On the suitability of Extreme Learning Machine for gene classification using feature selection

J. Sánchez-Monedero, M. Cruz-Ramírez, F. Fernández-Navarro, J. C. Fernández, P.A. Gutiérrez, C. Hervás-Martínez
Department of Computer Science and Numerical Analysis
University of Cordoba
Córdoba, Spain
Email: {i02samoj,i42crram,i22fenaf,jcfernandez,pagutierrez,chervas}@uco.es

***Abstract*—This paper studies the suitability of Extreme Learning Machines (ELM) for resolving bioinformatic and biomedical classification problems. In order to test their overall performance, an experimental study is presented based on five gene microarray datasets found in bioinformatic and biomedical domains. The Fast Correlation-Based Filter (FCBF) was applied in order to identify salient expression genes among the thousands of genes in microarray data that can directly contribute to determining the class membership of each pattern. The results confirm that the ELM classifier is a promising candidate for improving Accuracy and Minimum Sensitivity.**

***Keywords*-Extreme Learning Machine; Neural Network; bioinformatic; gene microarray; feature selection**

I. Introduction

The importance of the use of Artificial Neural Networks (ANNs) in classifying microarray gene expression has been indicated as an alternative to other techniques in several research studies [1], [2] due to their flexibility and high degree of Accuracy in fitting to experimental data. This study focuses on the Extreme Learning Machine (ELM) [3], [4] with different basis functions. ELM classifiers have been successfully employed in different pattern recognition problems including the classification of microarray genes [5].

ELM both avoids some problems like local minima, improper learning rates and over-fitting, which are commonly faced by gradient-descent-based algorithms [4] (such as BP [6] or $iRprop^+$ [7] algorithms) and it also completes the training very fast. In a previous work on gene classification by Zhang et. al. [5], ELM was employed with a sigmoid activation function. This study contemplates the suitability of the Radial Basis Function (RBF) for ELM (ELM-RBF) in the classification of microarray genes and compares this case to the one using sigmoid activation.

The motivation for applying feature selection (FS) techniques has shifted from optional status to become a real prerequisite for model building. The main reason is the high-dimensional nature of many modelling tasks in this field. A typical microarray dataset may contain thousands of genes but only a small number of samples (often less than two hundred). In addition, feature selection can minimize the effect of noise introduced by some variables.

Based on the generation procedure, FS can be divided into individual feature ranking (FR) and feature subset selection (FSS). FR measures feature-class relevance and then rank features by considering their scores and selecting the top-ranked ones. In contrast, FSS attempts to find a set of features with good performance. Hybrid models were proposed to handle large datasets and to take advantage of the above two approaches (FR, FSS). In this work, the relevant features are obtained by the Fast Correlation-Based Filter (FCBF), a hybrid approach proposed in [8].

The advantage of ELM-RBF is that it is an easy method for the user to work with. The only parameter the user needs to adjust in ELM is the number of nodes, and this is automatically chosen by the applied cross-validation procedure. The parameters related to the RBF basis function are automatically configured analytically.

This paper is organized as follows: Section II briefly explains the concepts of Accuracy and Minimum Sensitivity used for measuring classifier performance in multiclass problems; Section III presents the ELM algorithm considered in this work; Section IV introduces the feature selection algorithm used in this paper; Section V describes the experiments carried out and discusses the results obtained. Finally, Section VI completes the paper with its main conclusions and the future directions suggested by this study.

II. Classifier performance measurement

A classification problem with Q classes and N training patterns is considered with g as a classifier obtaining a $Q \times Q$ confusion matrix $M(g) = \left\{ n_{ij};\ \sum_{i,j=1}^{Q} n_{ij} = N \right\}$ where n_{ij} represents the number of times the patterns are predicted by classifier g to be in class j when they really belong to class i.

Let us denote the number of patterns associated with class i by $n_i = \sum_{j=1}^{Q} n_{ij}, \quad i = 1, \ldots, Q$. First two scalar measures are defined that take the elements of the confusion matrix into consideration from different points of view. Let $S_i = n_{ii}/n_i$ be the number of patterns correctly predicted to be in class i with respect to the total number

978-1-4244-8136-1/10/$26.00 ©2010 IEEE

of patterns in i class (Sensitivity for class i). Therefore, the Sensitivity for class i estimates the probability of correctly predicting a class i example. From the above quantities the Minimum Sensitivity (MS) of the classifier is defined as the minimum value of the sensitivities for each class, $MS = \min\{S_i;\ i = 1, \ldots, Q\}$. The Correct Classification Rate or Accuracy is defined, $C = (1/N)\sum_{j=1}^{Q} n_{jj}$, which is the rate of all the correct predictions. Note than this approach do not consider unclassifiable patterns.

