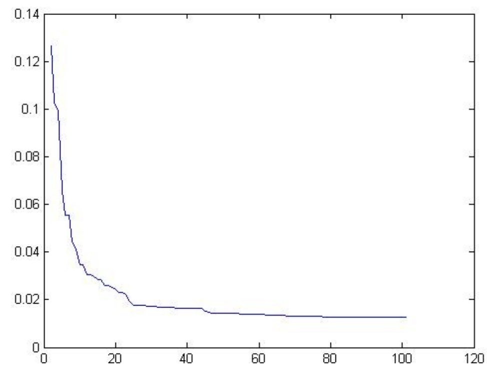

Figure 7. A sample of convergence trend for Iris data sets for propose method

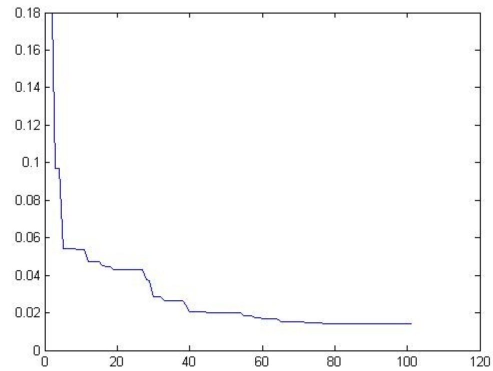

Figure 9. A sample convergence trend for Glass data sets for propose method

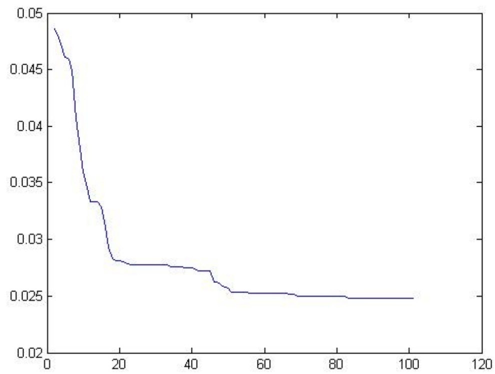

Figure 10. A sample convergence trend for Abalone data sets for propose method

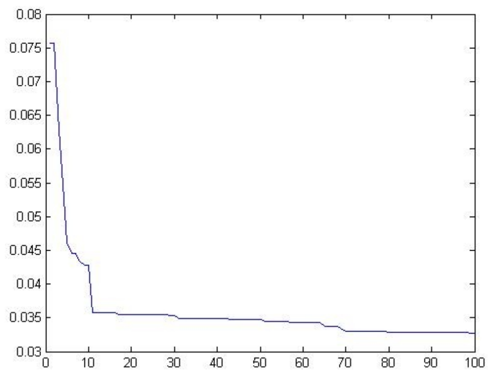

Figure 11. A sample convergence trend for WDBC data sets for propose method

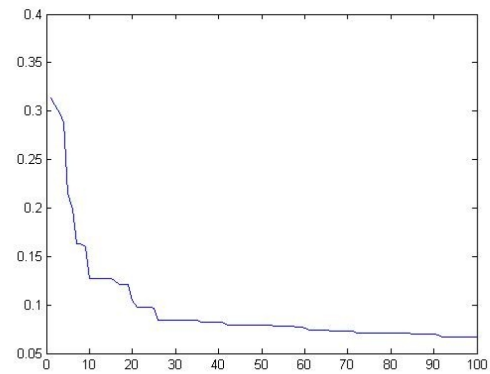

Figure 12. A sample convergence trend for PIMA data sets for propose method

Table 6. The Comparison of results for Wine Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
Wine	ICA-ANN[8]	0.0393	0.0399	0.9670	0.9674
Wine	QCA-ANN[15]	0.0204	0.0233	0.9829	0.9812
Wine	COA-ANN[16]	0.0062	0.0311	0.9986	0.9735
Wine	CCOA-ANN[17]	0.0027	0.0138	0.9994	0.9893
Wine	Stud GA-ANN	0.0032	0.0234	0.9989	0.9830
Wine	**Proposed method**	**9.5755e-04**	**0.0051**	**0.9994**	**0.9962**

