

AN OPTIMIZED EXTREME LEARNING MACHINE USING ARTIFICIAL CHEMICAL REACTION OPTIMIZATION ALGORITHM

TỐI ƯU HÓA MÁY HỌC CỰC TRỊ SỬ DỤNG THUẬT TOÁN PHẢN ỨNG HÓA HỌC NHÂN TẠO

Tran Thuy Van

ABSTRACT

Extreme Learning Machine (ELM) is a simple learning algorithm for single-hidden-layer feed-forward neural network. The learning speed of ELM can be thousands of times faster than back-propagation algorithm, while obtaining better generalization performance. However, ELM may need high number of hidden neurons and lead to ill-condition problem due to the random determination of the input weights and hidden biases. In order to surmount the weakness of ELM, this paper proposes an optimization scheme for ELM based on artificial chemical reaction optimization algorithm (ACROA). By using ACROA to optimize the hidden biases and input weights according to both Root mean squared error and the Norm of output weights, the classification performance of ELM will be improved. The experimental result on several real benchmark problems demonstrates that the proposed method can attain higher classification accuracy than traditional ELM and other evolutionary ELMs.

Keywords: Extreme learning machine (ELM), artificial chemical reaction optimization algorithm (ACROA), single-hidden-layer feed-forward neural network (SLFN); learning algorithm; classification.

TÓM TẮT

Máy học cực trị (ELM) là một thuật toán học đơn giản ứng dụng cho các mạng nơ-ron truyền thẳng một lớp ẩn. Tốc độ học của ELM nhanh hơn gấp nghìn lần so với thuật toán lan truyền ngược, trong khi đó nó đạt được hiệu suất cao hơn. Tuy nhiên, vì các trọng số nút vào và các sai lệch nút ẩn được lựa chọn ngẫu nhiên, nên thuật toán ELM có thể cần nhiều nơ-ron ở lớp ẩn và dẫn đến vấn đề nhiều điều kiện ràng buộc. Để giải quyết mặt hạn chế này của ELM, bài báo này đề xuất một chiến lược tối ưu cho ELM trên cơ sở thuật toán tối ưu phản ứng hóa học nhân tạo (ACROA). Bằng việc sử dụng ACROA để tối ưu hóa các trọng số vào và sai lệch của các nút ẩn trên cơ sở hai tiêu chuẩn định mức trọng số đầu ra và lỗi bình phương trung bình, hiệu suất phân loại của ELM được cải thiện. Kết quả thực nghiệm trên vài tập mẫu chuẩn trong thực tế chứng minh rằng phương pháp đã đề xuất đạt độ chính xác phân loại cao hơn ELM gốc và các ELM tiến hóa khác.

Từ khóa: Máy học cực trị (ELM), thuật toán tối ưu phản ứng hóa học nhân tạo (ACROA), mạng nơ-ron truyền thẳng một lớp ẩn (SLFN), thuật toán học, sự phân loại.

Hanoi University of Industry

Email: tranthuyvan.hau@gmail.com

Received: 25/11/2019

Revised: 20/6/2020

Accepted: 21/10/2020

Nomenclature

ACROA	Artificial Chemical Reaction Optimization Algorithm
ELM	Extreme Learning Machine
SLFN	Single-hidden-Layer Feed-forward Neural network
CGLAs	Classical Gradient-based Learning Algorithms
PSO	Particle Swarm Optimization
DE	Differential Evolutionary
RMSE	Root Mean Squared Error
MP	Moore-Penrose

1. INTRODUCTION

Classical gradient-based learning algorithms (CGLAs) such as Levenberg-Marquardt and back propagation were widely applied for training single-hidden-layer feed-forward neural network (SLFN) [1]. Nonetheless, the CGLAs are able to be dropped toward a local minimum and time consuming due to inappropriate learning steps [2]. To deal with the drawbacks of Levenberg-Marquardt and back propagation algorithms, an extreme learning machine (ELM) was proposed by Huang et al. [1, 3] in 2004. In ELM, the input weights and hidden biases are randomly chosen, and the corresponding output weights will be determined analytically through Moore-Penrose (MP) generalized inverse [4]. The ELM tends to reach the smallest norm of output weights and attains the smallest training error [5, 21]. So, the ELM has faster learning speed and better generalization performance than those of the CGLAs. Moreover, ELM can avoid local minima and time consuming [6, 22].

