Fully Complex-valued ELM Classifiers for Real-valued Classification Problems

R. Savitha[1], S. Suresh[2], N. Sundararajan[1], and H. J. Kim[3]

[1] School of Electrical and Electronics Engineering,
Nanyang Technological University, Singapore
[2] School of Computer Engineering,
Nanyang Technological University, Singapore
[3] CIST, Korea University, Seoul

Abstract. The orthogonal decision boundaries of the complex-valued neural networks provide them with an excellent ability to perform classification tasks with good generalization ability. Complex-valued classifiers suffer from issues like slow convergence due to gradient descent based learning, and misclassification due to choice of activation functions. In this paper, we present two fast learning neural network classifiers with a single hidden layer: the 'Phase Encoded Complex-valued Extreme Learning Machine (PE-CELM)' and the 'Bilinear Branch-cut Complex-valued Extreme Learning Machine (BB-CELM)'. The proposed classifiers use the phase encoded transformation and the bilinear transformation with a branch-cut at 2π as the activation functions in the input layer map the real-valued features to the complex domain. The neurons in the hidden layer employ the fully complex-valued activation function of the type of a hyperbolic secant function. The parameters of the hidden layer are chosen randomly and the output weights are estimated as the minimum norm least square solution to a set of linear equations. The universal classification ability of these classifiers are evaluated on a set of benchmark classification problems. Results highlight the superior classification ability of these classifiers with least computational effort.

1 Introduction

Complex-valued neural networks have better computational ability than real-valued networks [2]. Their inherent orthogonal decision boundary [4] that provides them with an exceptional ability to perform real-valued classification tasks [3] motivates researchers to develop complex-valued classifiers.

Complex-valued classifiers available in the literature include the Multi Layer Multi Valued Network (MLMVN) [5] and the single layer complex-valued neural network with phase encoded input features [6] referred to as 'Phase Encoded Complex Valued Neural Network (PE-CVNN)'. A multi-valued neuron used in the MLMVN [5] uses multiple-valued threshold logic to map the complex-valued input to C discrete outputs using a piecewise continuous activation function, where C is the total number of classes. The transformation used in the MLMVN is not unique, leading to misclassification. The misclassification is further enhanced by the increase in number of sectors inside the unit circle in multi-category classification problems with more number of classes (C). Moreover, the MLMVN uses a derivative-free global error correcting rule for network parameter update that requires significant computational effort.

In the PE-CVNN presented in [6], [7], the real-valued input feature is phase encoded in $[0, \pi]$ to obtain the complex-valued input feature. This transformation retains the relational property and spatial relationship among the real-valued input features [6]. However, the activation functions used in [6], [7] are similar to the split complex-valued activation functions and do not preserve the phase information of the error signal during the backward computation. This might result in inaccurate estimation of the decision function while performing classification problems. Moreover, the gradient descent based learning, presented in [6], also requires significant time to train the classifier.

In this paper, we propose two complex-valued classifiers that require considerably insignificant computational effort compared to the other classifiers. The classifiers have a non-linear input/hidden layer and a linear output layer. The neurons at the input layer transform the real-valued input features to the complex domain ($\Re \rightarrow \mathbb{C}$). The bilinear transformation [5] with a branch-cut (δ) and the phase encoding transformation presented in [6] are used as the transformations in the input layer. The network employing the bilinear transformation with a branch-cut in the input layer is referred to as 'Bilinear Branch-cut Complex-valued Extreme Learning Machine (BB-CELM)' and the network employing the phase encoding transformation in the input layer is referred to as 'Phase Encoded Complex-valued Extreme Learning Machine (PE-CELM)'. The neurons in the hidden layer of these networks use the fully complex-valued activation function of the type of a hyperbolic secant function [9]. Similar to the C-ELM [8], the parameters of the hidden neurons are chosen randomly and the output parameters of the networks are estimated analytically.

The performances of both the BB-CELM and the PE-CELM are studied in comparison with other complex-valued and a few real-valued classifiers on three multi-category benchmark classification problems from the UCI repository [10]. It will be observed from the performance study that the proposed classifiers outperform the other classifiers available in the literature.

The paper will be organized as follows: Section 2 presents the detailed description of the BB-CELM and the PE-CELM classifiers. Section 3 presents the performance results of these classifiers in comparison with other classifiers on a set of multi-category benchmark classification problems. Finally, Section 4 summarizes the conclusion from the study.

