ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 10, March 2018

# An Overview of Emerging Developments in Extreme Learning Machines

*By*

*Parminder Kaur*

*Assistant Professor*

*Khalsa College for Women, Civil Lines, Ludhiana*

*Email id: parminder157@gmail.com*

## Abstract:

In the field of computational intelligence, Extreme Learning Machines have generated a lot of interest amongst the researchers. This is mainly due to its training speed, simplicity and ease of use. Their ability to solve computationally complex problems with ease coupled with ability to avoid problems related to local minima, generally encountered in the conventional computational techniques has led to the popularity of the Extreme Learning machines. These are now finding applications for solution of a galaxy of problems related to classification, regression, pattern recognition, forecasting and diagnosis, image processing, besides many others. This paper examines the current state of research in the field of Extreme Learning machines. It discusses the major developments and advances in ELM and serves as a preliminary point for the new researchers in the arena of ELM.

**Keywords:** Computational Intelligence, Extreme Learning Machines, Learning algorithms, Single hidden layer feed-forward ANN.

## Introduction

The last decade or so has witnessed enormous research in the field of machine learning to solve computationally complex problems. However, one of the drawbacks of the conventional methods of machine learning algorithms has been their underlying assumption of the training as well as the testing samples possessing the same data distribution.

Feed Forward Neural Networks have been widely used for different applications in the past decade. Their ability to approximate complex nonlinear mappings from the input samples and offer models for a variety of natural occurring as well as artificial phenomenon, which the other conventional parametric techniques find it hard to handle, has led to the popularity of the Feed forward ANNs. However, one of the major drawbacks of these neural networks has been the interdependency amongst the various parameters of these networks. The tuning of these interdependent parameters is a time consuming process which in turn makes these networks relatively slower in speed. Single hidden layer feed forward Artificial Neural Networks have been extensively studied by the researchers for their learning abilities and fault tolerant capabilities[1-6]. A majority of the learning algorithms employed for training these networks are relatively slow and also have a tendency to get stuck in the local minima.ELMs have been developed to improve the efficiency of the SLFANNs.Unlike the conventional learning algorithms which require manual tuning of the parameters like the learning rate, learning epochs etc., the ELMs are implemented automatically without the intervention of the users.

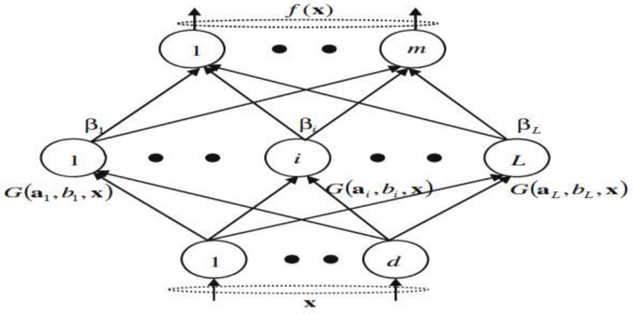
$f(\mathbf{x})$
1
$m$
$\beta_1$
$\beta_i$
$\beta_L$
1
$i$
$L$
$G(\mathbf{a}_1, b_1, \mathbf{x})$
$G(\mathbf{a}_i, b_i, \mathbf{x})$
$G(\mathbf{a}_L, b_L, \mathbf{x})$
1
$d$
$\mathbf{x}$

Fig 1: Single Layer Feed Forward Artificial Neural Network

In ELMS, the learning parameters of the hidden nodes can be independently assigned randomly and the output weights can also be determined through a simple generalized operation. The training phase in the ELMs can thus be accomplished easily and speedily. ELMs have been found to achieve good generalization performance. Moreover, ELMS have been found to be good universal approximators with additive activation function [7-9]. These characteristics have enabled application of ELMs to a large number of real world applications like regression, image processing, forecasting, classification etc. [10-14].

## Basic ELM Algorithm:

Extreme learning machine (ELM) is a learning method for training single hidden layer feed-forward ANN (SLFANN), which generates input weights and biases randomly. Later, the output weights are calculated analytically [15].Fig 2 depicts the architecture of Extreme Learning Machine.

