

## An Efficient Extreme Learning Machine Based on Fuzzy Information Granulation

<https://doi.org/10.3991/ijet.v12i06.7089>

Yuehua Gao

Chongqing University of Science & Technology, Chongqing, China

gaoyuehua2796@163.com

Tianyi Chen

Third Research Institute of China Aerodynamics Research Center, Mianyang, China

wx\_cty@163.com

**Abstract**—In order to improve learning efficiency and generalization ability of extreme learning machine (ELM), an efficient extreme learning machine based on fuzzy information granulation (FIG) is put forward. The new approach not only improves the speed of basic ELM algorithm that contains many hidden nodes, but also overcomes the weakness of basic ELM of low learning efficiency and generalization ability by getting rid of redundant information in the observed values. The experimental results show that the proposed method is effective and can produce desirable generalization performance in most cases based on a few regression and classification problem.

**Keywords**—extreme learning machine (ELM), fuzzy information granulation (FIG), neural networks, support vector machine (SVM)

### 1 Introduction

Extreme learning machine (ELM) [1], proposed by Dr. Huang in 2004, works for the "generalized" single-hidden layer feed-forward networks (SLFNs) [1 2], but there is no need to tune the hidden layer (or called feature mapping) in ELM [3]. Due to the fact that ELM's input weights and hidden neurons' biases do not need to be adjusted during training and one may simply randomly assign values to them, the learning phase of many applications can be completed within seconds [4]. Compared with some classical methods, such as neural networks [5] and support vector regression (SVR) [6], ELM is proved to require fewer optimization constraints and result in simpler implementation, faster learning, and better generalization performance.

Due to the simplicity of ELM's implementations, ELM has been extensively used in classification and regression applications. However, because of some ELM's parameters (the input weights and hidden biases) are randomly chosen, ELM's learning efficiency and generalization ability can't be guaranteed. Usually, there are two ways to solve this problem. One is to find the best parameters and some scholars have finished by some intelligence algorithm. The other is to increase the number of the hid-

den units so as to make these randomly generated parameters approach the best parameters [7]. Nevertheless, the hidden units's increase will add up to the amount of calculation and is inconvenient for the implementation.

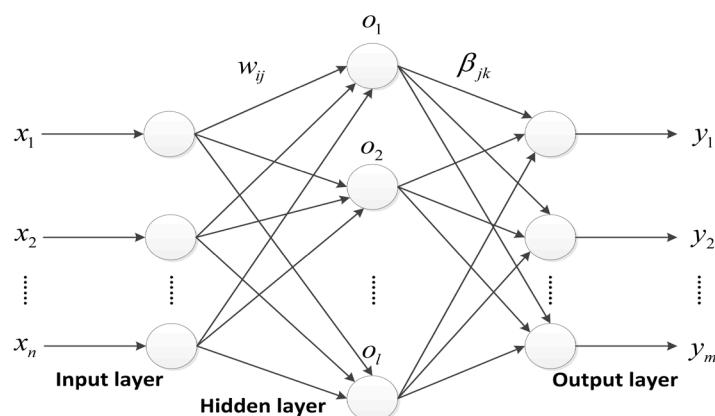
In recent years, information granulation (IG) has been used increasingly as an effective technique to get rid of redundant information in the observed values. IG is the process of forming meaningful entities of information, and fuzzy modeling can be conveniently adopted for information granulation, i.e. fuzzy information granulation (FIG). The FIG approach can transform primary data into a sequence of granules by setting the size of the granulation window, generating the granulated sets [8]. Historical observation data is selected to be fuzzy information granulated and this process can improve training time, training efficiency and ensure test accuracy.

Enlightened by the idea of FIG-SVM (support vector machine based on fuzzy information granulation) algorithm [9], this paper puts forward an efficient extreme learning machine (ELM) based on fuzzy information granulation (FIG) called as FIG-ELM. The FIG-ELM was able to achieve good generalization performance and prediction accuracy with less training time. In this paper, two regression problems and two classification problems are used to examine FIG-ELM's performance and the results show that the FIG-ELM algorithm is superior to conventional ELM, SVM, FIG-SVM, BP, LVQ and DT.[10]

The paper is organized as follows. In Section 2, the extreme learning machine (ELM) is introduced. An Efficient Extreme Learning Machine Based on Fuzzy Information Granulation (FIG-ELM) is discussed in Section 3, and in Section 4 simulation results of some recognized benchmark regression and classification problems are given. In Section 5, conclusions are brought forward.