Table 7. The Comparison of results for the Iris Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
Iris	ICA-ANN[8]	0.0292	0.0307	0.9787	0.9771
Iris	QCA-ANN[15]	0.0141	0.0224	0.9895	0.9822
Iris	COA-ANN[16]	0.0402	0.0209	0.9847	0.9879
Iris	CCOA-ANN[17]	0.0121	0.0288	0.9922	0.9793
Iris	Stud GA-ANN	0.0175	0.0026	0.9868	0.9981
Iris	**Proposed method**	**0.0100**	**0.0354**	**0.9927**	**0.9749**

Table 8. The Comparison of results for the Glass Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
Glass	ICA-ANN[8]	0.0277	0.0310	0.9730	0.9596
Glass	QCA-ANN[15]	0.0248	0.0223	0.9743	0.9777
Glass	COA-ANN[16]	0.0320	0.0398	0.9799	0.9724
Glass	**CCOA-ANN[17]**	**0.0067**	**0.0227**	**0.9945**	**0.9604**
Glass	Stud GA-ANN	0.0148	0.0232	0.9850	0.9773
Glass	Proposed method	0.0094	0.0278	0.9900	0.9765

Table 9. The Comparison of results for the Abalone Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
Abalone	ICA-ANN[8]	0.0335	0.0385	0.6182	0.5956
Abalone	QCA-ANN[15]	0.0280	0.0288	0.6865	0.6936
Abalone	COA-ANN[16]	0.0249	0.0277	0.7169	0.7233
Abalone	**CCOA-ANN[17]**	**0.0238**	**0.0248**	**0.7423**	**0.7319**
Abalone	Stud GA-ANN	0.0275	0.0291	0.6898	0.6862
Abalone	Proposed method	0.0257	0.0265	0.7246	0.7069

Table 10. The Comparison of results for the WDBC Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
WDBC	ICA-ANN[8]	0.0498	0.0509	0.9673	0.9649
WDBC	QCA-ANN[15]	0.0324	0.0409	0.9655	0.9615
WDBC	COA-ANN[16]	0.0594	0.0655	0.9245	0.9655
WDBC	CCOA-ANN[17]	0.0330	0.0472	0.9741	0.9478
WDBC	Stud GA-ANN	0.0365	0.0479	0.9617	0.9523
WDBC	**Proposed method**	**0.0324**	**0.0413**	**0.9655**	**0.9687**

Table 11. The Comparison of results for the PIMA Dataset

Data set	Algorithm	MSE (Train)	MSE (Test)	Corrected Classified (train)	Corrected Classified (test)
PIMA	ICA-ANN[8]	0.0743	0.0767	0.7728	0.7272
PIMA	QCA-ANN[15]	0.0704	0.0823	0.9513	0.9552
PIMA	COA-ANN[16]	0.0756	0.0895	0.9456	0.9527
PIMA	CCOA-ANN[17]	0.0651	0.0732	0.9500	0.9601
PIMA	Stud GA-ANN	0.0667	0.0865	0.9536	0.9246
PIMA	**Proposed method**	**0.0597**	**0.0619**	**0.9466**	**0.9478**

Figure 11 show the Mean square error for Wine, Iris and Glass dataset with ICA-ANN, QCA-ANN, CCOA-ANN, COA-ANN, Stud GA-ANN and proposed method.