For overcoming the above weakness of ELM, some nature-inspired population-based methods with global search capabilities had been successfully applied by optimizing the hidden biases and input weights, such as the combination of genetic algorithm and ELM [23], differential evolutionary (DE) [8], particle swarm optimization (PSO) [7]. In [6], an evolutionary ELM (E-ELM) was proposed which used

advantages of both ELM algorithm and DE algorithm. A modified DE was used to search for the optimal hidden biases and input weights, and the output weights were analytically determined by using MP generalized inverse. Thus E-ELM was able to obtain better generalization performance with much more compact networks. In the literature [9], a hybrid algorithm was proposed to optimize the hidden biases and input weights which could train the network to be more suitable for some prediction problems, namely evolutionary ELM based on PSO (PSO-ELM). Another hybrid evolutionary approach was proposed by Pacifico et al. [10] to select the optimal hidden biases and input weights of ELM by using PSO combining local best topology and clustering strategies.

Recently, a novel meta-heuristic optimization method was suggested by Alatas, namely artificial chemical reaction optimization algorithm (ACROA) [11]. ACROA is developed based on the chemical reactions of molecules and the second law of thermodynamics, so a system tends to the lowest enthalpy and the highest entropy [12]. In the ACROA, enthalpy or entropy can be used as objective function for minimization or maximization problem. ACROA is different from genetic algorithm [13] and PSO in solution mechanism of optimization and search. The ACROA has fewer parameters, and is more robust. Thus, ACROA method is adapted to solve the optimization problems. The successful application of the ACROA for the mining of classification rules can be indicated in [14].

In this paper, an optimization scheme for ELM based on ACROA is proposed to overcome the weakness of ELM, and to maximize ELM classifier's generalization performance. Firstly, CRO algorithm is used to optimize the hidden biases and input weights according to both the norm of output weights and the root mean squared error (RMSE) on validation set. Consequently, the corresponding output weights can be determined analytically. Secondly, the proposed method is compared with other methods over some benchmark classification problems available in the public repository. The experimental results show that the proposed method can attain higher classification accuracy than both other evolutionary ELMs and original ELM, while cost time is shorter than other evolutionary ELMs.

2. EXTREME LEARNING MACHINE

In the ELM for SLFN [3, 4], the hidden biases and input weights are randomly generated, and the output weights are analytically determined with a given number of hidden neurons. For a classification problem, a set of N arbitrary distinct samples can be expressed as $\Omega = \{(z_j, q_j) | z_j \in R^n; q_j \in R^m; j = 1, 2, \dots, N\}$, where $z_j = [z_{j1}, z_{j2}, \dots, z_{jn}]^T$ is an n-dimensional features vector of sample j^{th} , and $q_j = [q_{j1}, q_{j2}, \dots, q_{jm}]^T$ is a coded class label vector. Then a standard SLFN with L hidden neurons and activation function $\mu(\cdot)$ can approximate the samples set Ω with zero error. This means that the SLFN is mathematically modeled as the following linear system [3]:

$$\sum_{i=1}^L w_i \mu(v_i z_j + b_i) = q_j; j = 1, 2, \dots, N, \tag{1}$$

where $v_i = [v_{i1}, v_{i2}, \dots, v_{in}]$ is the weight vector connecting the input nodes and the i^{th} hidden node, $w_i = [w_{i1}, w_{i2}, \dots, w_{im}]^T$ is the weight vector connecting the i^{th} hidden node and the output nodes, and b_i is the bias of the i^{th} hidden node. It should be noted that many activation functions can be used for hidden neurons in original ELM classifier, such as sigmoidal, sine, tri-angular basis and radial basis.