2 Description of the Classifiers

In this section, we present the detailed description of two fast learning classifiers, the Bilinear Branch-cut Complex-valued Extreme Learning Machine (BB-

CELM) and the Phase Encoded Complex-valued Extreme Learning Machine (PE-CELM) in the complex-domain.

2.1 Classification problem definition

Let us assume N random observations $\{(\mathbf{x}_1, c_1), \cdots, (\mathbf{x}_t, c_t), \cdots, (\mathbf{x}_N, c_N)\}$, where $\mathbf{x}_t \in \Re^m$ are the m-dimensional real-valued input features of tth observation and $c_t \in \{1, 2, \cdots, C\}$ is its class label. The coded class label in the complex domain $\mathbf{y}^t$ are obtained using:

$$y_l^t = \begin{cases} 1+i, & \text{if } c_t = l, \\ -1-i, & \text{otherwise,} \end{cases} \quad l = 1, 2, \cdots, C \tag{1}$$

Now, the classification problem in the complex domain can be viewed as finding the decision function (F) that maps the real-valued input features to the complex-valued coded class labels, i.e., $F : \Re^m \to \mathbb{C}^C$, and then predicting the class labels of new, unseen samples with certain accuracy.

2.2 Fast learning complex-valued classifiers

The basic building block of the PE-CELM and the BB-CELM classifiers is the complex-valued extreme learning machine [8]. The PE-CELM and BB-CELM are the single hidden layer networks, with a non-linear input/hidden layer and a linear output layer as shown in Fig. 1.

The neurons in the input layer employ these transformations to map the real-valued input features to the complex domain ($\Re \to \mathbb{C}$):

- **The bilinear transformation with a branch cut:** As the bilinear transformation [5] results in aliasing at 0 and 2π, we introduce a branch cut around 2π. The transformation thus obtained is termed as a bilinear transformation with a branch cut. The network using this transformation (Eq. (2)) at the input layer is referred to as 'Bilinear Branch-cut Complex-valued Extreme Learning Machine (BB-CELM)'.

$$z_l = exp(2\pi x_l - \delta) \tag{2}$$

- **The phase encoded transformation [6]:** The network with the phase encoded transformation (Eq. (3)) at the input layer is called 'Phase Encoded Complex-valued Extreme Learning Machine (PE-CELM)'. [1]

$$z_l = exp(\pi x_l) \tag{3}$$

The neurons in the hidden layer of the BB-CELM/PE-CELM classifiers employ the fully complex-valued activation function of the type of a hyperbolic secant function [9], and their responses are given by

$$h_j = sech\left(\mathbf{u}_j^T (\mathbf{z}_t - \mathbf{v}_j)\right); j = 1 \cdots K \tag{4}$$

where $\mathbf{u}_j$ is the complex-valued scaling factor and $\mathbf{v}_j$ is the center of the j-th neuron, and $sech(x) = 2/(e^x + e^{-x})$.

[1] It must be noted here that all the input features are scaled in $[0, 1]$, *i.e.,* $\mathbf{x}_t \in \Re[0, 1],\ t = 1 \cdots N$.

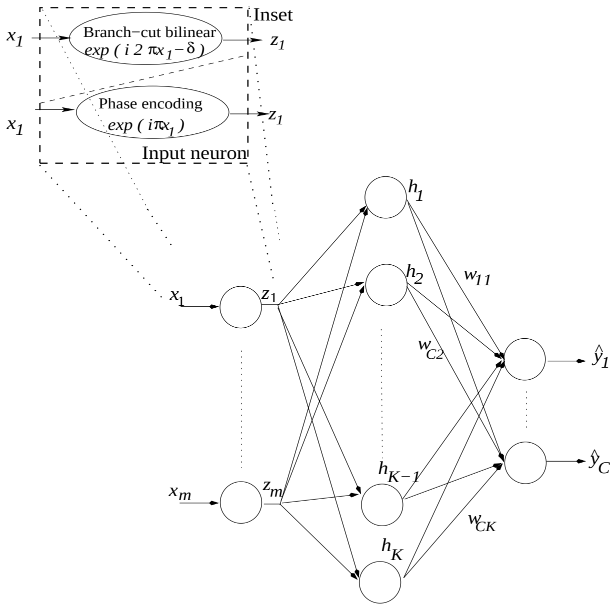

Fig. 1. The architecture of a BB-CELM/PE-CELM. The transformations are shown in the inset

The neurons in the output layer of the BB-CELM/PE-CELM classifiers employ linear activation functions and the output of the classifiers is given by:

$$\widehat{y}_l = \sum_{j=1}^{K} w_{lj} h_j \tag{5}$$

where w_{lj} are the complex-valued weight connecting the l-th output neuron and the j-th hidden neuron.