## 2 Extreme Learning Machine

The typical single hidden layer feed-forward networks (SLFNs) can be expressed as:



**Fig. 1.** SLFN structure

Where  $x$  and  $y$  respectively represent the input element and the output element,  $W_{ij}$  is the weight vector connecting the  $i$ -th input node and the  $j$ -th hidden node,  $\beta_{jk}$  is the weight vector connecting the  $j$ -th hidden node and the  $k$ -th output node.

For  $Q$  arbitrary distinct samples  $(x_i, y_i)$ ,  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ ,  $y_i = [y_{i1}, y_{i2}, \dots, y_{in}]^T \in R^m$ ,  $i=1,2,\dots,Q$ ,  $g(x)$  is activation function and  $b$  is the threshold. The ELM's mathematic model is:

$$T = \sum_{i=1}^l \beta_i g_i(x_j) = \sum_{i=1}^l \beta_i g(w_i \cdot x_j + b_i), j=1,2,\dots,Q \quad (1)$$

where  $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$  is the weight vector connecting the  $i$ -th input node and the hidden nodes,  $x_j = [x_{j1}, x_{j2}, \dots, x_{nj}]^T$  is the input matrix,  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{in}]^T$  is the weight vector connecting the  $i$ -th hidden node and the output nodes and  $b_i$  is the threshold of the  $i$ -th node,  $i=1,2,\dots,l$ .

Eq. (1) can be rewritten as follows

$$H\beta = T' \quad (2)$$

where  $T'$  is the transposition of matrix  $T$  and  $H$  is called the hidden layer output matrix in neural network.

$$\begin{aligned} H(w_1, w_2, \dots, w_l, b_1, b_2, \dots, b_l, x_1, x_2, \dots, x_Q, ) = \\ \left[ \begin{array}{c} g(w_1 \cdot x_1 + b_1)g(w_2 \cdot x_1 + b_2)g(w_l \cdot x_1 + b_l) \\ g(w_1 \cdot x_2 + b_1)g(w_2 \cdot x_2 + b_2)g(w_l \cdot x_2 + b_l) \\ \vdots \\ g(w_1 \cdot x_Q + b_1)g(w_2 \cdot x_Q + b_2)g(w_l \cdot x_Q + b_l) \end{array} \right]_{Q \times l} \end{aligned} \quad (3)$$

According to Eq. (1) and Eq. (2), matrix  $H$  is known when activation function  $g(x)$  is given and the parameters values ( $w, b$ ) are randomly obtained. So, the weight vector  $\beta$  can be solved by the following equation:

$$\min_{\beta} \|H\beta - T'\| \quad (4)$$

The solution is:

$$\hat{\beta} = H^+ T' \quad (5)$$

where  $H^+$  is matrix  $H$ 's Moore-Penrose generalized inverse.

### 3 An Efficient Extreme Learning Machine Based on Fuzzy Information Granulation (FIG-ELM)

#### 3.1 Fuzzy Information Granulation (FIG)

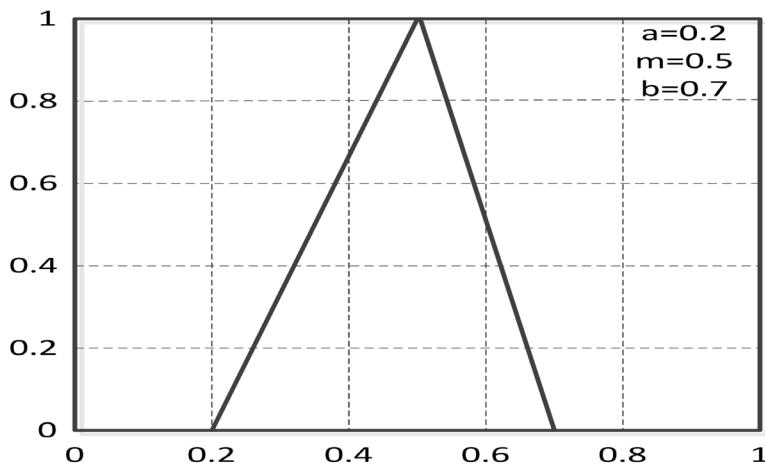
The concept of fuzzy information granulation (FIG) was suggested by Dr.Lotfi and A. Zadehin the 1960s. The FIG approach is implemented to transform the original data into a sequence of granules, gaining more general view at the data that retains only the most dominant components of the original temporal series [8].