The (1) can be rewritten compactly in a matrix form as follows:

$$HW = Q, \tag{2}$$

where H, W, and Q are the hidden layer output matrix, the output weights matrix, and the coded class label matrix, respectively. These matrices can be represented as follows:

$$H(v_1, \dots, v_L, b_1, \dots, b_L, z_1, \dots, z_N) = \begin{bmatrix} \mu(v_1 z_1 + b_1) & \dots & \mu(v_L z_1 + b_L) \\ \vdots & \dots & \vdots \\ \mu(v_1 z_N + b_1) & \dots & \mu(v_L z_N + b_L) \end{bmatrix}_{N \times L};$$

$$W = \begin{bmatrix} w_1^T \\ \vdots \\ w_L^T \end{bmatrix}_{L \times m}; Q = \begin{bmatrix} q_1^T \\ \vdots \\ q_N^T \end{bmatrix}_{N \times m}. \tag{3}$$

Thus, for the given linear system as in (2), the output weights are determined by finding the least-square solution [15]. The minimum norm least-square solution of the above linear system can be represented as follows [4]:

$$\hat{W} = H^+ Q, \tag{4}$$

where H^+ is the Moore-Penrose (MP) generalized inverse [16] of matrix H. The solution \hat{W} is unique, and has the smallest norm among all the least-square solutions of (2). This implies that the smallest training error can be reached, and ELM tends to obtain good generalization performance by using the MP generalized inverse method [6]. Moreover, since all the parameters of SLFN need not be tuned, the ELM algorithm converges much faster than the CGLAs.

3. ACROA METHOD

ACROA is a stochastic and adaptive search method. Its optimization is based on a chemical reaction process that leads to the transformation of chemical substances into another. The principle of ACROA contains five steps (more details about five steps can be found in [11, 14].

Step 1: Optimization problem and initial parameters. The optimization problem is defined as

$$\begin{aligned} &\text{minimize}\{f(\alpha)\}; \alpha_p \in H_p = [\theta_p^l, \theta_p^u]; \\ &p = \{1, 2, \dots, M\} \end{aligned} \tag{5}$$

where $f(\alpha)$ is a objective function, $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_M]$ is a decision variables vector, H_p is the feasible range of values for p^{th} decision variable, M is the number of decision variables, and θ_p^u and θ_p^l are the upper and lower bounds of the p^{th} decision variable, respectively. The different

encoding type of molecules is used appropriately for each optimization problem. Also, the parameter *ReacNum* is initialized in this step.

Step 2: Initialization and evaluation for reactants. The reactants are initialized uniformly in the possible solution region. The association rules are represented, and the value of objective function is evaluated.

Step 3: Application of elementary reactions. In the ACROA, there are five elementary reactions, namely decomposition reaction, redox1 reaction, synthesis reaction, displacement reaction, and redox2 reaction.

Step 4: Updating reactants. The chemical equilibrium is tested, and the new reactants are updated by evaluating objective function value.

Step 5: Checking termination criterion. Step 3 and step 4 will be repeated until the termination criterion is met.

4. OPTIMIZED ELM USING ACROA (AC-ELM)

In this section, ACROA will be used to optimize the hidden biases and input weights of ELM with the prefixed number of hidden neurons. The flow chart of AC-ELM is shown in Fig. 1, and consists of the following detailed steps:

The first, the set of initial molecules (*Pop*) is randomly generated, in which each molecular structure represents one ELM model. Each molecular structure in this solution set is composed of a vector of hidden biases and input weights:

$$\omega_k = \begin{bmatrix} v_{11}, v_{21}, \dots, v_{n1}, v_{12}, v_{22}, \dots, v_{n2}, \dots, \\ v_{1L}, v_{2L}, \dots, v_{nL}, b_1, b_2, \dots, b_L \end{bmatrix}, \quad (6)$$

where ω_k is the k^{th} molecular structure of the molecules set, and $k = 1, 2, \dots, \text{PopSize}$. All elements in the molecular structure are randomly initialized within the range of $[-1, 1]$.

The second, instead of the whole training samples set as used in the literatures [6, 17], the corresponding fitness function of each molecular structure is only adopted as RMSE on the validation samples set to avoid the over-fitting of the SLFN:

$$f(\cdot) = \sqrt{\frac{1}{N_V} \sum_{j=1}^{N_V} \left\| \sum_{i=1}^L w_i \mu(v_i z_j + b_i) - q_j \right\|_2^2}, \quad (7)$$

where N_V is the number of the validation samples ($N_V < N$), and $\|\cdot\|_2$ is the Euclidean norm. Then, the fitness function of each molecular structure is evaluated. For each molecular structure, the corresponding output weights are determined according to (4) on the training samples set.