Our objective is to properly configure ELM in order to get the optimum classifier for both Accuracy and Minimum Sensitivity. Note that it is possible to find here classifiers with a high level of C but low values of MS, particularly in imbalanced dataset problems [9].

III. Extreme learning machine

This Section briefly presents Extreme Learning Machine algorithms for single-layer feedforward neural networks (SLFN). Let us consider the training set given by N samples $D = \{(\mathbf{x}_j, \mathbf{y}_j) : \mathbf{x}_j \in R^K, \mathbf{y}_j \in R^Q, j = 1, 2, \ldots, N\}$, where $\mathbf{x}_j$ is a $K \times 1$ input vector and $\mathbf{y}_j$ is a $Q \times 1$ target vector in a multiclass problem (K is the number of variables of each pattern and Q the number of classes).

Let us consider a SLFN with M nodes in the hidden layer given by $f(\mathbf{x}, \boldsymbol{\theta}) = (f_1(\mathbf{x}, \boldsymbol{\theta}_1), f_2(\mathbf{x}, \boldsymbol{\theta}_2), \ldots, f_Q(\mathbf{x}, \boldsymbol{\theta}_Q))$:

$$f_l(\mathbf{x}, \boldsymbol{\theta}_l) = \beta_0^l + \textstyle\sum_{j=1}^{M} \beta_j^l \phi_j(\mathbf{x}, \mathbf{w}_j), l = 1, 2, \ldots, Q,$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_Q)^T$ is the transpose matrix containing all the neural net weights, $\boldsymbol{\theta}_l = (\boldsymbol{\beta}^l, \mathbf{w}_1, \ldots, \mathbf{w}_M)$ is the vector of weights of the l output node, $\boldsymbol{\beta}^l = \beta_0^l, \beta_1^l, \ldots, \beta_M^l$ is the vector of weights of the connections between the hidden layer and the lth output node, $\mathbf{w}_j = (w_{1j}, \ldots, w_{Kj})$ is the vector of weights of the connections between the input layer and the jth hidden node, Q is the number of classes in the problem, M is the number of basis function units, RBFs in this case, in the hidden layer, $\mathbf{x}$ is the input pattern and $\phi_j(\mathbf{x})$ a generic basis function.

Suppose that a SLFN is being trained with M-nodes in the hidden layer to learn the N samples of set D. The linear system $f(\mathbf{x}_j) = \mathbf{y}_j, j = 1, 2, \ldots, N$, can be written in a more compact format as $\mathbf{H}\boldsymbol{\beta} = \mathbf{Y}$, where $\mathbf{H}$ is the hidden layer output matrix of the network:

$$\mathbf{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{w}_1, \ldots, \mathbf{w}_M) = \begin{bmatrix} \phi(\mathbf{w}_1 \cdot \mathbf{x}_1) & \cdots & \phi(\mathbf{w}_M \cdot \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi(\mathbf{w}_1 \cdot \mathbf{x}_N) & \cdots & \phi(\mathbf{w}_M \cdot \mathbf{x}_N) \end{bmatrix}_{N \times M},$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_M \end{bmatrix}_{M \times Q} \quad \text{and } \mathbf{Y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_N \end{bmatrix}_{N \times Q}.$$

The minimum norm least-square (LS) solution is unique and has the smallest norm among all the LS solutions.

In the case of RBF, the hidden layer output is composed of M kernels, which are usually Gaussian:

$$\phi_j(\mathbf{x}) = \phi_j(\mu_j, \sigma_j, \mathbf{x}) = \exp\left(\frac{\| \mathbf{x} - \mu_j \|^2}{\sigma_j}\right),$$

where μ_j is the kernel center and σ_j the kernel width. The ELM have been proposed for training ANNs with RBF nodes in [10], where both kernel centers μ_j and widths σ_j are arbitrarily assigned and output weights $\hat{\boldsymbol{\beta}}$ are calculated by using the Moore-Penrose (MP) generalized inverse of the hidden layer output matrix $\mathbf{H}$. On the other hand, in Optimally Pruned ELM (OPELM) [11] the Gaussian kernels have their centers taken randomly from the data points, like in [12], and widths randomly drawn between 20% percentile and 80% percentile of the distance distribution in the input space, as suggested in [13].

