IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

Randomness in Circular - Complex Extreme Learning Machine Vs Voting Based Extreme Learning Machine with Accuracy Based Ensemble Pruning: A Review

Ashish Gupta

Department of Computer Science and Engineering,

Mahamaya College of Agricultural Engineering and Technology, Ambedkar Nagar, Uttar Pradesh, India

Abstract: Extreme Learning Machine is a fast single layer feed forward neural network for real valued classification. It suffers from the problem of instability and over fitting. Extreme Learning Machine (ELM) has recently emerged as a fast classifier giving good performance. Voting based Extreme Learning Machine; VELM reduces this performance variation in Extreme Learning Machine by employing majority voting based ensembling technique. Circular–Complex Extreme Learning Machine (CC-ELM) is recently proposed complex variant of ELM which has fully complex activation function. It has been shown that CC-ELM outperforms real valued and other complex valued classifiers. In both CCELM & ELM parameters between input and hidden layer are initialized randomly and the weights between hidden and output layer are obtained analytically. Due to this randomization, the performance of both ELM & CC-ELM fluctuates. In this paper, performance fluctuation due to random parameter of CC-ELM and the circular transformation function have been analyzed first, then by using an Ensemble approach namely Bagging, a variants Bagging.C1 is proposed to bring the stability in the performance of CC-ELM. In Bagging.C1 various data samples are generated by using random parameters of circular transformation function. This work further proposes and evaluates Voting based Extreme Learning Machine with Accuracy based ensemble Pruning, VELM_AP. VELM_AP generates component classifier in the same way as VELM. Performance of proposed classifier ensemble is evaluated using a set of benchmark real-valued classification problems from the University of California, Irvine machine learning repository.

Keywords: Classification, Complex-Valued Neural Networks, Extreme Learning Machine, Ensemble Pruning

I. INTRODUCTION

Extreme learning machine is a fast classifier with a good prediction ability which was recently proposed by Huang et al [1]. It is a single layer feed-forward neural network in which the input weights and hidden layer biases are initialized randomly and the weights between hidden layer and output layer are determined analytically using Moore-Penrose generalized inverse H† of the output matrix of hidden layer. Unlike other traditional gradient descent learning algorithm such a back propagation ELM does not require tuning of the parameters like learning rate, learning epoch etc and can provide better performance in terms of learning speed, reliability and generalization Tian et. al [2]. It does not have to encounter problems like stopping criteria and local minima Wang et. al. [3].

With the evolution of technologies that include the processing of complex-valued signals like, signal processing, adaptive array processing [4], image processing [5] it has been a necessity to create and develop complex-valued neural networks. R.Savitha et. al [6] proposed a Fast learning circular complex valued extreme learning machine (CC-ELM) for the classification of real valued data sets in complex domain.CC-ELM uses sin(z) as circular transformation function and sech() as fully complex Gaussian type activation function. Circular transformation function, an orthogonal decision boundary of CC-ELM at hidden layer and output layer shows better performance and prediction ability of real valued data than ELM and various complex valued classifiers [7]. Various variants of ELM also have been proposed in literature to enhance the performance like OS-ELM [8], I-ELM [9], W-ELM [10], C-ELM [11] etc. However these variants could not address some drawbacks of ELM [12] which are mentioned below:

1. The ELM and its variants randomly select the input weights and the hidden layer bias. This causes instability in the predictions of ELM.
2. The ELMs suffer from the over fitting problem. This is due to more number of hidden nodes on larger datasets and the complexity of the input instances distribution.
3. When the size of the dataset increases the order of the matrix H also increases. This implies that when larger datasets are used large memory is required to calculate the Moore-Penrose inverse.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

Hansen and Salmon have proved in [13] that the above mention problems can be solved by using ensemble learning. The ensemble learning aims at reducing the risk of modelling error by combining several base learners. Each base learner in the ensemble can be generated either by creating diversity in the dataset used for each classifier by using subsets of a larger dataset or the whole dataset by creating diversity in terms of settings in the learning algorithm [14]. Yu Liu [15] proved that variation among components input weights and initial parameter forces those components to have diverse output space which increases the diversity and generalization ability of an ensemble model. Jiuwen Cao et. al. [16] also have proved that large number of multiple realization of ELM reduces the misclassified sample, have the lower variance and is able to correctly classify test sample with probability of one.

Ensemble of weak learner model may differ in terms of 1).the base classifier used for prediction 2). Approach used to create various model either by selecting different realization of model or by using different samples of data set. 3).ensembling approach is used and 4). The ensemble pruning algorithm is used. The two popular methods for an ensemble are Bagging [17], Boosting [18] and its variants. In the literature variants of ELM based on Bagging are, V-ELM, in this it is proved that ability to correctly classify a test data with a large number of training model is probability of one and the final prediction is done by majority voting .