The third, as investigated by Bartlett et al. [18] and Zhu et al. [6], neural networks tend to get the weights of smaller norm to reach better generalization performance. In order to obtain the best molecular structure for the population of molecules, the RMSE on the validation samples set along with the norm of output weights are considered. Thus, the

generalization performance of SLFN is significantly improved. The corresponding details are described as follows:

$$\omega_b = \begin{cases} \omega_k, & \left[\begin{array}{l} (f(\omega_b) - f(\omega_k) \geq \varepsilon f(\omega_b)) \text{ or} \\ (|f(\omega_b) - f(\omega_k)| < \varepsilon f(\omega_b)) \text{ and} \\ (\|W_{\omega_k}\|_2 < \|W_{\omega_b}\|_2) \end{array} \right] \\ \omega_b, & \text{else} \end{cases}, \quad (8)$$

where ε is a tolerance rate, and $f(\omega_k)$ and $f(\omega_b)$ are the corresponding fitness functions for the k^{th} molecular structure and the best molecular structure of all molecules, respectively. W_{ω_k} is the matrix of the corresponding output weights when the hidden biases and input weights are set as the k^{th} molecular structure, and W_{ω_b} is the best molecular structure of all molecules attained by MP generalized inverse.

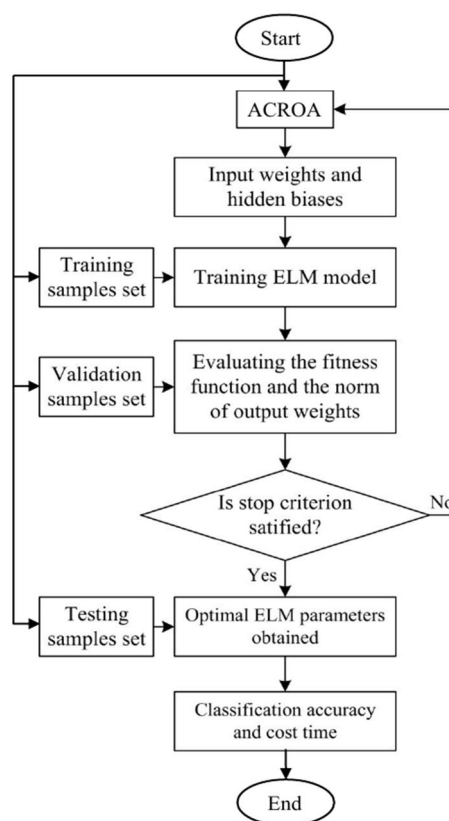


Fig. 1. The flow chart of AC-ELM

The fourth, in the iteration, the new molecules are added into the population by occurring uni-molecular collision or inter-molecular collision. According to the literatures [3, 4], all elements in the molecular structure should be bounded within the range of $[-1, 1]$. Therefore, the normalization of the elements of these new molecules in the ACROA is needed, and it is performed as follows:

$$v_{si} = \begin{cases} -2 - v_{si}, & v_{si} < -1 \\ 2 - v_{si}, & v_{si} > 1 \end{cases}; i = 1, 2, \dots, L; s = 1, 2, \dots, m, \quad (9)$$

$$b_i = \begin{cases} -2 - b_i, & b_i < -1 \\ 2 - b_i, & b_i > 1 \end{cases}; i = 1, 2, \dots, L. \quad (10)$$

Finally, the process of the above optimization is reiterated until the stopping criterion is met. Therefore, the optimal ELM with the obtained hidden biases and input weights is applied to the testing samples set.

5. EXPERIMENT RESULTS

In this section, the experimental results are presented on four classic classification problems from UCI machine repository [19] to validate our proposed method. These benchmark data sets present different degrees of difficulties and different number of classes. The specification of these problems is listed in Table 1. For each trial of simulation, the training, validation and testing data sets are randomly regenerated from its whole data set for all the algorithms [6].