The present work also includes the Gaussian RBF to the ELM source code[1]. The initialisation of the Gaussian parameters is carried out in the same way as described in OPELM.

One of the advantages of ELM over other methods is that the only parameter that the user must properly adjust is the number of hidden nodes. In this work, a cross-validation process has been designed which evaluates the number of nodes suitable for maximizing both Accuracy (C) and Minimum Sensitivity (MS). Cross-validation is performed by testing a range of M number of hidden nodes in a 10-fold procedure using only the training data. For each M value to be tested, the cross-validation procedure is run three times with the same data partition. Therefore the total number of executions over the training data is 30. The M considered as optimal is the one shown in the following equation:

$$\hat{M} = \arg\max_{M_i} \frac{\overline{C}_{M_i} + \overline{MS}_{M_i}}{2},$$

where $\overline{C}_{M_i}$ is the mean C and $\overline{MS}_{M_i}$ is the mean MS obtained by the algorithm using M_i number of nodes in the hidden layer.

IV. Feature Selection: Fast Correlation-Based Filter (FCBF)

The limitations of FR and FSS approaches, in high-dimensional spaces, clearly suggest the need for a hybrid model. The FCBF method can be labeled as this kind of framework, Hybrid-Generation Feature Selection.

In feature subset selection, it is a fact that two types of features are generally perceived as being unnecessary: features that are irrelevant to the target concept, and features that are redundant given other features.

Notions of feature redundancy are normally in terms of feature correlation. It is widely accepted that two features are

[1] For ELM source codes, refer to http://www.ntu.edu.sg/eee/icis/cv/egbhuang.htm

redundant to each other if their values correlate completely. There are two widely used types of measures for correlations between two variables: linear and non-linear. In the linear type, the Pearson correlation coefficient is used, and in non-linear cases, many measures are based on the concept of entropy, or the measure of the uncertainty of a random variable. Symmetrical uncertainty (SU) is frequently used, defined as

$$SU(\mathbf{x},\mathbf{y}) = 2\left[\frac{IG(\mathbf{x}|\mathbf{y})}{H(\mathbf{x})+H(\mathbf{y})}\right],$$

where $H(\mathbf{x}) = -\sum_i^p p(x_i)\log_2(p(x_i))$ is the entropy of a variable $\mathbf{x}$ and $IG(\mathbf{x}|\mathbf{y}) = H(\mathbf{x}) - H(\mathbf{x}|\mathbf{y})$ is the information gain from $\mathbf{x}$ provided by $\mathbf{y}$. Both of them are between pairs of variables. However, this may not be straightforward for determining feature redundancy when one is correlated with a set of features. [14] applies a technique based on cross-entropy, called Markov blanket filtering, to eliminate redundant features.

FCBF calculates SU-correlation between any feature F_i and class C generating a list in descending order, and decides heuristically that a feature F_i is relevant if it is highly correlated with class C, i.e., if $SU_{i,c} > \delta$, where δ is a relevance threshold which can be determined by users. The selected relevant features are then subject to redundancy analysis. Similarly, FCBF evaluates the SU-correlation between individual features for redundancy analysis based on an approximate Markov blanket concept. For two relevant features F_i and F_j $(i \neq j)$, F_j can be eliminated if $SU_{i,c} \geq SU_{j,c}$ and $SU_{i,j} \geq SU_{j,c}$. The iteration starts from the first element in the ranking and continues as follows. For all the remaining features, if F_i happens to form an approximate Markov blanket for F_j, F_j will be removed from the list. After one round of filtering features based on F_i, the algorithm will take the remaining feature right next to F_i as the new reference to repeat the filtering process. The algorithm stops until no more features can be eliminated.

V. Experiments

This section presents the experimental results and analysis of ELM Gaussian RBF models on five public microarray datasets with high dimensionality/small sample size. The features of each dataset are shown in Table I.

A. Microarray data

These datasets were taken from bioinformatics and biomedical domains. They are often used to validate the performance of classifiers and gene selectors. Due to high dimensionality and small sample size, gene selection is an essential prerequisite for further data analysis. The selected datasets were: Breast [15], CNS [16], Colon [17], GCM [18] and Leukemia [19].

