# MODIFICATION OF HIDDEN LAYER WEIGHT IN EXTREME LEARNING MACHINE USING GAIN RATIO

Fetty Tri Anggraeny and Intan Yuniar Purbasari
Department of Informatics Engineering, University of Pembangunan Nasional "Veteran" East Java, Surabaya, Indonesia
e-Mail: fetty.ta@gmail.com

**ABSTRACT**

Extreme Learning Machine (ELM) is a method of learning feed forward neural network quickly and has a fairly good accuracy. This method is devoted to a feed forward neural network with one hidden layer where the parameters (i.e. weight and bias) are adjusted one time randomly at the beginning of the learning process. In neural network, the input layer is connected to all characteristics/features, and the output layer is connected to all classes of species. This research used three datasets from UCI database, which were Iris, Breast Wisconsin, and Dermatology, with each dataset having several features. Each characteristic/feature of the data has a role in the process of classification levels, starting from the most influencing role to non-influencing at all. Gain ratio was used to extract each feature role on each datasets. Gain ratio is a method to extract feature role in order to develop a decision tree structure. In this study, ELM structure has been modified, where the random weights of the hidden layer were adjusted to the level of each feature role in determining the species class, so as to improve the level of training and testing accuracy. The proposed method has higher classification accuracy rate than basic ELM on all three datasets, which were 99%, 96%, and 82%, respectively.

**Keywords:** extreme learning machine, feature weight, information gain.

## INTRODUCTION

Extreme Learning Machine (ELM) is a learning method for a single hidden layer feed-forward neural network (SLFN) that resolves concerns raised by the use of back propagation methods. The learning stages in backpropagation take much longer than ELM despite using the same neural network configuration, i.e. one hidden layer. This is because the Feed Forward Neural Network uses a learning algorithm based on gradient that works slowly, and all the adjustable parameters are repeatedly adjusted in the learning process until an iteration stopping criteria is reached (Huang et al., 2004). Meanwhile, the learning stage in ELM requires only one iteration and the weight parameters are set once randomly, despite having only one hidden layer (Huang et al., 2006). Although ELM gives quickly learning process, it has a fairly good accuracy too.

ELM has been gaining high attention from researchers since its announcement (Huang et al., 2015). It is not only researched within the scope of classification problems, but also in the scope of regression and clustering problems. With its advantages, ELM is considered appropriate to resolve various problems with big data and real-time applications, such as in medical field, image processing, computer vision, etc. They paper show that ELM and its variants are efficient, accurate and easy to implement, also in hardware needs (robot).

ELM research result by Huang et al. (2004) showed that ELM has a higher accuracy than other methods, such as SVM, AdaBoost, C4.5, and RBF. Two datasets used were real diabetes medical diagnosis dataset and forest type dataset.

Data in classification problems consists of several features that represent an object of a specific class species. Each feature has a role level (i.e. a weight feature) which can be categorized as high and low role levels. Weighting is generally performed in feature selection stage and aims to analyze the data and generate the level of a feature role in the classification process. There are two approaches in feature selection process: filter approach and wrapper approach (Karegowda et al., 2010). The filter approach is carried out separately from the classification engine, and is an important preprocessing stage. Since it is separated from the classification engine, the outputs of this feature selection approach can be used by different classification engines. The wrapper feature selection approach uses a classification engine to determine the role levels of each feature in the classification process. In other words, the filter approach is simpler and faster than the wrapper approach. Some of the feature selection methods using filter approach, among others, are gain ratio (Karegowda et al., 2010, Priyadarsini et al., 2011, and Anggraeny et al., 2013), particle swarm intelligence (PSO) (Yang et al., 2007 and Huang et al., 2008), and differential evolution (Khushaba et al., 2011). Some feature selection methods using wrapper approach are ant colony optimization (ACO) (Kanan et al., 2008) and sequential floating forward selection (SFFS) (Liao et al., 2010).

Karegowda et al. (2010) used gain ratio (GR) as feature selection technique, and Radial Basis Function Network (RBF) and Back Propagation Neural Network (BPN) as classifier. The research result showed that classification accuracy for BPN is about 72.88%, GR-BPN is 78.21%, RBF is 81.20%, and GR-RBF is 86.46%.

Anggraeny et al. (2013) used gain ratio (GR) as feature selection technique and Voting of Neural Network Particle Swarm Optimization (VNNPSO) as classifier. The result showed that GR-VNNPSO did not improve all datasets accuracy rate. The method improved classification accuracy on dermatology dataset about 13.28% and reduced accuracy rate about 0.45% on iris dataset and 3.49% on breast cancer Wisconsin. Although only one dataset showed better accuracy, the increasing value was much higher than the decreasing value.

