An Upgraded Bat Algorithm for Tuning Extreme Learning Machines for Data Classification

Adis Alihodzic
University of Sarajevo
Zmaja od Bosne 33-35
Sarajevo, BiH 71000
adis.alihodzic@pmf.unsa.ba

Eva Tuba
University of Belgrade
Studentski trg 16
Belgrade, Serbia 11000
etuba@acm.org

Milan Tuba*
Jonh Naisbitt University
Bulevar umetnosti 29
Belgrade, Serbia 11070
mtuba@acm.org

ABSTRACT

The learning time of the synaptic weights for feedforward neural networks tend to be very long. In order to reduce the learning time, in this paper we propose a new learning algorithm for learning the synaptic weights of the single-hidden-layer feedforward neural networks by combining the upgraded bat algorithm with the extreme learning machine. The proposed approach can efficiently search for the optimal input weights as well as the hidden biases, leading to the reduced number of evaluations needed to train a neural network. The experimental results based on classification problems and comparison with other approaches from literature have shown that the proposed algorithm produces a satisfactory performance in almost all cases and that it can learn the weight factors much faster than the traditional learning algorithms.

CCS CONCEPTS

•Computing methodologies → Heuristic function construction; Neural networks; Bio-inspired approaches;

KEYWORDS

Swarm intelligence, bat algorithm, extreme learning machine

ACM Reference format:
Adis Alihodzic, Eva Tuba, and Milan Tuba. 2017. An Upgraded Bat Algorithm for Tuning Extreme Learning Machines for Data Classification. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3067695.3076088

1 INTRODUCTION

During last few decades artificial neural networks (ANN) have been increasingly used for various practical applications. Single-hidden-layer feedforward neural networks (SLFN) are widely used for classification problems [8]. Back-propagation (BP) algorithm is used for optimizing the network performance by training the SLFN. The Levenberg-Marquardt (LM) algorithm has also wide applications in training the SLFNs. Although these algorithms have good performance they have their shortcomings such as slow convergence and getting stuck into local minima. In order to avoid these drawbacks, in this paper we investigate the use of the extreme learning machine (ELM) which as a tuning-free method is a surprisingly efficient for learning neural networks with a single hidden layer. In the ELM, the input weights and the hidden biases are randomly generated, while the output weights, unlike the traditional neural networks, are algebraically determined by solving a linear system of equations. Solving a general linear system $Ax = b$, where A may be singular and may even not be square is a delicate issue, though it becomes simpler by using the Moore-Penrose generalized inverse method. By combining with the Moore-Penrose generalized inversion, ELM not only become much faster than the classical neural networks, but also has a better performance for the training phase.

Even though the ELM can obtain good results during the process of training and testing of neural networks, the main disadvantage of these networks is the non-optimal tuning of the input weights and biases. Moreover, the extreme learning machines, in order to adequately adjust the weight coefficients, often require a larger number of hidden neurons in comparison to the number of hidden neurons used by conventional learning algorithms.

Swarm intelligence and other nature-inspired algorithms, based on a random selection can produce better results compared to the conventional algorithms hence it can be expected that their incorporation into the extreme learning machine can improve the performance. For example, differential evolution (DE) is one of the first such algorithms that combined with the extreme learning machines produced good performance [2]. Particle swarm optimization (PSO) was also used to select input weights and hidden biases in a meaningful way with the aim of achieving the best performance of ELM in [3].

Further improvements are still possible and in this paper a new method called UBA-ELM combining the ELM with an upgraded bat algorithm (UBA) is proposed as a novel learning method for tuning extreme learning machine so as to perform better classification. In the proposed method, the upgraded bat algorithm is used to optimize input weights and hidden biases according to the root mean squared error (RMSE), while the output weights are algebraically calculated by the Moore-Penrose (MP) generalized inverse.

2 THE PROPOSED UBA-ELM

Bat algorithm (BA) is a new population based metaheuristic approach proposed by Xin-She Yang [11] and it was used for classification problem [9], [10]. In order to enhance the search performance of the BA and also to provide a better combination with ELM and

*This research is supported by the Ministry of Education, Science and Technological Development of Republic of Serbia, Grant No. III-44006

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
GECCO '17 Companion, Berlin, Germany
© 2017 Copyright held by the owner/author(s). 978-1-4503-4939-0/17/07…$15.00
DOI: http://dx.doi.org/10.1145/3067695.3076088

to avoid being trapped into local optima, we introduced an upgraded bat algorithm (UBA) with four modifications to the basic bat algorithm.