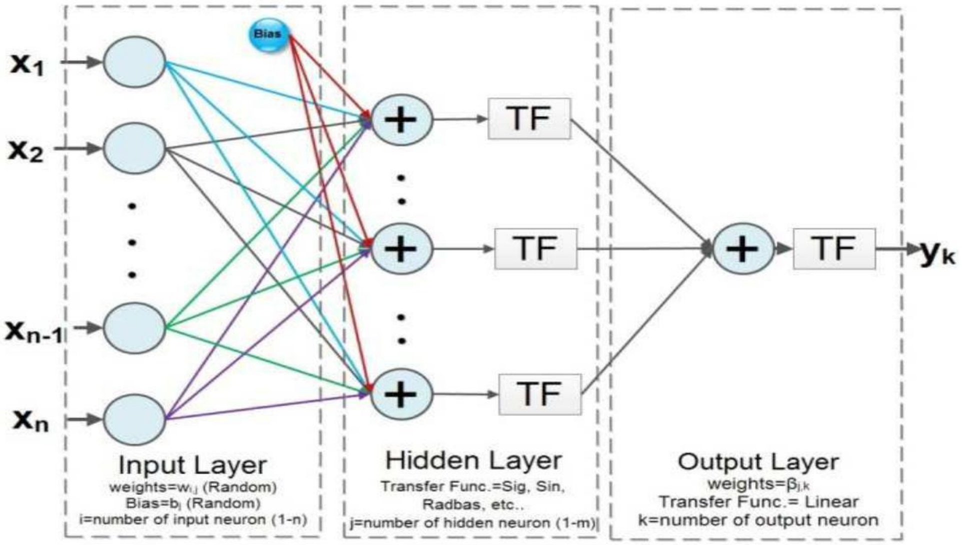

Fig 2: Architecture of Extreme Learning Machine

For a network with m neurons in the hidden layer and the input layer with n neurons denoted by $x_i$, the k number of outputs in the output layer denoted by $y_k$ can then be mathematically expressed as [16]

$$g($$

, are the input weights, , are the output weights. The bias of the hidden layer neurons is denoted by while the activation function for the network is denoted by g. The mathematical model expressed above can be rewritten as:

$$\mathbf{H} \quad = \mathbf{y}$$

Where H is the hidden layer output matrix and can be expressed as:

$$H= \begin{matrix} ( , \ + ) \ldots\ldots\ldots\ldots\ldots\ldots ( ,+ ) \\ . \\ . \\ . \\ . \\ ( , \ + ) \ldots\ldots\ldots\ldots\ldots\ldots ( ,+ \end{matrix}$$

Here, the only unknown parameters are **,** the output weights, which then can be found using the Moore-Penrose equation given by: $\boldsymbol{' = H^{+} y}$

Here, $\boldsymbol{H^{+}}$ is the Moore-Penrose inverse matrix of H [17].

The basic steps followed for use of ELM network for the purpose of classification are depicted in Fig 3.

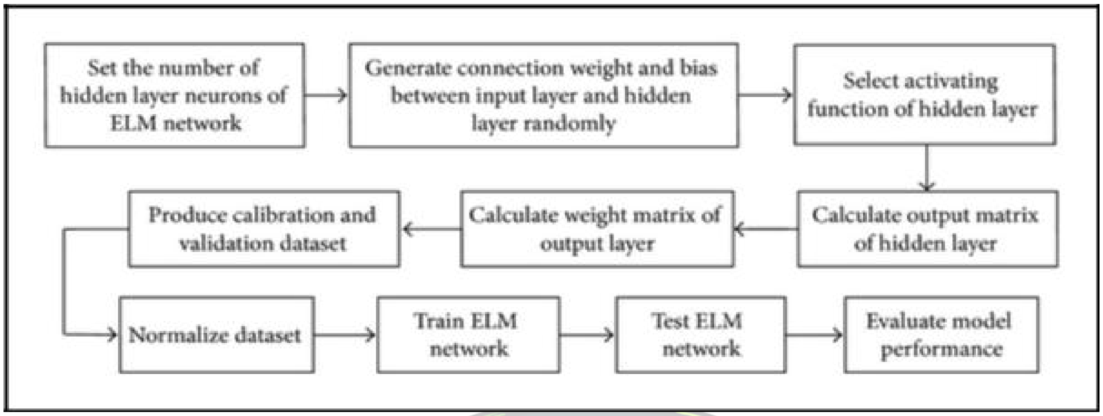
Set the number of hidden layer neurons of ELM network
Generate connection weight and bias between input layer and hidden layer randomly
Select activating function of hidden layer
Calculate output matrix of hidden layer
Calculate weight matrix of output layer
Produce calibration and validation dataset
Normalize dataset
Train ELM network
Test ELM network
Evaluate model performance

Fig 3: Process followed for developing ELM network for classification purpose.