The class labels can be estimated from the outputs using:

$$\widehat{c} = \max_{l=1,2,\cdots,C} \quad \text{real} \quad (\widehat{y}_l) \tag{6}$$

Eq. (5) can be written in a matrix form as

$$\widehat{Y} = WH \tag{7}$$

where W is the matrix of all output weights connecting the hidden layer, and H is the $K \times N$ matrix of the response of the hidden neurons for the samples in the training data set given by

$$H(V, B, Z) = \begin{bmatrix} sech\,(\mathbf{u}_1 \|\mathbf{z}_1 - \mathbf{v}_1\|) & \cdots & sech\,(\mathbf{u}_1 \|\mathbf{z}_N - \mathbf{v}_1\|) \\ \vdots & \vdots & \vdots \\ sech\,(\mathbf{u}_K \|\mathbf{z}_1 - \mathbf{v}_K\|) & \cdots & sech\,(\mathbf{u}_K \|\mathbf{z}_N - \mathbf{v}_K\|) \end{bmatrix} \tag{8}$$

Similar to the C-ELM [8], the parameters of the hidden neurons ($\mathbf{u}_j$, $\mathbf{v}_j$) are chosen randomly and the output weights W are estimated by the least squares method according to:

$$W = YH^{\dagger} \tag{9}$$

where $H^{\dagger}$ is the Moore-Penrose pseudo-inverse of the hidden layer output matrix, and Y is the complex-valued coded class label.

The proposed PE-CELM/BB-CELM algorithm can be summarized as:

- For a given training set (X, Y), select the appropriate number of hidden neurons K.
- Choose the scaling factor U and the neuron centers V randomly.
- Calculate the output weights W analytically: $W = TY_K^{\dagger}$.

In this paper, selection of an appropriate number of hidden neurons is done using the addition/deletion of neurons to obtain an optimal performance, as discussed in [11] for real-valued networks.

3 Performance evaluation

In this section, the performances of the PE-CELM and the BB-CELM are evaluated on a set of benchmark multi-category classification problems from the UCI machine learning repository [10]. The performance results are compared with other complex-valued classifiers, viz., MLMVN [5] and PE-CVNN [7]. To highlight the advantages of the orthogonal decision boundaries, the performances are also compared with a few real-valued classifiers, viz., the Minimal Resource Allocation Network (MRAN) [12], the Growing and Pruning Radial Basis Function Network (GAP-RBFN) [13], the Online Sequential Extreme Learning Machine (OS-ELM) [14], the Support Vector Machines (SVM) [15], the Self-regulatory Resource Allocation Network (SRAN) [16], and the Real Coded Genetic Algorithm Extreme Learning Machines (RCGA-ELM) [18].

The average (η_a) (Eq. (10)) and over-all (η_o) (Eq. (11)) classification efficiencies [19] of the classifiers derived from their confusion matrices are used as the performance measures for comparison in this study.

$$\eta_a = \frac{1}{C}\sum_{i=1}^{C}\frac{q_{ii}}{N_i} \times 100\% \tag{10}$$

$$\eta_o = \frac{\sum_{i=1}^{C} q_{ii}}{\sum_{i=1}^{C} N_i} \times 100\% \tag{11}$$

where q_{ii} is the total number of correctly classified samples in the class c_i and N_i is the total number of samples belonging to a class c_i in the data set.

The number of classes, the number of features, the size of the training and the testing data sets for the three multi-category benchmark classification problems

Table 1. Description of benchmark data sets selected from [10] for performance study

Problem	No. of features	No. of classes	No. of samples	
			Training	Testing
Image Segmentation (IS)	19	7	210	2,100
Vehicle Classification (VC)	18	4	424	422
Glass Identification (GI)	9	7[a]	109	105

[a] Actual number of classes 7, but no samples in one class

considered for the study, are presented in Table 1. The image segmentation data set is a well-balanced data set. The data set for the glass identification problem is a highly unbalanced data set, with no samples in one class. The vehicle classification data set is also not a well-balanced data set.