For given time series, all time series X can be regard as a window for fuzzification. The task of fuzzification is to create a fuzzy granule P on X, which can reasonably describe inkling G of X [9]. So, the definition of data is:

$$g @x \text{ is } G \quad (6)$$

Fuzzification essentially is a process to ensure a function A, which A is the membership function of G. Generally speaking, there are some common forms of fuzzy granules: triangular type, trapezoidal type, gaussian type, parabolic type and so on. In this paper, triangular type was chosen. The structure of triangular type is shown in the Figure 2. The membership function can be constructed as:

$$A(x, a, m, b) = \begin{cases} 0, & x < a \\ \frac{x-a}{m-a}, & a \leq x \leq m \\ \frac{b-x}{b-m}, & m < x \leq b \\ 0, & x > b \end{cases} \quad (7)$$



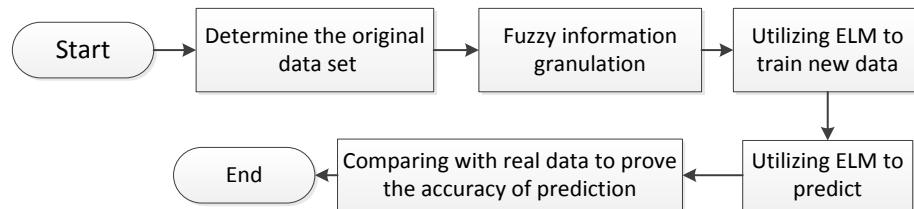
**Fig. 2.** Triangular type structure

### 3.2 FIG-ELM

ELM need not spend much time to tune the input weights and hidden biases of the SLFN by randomly choosing these parameters. Since the output weights are computed based on the input weights and hidden biases, there inevitably exists a set of non-optimal or unnecessary input weights and hidden biases. Thus, in order to make these randomly generated parameters approach the best parameters, we need to increase the number of the hidden units at the cost of increasing the training time.

In view of the situation above, an efficient approach named FIG-ELM combining ELM with FIG is proposed. In order to improve ELM's training time, learning efficiency and generalization ability , FIG is exploited to dispose the original data set, namely, get rid of redundant information in the observed values at first, and then ELM is used to do train granulated data for prediction.

The whole arithmetic processes can be shown in Figure 3:



**Fig. 3.** Flow chart

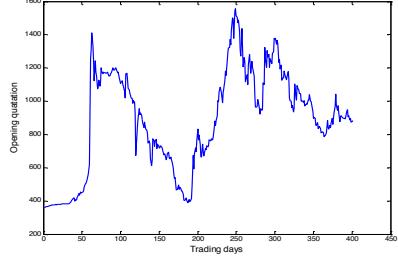
## 4 Experimental Results and Analyses

In order to prove the capabilities of the approach proposed in this paper, some regression problems and classification problems are selected. All the programs are run in MATLAB 2010a environment. In every simulation, there are three different fuzzy particle after granulation: Low, R, Up, and we all only choose fuzzy particle Low for prediction to facilitate the process.

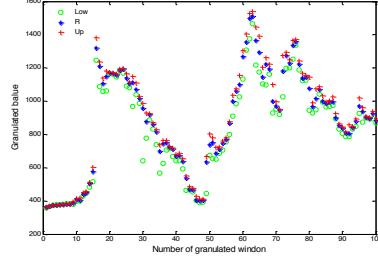
### 4.1 Regression Problems

**Real-World Problem:** The performances of SVM, FIG-SVM, ELM, FIG-ELM are compared on a real-world benchmark data set (Shanghai composite index). We choose 400 groups of data.