In the experiment of this paper, all the input attributes and output classes have been normalized to the ranges [0, 1] and [-1, 1], respectively. The input weights and the biases of ELM have been obtained into the range [-1, 1]. The sigmoidal function $\mu(x) = 1/(1 + e^{-x})$ is used as the activation function for ELM [4].

In order to evaluate the performance and effectiveness of proposed AC-ELM method, the AC-ELM method is compared with original batch ELM [4], the evolutionary ELM (E-ELM) [6], and the evolutionary ELM based on PSO (PSO-ELM) [9]. All the simulated results are carried out in MATLAB 7.10 environment.

Table 1. Specification of four classification problems

Problems	Attributes	Classes	Number of samples			
			Training	Validation	Testing	Total
Cancer	30	2	229	170	170	569
Credit	14	2	270	210	210	690
Diabetes	8	2	252	258	258	768
Glass	9	6	114	50	50	214

Four algorithms, the ELM, the PSO-ELM, the E-ELM, and the AC-ELM, were used to classify four data sets (in Table 1). For PSO, the parameters were fixed for all data sets as in Table 2, according to the literatures [10, 20]. Similar to PSO, the population sizes and maximum learning epochs of DE were set to 50 and 100, respectively, and some other parameters were fixed with the values given in Table 3 [6]. To make a fair comparison, the values of the ACROA were chosen to be the same, e.g., the initial population is set by $ReacNum = 50$, and the termination criterion is considered as 100 iterations. The performance of all methods is evaluated by using the average and standard deviation (Dev) of the testing accuracy in 50 trials.

Table 2. PSO parameters for all simulations

Parameters	Value
Swarm Size (s)	50
Acceleration Factors (c_1)	1.9
Acceleration Factors (c_2)	1.9

Inertia Factor (w)	0.8 to 0.3
Maximum Number of Iterations	100
Number of Trials	50

Table 3. DE parameters for all simulations

Parameters	Value
Population Size (NP)	50
Constant Factor (F)	0.9
Crossover Constant (CR)	0.7
Tolerance Rate (ϵ)	0.03
Maximum Learning Epochs	100
Number of Trials	50

First of all, the simulation of the original ELM classifier is represented for four classification problems. The number of neurons in the hidden-layer is considered in the range [1, 100]. Fig. 2 shows the training and testing accuracies depend on the number of hidden nodes for all data sets.

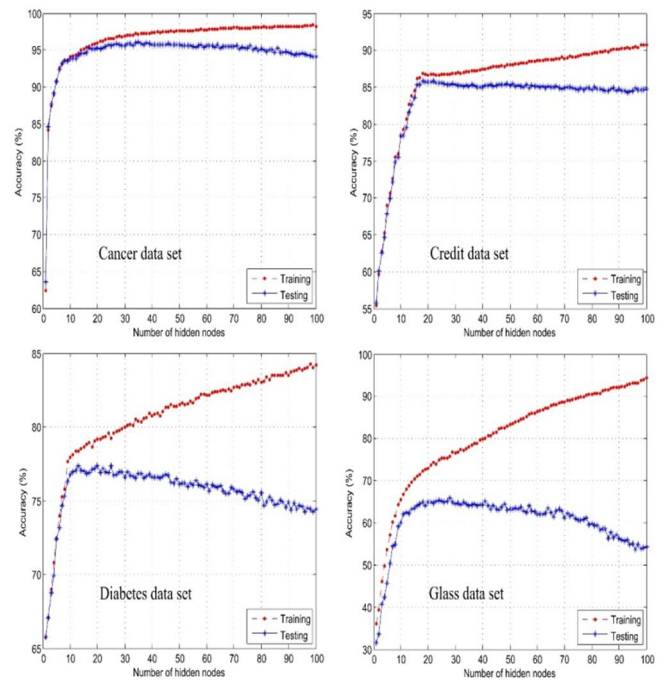


Fig. 2. The training and testing accuracies of ELM depend on the number of hidden nodes

As seen from the results in Fig. 2, the training accuracies increase when the number of hidden nodes increases. However, the testing accuracies only obtain maximum values with the number of hidden nodes in the range [10, 30], and they obtain lower value with other numbers of hidden nodes. Specifically, the highest results of the testing accuracies, together with the corresponding number of nodes in the hidden layer, are presented in Table 4. In Table 4, the corresponding performances of three evolutionary ELMs on all classification problems are also shown. Note that for all data sets and algorithms, the best results (according to the empirical analysis) are emphasized in bold.