In these 5 microarray datasets, all expression values of genes are real numbers. For convenience, they were standardized before our experiments, that is, the mean and standard deviation of each gene represented were zero and one, respectively, after the standardized operation had been performed.

B. Alternative Statistical and Artificial Intelligence methods used for comparison purposes

Different state-of-the-art Statistical and Artificial Intelligence algorithms have been used for comparison purposes. Specifically, the results of the following algorithms have been compared with the ELM and OPELM using sigmoid and RBF basis functions (ELM-SIG, ELM-RBF, OPELM-SIG, OPELM-RBF):

1) A Gaussian Radial Basis Function Network (RBFN) which derives the centres and width of hidden units using k-means, and combines the outputs obtained from the hidden layer using logistic regression.
2) The MultiLogistic (MLogistic) algorithm. It is a method for building a multinomial logistic regression model with a ridge estimator to guard against overfitting by penalizing large coefficients.
3) The SimpleLogistic (SLogistic) algorithm. It is based on applying the LogitBoost algorithm with simple regression functions and determining the optimum number of iterations using five fold cross-validation.
4) The C4.5 classification tree inducer.
5) The Logistic Model Tree (LMT) classifier which combines linear logistic regression and tree induction.

These algorithms have been selected because many of these approaches have also been tested previously in the classification problem of microarray gene expression. A detailed description and some previous results of these methods can be found in [20], [21].

C. Experimental design

The experimental design was conducted using a holdout cross validation procedure with approximately 75% of the instances for the training dataset and 25% of them for the generalization dataset. In order to evaluate the stability of the methods, the ELM and OPELM algorithms are run 30 times. The evaluation of the different models has been performed by measuring the Correctly Classified Rate (C) or Accuracy, and the Minimum Sensitivity (MS).

For ELM and OPELM, the number of hidden nodes gradually increases in intervals of 1 within the interval $[5, 200]$ and the nearly optimal number of nodes for ELM and OPELM are then selected based on the cross-validation method described in Section III.

The ELM-RBF method has been implemented in Matlab. The OPELM Matlab version was used while WEKA [20] was used to obtain the results of the remaining methods.

For the FCBF feature selection, WEKA [20] was used with default parameters, including the relevance threshold δ .

Table I
CHARACTERISTICS OF THE FIVE DATASETS USED FOR THE EXPERIMENTS: NUMBER OF INSTANCES (Size) AND NUMBER OF REAL (R) INPUT VARIABLES, TOTAL NUMBER OF INPUTS (#In.), NUMBER OF CLASSES (#Out.), AND PER-CLASS DISTRIBUTION OF THE INSTANCES (Distribution)

Dataset	Source	Size	R	In	Out	Distribution
Breast	Van't Veer et al [15]	97	493	493	2	(46,51)
CNS	Pomeroy et al [16]	60	170	170	2	(21,39)
Colon	Alon et al [17]	62	59	59	2	(40,22)
GCM	Ramaswamy et al [18]	190	264	264	14	(20,11,11,11,11,30,10,10,11,22,11,11,10,11)
Leukemia	Golub et al [19]	72	203	203	2	(25,47)

Table II
COMPARISON OF THE PROPOSED METHOD TO OTHER PROBABILISTIC METHODS: RESULTS OF ACCURACY ($C_{\mathrm{G}}(\%)$) AND SENSITIVITY (MS_G) ON THE GENERALIZATION SET

		RBFN	MLogistic	SLogistic	C4.5	LMT	OPELM		ELM	
Dataset	Metric						SIG	RBF	SIG	RBF
Breast	C_{G}	80.00	84.00	*84.00*	64.00	84.00	73.73	83.87	67.20	**87.20**
CNS	C_{G}	86.66	**100.00**	80.00	60.00	80.00	75.78	80.00	74.89	*97.11*
Colon	C_{G}	**87.50**	75.00	*81.25*	75.00	75.00	74.38	*81.25*	74.17	80.00
GCM	C_{G}	**82.69**	80.77	71.15	48.08	67.31	58.85	74.62	66.54	*81.31*
Leukemia	C_{G}	94.44	94.44	83.33	83.33	83.33	90.19	*94.63*	89.07	**96.30**
	$\overline{C}_{\mathrm{G}}(\%)$	86.26	*86.84*	79.95	66.08	77.93	74.59	82.87	74.37	**88.38**
	$\overline{R}_{C_{\mathrm{G}}}$	*2.9*	3.3	4.70	8.20	5.60	7.00	3.70	7.60	**2.00**
Breast	MS_{G}	75.00	**83.33**	**83.33**	41.66	**83.33**	66.43	78.42	60.64	81.75
CNS	MS_{G}	60.00	**100.00**	60.00	60.00	60.00	65.67	60.00	59.33	*95.33*
Colon	MS_{G}	83.33	70.00	66.66	50.00	50.00	64.78	75.78	67.78	74.44
GCM	MS_{G}	**33.33**	**33.33**	**33.33**	0.00	**33.33**	1.11	0.00	6.67	**33.33**
Leukemia	MS_{G}	91.66	91.66	66.66	83.33	66.66	86.11	91.94	81.67	**94.44**
	$\overline{MS}_{\mathrm{G}}(\%)$	68.66	*75.66*	62.00	47.00	58.66	56.82	61.23	55.22	**75.86**
	$\overline{R}_{MS_{\mathrm{G}}}$	3.90	*2.70*	5.10	7.60	5.60	5.80	4.70	7.00	**2.60**