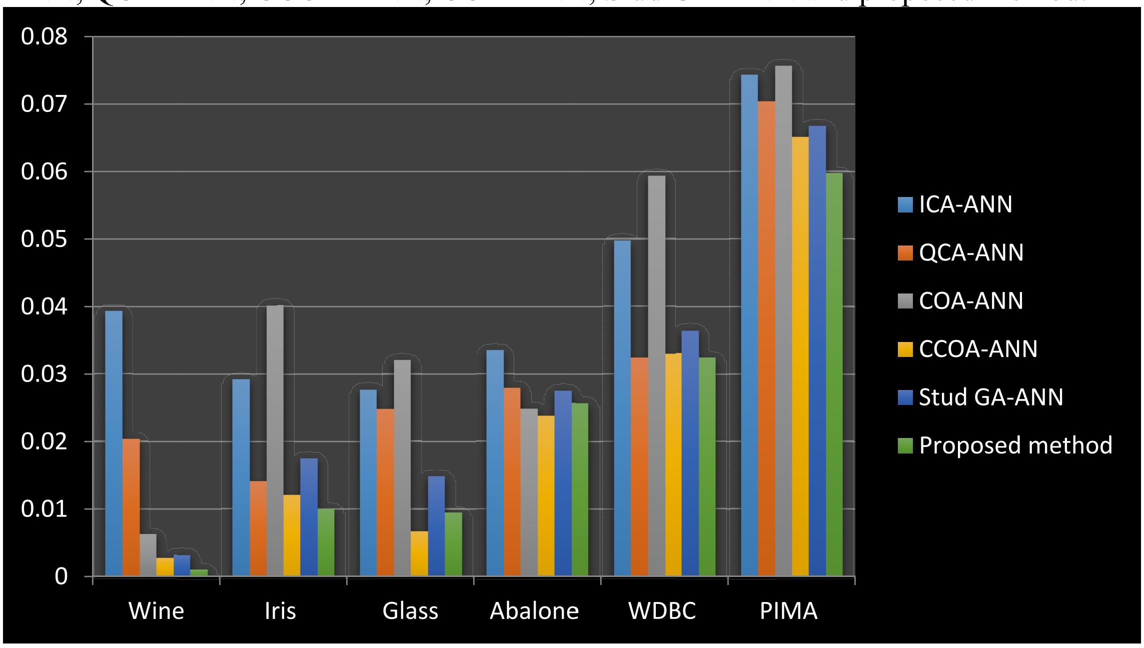

Figure 11.Mean square error for 4 dataset with 5 evolutionary algorithms

As can be seen in figure 11, the proposed method has appropriate response in comparison to other algorithms on Iris and Wine datasets. Only algorithm CCOA gains suitable response versus proposed method in Glass dataset whereas CCOA and COA were better than our proposed in Abalone dataset. Our proposed method and CCOA with containing the local chaotic searching systems are capable to find optimal response in each generation. Improved approaches outperforms its basic method in each generation and every generation solves better than its rudimentary algorithm that is why these have good degree of convergence versus its primary algorithm. Training of neural network by evolutionary algorithms is one of the problem in which the rate of exploit must be greater than the rate of explore that is why our proposed algorithm and CCOA have better results.

6. Conclusion

Determination of the weights of the neural network has a considerable impact on increasing the efficiency of this method. Using evolutionary algorithms for determining the optimal weights has had acceptable results. In this article chaotic Stud GA was used for training a multi-layer perceptron neural network (in the application of classification). The results from the implementation of the proposed method were compared with imperialist competitive algorithm, quad countries algorithm, Stud GA and cuckoo optimization algorithm. The results show that the proposed method with the specified parameters has better results than four other methods. To continue working, using the proposed method is suggested for other structures of neural network such as RBF, determination of network topology and development of the proposed method for time-series forecasting systems.

References

[1] L. V. Fausett, Fundamentals of neural networks: Prentice-Hall, 1994.

[2] X. Yao, "Evolutionary Artificial Neural Networks," Neural Systems, vol. 6, p. 19, 1993.

[3] O. Tatt Hee and N. A. M. Isa, "Adaptive Evolutionary Artificial Neural Networks for Pattern Classification," Neural Networks, IEEE Transactions on, vol. 22, pp. 1823-1836, 2011.

[4] Q. Xilong, F. Jian, and S. Wenfang, "Parallel Genetic Algorithm Model Based on AHP and Neural Networks for Enterprise Comprehensive Business," in Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on, 2008, pp. 897-900.

[5] Z. Chunkai, S. Huihe, and L. Yu, "Particle swarm optimisation for evolving artificial neural network," in Systems, Man, and Cybernetics, 2000 IEEE International Conference on, 2000, pp. 2487-2490 vol.4.