The first modification was to modify the pulse emission rate vector. With the new form of pulse rate function with appropriately tuned parameters in the early stages of the bat algorithm a large part of agents (bats) will be redirected to the diversification.

The second modification was to change the frequency, while the third modification relates to the modification of the equation for the velocity of an individual bat using some form of crossover.

The fourth modification allows that when some solution gets trapped into a local optimum, modification or replacement of this stagnant solution will be carried out if a predetermined number of allowed trials is exceeded.

To validate and test this UBA-ELM we tested and compared it with the pure bat algorithm and our upgraded algorithm outperformed it in all cases.

3 EXPERIMENTAL RESULTS

In order to test the performance of the proposed upgraded bat algorithm (UBA-ELM) with the state-of-the-art algorithms: (i) genetically optimized ELM (GO-ELM) [7], (ii) the original ELM [4], (iii) improved genetic algorithm for SLFN (IGA-SLFN) [5], (iv) the self-adaptive evolutionary ELM (SaE-ELM) [1], and (v) the SLFN trained using the Levenberg-Marquardt algorithm (LM-SLFN), 7 well-known benchmark problems were used [6].

In the Table 1 the comparison results for five algorithms are presented for average of the 30 runs on the 7 data sets as well as the number of hidden neurons. The best performance attributes are highlighted in bold. From the analysis of the results given in Table 1, it can be confirmed that the UBA-ELM is superior in the comparison with the other algorithms for almost all statistical parameters. Namely, UBA-ELM shows the lowest mean of the RMSE in all cases. Also, it has the lowest standard deviation of the RMSE, except for the benchmark problems Cancer and Ailerons. When the training time is considered, our proposed UBA-ELM is also superior with respect to the other algorithms except the ELM. GO-ELM, IGA-SLFN, SaE-ELM and LM-SLFN algorithms use several tests to select the optimal number of hidden neurons and as a consequence the mentioned algorithms take much more CPU time.

Our proposed UBA-ELM algorithm also requires a smaller number of neurons at hidden layer. Since the convergence of the UBA-ELM is fast, it can be successfully used for applications where a short training time is crucial.

From the analysis of the Table 1 it is clear that for the tested data sets the proposed UBA-ELM algorithm achieves best testing accuracy with least hidden neurons. It achieves better performance by using upgraded bat algorithm to pick the input weight factors. The proposed UBA-ELM algorithm has the best generalization performance and is most stable for all benchmark problems with respect to all other tested state-of-the-art algorithms.