One successful approach in this sense is the use of 'randomization,' that is, the stochastic assignment of a subset of the NNs' weights in order to derive a simpler (often linear) optimization problem to solve. This idea has been applied countless times over the years, and it has resulted in three broad families of NN models that we classify as follows: (1) feed-forward networks with random weights (RW-FFN), (2) recurrent NNs with random weights (i.e., reservoir computing, RC), and (3) randomized kernel approximations. Although the literature on these three methods is generally kept separate, they all share two fundamental ideas that contribute to their success. First, in all three cases, randomization is used to define a (generally data independent) feature map, which transforms the input into a highly dimensional space where learning is supposed to be simpler. Secondly, the resulting optimization problem is cast as a standard linear least-squares, which is by far the simplest, most studied and scalable learning procedure to date. In a sense, many of these methods are direct descendants of the seminal work of Cover on the linear separability of patterns, and they are elegantly summarized in the recent quote from Rahimi and Recht, that "randomization is […] cheaper than optimization."

However very less research work has been conducted about the performance ability of ensemble classifier in a complex domain. In this paper for CVNN variants of Bagging is evaluated. Two complex-valued ensemble based classifiers namely, V-CELM.C1 (Voting based-Complex Extreme Learning Machine.C1) and V-CELM.C2 (Voting based-Complex Extreme Learning Machine.C2) are implemented using variants of Bagging in the complex domain named as Bagging.C1. For evaluating the performance of the ensemble based classifiers the simulations has run on datasets retrieved from the open UCI repository. Eight multi-class and six binary-class datasets have chosen for the experiments. Furthermore, VELM-AP which is an extension of VELM. Voting based Extreme Learning Machine; VELM reduces this performance variation in Extreme Learning Machine by employing majority voting based ensembling technique. VELM improves the performance of ELM at the cost of increased redundancy. This problem can be reduced using ensemble pruning techniques. This work proposes and evaluates Voting based Extreme Learning Machine with Accuracy based ensemble Pruning, VELM_AP. VELM_AP generates component classifier in the same way as VELM. VELM [4], uses majority voting to get combined outcome of independent component classifiers of the ensemble. It improves the performance of ELM at the cost of increases redundancy. Ensemble pruning techniques [11]–[18] can be employed to a get a sub-ensemble containing accurate and diverse classifiers. VELM_AP uses accuracy measure for pruning VELM. Along with the various experiments presented in this paper to study the effectiveness of our classifiers, we have also used the Wilcoxon ranksum method to strengthen our claims of building an efficient complex based ensemble classifier. In the next section this paper discusses related work i.e. CC-ELM, VELM and ensemble pruning techniques. After this section, this paper describes the proposed work. After that, this paper describes the experimental setup and results obtained. The last section consists of conclusion and future work.

II. RELATED WORK

This section contains the brief review of the fundamental topics which were proposed earlier and are important from the perspective of the proposed work.

A. Circular Complex-Valued Extreme Learning Machine

CC-ELM has chosen as base classifier among other complex-valued extreme learning machines due to it efficient transformation mapping and better accuracy in prediction. For creating a complex-valued extreme learning machine the basic assumption is that a set of N observations namely O, can be represented as : O = {(xi,ci)|xi ϵ Rn, i = 1,2,…..,N}, where xi is the n-dimensional real-valued input vector and ci ϵ CV, where CV = {1,2,…,C} is the class vector containing coded class labels with C number of classes. The following function is used to obtain the coded-class label, ccrt:

$$\text{ccrt} = \begin{cases} \mathrm{i}+1, \mathrm{ci}=\mathrm{r} \\ -\mathrm{i}-1, \text{otherwise} \end{cases} \qquad r=1,2,\ldots,C \qquad \ldots(1)$$

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

The transformation function used in CC-ELM to map the real-valued input features into complex domain is the sin () function $z_1=\sin(ax_1+ibx_1+\alpha_1)$ ...(2)
The sin function is analytic and almost bounded everywhere [16] which makes it a suitable function to be used in the transformation part of the CC-ELM classifier. a and b($0 < a,b \leq 1$) are real-valued non-zero transformation constant and α ($0< \alpha< 2\pi$) is the non-zero translational/rotational bias.
The CC-ELMs use a fully complex valued activation function in the hidden layer of the type of hyperbolic secant function [14]. The responses of the sech activation are given as in (3):

$$hj = \mathrm{sech}(ujT(zt - vj));\ j=1,\ldots, K \qquad ...(3)$$

where uj and vj ϵ Cn are complex-valued scaling factor and complex-valued center of the j-th hidden neuron respectively.
The output layer neurons employ a linear activation function in the CELM. The output of the CELM network with K hidden neurons is:

$$\tilde{Y}_n= \sum_{j=1}^{K} w_{nj}h_j \qquad ...(4)$$

where wnj are the weights connecting the n-th output neuron with the j-th hidden neuron. Equation (4) can also be written as in Equation (5).