[6] K. Socha and C. Blum, "An ant colony optimization algorithm for continuous optimization: application to feed-forward neural network training," Neural Computing and Applications, vol . 16 ,pp. 235-247, 2007.

[7] C. Ozturk and D. Karaboga, "Hybrid Artificial Bee Colony algorithm for neural network training," in Evolutionary Computation (CEC), 2011 IEEE Congress on, 2011, pp. 84-88.

[8] M. T. Mahmoudi, N. Forouzideh, C. Lucas, and F. Taghiyareh, "Artificial Neural Network Weights Optimization based on Imperialist Competitive Algorithm," presented at the Computer Science and Information Technologies Lviv, Ukraine, 2009.

[9] S. Kulluk, L. Ozbakir, and A. Baykasoglu, "Training neural networks with harmony search algorithms for classification problems," Engineering Applications of Artificial Intelligence, vol. 25, pp. 11-19, 2012.

[10] F. Jin and G. Shu, "Back propagation neural network based on artificial bee colony algorithm," in StrategicTechnology (IFOST), 2012 7th International Forum on, 2012, pp. 1-4.

[11] S. Guangmin, Z. Canhui, Z. Weiwei, and Y. Guangyu, "A new recognition method of vehicle license plate based on Genetic Neural Network," in Industrial Electronics and Applications (ICIEA), 2010 the 5th IEEE Conference on, 2010, pp. 1662-1666.

[12] M. Yaghini, M. M. Khoshraftar, and M. Fallahi, "A hybrid algorithm for artificial neural network training," Engineering Applications of Artificial Intelligence, vol. 26, pp. 293-301, 2013.

[13] M. Sheikhan and N. Mohammadi, "Neural-based electricity load forecasting using hybrid of GA and ACO for feature selection," Neural Computing and Applications, vol. 21, pp. 1961-1970, 2012.

[14] M. Abdechiri, K. Faez, and H. Bahrami, "Neural Network Learning Based on Chaotic Imperialist Competitive Algorithm," in Intelligent Systems and Applications (ISA), 2010 2nd International Workshop on, 2010, pp. 1-5.

[15] F. Zabihi, k. faez, and m. s. hosseini, "Quad Countries Algorithm used for determine the weights of multi-layer Perceptron Neural Network," presented at the 18th National CSI Computer Conference, 2013.

[16] H. Ebrahimpour-Komleh and S. J. Mousavirad, "Cuckoo Optimization Algorithm for FeedForward Neural Network Training," presented at the 21stIranian Conference on Electrical Engineering (ICEE2013), 2013.

[17] F. razavi, K. faez, and F. zabihi, "Improving Cuckoo optimization algorithm in order to measure multilayered perceptron neural network weights," presented at the 5th Iranian Conference onBioinformatics(ICB 2014), 2014.

[18] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning: Addison-Wesley Longman Publishing Co., Inc., 1989.

[19] S. Wang, S. Yin, and M. Jiang, "Neural networks based on evolutional algorithm for residential loan," in Control and Decision Conference, 2008. CCDC 2008. Chinese, 2008, pp. 2516-2520.

[20] W. Khatib and P. J. Fleming, "The Stud GA: A Mini Revolution?," presented at the Proceedings of the 5th International Conference on Parallel Problem Solving from Nature, 1998.

[21] G.-G. Wang, A. H. Gandomi, and A. H. Alavi, "Stud krill herd algorithm," Neurocomput., vol. 128, pp. 363-370, 2014.

[22] A. H. Gandomi and X.-S. Yang, "Chaotic bat algorithm," Journal of Computational Science, vol. 5, pp. 224-232, 2014.

[23] D. Yang, Z. Liu, and J. Zhou, "Chaos optimization algorithms based on chaotic maps with different probability distribution and search speed for global optimization," Communications in Nonlinear Science and Numerical Simulation, vol .19 ,pp. 1229-1246, 2014.

[24] A. Kaveh, R. Sheikholeslami, S. Talatahari, and M. Keshvari-Ilkhichi, "Chaotic swarming of particles: A new method for size optimization of truss structures," Advances in Engineering Software, vol. 67, pp. 136-147, 2014.

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