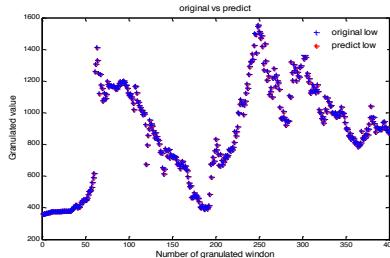
There are 100 hidden nodes assigned for ELM, FIG-ELM and SVM's, FIG-SVM's parameters be are as  $C=10$  and  $g=1.5$ . Figure 4 and Figure 5 show the original data and the new data disposed by FIG respectively. Figure 6 shows the fitting results obtained by these four learning algorithms. Table 1 summarizes the results of real-world regression problem, with regard to the training time, maximum absolute error, least absolute error and the average error for making the proposed algorithm's performance prominent.



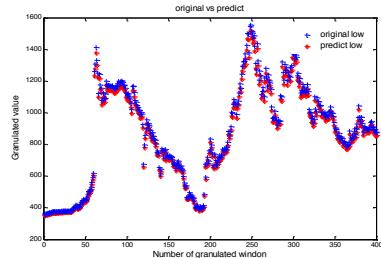
**Fig. 4.** Original data



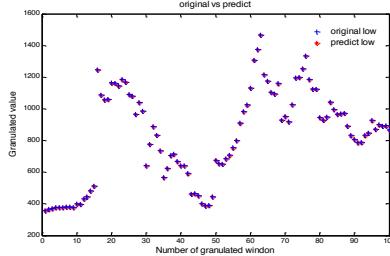
**Fig. 5.** Granulation data



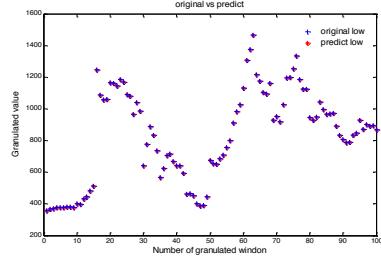
**(a) SVM**



**(b) ELM**



**(c) FIG-SVM**



**(d) FIG-ELM**

**Fig. 6.** Prediction performances of four algorithms

**Table 1.** Performance comparison between four algorithms

Algorithms	Train Time (s)	Maximum Absolute Error	Least Absolute Error	Average Absolute Error
SVM	2.39	0.98	0.02	0.55
FIG-SVM	1.57	0.96	0.05	0.43
ELM	0.73	36	7	21
FIG-ELM	0.45	5.2*10-7	1.1*10-7	3.3*10-7

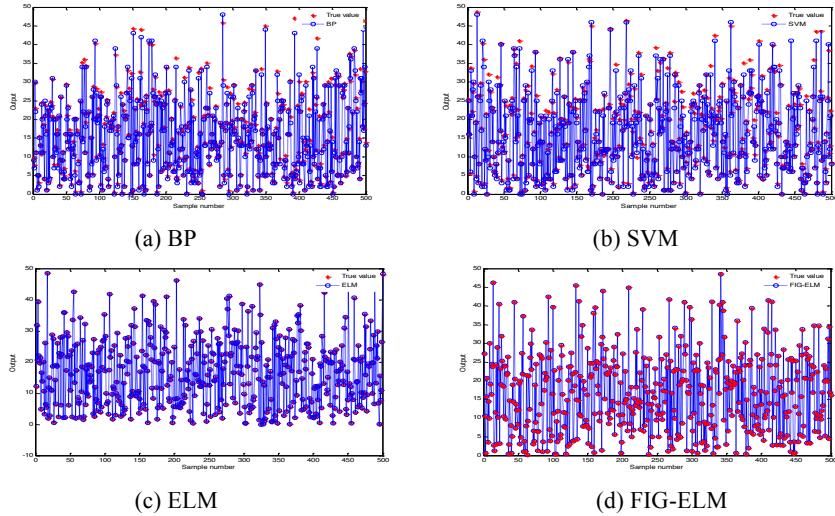
It can be seen from Table 1 that FIG-ELM takes the least time and has the optimal performance. FIG-ELM learning algorithm spends 0.45s obtaining the average absolute error ( $3.3*10^{-7}$ ). However, it respectively takes 2.39s, 1.57s and 0.73s for SVM, FIG-SVM and ELM algorithms to reach a much higher average absolute error(0.55,

0.43 and 21). In addition, it's clear to observe from Figure 6 that the prediction results obtained by FIG-ELM almost coincides with the true values. Obviously, it can be seen that the proposed algorithm outperforms the contrastive algorithms.