$$\tilde{Y}= WH \qquad ...(5)$$

where W is the matrix of all output weights connecting the hidden and output layer neurons. H is the response matrix of the hidden layer and is given as in Equation (6).

$$H(V,U,Z) = \begin{bmatrix} \mathrm{sech}(U1T\|Z1 - V1\|) & \ldots & \mathrm{sech}(UkT\|Zn - V1\|) \\ \ldots & \ldots & \ldots \\ \mathrm{sech}(U1T\|Z1 - Vk\|) & \ldots & \mathrm{sech}(UkT\|Zn - Vk\|) \end{bmatrix} \qquad ...(6)$$

Here H is a K × N matrix, where K is the number of hidden neurons and N is the number of samples to be trained. The parameters (uj,vj) chosen randomly and The output weights W are calculated by the least squares method according to Equation (7):

$$W = YH\dagger, \qquad ...(7)$$

Where H† is the Moore-Penrose pseudo-inverse of the hidden layer output matrix and Y is the complex-valued coded class label.
From the outputs, the class labels are estimated as below:

$$C' = \max_{l=1,2,3,\ldots,c} \mathrm{real}\ (\tilde{Y}) \qquad ....(8)$$

B. Voting Based Extreme Learning Machine
ELM suffers from the problem of instability and over fitting. The instability problem arises due to random initialization of weights between the input and hidden layer. V-ELM [4] solves this problem, by generating a number of classifiers, succeeded by majority voting for finding the prediction of the ensemble. Independent component classifiers of the ensemble are generated by randomly assigned different weights between input and hidden layer. For n testing instances, the output label (L_{test}) corresponding to all the component classifiers is obtained. The final predicted output (FP) is given by:
$FP = \mathrm{mode}(L_{test})$
Here, $L_{test} = [L^1_{test}, L^2_{test}, \ldots ,L^{NCE}_{test}]$
The mode operation calculates the class to which the maximum numbers of classifiers are voting. Taking an example of binary classification where, an instance belongs to either positive class or negative class. Number of classifier, NCE =100. Let for any test instance 70 classifiers vote for positive class whereas, 30 vote for negative output. Then the final output of V-ELM is positive class.

C. Ensemble Pruning
Many is better than all. Instead of using all component classifiers, a subset of accurate and diverse classifiers may give equal or better performance. A number of ensemble pruning methods have been proposed. Ensemble pruning techniques are mainly classified in three categories: Order based, Clustering based and Optimization based. Ordering based ensemble pruning technique orders the component classifiers of the ensemble as per their importance which is quantified by suitable metric. The final sub ensemble is constructed by choosing first few classifiers as per their ordering. The number of classifiers in the pruned ensemble is determined by setting threshold. Some of the ordering based pruning techniques are Reduce Error Pruning, Kappa pruning, Complementariness Pruning, Margin distance pruning etc. The optimization based pruning techniques give better solution compared to other techniques but they are computationally intensive. The author in [21] uses Accuracy and reduce error pruning technique to get an optimally

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

pruned ensemble. Backtracking in reduce error pruning increases computational overhead and guarantees that the pruned ensemble will have greater or equal accuracy than the full ensemble.

III. PROPOSED WORK

CC-ELM BASED ENSEMBLE METHODS (Bagging.C1)

In this paper it has been tried to significantly improve the classification performance of CC-ELM by incorporating variants of bagging using the random parameter of circular transformation function. Major focus has been rendered to the method by which datasets are chosen for training by each base learner in the ensemble. The description of the variants of bagging namely: 'Bagging.C1' is given below:

V-CELM.C1 using Bagging.C1: To bring diversity in order to implement an ensemble (i) data can be diverse and the classifier constant, (ii) data is kept constant and classifier varied or (iii) bring both data and classifier diversity together. In the algorithm Bagging.C1 we follow the latter method, i.e., to incorporate both data and classifier diversity. In each iteration of Bagging. C1, a new dataset is provided to the base classifier.

Algorithm for Bagging.C1

Input: O_{DS}: Original training set = $\{(x_i , c_i)| x_i \in R_n, c_i \in CV, i = 1,2,\ldots,N\}$; J: Number of iterations; n: size of O_{DS} and Bootstrap, CC-ELM: Base classifier

Output: final-CLP: Final class prediction using majority voting

Algorithm

1. for j = 1 to J
2. L_j = Random Transformation Sample (O_{DS})
3. Generate a new CC-ELM_j model by randomly choosing U and V
4. Calculate the class prediction, CLP_j for CC-ELM_j

CLP_j = output (CC-ELM_j (L_j))

5. End
6. Final-CLP = $\max_{j=1,\ldots,J}$ (CLP)

Function Random Transformation Sample (O_{DS})