Priyadarsini et al. (2011) used gain ratio as feature subset selection, combined by Naïve Bayes as classifier and K-Means for clustering. Ranking method is used in adult dataset to select a subset of 7 attributes

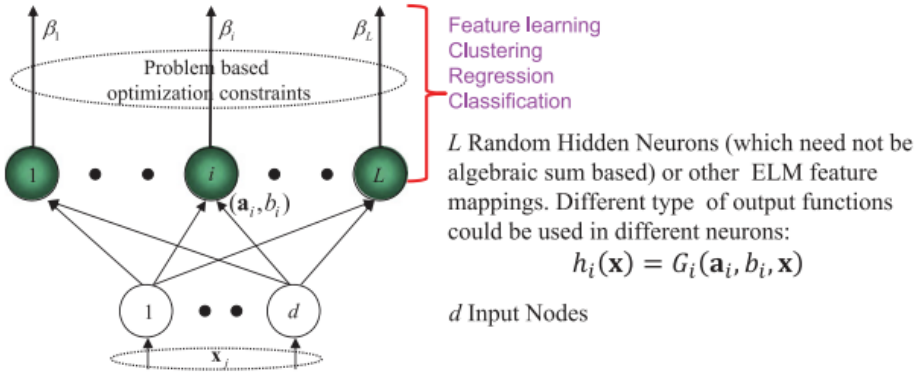


**Figure-1**. ELM Architecture (Huang et al., 2015).

from the original dataset of 10 attributes. Both on classification and clustering, the utility of the dataset is unaffected by the attribute reduction.

In this research, we will add feature weight using gain ratio method as a multiplier factor of hidden layer random weight in Extreme Learning Machine, in the hope that this modification of ELM weighting will increase the accuracy of training and testing data.

## METHODOLOGY

### Extreme Learning Machine

Extreme Learning Machine (ELM) is a learning method in single hidden layer feed-forward neural network which is faster and generally has a higher accuracy than backpropagation (Huang et al., 2004). The configuration of the neural network consists of *d* input nodes in accordance with the number of features, *L* hidden nodes, and *m* output nodes in accordance with the number of classes (Figure 1). Unlike backpropagation, ELM is not only aims to achieve a minimum error learning, but also the smallest norm of weight. The function of the output of ELM is (Huang et al., 2015):

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x) = h(x)\beta \quad (1)$$

where $\beta = [\beta_1, \cdots, \beta_L]^T$ is the output weight between $L$ hidden layer nodes and $m \geq 1$ output nodes, and $h(x) = [h_1(x), \cdots, h_L(x)]$ is the ELM nonlinear feature mapping (Figure 2), where $h_i(x)$ is the output of the *i*-th node to the hidden layer. The output function in hidden nodes may use a different activation function for each node. The output function in hidden nodes is notated as follows:

$$h_i(x) = G(a_i, b_i, x), \quad a_i \epsilon R^d, \; b_i \epsilon R \quad (2)$$

where $G(a_i, b_i, x)$, with hidden nodes parameter $(a, b)$ as a non-linear activation function, $x_j$ is the *j*-th input value, $a_i$ is random weight of *i*-th input layer, and $b_i$ is bias of the *i*-th hidden node.

$$G(a_i, b_i, x) = G(a_i . x_j + b_i) \quad (3)$$

ELM consists of two main stages: random mapping feature and linear completion of parameters. In the first phase, ELM randomly initializes the weights of hidden layer nodes to map input data into ELM feature space. The hidden node's parameters (a, b), are randomly initialized by a probability distribution. In the second phase, the weights connecting the hidden layer and output layer (β) are resolved by minimizing the error output:

$$\min_{\beta \epsilon R^{L \times m}} \|H\beta - T\|^2 \quad (4)$$

where *H* is the matrix of hidden layer output, *T* is target data training matrix, and $\|.\|$ denotes Frobenius norm.

### Gain Ratio

Gain ratio is an improvement of information gain. Information gain is used to form the induction of a decision tree (ID3), while the gain ratio is used on C4.5 algorithm, which is an improvement algorithm of ID3 (Asha et al., 2010). Information gain produces bias; it prefers features with many variations of values rather than features which has little variation despite being more informative. For example, let us look at a unique feature of a student data, such as its ID, in student table of a database. A separation using student ID creates a lot of partitions, as each data record has a unique value, which is student ID (Asha et al., 2010).

Let *S* is a data sample set and *m* is the number of classes. The entropy or information approximation to classify a sample is:

$$I(S) = -\sum_{i=1}^{m} p_i log_2(p_i) \quad (5)$$

where $p_i$ is the sample probability with a $class_i$ conclusion.