**Approximation of nonlinear function:** In this example, four algorithms (BP, SVM, ELM, and FIG-ELM) are used to approximate the nonlinear function. The nonlinear function is represented as:  $y=x^{12}+x^{22}$  (8)

The training data set and testing data set are randomly generated, and 1500 groups of data are used for training and 500 groups for testing.

There are 100 hidden nodes assigned for BP, ELM, FIG-ELM and SVM's parameters be are as  $C=10$  and  $g=1.5$ . Moreover, FIG's number of granulation window is 300, namely, FIG-ELM's 300 groups of training data can represent 1500 groups. The training time, average absolute error and testing mean square error (MSE) of these four algorithms are shown in Table 2. Figure 7 shows the true value of the nonlinear function and the approximated value obtained by these four learning algorithms.



**Fig. 7.** Outputs of BP, SVM, ELM and FIG-ELM learning algorithms

**Table 2.** Performance comparison of nonlinear function

Algorithms	Train Time (s)	Average Absolute Error	MSE
BP	2.07	0.467	0.66
SVM	0.62	0.552	0.88
ELM	0.045	$2.5 \times 10^{-3}$	$3.7 \times 10^{-3}$
FIG-ELM	0.02	$1.45 \times 10^{-4}$	$5.7 \times 10^{-4}$

As can be seen from Figure 7 (a) and (b), the BP and SVM learning algorithms' testing error is obvious. We can see that FIG-ELM has the best prediction performance from Figure 7 and Table 2, it not only has the shortest training time, but also the minimum MSE.

## 4.2 Classification Problems

**Small-scale Real Classification Application:** The performance comparison of the proposed FIG-ELM algorithm and many other popular algorithms (BP, SVM, LVQ and DT) has been conducted through a real medical diagnosis problem: Breast Cancer, was produced in the University of Wisconsin-Madison. There are 10 attributes in total and the hidden nodes are set as 100. The total data are 569 groups, and 500 groups of data are randomly selected for training and the rest 69 groups for testing. Similarly, FIG's number of granulation window is 100, namely, FIG-ELM's 100 groups of training data can represent 500 groups. One hundred trials have been conducted to compare the performance of these algorithms. Simulation results, including the average training accuracy, the average testing accuracy, and the training time, are shown in Table 3.

**Table 3.** Performance comparison in real medical diagnosis application: breast cancer

Algorithms	Training Time (s)	Training Accuracy (%)	Testing Accuracy (%)
BP	2.06	81.16	73.9
SVM	1.03	100	67.97
LVQ	0.33	89	91.33
DT	0.25	87.85	90
FIG-ELM	0.07	95.74	95.65

It's easy to see from Table 3 that SVM's testing accuracy is lowest though it obtains the best training accuracy. BP performs very poor in this case, with testing accuracy being at 0.739 and longest training time. LVQ and DT algorithms obtain better performance at training time, training and testing accuracy, but is still not very well. By contrast, FIG-ELM algorithm runs the fastest among these five algorithms and obtains the best performance.

**Medium Size Classification Problem—Character Recognition:** The total data are 6000 groups, and 5000 groups of data are randomly selected for training and the rest 1000 groups for testing. In addition, FIG's number of granulation window is 500. One hundred trials have been conducted to compare the performance of these algorithms. Simulation results, including the average training accuracy, the average testing accuracy, the training time and the number of hidden nodes are shown in Table 4.