1. Select random numbers $a \in [0,1]$,$b \in [0,1]$,$\alpha \in [0,2\pi]$
2. for i = 1,…, N
3. $z_i = \sin (ax_i + bx_i + \alpha)$
4. $cc_r^i = \begin{cases} i + 1, ci = r \\ -i - 1, \text{otherwise} \end{cases}$ r=1,2,….,C
5. end
6. return L= $\{(z_i,cc_i) \mid z_i \in C^n, cc_i \in CC, i=1,2,\ldots,N\}$, CC is the complex coded class label matrix

The original dataset ODs which is in real-valued format is transformed to complex-valued dataset by the transformation function: $z_i = \sin (ax_i + bx_i + \alpha)$ where a and b ($0 < a, b \leq 1$) is real-valued non-zero transformation constant and is the non-zero translational/rotational bias. The values of a, b and are randomly chosen for each classifier in the ensemble thereby generating random samples of the original dataset. The classifier specifications itself are changed for each classifier in the ensemble by varying the parameters u and v of the CC-ELM classifier. We use random values of a, b and for changing the dataset for each classifier. The impact of transformation function on the classification ability of complex valued neural networks has been stressed on [14], [15] and [16]. Thus creating diverse datasets from the original dataset by randomizing the parameters a, b and are supposed to bring good performance results in bagging. Subsequent to training, in the testing phase when a new instance is provided to the ensemble, the class prediction is performed by a majority voting of the class predictions by the individual CC-ELM classifiers in the ensemble. We generate an ensemble based classifier termed as V-CELM.C1 which implements Bagging.C1 algorithm.

V-CELM.C2: In V-CELM.C2, instead of providing random samples of the original dataset OD for each classifier in the ensemble, we produce only one sample and use the same across all the CC-ELM classifiers. The diversity is only in terms of the parameters u and v of the CC-ELM classifier ineach iteration of the ensemble. The remaining steps are the same as for Bagging.C1.

In order to reduce the redundant classifiers in VELM this work proposes and evaluates ordering based ensemble pruning algorithm. This work uses G-mean metric to quantify the importance of component classifiers of the ensemble.

DOI 10.17148/IARJSET.2019.6510

Vol. 6, Issue 5, May 2019

Compared to overall accuracy, G-mean is a better accuracy metric when data is not balanced. The proposed work assumes that all the ELM based classifiers are diverse as the weights between the input and hidden neurons are assigned randomly. The pseudo code of proposed algorithm is as follows:

Algorithm for VELM_AP

Training Phase

I. Generate NCE ELM based classifiers.

II. Find the output of all component classifiers of the pruned ensemble for training dataset. Compute accuracy on training data.

III. Arrange the classifiers in decreasing order of training G-mean.

IV. To avoid tie condition during voting select odd number of top classifiers from the ordered list of classifiers to make the pruned ensemble

Testing Phase

I. Find the output of component classifiers of the pruned ensemble for test dataset.

II. Perform majority voting of outcomes of the classifiers in the pruned ensemble to get the final outcome,

III. Do performance evaluation using the final outcome.

IV. EXPERIMENTAL STUDY AND RESULT ANALYSIS

To evaluate the efficacy of the proposed ensemble techniques Four experiments has conducted in this study and for the proposed method the simulation is done on The MATLAB 7.10.0 (R2010a) running on Core 2 DUO PC. The experiments have used a total 14 datasets obtained from the UCI repository [23] of which 8 and 6 are multi-class datasets and binary class datasets respectively. The information regarding the datasets is provided in Table No. 1.

Table 1: Datasets used and their description

Data-set	Category	Instances	Testing Instances	Training Instances	Classes	Features
Balance	Multi	625	225	400	3	40
Cancer	Binary	683	383	300	2	9
Cancer1	Binary	569	269	300	2	30
Ecoli	Multi	336	168	168	8	7
Heart	Binary	270	170	100	2	13
Image	Multi	2310	2100	210	7	19
Ionosphere	Binary	351	251	100	2	34
Optical Digits	Multi	5620	1797	3823	10	64
Pen Digits	Multi	10992	3498	7494	10	16
Pima	Binary	768	368	400	2	8
Spambase	Binary	4601	2301	2300	2	7
Vehicle	Multi	846	422	424	4	18
Waveform	Multi	5000	2000	3000	3	40
Wine	Multi	178	78	100	3	13

Here the datasets of various sizes have been used across our experimental study. The number of training instances and testing instances used for each dataset has been mentioned in the Table. We use twenty base CC-ELM classifiers throughout our experiments (J=20). The number of hidden nodes generated for each CC-ELM classifier in the ensemble is as per given in [16]. In order to Prove the efficiency of the ensemble methods over the single classifier models; two complex-valued ensemble based classifiers V-CELM.C1, V-CELM.C2 and one voting based classifier VELM-AP are compared with the CC-ELM classifier in four major aspects:

1. The impact of building ensemble methods over a single neural network based classifier
2. Average testing accuracy
3. The impact of stability in the performance of an ensemble classifier over a single classifier model.
4. Statistical test namely Wilicoxon test to ensure the performance overhand of the ensemble techniques.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 6, Issue 5, May 2019

We mention each experiment, their respective results and a detailed study of the proposed ensemble methods based on these results.