The algorithmic steps followed for Extreme Learning Machine are listed below:

**Input:** Given a set of samples for training, N ={$(x_i,t_i)$| $x_i$∈ , ∈ , = 1, 2, 3, … … … . , }, activation function g(x) and N′ hidden number of nodes.

**Output:** the output weight β

1. Randomly assign input weights $w_i$ and bias $b_i$ , i= 1, 2, ……., N’
2. Calculate the hidden layer output matrix H:

( , + ) … … … … … … ( ,+ )

H=

( , + )… … … … ( , +

3. Calculate the output weight: ***’ = H⁺ y***

Where $\boldsymbol{H^{+}}$ is the Moore-Penrose inverse matrix of T =[$t_1$,……,$t_N$]’

## Advances in Extreme Learning Machine Algorithm

**Huang et al [18]** proposed the original ELM for SLFANNs wherein values for weights and biases were assigned randomly. The output was calculated through mathematical transformation. Over the years a number of variants of ELM have been proposed

**Incremental ELM (I-ELM)**: Huang et al [19] proposed an incremental ELM in which nodes of the hidden layer were added to the hidden layer one. I-ELM added nodes randomly to the hidden layer one by one. The output weights of the existing hidden nodes were freezed when a new hidden node was added. I-ELM is not only efficient for SLFANN with continuous activation functions, but also for SLFNs with piecewise continuous activation functions.

**Convex Incremental ELM (CI-ELM)**:CI-ELM [20] recalculated the output weights of the existing hidden nodes after a new hidden node was added. CI-ELM could achieve faster convergence rates and more compact network architectures than I-ELM while retaining the I-ELM’s simplicity and efficiency.

**Enhanced Incremental ELM (EI-ELM)**: EI-ELM [21]allowed maximum number of hidden nodes, no control parameters need to be manually set by users. Different from the original I-ELM, EI-ELM picked the optimal hidden, node which led to the smallest residual error at each learning step among several randomly generated hidden nodes. EI-ELM could achieve faster convergence rate and much more compact network architect.

**Error Minimized ELM**: Feng et al [22] proposed error minimized ELM. These are capable of growing hidden nodes one by one or group by group to automatically determine the number of hidden nodes in generalized SLFANNs. In the mean time, the output weights are updated regularly and this leads to considerable reduction of computational complexity. The EM-ELMs are therefore considered to be efficient implementation of the original ELM. Some of the areas where EM-ELMs have been successfully implemented include real benchmark regression and classification problems.

**Two Stage ELMs (TS-ELM):**Lan et al. [23] introduced a systematic two stage algorithm (named TS-ELM). In the first stage, a forward recursive algorithm was applied to select the hidden nodes from the candidates randomly generated in each step and add them to the network until the stopping criterion was met. In the second stage, the selected hidden nodes were reviewed to eliminate the insignificance nodes from the network, which drastically reduced the network complexity. TS-ELM with smaller network structure could achieve better or equivalent performance to EI-ELM.

**Online Sequential ELMs (OS-ELM):**Liang et al. [24] presented a sequential learning algorithm referred to as online sequential extreme learning machine (OS-ELM), which can handle both additive and RBF nodes in a unified framework. In OSELM with additive nodes, the input weights linking the input nodes to hidden nodes and biases were randomly generated, and then, the output weights were analytically determined based on the output of hidden nodes. Unlike other sequential learning algorithms, OS-ELM only required the number of hidden nodes to be specified as the conventional ELM.

**Ordinal ELM (O-ELM):**Deng et al. [25] presented an encoding based ordinal regression framework and three ELM-based ordinal regression algorithms. The paper designed an encoding-based framework for ordinal regression which included three encoding schemes: single multi-output classifier, multiple binary classifications with one-against-all decomposition method, and one-against-one method. Based on the framework, the SLFN was redesigned for ordinal regression problems, and the algorithms were trained by the extreme learning machine.

**Symmetrical ELM (S-ELM):**Li et al.[26] proposed a fully complex extreme learning algorithm(named C-ELM). In C-ELM, the ELM algorithm was extended from the real domain to the complex domain. Similar to ELM, the input weights and hidden layer biases of C-ELM were randomly chosen based on some continuous distribution probability, and then, the output weights were simply analytically calculated instead of being iteratively tuned. Then, C-ELM is used for equalization of a complex nonlinear channel with QAM signals.