The best result is in bold face and the second best result in italics

D. Results

Table II shows the results of the Correct Classification Rate (C_{G}) and Minimum Sensitivity (MS_{G}) in the generalization set for each dataset and the RBFN, MLogistic, SLogistic, C4.5, LMT and OPELM and ELM with sigmoid and RBF basis functions methods. The result for the ELM' related methods model is the mean result of the 30 executions of the ELM and OPELM (stochastic approaches). Based on the C_{G} and MS_{G}, the ranking of each method in each dataset is obtained ($R = 1$ for the best performing method and $R = 9$ for the worst one). The mean Accuracy and minimum Sensitivity ($\overline{C}_{\mathrm{G}}$ and $\overline{MS}_{\mathrm{G}}$) as well as the mean ranking ($\overline{R}_{C_{\mathrm{G}}}$ and $\overline{R}_{MS_{\mathrm{G}}}$) are also included in Table II.

From the analysis of the descriptive results, it can be seen that the ELM-RBF method obtained the best results in mean for the five datasets not only in best Accuracy ($\overline{C}_{\mathrm{G}} = 88.38\%$) and minimum Sensitivity ($\overline{MS}_{\mathrm{G}} = 75.86\%$) values but also in ranking ($\overline{R}_{C_{\mathrm{G}}} = 2.00$ and $\overline{R}_{MS_{\mathrm{G}}} = 2.60$). MLogistic obtained the second best results in Accuracy ($C_{\mathrm{G}} = 86.84\%$) and Minimum Sensitivity ($MS_{\mathrm{G}} = 75.66\%$).

Based on these results, a second conclusion is that sigmoid basis functions are not suitable for ELM' methods in these databases since they obtain bad results in C_{G}, MS_{G} and ranking compared to all the methods except for C4.5.

To determine the statistical significance of the rank differences observed for each method in the different datasets, a non-parametric Friedman test [22] has been carried out with the C_{G} and MS_{G} ranking of the best models in the generalization sets (since a previous evaluation of the C_{G} and MS_{G} values results in rejecting the normality and the equality of variances hypothesis. The test shows that the effect of the method used for classification is statistically significant at a significance level of $\alpha = 5\%$, as the confidence interval is $C_0 = (0, F_{0.05} = 2.44)$ and the F-distribution statistical values are $F^* = 7.67 \notin C_0$ for C_{G} and $F^* = 2.69 \notin C_0$ for MS_{G}. Consequently, the null-hypothesis stating that all algorithms perform equally in mean ranking is rejected.

It has been noted that the approach that compares all classifiers to each other in a post-hoc test is not as sensitive as the approach comparing all classifiers to a given classifier (a control method). The Bonferroni-Dunn test is an example of this latter type of comparison with a control method and it considers that the performance of any two classifiers is deemed to be significantly different if their mean ranks differ

Table III
JUSTIFICATION OF THE APPROACHES: CRITICAL DIFFERENCE (CD) VALUES AND DIFFERENCES IN RANKINGS IN THE BONFERRONI-DUNN TEST, USING ELM-RBF AS THE CONTROL METHOD