A. The impact of building CC-ELM base classifier over the proposed ensemble methods

To enhance the performance of CC-ELM Bagging.C1 has been used to generate the base classifier with diversity. In this section we study the power of ensemble method of CC-ELM base classifier with data set Waveform (DS3).

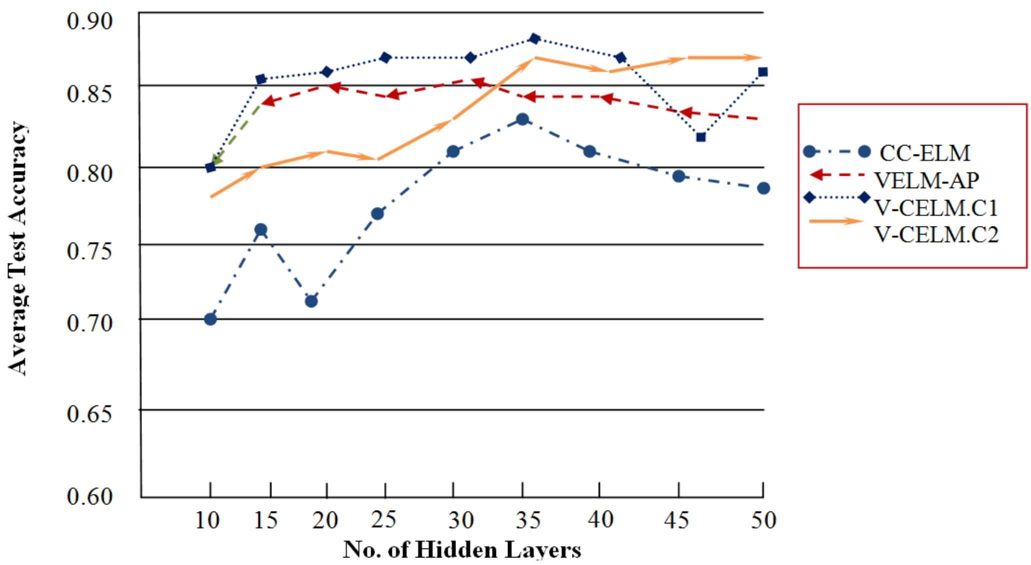

Fig.1: The impact of construction of CC-ELM sub classifier on Ensemble Waveform data set (DS3)

The experiment is performed by varying the hidden nodes of CC-ELM from 10 to 50 and the results are shown for the testing accuracy in Fig. 1. The CC-ELM is evaluated our proposed algorithms and From the results it is vivid that the testing accuracy of the ensemble methods are far better than that of a single CC-ELM model. It implies that while building an ensemble rather than using a single model better prediction capability can be achieved with less number of computational resources.

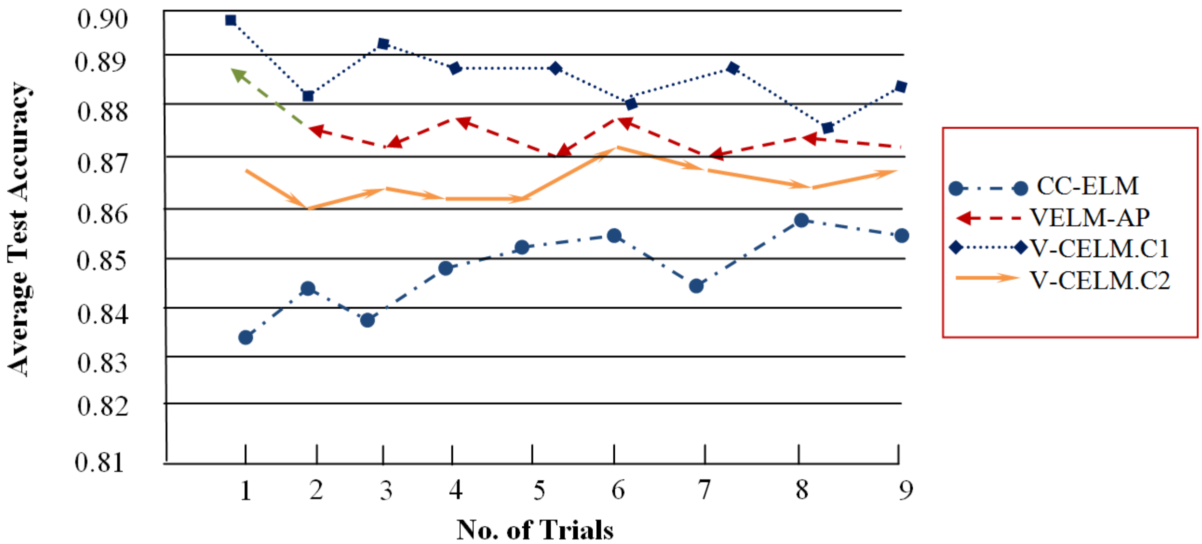

Fig.2: The Experimental result in average test accuracy with varying number of Training numbers of Spambase data set (DS9)

In Fig. 2 performance of dataset Spambase (DS9) is evaluated with varying no. of training numbers, to analyze the performance fluctuation due to the random parameter of circular transformation function. The result clearly depicts that performance vary too much when we use a single model. The result also describes that randomization of circular transformation function a, b and α creates more variation than the random parameter between input and hidden layer u and v. When ensemble of base classifier has used both performance and stability increases highly.