The number of hidden neurons chosen according to the discussion in Section 2.2, the training time and the testing performance measures of the BB-CELM and the PE-CELM classifiers, in comparison with other complex-valued and a few real-valued classifiers, are presented in Table 2. The results for the SRAN and RCGA-ELM is reported from [16] and [18], respectively. For the other real-valued classifiers, the results are reproduced from [19]. The results for the PE-CVNN are reproduced from [7] (single layered network) and those of the MLMVN are generated using the toolbox available in the author's web site [1].

From the results, it can be observed that the BB-CELM and PE-CELM classifiers outperform other complex-valued classifiers. The classification performances of the PE-CVNN and the MLMVN classifiers are affected by their activation functions. The activation function of the PE-CVNN is similar to the split complex-valued activation function that does not exploit the complete advantage of the orthogonal decision boundaries of the complex-valued neural networks. On the other hand, the MLMVN maps the complex-valued inputs to C discrete outputs in the unit circle, which increases with C and affects the classification performance.

It can be also observed that the proposed classifiers perform better than the real-valued classifiers, especially in the classification of the unbalanced data sets. This is because of the orthogonal decision boundaries, which is inherent in complex-valued neural networks. It can also be observed that the BB-CELM and the PE-CELM require significantly lesser computational effort, compared to all the other classifiers performing the classification task.

Comparing the performances of the PE-CELM and the BB-CELM classifiers, it is observable that the BB-CELM classifier performs better than the PE-CELM classifier. The phase encoding transformation used in the PE-CELM classifier maps the real-valued input features to only two quadrants of the unit circle pro-

[1] http://www.eagle.tamut.edu/faculty/igor/Downloads.htm

Table 2. Performance results for benchmark multi-category classification problems

Dom -ain	Algo.	Image Segmentation				Vehicle Classification				Glass Identification			
		K	Time (sec.)	Testing		K	Time (sec.)	Testing		K	Time (sec.)	Testing	
				η_o	η_a			η_o	η_a			η_o	η_a
Real	MRAN	76	783	86.52	86.52	100	520	59.94	59.83	51	520	63.81	70.24
	GAP-RBF	83	365	87.19	87.19	81	452	59.24	58.23	75	410	58.29	72.41
	OS-ELM	100	21	90.67	90.67	300	36	68.95	67.56	60	15	67.62	70.12
	SVM[a]	96	721	90.62	90.62	234	550	68.72	67.99	102	320	64.23	60.01
	SMC-RBF	43	142	91	91	75	120	74.18	73.52	58	97	78.09	77.96
	RCGA-ELM	50	-	91	91	75	-	74.2	74.4	60	-	78.1	-
	SRAN	47	22	92.3	92.3	55	113	75.12	76.86	59	28	86.21	80.95
Comp.	MLMVN	80	1384	83	83	90	1396	78	77.25	85	1421	73.24	66.83
	PE-CVNN	-	-	93.2[b]	-	-	-	78.7[c]	-	-	-	65.5[b]	-
	BB-CELM	**65**	**0.03**	**92.5**	**92.5**	**100**	**0.11**	**80.3**	**80.4**	**70**	**0.08**	**88.16**	**81**
	PE-CELM	**75**	**0.03**	**92.1**	**92.1**	**100**	**0.11**	**80.8**	**81.1**	**70**	**0.08**	**86.35**	**80**

[a] Support vectors

[b] -A single layer network is used in [7]. In [7], 75% of the total samples are used in training. In our work, we use only 10% of the samples in training.

[c] -In [7], 75% of the total samples are used in training. In our work, we use only 50% of the samples in training.

hibiting it from exploiting the advantages of the orthogonal decision boundaries completely. The bilinear transformation with a branch cut around 2π used in the BB-CELM classifier uses all the four quadrants of the complex plane.