Let feature/attribute A has a value variation of *v*. Let $s_{ij}$ is the number sample class $C_i$ in subset $S_j$. $S_j$ consists of samples in *S* having a value $a_j$ from *A*. The entropy based on the division of the subset of attribute A is:

$$E(A) = -\sum_{i=1}^{m} I(S) \frac{s_{1i} + s_{2i} + \cdots s_{mi}}{s} \quad (6)$$

Gain information to branch attribute A is:

$$Gain(A) = I(S) - E(A) \quad (7)$$

C4.5 uses the gain ratio by applying a normalization of the gain information obtained from:

$$SplitInfo(S) = -\sum_{i=1}^{v} (|S_i|/|S|) log_2 (|S_i|/|S|) \quad (8)$$

Gain ratio is then computed using the following formula:

$$GainRatio(A) = Gain(A)/SplitInfo(S) \quad (9)$$

Attribute with the highest gain ratio is selected as a splitting attribute.

**Feature Weight Adjusted on Extreme Learning Machine**

The proposed approach is completely described in the following FW-ELM algorithm in Figure 2. ELM preferred to use only random weights as parameter for hidden layer nodes. In the proposed method, first we calculate the feature role using gain ratio method for all features in dataset. These features role will be used as multipliers factor together with ELM random weighting as the input of neural network structure.

This research added one weight, i.e. feature role, which was obtained using gain ratio method. With the additional feature weight, the formulation in hidden layer becomes:

$$h_i(x) = G\left(a_i, b_i, x, f_j\right), \quad a_i \epsilon R^d, \; b_i \epsilon R \tag{10}$$

$$G\left(a_i, b_i, x, f_j\right) = G(f_j . a_i . x_j + b_i) \tag{11}$$

Where $f_j$ is the feature weight of *j*-th input (feature) using gain ratio method.

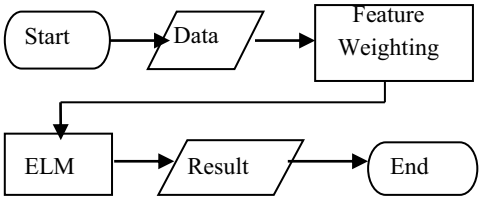


**Figure-2**. Methodology

**RESULTS and DISCUSSION**

Several datasets from UCI database were used for testing purpose as listed in Table 1, which were Iris, Breast Wisconsin, and Dermatology. The phase of feature weight computation using gain ratio gave results in the form of an order of importance of features from the largest to the smallest, as listed in Table 2, along with their respective gain ratio values.

Trials were performed 10 times for each dataset, and all data were used both as training and testing. The parameter assessed was the level of accuracy. Table 3 shows a comparison of SLFN tests between Feature Weighted-ELM (FW-ELM), ELM, and GR-VNNPSO (Anggraeny et al, 2013), in terms of classification accuracy (%) and computing time (s). For the original ELM method by Huang et al (2004), the datasets used in their published paper were different from ours. However, we were able to obtain their source code from Huang's website[1] and run it on our datasets. The comparison between ELM and FW-ELM aimed to investigate the effect of variable addition, i.e. feature weight, in ELM architecture. In the case of comparison between FW-ELM and GR-VNNPSO, the latter applied gain ratio before classification process but using different classifier method. This trial was performed using all features in each classifier.

Based on trial results shown in Table 3, the FW-ELM gave a relatively better accuracy than ELM. This showed that the addition of feature weight parameters in the ELM configuration was able to improve the accuracy on all three datasets significantly. In terms of training time, FW-ELM was faster than ELM, provided that feature weight computation process is carried out outside the classification engine. Compared with GR-VNNPSO, FW-ELM has less accuracy on all dataset but faster in computing time.

**Table-1**. Dataset characteristics

| Dataset | feature | class | ∑ data |
|---|---|---|---|
| Iris | 4 | 3 | 150 |
| Breast Wisconsin | 9 | 2 | 699 |
| Dermatology | 34 | 15 | 366 |

Source: (Blake et al., 1998)

**Table-2**. Order of features according to gain ratio value

| **Dataset** | **#feature** | **Gain Ratio** |
|---|---|---|
| Iris | 4 | 0.871: 4, 0.734: 3, 0.381: 1, 0.242: 2 |
| Breast Wisconsin | 9 | 0.675: 2, 0.66: 3, 0.564: 6, 0.543: 7, 0.505: 5, 0.466: 8, 0.459: 1, 0.443: 4, 0.198: 9 |
| Dermatology | 34 | 0.7715: 31, 0.7254: 27, 0.7237: 33, 0.7221: 6, 0.7111: 29, 0.7094: 12, 0.7019: 15, 0.6829: 25, 0.6741: 8, 0.628: 30, 0.6012: 22, 0.5919: 20, 0.5303: 21, 0.5297: 34, 0.527: 7, 0.438: 9, 0.4291: 24, 0.3993: 10, 0.3707: 28, 0.3251: 14, 0.325: 16, 0.3171: 26, 0.2941: 23, 0.2911: 11, 0.2674: 5, 0.1978: 3, 0.1769: 2, 0.1687: 19, 0.1599: 13, 0.1491: 4, 0.098: 1, 0.0959: 18, 0.0833: 17, 0.0598: 32 |