Table 2: Average test accuracy results of CC-ELM and ensemble methods for multi-class datasets

Data-set	Hidden node	V-CELM.C1	V-CELM.C2	VELM-AP	CC-ELM
Balance	15	85.743	86.098	84.075	83.663
Ecoli	15	88.580	86.580	84.537	85.638
Image	70	95.846	94.377	92.354	92.755
Optical	70	94.180	93.031	91.048	92.427
Pen	60	87.897	86.145	84.132	81.193
Vehicle	90	81.161	79.845	77.824	79.523
Waveform	45	89.097	87.237	85.254	85.072
Wine	20	94.723	94.124	92.137	91.671

The performance results for other datasets mentioned in Table No. 4 match with those that we have provided for the Waveform multi-class dataset. The results are not displayed due to the limitation in space to present them.

Table 3: Average test accuracy results of CC-ELM and ensemble methods for binary class datasets

Data-set	Hidden node	V-CELM.C1	V-CELM.C2	VELM-AP	CC-ELM
Pima	25	76.637	76.182	75.227	76.486
Spambase	60	87.264	86.470	85.512	84.758
Heart	20	84.077	83.747	82.881	83.473
Iono	20	81.021	80.013	81.156	79.845
CANCER1	25	86.317	85.185	84.228	84.571
CANCER	25	96.880	95.811	94.934	95.792

B. The impact of ensemble methods on the testing accuracy of base classifier CC-ELM

In this section, we intend to analyze the testing accuracy of the proposed ensemble methods over the original CC-ELM classifier. For this we have chosen to show the results of testing accuracy over the 14 datasets including multiclass and binary class datasets already described in Table no. 1. For each dataset, the experiment was run for 50 times and the average of the testing accuracy over the fifty results was retrieved and is presented in Table no. 2 for multi-class datasets and Table no. 3 for binary class datasets. The tables clearly represent the efficacy of our methods. In almost all the datasets, the test accuracy of the ensemble methods are quite better than that of the basic CC-ELM classifier as a single model. Since we have used datasets of various sizes across our experiment, our results clearly indicate that the ensemble methods we proposed can outperform the single CC-ELM model in almost all the cases whether the data range be small, medium and large. We had intended to check the importance of using random values of a, b and α in producing random datasets for V-CELM.C1. The results show that this technique is effective. And our Random Transformation Sample() method is indeed efficient. The diversity in terms of classifier can bring significant changes to the classification ability for an ensemble.

C. Testing the stability of the ensemble methods over CC-ELM base classifier

In this experiment, we test the stability of the ensemble methods over various network size and data size. To demonstrate that our proposed methods have better stability over changing conditions of network and data ranges, we conduct experiments varying the datasets of large, medium and small sizes over different network sizes of 30, 40 and 50 hidden nodes. The datasets chosen are spambase (large), pen digit (medium) and vehicle (small). The experiment is run on each dataset for CC-ELM and two ensemble methods ten times each. The results are presented in Table no. 4. The observations from the table are briefly summarized below:

Table 4: stability test by varying the hidden node

	V-CELM.C1			V-CELM.C2			VELM-AP			CC-ELM		
Data-Set	h_30	h_40	h_50	h_30	h_40	h_50	h_30	h_40	h_50	h_30	h_40	h_50
Pen Digit	82.0	85.8	85.1	82.6	85.1	85.3	84.6	84.1	84.3	74.1	79.0	81.4
Spam Base	88.6	88.1	89.6	88.5	89.0	89.5	89.5	87.0	88.5	85.8	87.4	88.0
Vehicle	79.4	80.6	81.8	80.6	81.2	82.4	82.2	79.1	81.4	78.5	80.1	81.2

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

1. As the number of hidden nodes increases, the testing accuracy of all the classifiers increases respectively in the case of any dataset irrespective of its size. The entire proposed ensemble methods can be seen to have higher testing accuracies than that of the original CC-ELM method.
2. It can be seen that all the ensemble methods presented here are better in stability for larger datasets when compared to the original-ELM which presents more fluctuations. The bagging methods which use stochastic replacement of the dataset for each base classifier provides more stable results for large data instances.
3. The medium size spam dataset exhibit far better results for the ensemble methods when compared to both large and small datasets. This is however obvious as Spambase as we have chosen is a binary class dataset and the performance overhand are quite reasonable.