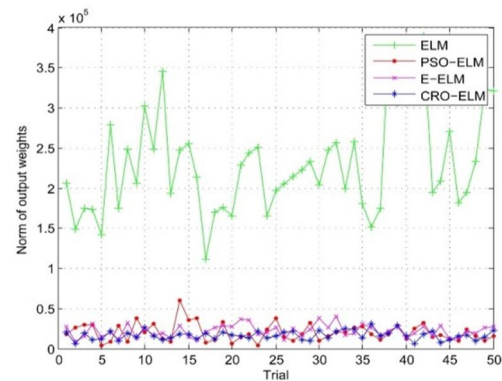
Table 4. Performance of four algorithms on all data sets

Problems	Algorithms	Hidden Nodes	Average Accuracy (%) ± Standard Deviation		Cost Time (s)	Norm of Output Weights
			Training	Testing		
Cancer	ELM	36	96.36 ± 0.47	94.74 ± 1.28	0.0278	2.3452x10 ⁵
	PSO-ELM	16	95.50 ± 0.33	95.23 ± 1.01	16.4638	2.1947x10 ⁴
	E-ELM	16	94.57 ± 0.89	95.15 ± 1.52	12.2285	2.3048x10 ⁴
	AC-ELM	16	95.89 ± 1.28	95.73 ± 1.35	1.5651	1.7653x10⁴
Credit	ELM	20	85.87 ± 0.84	84.67 ± 2.25	0.34082	4.7832x10 ⁶
	PSO-ELM	16	86.85 ± 0.46	85.96 ± 1.77	18.8695	4.8932x10⁵
	E-ELM	16	84.77 ± 1.38	86.15 ± 1.75	12.0234	6.5274x10 ⁵
	AC-ELM	16	86.95 ± 1.69	86.42 ± 1.48	1.643168	6.1132x10 ⁵
Diabetes	ELM	15	77.56 ± 1.29	76.14 ± 2.29	0.0345	7.8732x10 ¹
	PSO-ELM	12	78.82 ± 0.76	76.87 ± 1.62	27.8762	4.6402x10 ¹
	E-ELM	12	76.81 ± 1.97	76.91 ± 1.71	17.4382	5.4987x10 ¹
	AC-ELM	12	77.92 ± 1.62	77.25 ± 1.38	2.2376	4.1295x10¹
Glass	ELM	30	75.21 ± 2.54	64.37 ± 6.79	0.0201	4.8732x10 ⁵
	PSO-ELM	12	70.98 ± 1.98	65.31 ± 5.08	8.5836	1.3106x10⁴
	E-ELM	12	66.53 ± 3.05	65.12 ± 4.92	8.7601	2.4382x10 ⁴
	AC-ELM	12	70.29 ± 5.42	65.59 ± 4.47	1.2139	1.8762x10 ⁴

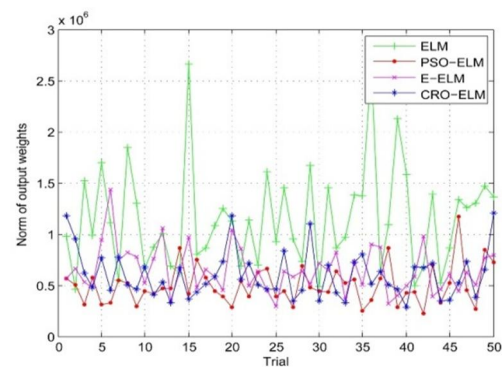
From Table 4, it can be seen that the testing accuracy of AC-ELM algorithm is the highest, compared with the other three algorithms on all the data sets. The training accuracy of the AC-ELM is highest on Credit data set only, while the training accuracy of the ELM and PSO-ELM are highest on two data sets (Cancer, Glass) and one data set (Diabetes), respectively. The cost time of the AC-ELM is less than the PSO-ELM and the E-ELM on all the data sets. Specially, the number of hidden nodes which used to attain these results in the AC-ELM is less than that in the ELM, and the same in both the PSO-ELM and the E-ELM. Clearly, the global and local search ability of ACROA advantages reducing the