Table 1: Results for the six optimization methods

Data set.	Method	Testing RMSE Mean	Testing RMSE Std	Training time (s)	HN
Boston Housing	GO-ELM	0.3503	0.0565	4.5463	21.15
	ELM	0.4652	0.1901	**0.0020**	20.00
	IGA-SLFN	0.5208	0.1278	1.6540	28.00
	SaE-ELM	0.4594	0.1628	8.0205	15.00
	LM-SLFN	0.5182	0.1677	1.0935	15.00
	UBA-ELM	**0.2249**	**0.0329**	0.4260	**15**
Automobile MPG	GO-ELM	0.2608	0.0490	4.3751	21.75
	ELM	0.2610	0.0443	**0.0025**	19.00
	IGA-SLFN	0.4874	0.1582	1.3098	23.00
	SaE-ELM	0.2682	0.0491	7.5290	15.00
	LM-SLFN	0.4030	0.1109	0.7050	17.00
	UBA-ELM	**0.1557**	**0.0083**	0.0470	**10**
Cancer	GO-ELM	0.5782	**0.0229**	4.0442	18.25
	ELM	0.5864	0.0240	**0.0015**	11.00
	IGA-SLFN	0.5706	0.1478	2.1523	22.00
	SaE-ELM	0.6169	0.0287	7.6500	15.00
	LM-SLFN	0.8288	0.2090	1.0375	15.00
	UBA-ELM	**0.4004**	0.2251	0.0450	**10**
Servo	GO-ELM	0.2320	0.0428	3.6212	21.75
	ELM	0.2401	0.0263	**0.0030**	24.00
	IGA-SLFN	0.5316	0.1419	1.0425	15.00
	SaE-ELM	0.2301	0.0261	6.6805	15.00
	LM-SLFN	0.2713	0.0566	0.5555	15.00
	UBA-ELM	**0.2137**	**0.0152**	0.1315	**15**
CPU	GO-ELM	0.1733	0.0281	3.7444	17.20
	ELM	0.1772	0.0712	**0.0010**	11.00
	IGA-SLFN	0.7492	0.3486	1.2839	22.00
	SaE-ELM	0.2449	0.0971	31.8785	19.00
	LM-SLFN	0.3098	0.1069	0.5935	18.00
	UBA-ELM	**0.0479**	**0.0074**	0.0167	**3**
Concrete Comp. Strength	GO-ELM	0.2738	0.0230	5.4051	22.05
	ELM	0.3152	0.0533	**0.0025**	13.00
	IGA-SLFN	0.7526	0.3365	1.7226	24.00
	SaE-ELM	0.3109	0.0312	38.7825	15.00
	LM-SLFN	0.3765	0.0661	1.5910	17.00
	UBA-ELM	**0.2278**	**0.0085**	0.1748	**8**
Ailerons	GO-ELM	0.0912	0.0007	27.4410	23.55
	ELM	0.0947	0.0019	**0.0590**	29.00
	IGA-SLFN	0.3809	0.2393	8.3769	15.00
	SaE-ELM	0.0917	**0.0006**	323.0840	27.00
	LM-SLFN	0.0940	0.0022	24.6795	22.00
	UBA-ELM	**0.0433**	0.0324	1.5769	**5**

REFERENCES

[1] Jiuwen Cao, Zhiping Lin, and Guang-Bin Huang. 2012. Self-Adaptive Evolutionary Extreme Learning Machine. *Neural Processing Letters* 36, 3 (2012), 285–305.

[2] Guorui Feng, Zhenxing Qian, and Xinpeng Zhang. 2012. Evolutionary selection extreme learning machine optimization for regression. *Soft Computing* 16, 9 (2012), 1485–1491.

[3] Elliackin M.N. Figueiredo and Teresa Bernarda Ludermir. 2014. Investigating the use of alternative topologies on perforance of the PSO-ELM. *Neurocomputing* 127 (2014), 4–12.

[4] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. 2006. Extreme learning machine: Theory and applications. *Neurocomputing* 70, 1 (2006), 489–501.

[5] Frank Hung-Fat Leung, Hak-Keung Lam, Sai-Ho Ling, and Peter Kwong-Shun Tam. 2003. Tuning of the structure and parameters of a neural network using an improved genetic algorithm. *IEEE Transactions on Neural Networks* 14, 1 (2003), 79–88.

[6] Moshe Lichman. 2013. UCI Machine Learning Repository. (2013). http://archive.ics.uci.edu/ml

[7] Tiago Matias, Francisco Souza, Rui Araujo, and Carlos Henggeler Antunes. 2014. Learning of a single-hidden layer feedforward neural network using an optimized extreme learning machine. *Neurocomputing* 129 (2014).

[8] Eva Tuba, Milan Tuba, and Marko Beko. 2016. Support Vector Machine Parameters Optimization by Enhanced Fireworks Algorithm. In *LNCS: International Conference in Swarm Intelligence*, Vol. 9712. Springer, 526–534.

[9] Eva Tuba, Milan Tuba, and Dana Simian. 2016. Adjusted bat algorithm for tuning of support vector machine parameters. In *IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2225–2232.

[10] Eva Tuba, Milan Tuba, and Dana Simian. 2016. Handwritten digit recognition by support vector machine optimized by bat algorithm. In *24th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision,(WSCG 2016)*. ACM, 369–376.

[11] Xin-She Yang. 2010. A New Metaheuristic Bat-Inspired Algorithm. In *Studies in Computational Intelligence: Nature Inspired Cooperative Strategies for Optimization*, Vol. 284. Springer Berlin Heidelberg, 65–74.