D. Statistical test: Wilcoxon test to verify the performance overhand of the proposed methods

Various statistical methods are used in literature to prove the efficiency of the neural network classifiers like Wilcoxon test, paired t-test etc. We use the Wilcoxon test in our experiments to analyze, evaluate and conclude that our methods are undeniably better than the original CC-ELM classifier. The experiments are conducted over 6 datasets where for each dataset the CC-ELM and one of the proposed methods are run for 10, 30 and 50 times. We thus generate six statistics for each ensemble method to ensure its efficacy over the CC-ELM. We denote the dataset for the original CC-ELM and our proposed method as Oi and Pi where i ϵ {1, 2, 3}. When i = 1, O1 and P1 are vectors of size 10, when i=2 O2 and P2 are vectors of size 30 and when i=3 O1 and P2 are size 50 vectors.

Table 5: CC-ELM and V-CELM.C1 rank test

Data-set	Wilicoxon test of CC-ELM and V-CELM.C1					
	10 trial		30 trial		50 trial	
	p_value1	h_value1	p_value1	h_value1	p_value1	h_value1
Balance	0.4125	0	1.482475	0	1.042967	1
Heart	0.0606	0	2.46E-18	1	2.36E-19	1
Optical	0.0116	1	2.46E-01	1	9.79E-07	1
Spam	0.0133	1	8.85E-04	1	3.11E-05	1
Wave	0.0114	1	7.01E-02	1	3.76E-08	1
Wine	0.0585	1	1.013842	1	1.013678	1

Table 6: CC-ELM and V-CELM.C2 rank test

Data-set	Wilicoxon test of CC-ELM and V-CELM.C2					
	10 trial		30 trial		50 trial	
	p_value1	h_value1	p_value1	h_value1	p_value1	h_value1
Balance	0.5599	0	0.699695	0	0.065558	0
Heart	0.2425	0	1.73E-10	1	3.47E-05	1
Optical	0.1023	1	2.01E-09	1	7.91E-14	1
Spam	0.0147	1	2.07E-10	1	3.26E-06	1
Wave	0.0013	1	3.82E-16	1	7.74E-17	1
Wine	0.0261	1	0.000433	1	0.001226	1

Table 7: CC-ELM and VELM-AP rank test

Data-set	Wilicoxon test of CC-ELM and VELM-AP					
	10 trial		30 trial		50 trial	
	p_value1	h_value1	p_value1	h_value1	p_value1	h_value1
Balance	0.5498	0	0.687586	0	0.064548	1
Heart	0.2334	1	2.03E-07	1	2.75E-08	1
Optical	0.0126	1	2.08E-11	1	7.83E-14	1
Spam	0.0047	1	1.78E-06	1	3.42E-07	0
Wave	0.0038	1	3.52E-14	1	8.47E-11	0
Wine	0.0382	1	0.001547	1	0.001846	1

For conducting the Wilcoxon test we use the MATLAB ranksum() function. The ranksum was calculated three times for each dataset with each proposed method against the original CC-ELM as ranksum(O1,P1), ranksum(O2,P2) and ranksum(O3,P3). The Table no. 5, Table no. 6 and Table no. 7, shows the p values and h values of Wilcoxon test tested between CC-ELM against V-CELM.C, V-CELM.C2 and VELM_AP. According to the Wilcoxon test when the value of p is too small, it shadows a doubt on the validity of the null hypothesis. Therefore the values of p and h clearly indicate the significant overhand of our proposed methods over the original CC-ELM. It can be observed from Table 2 that VELM_AP is better than CC-ELM. CC-ELM outperforms VELM-AP and its variants for all evaluated datasets. For further comparison of proposed classifier with CC-ELM; wilcoxon is conducted. The smaller the p value the improvement is more significant. Thus we have proved statistically that building an ensemble using CC-ELM can bring noteworthy improvement in the classification ability of the complex-valued neural network classifier.

V. CONCLUSION AND FUTURE WORK

In this paper, to reduce the performance variation and to enhance the stability, an ensemble based method on CC-ELM base classifier has implemented which also improves the generalization ability and class prediction capability of the fast learning CC-ELM classifier. For introducing the ensemble methods into a complex domain a popular ensemble techniques namely Bagging has chosen and made improvements so that it may be used to solve real-valued classification problems using CC-ELM. The newly proposed algorithms are tested for their classification ability against a single CC-ELM classifier to show the better performance of an ensemble. We found that the classifier based on the algorithm Bagging.C1 does due to the diversity we brought in the ensemble by randomly changing the values of a, b and α in the dataset. The experimental study further demonstrates that our approach is strong and robust. This paper proposes a new classifier, VELM-AP which is an extension of VELM. VELM gives better performance than ELM with increased computational and memory requirement. VELM_AP first creates NCE classifiers using ELM. VELM_AP then applies accuracy based ensemble pruning to reduce the redundant classifiers. The proposed classifier is evaluated using various datasets available at Keel repository. CC-ELM outperforms VELM-AP for all evaluated datasets taken from UCI repository. This is further illustrated from the result of wilcoxon rank test. The future work includes finding an approach to determine the optimal number of classifiers to be selected in the pruned ensemble. The future work also includes exploring other ensemble pruning techniques to enhance the performance of voting based extreme learning machine.