hidden neurons in the AC-ELM, and improving the testing accuracy. The results in Table 4 also show that the AC-ELM is able to obtain the smaller norm of the output weights than the PSO-ELM, the E-ELM, and the ELM on two data sets (such as Cancer and Diabetes). For Credit and Glass data sets, the smallest norm of the output weights is obtained by the PSO-ELM method. Besides, for four compared ELMs on all data sets, the norm values at each trial are surveyed, and represented in Fig. 3.

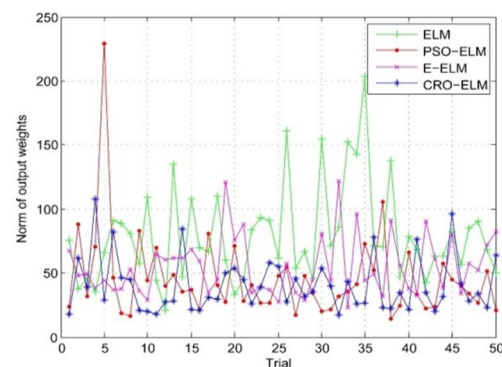
As seen from Fig. 3, the norm values of the output weights obtained by the PSO-ELM, the E-ELM, and the AC-ELM is almost less than those achieved by the ELM on all cases in each trial except on Diabetes classifications. In all cases, the norm values of the AC-ELM are steadier than those of the ELM, the PSO-ELM, and the E-ELM. Therefore, the proposed algorithm has the best generalization performance in all compared algorithms.



(a)



(b)



(c)

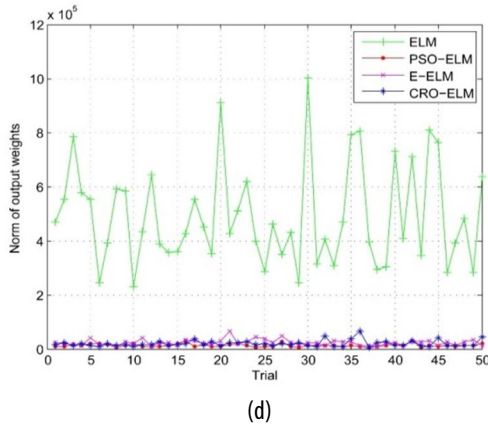


Fig. 3. The norm of output weights at each trial on Cancer, Credit, Diabetes, and Glass

- (a) For Cancer data set; (b) For Credit data set;
- (c) For Diabetes data set; (d) For Glass data set

6. CONCLUSIONS

In this paper, a novel learning algorithm based on hybridization of ACROA with ELM, namely AC-ELM is proposed. In the proposed algorithm, the hidden biases and input weights of the ELM were optimized by the ACROA, and the output weights of the ELM were analytically determined by using the smallest norm least-square scheme. Moreover, in the process of optimizing the hidden biases and input weights, the ACROA algorithm considered both the norm of the output weights and the RMSE on validation samples set. Therefore, the AC-ELM can search the global minimum, which represents the SLFN providing the best generalization performance. Finally, the performance of the tested algorithms was evaluated with well-known benchmark classification datasets. Experiment results show that the AC-ELM obtains higher testing accuracy on the various datasets than the ELM, the PSO-ELM and the E-ELM, while the AC-ELM obtains lower cost time.

Acknowledgment

The authors would like to thank the editor and the reviewers for their valuable comments.

REFERENCES

[1]. S. Haykin, 1999. *Neural Networks: A Comprehensive Foundation*, second ed. Englewood Cliffs, NJ, USA: Prentice Hall.

[2]. D.-S. Huang, 2004. *A constructive approach for finding arbitrary roots of polynomials by neural networks*. *IEEE Transactions on Neural Networks*, vol. 15, pp. 477-491.

[3]. G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, 2004. *Extreme learning machine: a new learning scheme of feedforward neural networks*. in *IEEE International Joint Conference on Neural Networks*, pp. 985-990.