Bonferroni-Dunn test (C_{G})									
	Compared Method								
Control Method	RBFN	MLogistic	SLogistic	C4.5	LMT	OPELM-SIG	OPELM-RBF	ELM-SIG	ELM-RBF
ELM-RBF	0.90	1.30	2.70	$6.20^{+}_{\bullet}$	3.60	$5.00^{+}_{\bullet}$	1.70	$5.60^{+}_{\bullet}$	-
Bonferroni-Dunn test (MS_{G})									
	Compared Method								
Control Method	RBFN	MLogistic	SLogistic	C4.5	LMT	OPELM-SIG	OPELM-RBF	ELM-SIG	ELM-RBF
ELM-RBF	1.30	0.10	2.50	$5.00^{+}_{\bullet}$	3.00	3.20	2.10	$4.40^{+}_{\circ}$	-

Bonferroni- Dunn Test: $CD_{(\alpha=0.1)}$= 4.33, $CD_{(\alpha=0.05)}$= 4.72
•, ∘: Statistical difference with α = 0.05 (•) and α = 0.10 (∘)
$+$: The difference is in favour of the Control Method

by at least the critical difference (CD):

$$CD = q\sqrt{\frac{K(K+1)}{6D}}, \quad (1)$$

where K and D are the number of classifiers and datasets. This test can be computed using Eq. (1) with appropriate adjusted values of q [23].

The results of the Bonferroni-Dunn test for $\alpha = 0.10$ and $\alpha = 0.05$ can be seen in Table III, using the corresponding critical values. From the results of this test, it can be concluded that ELM-RBF obtains a significantly better C_{G} and MS_{G} ranking mean than several of the other methods. For this reason, the ELM-RBF methodology is recommended to improve Accuracy and the Minimum Sensitivity values for these gene classification datasets.

A final comparison was performed considering the first and second methodologies according to the to the mean ranking results for C_{G} and MS_{G}. A Mann-Whitney test was done comparing mean results of ELM-RBF versus RBFN for C_{G} and mean results of ELM-RBF versus MLogistic for MS_{G}. Each test checked the null-hypothesis that data of the results in C_{G} and MS_{G} are independent samples from identical continuous distributions with equal medians, against the alternative that they do not have equal medians. The null-hypothesis was not rejected for both tests with $p-value = 0.75$ for the C_{G} test and $p-value = 0.70$ for the MS_{G} test.

The results of the number of hidden nodes selected are shown in Table IV. For OPELM, this table shows the initial number of hidden nodes used for the algorithm and the final mean number of hidden nodes after pruning the neural network. Regarding ELM, the ELM-RBF needs a higher number of hidden nodes than ELM with a sigmoid basis function. On the other hand, OPELM with RBF prunes the network and reduces the number of hidden nodes drastically. It seems that the pruning process applied by OPELM tends to eliminate too many neurons, resulting in a degradation in performance for some data sets (see Breast and CNS in Table II).

Table IV
NUMBER OF HIDDEN NODES FOR EACH ELM METHOD AND BASIS FUNCTION SELECTED BY THE CROSS-VALIDATION PROCEDURE ON THE TRAINING DATASET

	OPELM				ELM	
	SIG		RBF		SIG	RBF
Dataset	M_I	M_F	M_I	M_F	M	M
Breast	41	12.17	20	6.83	14	57
CNS	33	9.83	11	7.50	20	26
Colon	25	7.67	30	7.50	15	25
GCM	120	69.17	45	41.17	60	75
Leukemia	28	11.33	14	9.33	18	9

M number of hidden nodes for ELM
M_I initial number of hidden nodes for OPELM
M_F final mean number of hidden nodes for OPELM after prunning

VI. CONCLUSIONS

This paper has given a brief presentation of the Extreme Learning Machine algorithm. The RBF basis function is considered for the hidden layer of the ELM model (ELM-RBF). This algorithm is used for classification in five gene microarray analysis datasets. The Fast Correlation-Based Filter (FCBF) feature selection procedure was applied to the datasets.

Rather than randomly assigning ELM-RBF centers and widths, an alternative fast initialization procedure has been applied for these parameters [11]. The results are compared to several classification methods (including, among others, OPELM and alternative basis functions for ELM) in terms of Accuracy and minimum Sensitivity. The ELM is found to obtain the best performance results for these datasets.

Regarding ELM and OPELM, we conclude that the sigmoid basis function is not suitable for these datasets whereas the RBF basis function does give good performance results. In addition, it seems that the pruning process applied by OPELM tends to eliminate too many neurons so that there is a noticeably performance degradation in some data sets. As future work, it may be interesting to study how to reduce this performance degradation in OPELM.