**Table 4.** Performance comparison in medium size applications: character recognition

Algorithms	Hidden Nodes	Training Time (s)	Training Accuracy (%)	Testing Accuracy (%)
SVM	-	77.2	100	86.6
ELM	50	0.22	81.64	81.2
FIG-ELM	50	0.13	93.78	91.4
ELM	100	1.13	90.54	89.2
FIG-ELM	100	0.16	97.58	92.7
ELM	500	2.04	93.62	91.3
FIG-ELM	500	0.69	100	95.45

As observed from Table 4, compared with other algorithms, FIG-ELM's training time is unattainable. It's easy to find that SVM is over-fitting and has a long training time. ELM and FIG-ELM algorithms' training accuracy and testing accuracy are increasing with the hidden nodes' increasing, but ELM's training time is longer than FIG-ELM's clearly, which are at 2.04s and 0.69s when the number of hidden nodes are 500. Moreover, FIG-ELM's training and testing accuracy are the best. Hence, judging from the simulation results, FIG-ELM has stronger learning efficiency and generalization ability than ELM, especially when there is a large amount of data and need many hidden nodes.

## 5 Conclusions

This paper combines extreme learn machine (ELM) with fuzzy information granulation (FIG). The new approach not only improves the speed of basic ELM algorithm that contains many hidden nodes, but also overcomes the weakness of basic ELM of low learning efficiency and generalization ability. The effectiveness is demonstrated by simulating four examples: two regression problems (a regression dataset and a nonlinear function approximation) and two classification (Breast Cancer and Character Recognition). Experimental results show that FIG-ELM algorithm has a higher potential of enhancing the predictive accuracy, robustness and reducing the training time.

Furthermore, efforts will be further made to achieve a higher efficient level. We also need to improve ELM algorithm, such as online sequential ELM, etc.

## 6 References

- [1] Huang G.B., Zhu Q.Y., Siew C.K., Extreme learning machine: Theory and applications. Neurocomputing (S0925-2312), 2006, 70(1): 489-501. <https://doi.org/10.1016/j.neucom.2005.12.126>
- [2] Huang G.B., Ding X.J., Zhou H.M., Optimization method based extreme learning machine for classification. Neurocomputing, 2010, 74(1-3): 155-163. <https://doi.org/10.1016/j.neucom.2010.02.019>
- [3] Huang G.B., Ding X.J., Zhou H.M., et al. Extreme learning machine for regression and multiclass classification. IEEE Trans. On Systems, Man, and Cybernetics, 2012, 42(2): 513-529. <https://doi.org/10.1109/TSMCB.2011.2168604>
- [4] Huang G.B., Zhu Q.Y., C.-K. Siew, Extreme learning machine: A new learning scheme of feedforward neural networks. In Proceedings of International Joint Conference on Neural Networks (IJCNN2004), Budapest, Hungary, 2004.
- [5] Zhang R.D., Wang Q.S., Neural network based multi-step predictive control for nonlinear system. Control and Decision, 2005, 20(3): 332-336.
- [6] Cui G.M., Sun T., Zhang Y., Application of Support Vector Machine (SVM) in Prediction of Molten Iron Temperature in Blast Furnace. Control Engineering of China, 2013, 20(5): 809-817.
- [7] Huang G.B., Siew C.K., Extreme Learning Machine with Randomly Assigned RBF Kernels. International Journal of Information Technology, 2005.

- [8] Liao R.J., Zheng H.B., Grzybowski S. et al, Fuzzy information granulated particle swarm optimisation-support vector machine regression for the trend forecasting of dissolved gases in oil-filled transformers. IET Electric Power Applications, 2011, 5(2): 230- 237. <https://doi.org/10.1049/iet-epa.2010.0103>
- [9] Ding S.F., Han Y.Z., Yu J.Z. et al, A fast fuzzy support vector machine based on information granulation. Neural Comput & Applic, 2013, 23(suppl 1): S139-S144.
- [10] Han F., Yao H.F., Ling Q.H., An improved extreme learning machine based on particle swarm optimization. Neurocomputing, 2011, 116: 87-93.

## 7 Authors

**Yuehua Gao** is a assistant professor of Chongqing University of Science & Technology, Chongqing, China. He mainly works at machine design and control.

**Tianyi Chen** is a senior engineer at third research institute of China aerodynamics research center, Mianyang, China. His research interests including mechanical engineering, engineering management and automation.

Article submitted 01 May 2017. Published as resubmitted by the authors 06 June 2017.