VI. ACKNOWLEDGMENT

The author wishes to thank the reviewers and editors for their valuable suggestions and expert comments that help improve the quality as well as quantity of the paper. We would like to thank Er. Mahendra Rai, Head of Department (Soil & Water Conservation Engineering), MCAET, Ambedkar Nagar for providing continuous support.

REFERENCES

[1]. Z. Lu, X. Wu, X. Zhu, and J. Bongard, “Ensemble Pruning via Individual Contribution Ordering,” in The 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining Washington DC, 2010, no. 1, pp. 871–880.
[2]. T. Windeatt & C. Zor, “Ensemble Pruning Using Spectral Coefficients,” IEEE Trans. Neural Networks Learn. Syst., vol.24, no. 4, pp. 673–678.
[3]. L. Guo and S. Boukir, “Margin-based ordered aggregation for ensemble pruning,” Pattern Recognit. Lett., vol. 34, no. 6, pp. 603–609, 2013.
[4]. I. Partalas, G. Tsoumakas, and I. Vlahavas, “An ensemble uncertainty aware measure for directed hill climbing ensemble pruning,” Mach. Learn., vol. 81, no. 3, pp. 257–282, 2010.
[5]. G. Martinez-Muñoz, D. Hernández-Lobato, and A. Suarez, “An analysis of ensemble pruning techniques based on ordered aggregation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 2, pp. 245–259, 2009.
[6]. Y. Zhang, S. Burer, and W. N. Street, “Ensemble pruning via semi-definite programming,” J. Mach. Learn. Res., vol. 7, pp. 1315–1338, 2006.
[7]. L. Rokach, “Collective-agreement-based pruning of ensembles,” Comput. Stat. Data Anal., vol. 53, no. 4, pp. 1015–1026, 2009.
[8]. G. Martínez-Muñoz and A. Suárez, “Aggregation Ordering in Bagging,” in Proceedings of the {IASTED} International Conference on Artificial Intelligence and Applications, 2004, pp. 258–263.
[9]. Z. H. Zhou, J. Wu, & W. Tang, “Ensembling neural networks: Many could be better than all,” Artif. Intell., vol.137, no.1–2, pp. 239–263, 2002.
[10]. Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms. 2012.
[11]. M. Bhardwaj and V. Bhatnagar, “Towards an optimally pruned classifier ensemble,” Int. J. Mach. Learn. Cybern., pp. 1–20, 2014.
[12]. J. Cao, Z. Lin, G. Bin Huang, and N. Liu, “Voting based extreme learning machine,” Inf. Sci. (Ny)., vol. 185, pp. 66–77, 2012.
[13]. N. Liu and H. Wang, “Ensemble based extreme learning machine,” IEEE Signal Process. Lett., vol. 17, pp. 754–757, 2010.
[14]. J. Zhai, H. Xu, and X. Wang, “Dynamic ensemble extreme learning machine based on sample entropy,” Soft Computing, vol. 16. pp. 1493–1502, 2012.
[15]. G. Wang and P. Li, “Dynamic Adaboost ensemble extreme learning machine,” in ICACTE 2010 - 2010 3rd International Conference on Advanced Computer Theory and Engineering, Proceedings, 2010, vol. 3.
[16]. D. Wang, M. Alhamdoosh, Evolutionary extreme learning machine ensembles with size control, Neurocomputing 102 (2013) 98–110.
[17]. R. Savitha S. Suresh, N. Sundararajan, Fast learning Circular Complex-valued Extreme Learning Machine(CC-ELM) for real-valued classification problems, Information Sciences, Vol. 187 (2012) pp. 277–290.
[18]. J Zhai, H Xu and X Wang, “Dynamic ensemble extreme learning machine based on sample entropy”, Soft computing, vol. 16, 2012, pp. 1493—1502.
[19]. Machine learning repository http://archive.ics.uci.edu/ml/.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 6, Issue 5, May 2019

BIOGRAPHY

Ashish Gupta received the B.TECH degree in Computer Science and Engineering from Inderprastha Engineering College Ghaziabad, affiliated with Gautam Buddh Technical University, Lucknow, India. After qualifying GATE, he completed M.TECH in Computer Science and Engineering from Kanpur Institute of Technology, Kanpur, affiliated with Uttar Pradesh Technical University, Lucknow, India. His research interest is Mobile Computing. He worked as an In-Charge in Department of Computer Science & Engineering at MCAET, Ambedkar Nagar.