4 Conclusion

The exceptional universal classification ability of the complex-valued neural network is attributed to their inherent orthogonal decision boundaries. Issues in the complex-valued classifiers, available in the literature, include slow convergence due to the learning algorithm used and inaccurate classification due to the choice of activation function. In this paper, we present two efficient, fast learning complex-valued classifiers, the Bilinear Branch-cut Complex-valued Extreme Learning Machine (BB-CELM) and the Phase Encoded Extreme Learning Machine (PE-CELM). The performances of the BB-CELM and the PE-CELM are evaluated using three benchmark classification problems. From the performance study, it is evident that both the proposed classifiers outperform other classifiers with lesser computational effort. Moreover, the BB-CELM outperforms the PE-CELM as the transformation used in BB-CELM maps the real-valued input features into all the quadrants of the complex plane.

References

1. T. Kim, and T. Adali, Fully-complex multilayer perceptron network for nonlinear signal processing, Journal of VLSI Signal Processing 32 (1-2), 29-43 (2002).

2. T. Nitta, The computational power of complex-valued neuron, Artificial Neural Networks and Neural Information Processing ICANN/ICONIP (Istanbul, Turkey). Lecture Notes in Computer Science **2714**(2003), pp. 993–1000.
3. T. Nitta, Solving the XOR problem and the detection of symmetry using a single complex-valued neuron, Neural Networks **16**(8)(2003) 1101-1105.
4. T. Nitta, On the inherent property of the decision boundary in complex-valued neural networks, Neurocomputing **50**(2003) 291-303.
5. I. Aizenberg, and C. Moraga, Multilayer feedforward neural network based on multi-valued neurons (MLMVN) and a backpropagation learning algorithm, Soft Computing **11**(2)(2007) 169-193.
6. M. F. Amin, and K. Murase, Single-layered complex-valued neural network for real-valued classification problems, Neurocomputing **72**(4-6)(2009) 945-955.
7. M. F. Amin, M. M. Islam, and K. Murase, Ensemble of single-layered complex-valued neural networks for classification tasks, Neurocomputing **72**(10-12)(2009) 2227-2234.
8. M. B. Li, G. B. Huang, P. Saratchandran, and N. Sundararajan, Fully complex extreme learning machines, Neurocomputing **68**(1-4)(2005) 306-314.
9. R. Savitha, S. Suresh, and N. Sundararajan, A fully complex-valued radial basis function network and its learning algorithm, International Journal of Neural Systems **19**(4)(2009) 253 - 267.
10. C. Blake, and C. Merz, UCI Repository of Machine Learning Databases, Department of Information and Computer Sciences, University of California, Irvine, URL: <http://archive.ics.uci.edu/ml/>, 1998.
11. S. Suresh, S. N. Omkar, V. Mani, and T. N. Guru Prakash, Lift coefficient prediction at high angle of attack using recurrent neural network, Aerospace Science and Technology, **7**(8)(2003) 595-602.
12. Y. Lu, N. Sundararajan, and P. Saratchandran, A sequential learning scheme for function approximation using minimal radial basis function neural networks, Neural Computation, **9** (2) (1997) 461-478.
13. G.-B. Huang, P. Saratchandran, and N. Sundararajan, An efficient sequential learning algorithm for growing and pruning RBF (GAP-RBF) networks, IEEE Transactions on System, Man and Cybernetics B **34** (6) (2004) 2284-2292.
14. N.-Y. Liang, G. -B. Huang, P. Saratchandran, and N. Sundararajan, A fast and accurate on-line sequential learning algorithm for feedforward networks, IEEE Transactions on Neural Networks **17** (6) (2006) 1411-1423.
15. N. Cristianini, and J. S. Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, UK, 2000.
16. S. Suresh, Keming Dong, and H. J. Kim, A sequential learning algorithm for self-adaptive resource allocation network classifier, *Neurocomputing*, **73** (16-18) (2010), pp. 3012-3019.
17. H. J. Rong, G. B. Huang, N. Sundararajan, and P. Saratchandran, Online sequential fuzzy extreme learning machine for function approximation and classification problems, IEEE Trans. on Systems, Man, and Cybernetics Part B: Cybernetics, **39**(4)(2009) 1067-1072.
18. S. Suresh, R. Venkatesh Babu, and H. J. Kim, No-reference image quality assessment using modified extreme learning machine classifier, Applied Soft Computing **9**(2)(2009) 541-552.
19. S. Suresh, N. Sundararajan, and P. Saratchandran, A sequential multi-category classifier using radial basis function networks, Neurocomputing **71**(7-9)(2008) 1345-1358.