## Conclusion

The paper presents Extreme Learning Machine and its various variants as a result of the advances made in this field. The paper introduces Extreme Learning machines as an invaluable tool for problems like classification and regression which can be solved using the ELMs using less computational time, higher accuracy, greater ease and less complexity as compared to the other conventional techniques such as Back Propagation Artificial Neural Networks etc. The researchers can use this tool for further research in solving data of high dimensionality and exploring further its universal approximation capability which makes it a very efficient learning algorithm to implement.

## References

[1] Xu XZ, Ding SF, Shi ZZ, Zhu H (2012) Optimizing radial basis function neural network based on rough set and AP clustering algorithm. J Zhejiang Univ Sci A 13(2):131–138.

[2] Chen Y, Zheng WX (2012) Stochastic state estimation for neural networks with distributed delays and Markovian jump. Neural Netw 25:14–20.

[3]. Ding SF, Su CY, Yu JZ (2011) An optimizing BP neural network algorithm based on genetic algorithm. Artif Intell Rev36(2):153–162.

[4]. Francisco FN, Ce´sar HM, Gutie´rrez PA, Carbonero-Ruz M(2011) Evolutionary q-Gaussian radial basis function neural networks for multiclassification. Neural Netw 24(7):779–784

[5]. Ding SF, Jia WK, Su CY, Zhang LW (2011) Research of neural network algorithm based on factor analysis and cluster analysis.Neural Comput Appl 20(2):297–302

[6] Razavi S, Tolson BA (2011) A new formulation for feed forward neural networks. IEEE Trans Neural Netw 22(10):1588–1598.

[7]. Huang GB, Chen L, Siew CK (2006) Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Netw 17(4):879–892.

[8]. Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. Neurocomputing71:3060–3068.

[9]. Huang GB, Chen L (2007) Convex incremental extreme learning machine. Neurocomputing 70:3056–3062.

[10] Rong HJ, Ong YS, Tan AH, Zhu Z (2008) A fast pruned-extreme learning machine for classification problem. Neurocomputing72:359–366.

[11]. Huang GB, Ding X, Zhou H (2010) Optimization method based extreme learning machine for classification. Neurocomputing74:155–163.

[12]. Lim JS, Lee S, Pang HS (2013) Low complexity adaptive forgetting factor for online sequential extreme learning machine (OS-ELM) for application to nonstationary system estimations. Neural Comput Appl 22(3–4):569–576.

[13]. Huang GB, Zhou H, Ding X, Zhang R (2012) Extreme learningmachine for regression and multiclass classification. IEEE TransSyst Man Cybern Part B Cybern 42(2):513–529.

[14]. Wang L, Huang YP, Luo XY, Wang Z, Luo SW (2011) Imaged blurring with filters learned by extreme learning machine.Neurocomputing 74:2464–2474.

[15] G.-B. Huang, Q.-Y. Zhu and C.-K. Siew, “Extreme learning machine: Theory and applications”, Neurocomputing 70 (2006)489–501.

[16] S. Suresh, S. Saraswathi and N. Sundararajan, “Performance enhancement of extreme learning machine for multi-category sparse data classification problems”, Engineering Applications of Artificial Intelligence, 23 (2010) 1149-1157.

[17]. Huang, G.B., X. Ding, and H. Zhou, “Optimization methodbased extreme learning machine for classification”, Neurocomputing, 74(1-3) (2010) 155-163.

[18]. Huang GB, Zhu QY, Siew CK (2004) Extreme learning machine: a new learning scheme of feedforward neural networks. In: Proceedings of international joint conference on neural networks (IJCNN2004), vol 2, no 25–29, pp 985–990

[19]. Huang GB, Chen L, Siew CK (2006) Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Netw 17(4):879–892.

[20]. Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. Neurocomputing 71:3060–3068.

[21]. Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. Neurocomputing71:3460–3468.

[22]. Feng GR, Huang GB, Lin QP, Gay R (2009) Error minimized extreme learning machine with growth of hidden nodes and incremental learning. IEEE Trans Neural Netw 20(8):1352–1357

[23]. Lan Y, Soh YC, Huang GB (2010) Two-stage extreme learning machine for regression. Neurocomputing 73(16–18):3028–3038

[24]. Liang NY, Huang GB, Saratchandran P, Sundararajan N (2006) Afast and accurate on-line sequential learning algorithm for feedforward networks. IEEE Trans Neural Netw 17(6):1411–1423

[25].Deng WY, Zheng QH, Lian SG, Chen L, Wang X (2010) Ordinalextreme learning machine. Neurocomputing 74(1–3):447–456

[26]. Li MB, Huang GB, Saratchandran P, Sundararajan N (2005) Fully complex extreme learning machine. Neurocomputing68:306–314