**Table-3**. Average on accuracy (%) and computing time (s) of SLFN in FW-ELM, ELM, and GR-VNNPSO

| **Dataset** | **FW-ELM** | | **ELM** | | **GR-VNNPSO (Anggraeny et al, 2013)** | |
|---|---|---|---|---|---|---|
| | **accuracy (%)** | **computing time (s)** | **accuracy (%)** | **computing time (s)** | **accuracy (%)** | **computing time (s)** |
| Iris | 99.56 | 0.019 | 99.89 | 0.05 | 100 | 130.25 |
| Dermatology | 81.77 | 0.013 | 63.71 | 0.093 | 100 | 445.49 |
| Breast Wisconsin | 96.16 | 0.033 | 93.49 | 0.042 | 97.71 | 145.73 |

[1]http://www.ntu.edu.sg/eee/icis/cv/egbhuang.htm

There are features that have a big enough role in helping classification process and those that weaken of all the features possessed by a dataset. For the next improvement, this research can be expanded by adding a feature selection method, so that the features used in classification are those that have major roles in the classification process.

## CONCLUSION

In this research, we have modified ELM weighting training of neural network. This approach implemented gain ratio in order to calculate each feature weight in classification process. These weights were used as input to the neural network, and used as weight factor on ELM training process. Moreover, compared with two other methods, FW-ELM appeared to be a promising method as a classifier. In general, FW-ELM has successfully combines high accuracy classification and effectiveness on computing time into becoming a better classifier.

## REFERENCES

Anggraeny F.T., Widiasri M. 2013. Voting of Artificial Neural Network Particle Swarm Optimization Biclassifier using Gain Ratio Feature Selection. KURSOR Journal: Research on Computing and Its Application, Vol. 7, No. 2, pp. 69-74, ISSN: 0216-0544.

Blake C.L., Merz C.J. 1998. University of California at Irvine Repository of Machine Learning Databases, University of California, Irvine, 1998. http://www.ics.uci.edu/mlearn/MLRepository.html.

Huang C.L., & Dun J.F. 2008. A distributed PSO–SVM Hybrid System with Feature Selection and Parameter Optimization. Journal of Applied Soft Computing, Vol. 8, Issue 4, pp. 1381–1391.

Huang G.B., Zhu Q.Y., Siew C.K. 2004. Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks. Proceedings of IEEE International Joint Conference on Neural Network (IJCNN), Vol. 2, pp.985-990.

Huang G.B., Zhu Q.Y., Siew C.K. 2006. Extreme Learning Machine: Theory and Applications. Journal of Neurocomputing, Vol. 70, Issues 1-3, pp. 489-501.

Huang G., Huang G.B., Song S., You K. 2015. Trends in Extreme Learning Machine: A review. Journal of Neural Network, Vol. 61, pp. 32-48.

Kanan H.R., Faez K. 2008. An improved feature selection method based on ant colony optimization (ACO) evaluated on face recognition system. Journal of Applied Mathematics and Computation, Vol. 205, Issue 2, pp. 716–725.

Karegowda A.G., Manjunath A.S., Jayaram M.A. 2010. Comparative Study of Attribute Selection Using Gain Ratio Correlation based feature selection. International Journal of Information Technology and Knowledge Management, Vol. 2, No. 2, pp. 271-277.

Khushaba R.N., Al-Ani A., Al-Jumaily A. 2011. Feature Subset Selection using Differential Evolution and a Statistical Repair Mechanism. Journal of Expert Systems with Applications, Vol. 38, Issue 9, pp. 11515–11526.

Liao T.W. 2010. Feature extraction and selection from acoustic emission signals with an application in grinding wheel condition monitoring. Journal of Engineering Applications of Artificial Intelligence, Vol. 23, Issue 1, pp. 74–84.

Priyadarsini, R.P., Valarmathi, M.L., Sivakumari, S. 2011. Gain Ratio Based Feature Selection Method For Privacy Preservation. ICTACT Journal On Soft Computing, Vol. 01, Issue. 04, pp 201-205.

Yang H.C., Zhang S.B., Deng K.Z., DU P.J. 2007. Research into a Feature Selection Method for Hyperspectral Imagery Using PSO and SVM. Journal of China University of Mining & Technology, Vol. 17, Issue 4, pp. 473–478.