Acknowledgement

This work has been partially subsidized by the TIN 2008-06681-C06-03 project of the Spanish Inter-Ministerial Commission of Science and Technology (MICYT), FEDER funds and the P08-TIC-3745 project of the "Junta de Andalucía" (Spain). The research of Javier Sánchez-Monedero has been funded by the "Junta de Andalucía" Ph. D. Student Program.

References

[1] L. J. Lancashire, D. G. Powe, J. S. Reis-Filho, E. Rakha, C. Lemetre, B. Weigelt, T. M. Abdel-Fatah, A. R. Green, R. Mukta, R. Blamey, E. C. Paish, R. C. Rees, I. O. Ellis, and G. R. Ball, "A validated gene expression profile for detecting clinical outcome in breast cancer using artificial neural networks," *Breast cancer research and treatment*, vol. 120, no. 1, pp. 83–93, 2010.

[2] U. Maulik and A. Mukhopadhyay, "Simulated annealing based automatic fuzzy clustering combined with ann classification for analyzing microarray data," *Computers and Operations Research*, vol. 37, no. 8, pp. 1369–1380, 2010.

[3] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489 – 501, 2006.

[4] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 879–892, July 2006.

[5] R. Zhang, G.-B. Huang, N. Sundararajan, and P. Saratchandran, "Multi-Category Classification Using An Extreme Learning Machine for Microarray Gene Expression Cancer Diagnosis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 4, no. 3, pp. 485–495, 2007.

[6] Y. Chauvin and D. E. Rumelhart, *Backpropagation: Theory, Architectures, and Applications*. Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc, 1995.

[7] C. Igel and M. Hsken, "Empirical evaluation of the improved rprop learning algorithms," *Neurocomputing*, vol. 50, no. 6, pp. 105–123, 2003.

[8] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *Journal of machine learning research*, vol. 5, pp. 1205–24, 2004.

[9] J. Fernández Caballero, F. Martínez, C. Hervás, and P. Gutiérrez, "Sensitivity versus accuracy in multiclass problems using memetic pareto evolutionary neural networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 5, pp. 750 –770, may 2010.

[10] G. B. Huang and C. Slew, "Extreme learning machine: RBF network case," in *8th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, vol. 2, 2004, pp. 1029–1036.

[11] Y. Miche, A. Sorjamaa, P. Bas, O. Simula, C. Jutten, and A. Lendasse, "OP-ELM: Optimally Pruned Extreme Learning Machine," *IEEE Transactions on Neural Networks*, vol. 21, no. 1, pp. 158–162, January 2010.

[12] T. Poggio and F. Girosi, "A theory of networks for approximation and learning," *Laboratory, Massachusetts Institute of Technology*, vol. 1140, 1989.

[13] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)*, 1st ed. The MIT Press, December 2001.

[14] D. Koller and M. Sahami, "Toward optimal feature selection," in *13th Int. Conf. on Machine Learning*. Bari, IT: Morgan Kaufmann, 1996, pp. 284–292.

[15] L. J. Van't Veer, H. Dai, M. J. Van de Vijver, Y. D. He, A. A. M. Hart, M. Mao, H. L. Peterse, K. Van Der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, no. 6871, pp. 530–536, 2002.

[16] S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angelo, M. E. McLaughlin, J. Y. H. Kim, L. C. Goumnerova, P. M. Black, C. Lau, J. C. Allen, D. Zagzag, J. M. Olson, T. Curran, C. Wetmore, J. A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D. N. Louis, J. P. Mesirov, E. S. Lander, and T. R. Golub, "Prediction of central nervous system embryonal tumour outcome based on gene expression," *Nature*, vol. 415, no. 6870, pp. 436–442, 2002.

[17] U. Alon, N. Barka, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 96, no. 12, pp. 6745–6750, 1999.

[18] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C. H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. P. Mesirov, T. Poggio, W. Gerald, M. Loda, E. S. Lander, and T. R. Golub, "Multiclass cancer diagnosis using tumor gene expression signatures," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, pp. 15 149 – 54, 2001/12/18/ 2001.

[19] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–527, 1999.

[20] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed., ser. Data Management Systems. Morgan Kaufmann (Elsevier), 2005.

[21] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," *Machine Learning*, vol. 59, no. 1-2, pp. 161–205, 2005.

[22] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.

[23] Y. Hochberg and A. Tamhane, *Multiple Comparison Procedures*. John Wiley & Sons, 1987.