[4]. G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, 2006. *Extreme learning machine: Theory and applications*. *Neurocomputing*, vol. 70, pp. 489-501.

[5]. G.-B. Huang, D. Wang, and Y. Lan, 2011. *Extreme learning machines: a survey*. *International Journal of Machine Learning and Cybernetics*, vol. 2, pp. 107-122, 2011/06/01.

[6]. Q.-Y. Zhu, A. K. Qin, P. N. Suganthan, G.-B. Huang, 2005. *Evolutionary extreme learning machine*. *Pattern Recognition*, vol. 38, pp. 1759-1763.

[7]. J. Kennedy and R. Eberhart, 1995. *Particle swarm optimization*. in *IEEE International Conference on Neural Networks*, pp. 1942-1948.

[8]. R. Storn, K. Price, 1997. *Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces*. *Journal of Global Optimization*, vol. 11, pp. 341-359.

[9]. Y. X, Y. Shu, 2006. *Evolutionary Extreme Learning Machine - Based on Particle Swarm Optimization*. in *Advances in Neural Networks*. vol. 3971, J. Wang, Z. Yi, J. M. Zurada, B.-L. Lu, and Y. Hujun, Eds., ed: Springer Berlin Heidelberg, pp. 644-652.

[10]. L. D. S. Pacifico, T. B. Ludermitr, 2013. *Evolutionary extreme learning machine based on particle swarm optimization and clustering strategies*. in *International Joint Conference on Neural Networks (IJCNN)*, pp. 1-6.

[11]. B. Alatas, 2011. *ACROA: Artificial Chemical Reaction Optimization Algorithm for global optimization*. *Expert Systems with Applications*, vol. 38, pp. 13170-13180.

[12]. P. K. Nag, 1995. *Engineering Thermodynamics*. New Delhi: Tata McGraw-Hill Education.

[13]. D. Whitley, 1994. *A Genetic Algorithm Tutorial*. *Statistics and Computing*, vol. 4, pp. 65-85.

[14]. B. Alatas, 2012. *A novel chemistry based metaheuristic optimization method for mining of classification rules*. *Expert Systems with Applications*, vol. 39, pp. 11080-11088.

[15]. D. Lowe, 1989. *Adaptive radial basis function nonlinearities, and the problem of generalisation*. in *First IEE International Conference on Artificial Neural Networks (Conf. Publ. No. 313)*, pp. 171-175.

[16]. D. Serre, 2002. *Matrices: Theory and Applications*. New York: Springer.

[17]. B. Verma, R. Ghosh, 2003. *A Hierarchical Method for Finding Optimal Architecture and Weights Using Evolutionary Least Square Based Learning*. *International Journal of Neural Systems*, vol. 13, pp. 13-24.

[18]. P. L. Bartlett, 1998. *The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network*. *IEEE Transactions on Information Theory*, vol. 44, pp. 525-536.

[19]. M. Lichman. (Irvine, CA: University of California, School of Information and Computer Science, 2013). *UCI Machine Learning Repository*. Available: <http://archive.ics.uci.edu/ml>

[20]. M. Carvalho, T. B. Ludermitr, 2006. *An Analysis Of PSO Hybrid Algorithms For Feed-Forward Neural Networks Training*. in *Ninth Brazilian Symposium on Neural Networks, 2006*, pp. 6-11.

[21]. X. Li, W. Mao, W. Jiang, 2016. *Extreme learning machine based transfer learning for data classification*. *Neurocomputing*, Volume 174, Part A, Pages 203-210.

[22]. A. Lendasse, C. M. Vong, K.-A. Toh, Y. Miche, G.-B. Huang, 2017. *Advances in extreme learning machines (ELM2015)*. *Neurocomputing*, Volume 261, Pages 1-3.

[23]. Z. Yu, C. Zhao, 2017. *A Combination Forecasting Model of Extreme Learning Machine Based on Genetic Algorithm Optimization*. *International Conference on Computing Intelligence and Information System (CIIS)*, 21-23.

THÔNG TIN TÁC GIẢ

Trần Thủy Văn

Trường Đại học Công nghiệp Hà Nội