An Algorithm for Pruning Redundant Modules in Min-Max Modular Network with GZC Function

Jing Li[1], Bao-Liang Lu[1,⋆], and Michinori Ichikawa[2]

[1] Department of Computer Science and Engineering,
Shanghai Jiao Tong University, 1954 Hua Shan Road,
Shanghai 200030, China
jinglee@sjtu.edu.cn , blu@cs.sjtu.edu.cn

[2] Lab. for Brain-Operative Device, RIKEN Brain Science Institue,
2-1 Hirosawa, Wako-shi, Saitama, 351-0198, Japan
ichikawa@brain.riken.go.jp

Abstract. The min-max modular neural network with Gaussian zero-crossing function (M^3-GZC) has locally tuned response characteristic and emergent incremental learning ability, but it suffers from quadratic complexity in storage space and response time. Redundant Sample pruning and redundant structure pruning can be considered to overcome these weaknesses. This paper aims at the latter; it analyzes the properties of receptive field in M^3-GZC network, and then proposes a strategy for pruning redundant modules. Experiments on both structure pruning and integrated with sample pruning are performed. The results show that our algorithm reduces both the size of the network and the response time notably while not changing the decision boundaries.

1 Introduction

The min-max modular (M^3) neural network [1,2] is an alternative modular neural network model for pattern classification. It has been used in real-world problems such as part-of-speech tagging [3] and single-trial EEG signal classification [4]. The fundamental idea of M^3 network is divide-and-conquer strategy: decomposition of a complex problem into easy subproblems; learning all the subproblems by using smaller network modules in parallel; and integration of the trained individual network modules into a M^3 network.

Using linear discriminant function [5] as the base network module, the M^3 network (M^3-Linear) has the same decision boundaries as that of the nearest neighbor classifier (NN) [6]. And M^3-Linear is a specialization of M^3 network with Gaussian zero-crossing function (M^3-GZC) [7], so M^3-GZC can be viewed as a generalization of nearest neighbor classifier. The most attracting attributes of

[⋆] To whome correspondence should be addressed. This work was supported in part by the National Natural Science Foundation of China via the grants NSFC 60375022 and NSFC 60473040.

L. Wang, K. Chen, and Y.S. Ong (Eds.): ICNC 2005, LNCS 3610, pp. 293–302, 2005.
© Springer-Verlag Berlin Heidelberg 2005

M^3-GZC are its locally tuned response characteristic and emergent incremental learning ability. But it suffers from quadratic complexity in space and time, and may be inefficient in large-scale, real-world pattern classification problems.

To decrease the storage space and response time of M^3-GZC network, two ways of redundancy pruning can be considered. One is sample pruning, which is inspired by pruning strategies in NN [8,9,10,11,12,13,14,15]. We have proposed the Enhanced Threshold Incremental Check algorithm [16] for M^3-GZC network in our previous work. The other way is structure pruning, which is correlative with detailed network and can not borrow ideas from NN. In this paper we will analyze the structure of M^3-GZC network and propose a pruning algorithm.

The rest of the paper is organized as follows: In Section 2, M^3-GZC network is introduced briefly. In Sections 3 and 4 we analyze the properties of receptive field and redundant modules in M^3-GZC network. In Section 5 pruning algorithm is described. Experiments are presented in Section 6. Finally, conclusions are presented in Section 7.

2 Min-Max Modular Network with GZC Function

Let $\mathcal{T}$ be the training set for a K-class problem,

$$\mathcal{T} = \{(X_l, D_l)\}_{l=1}^{L}, \tag{1}$$

where $X_l \in R^n$ is the input vector, $D_l \in R^K$ is the desired output, and L is the total number of training data.

According to the min-max modular network [1,2], a K-class problem defined in equation (1) can be divided into $K \times (K-1)/2$ two-class problems that are trained independently, and then integrated according to a module combination rule, namely the minimization principle. Fig.1(a) shows the structure of M^3 network to a K-class problem, where L_i denotes the number of data belonging to class C_i.

A two-class problem can be further decomposed into a number of subproblems and be integrated according to the minimization principle and the maximization principle. These subproblems can be learned by some base network modules, such as SVM[17], back-propagation algorithm[3,4], and so on. Suppose the training set of each subproblem has only two different samples, and the base network module is Gaussian zero-crossing discriminate function as defined in equation (2), the corresponding network is called M^3-GZC network.

$$f_{ij}(x) = exp\left[-\left(\frac{\|x-c_i\|}{\sigma}\right)^2\right] - exp\left[-\left(\frac{\|x-c_j\|}{\sigma}\right)^2\right], \tag{2}$$

where x is the input vector, c_i and c_j are the given training inputs belonging to class C_i and class C_j $(i \neq j)$, respectively, $\sigma = \lambda\|c_i - c_j\|$, and λ is a user-defined constant.

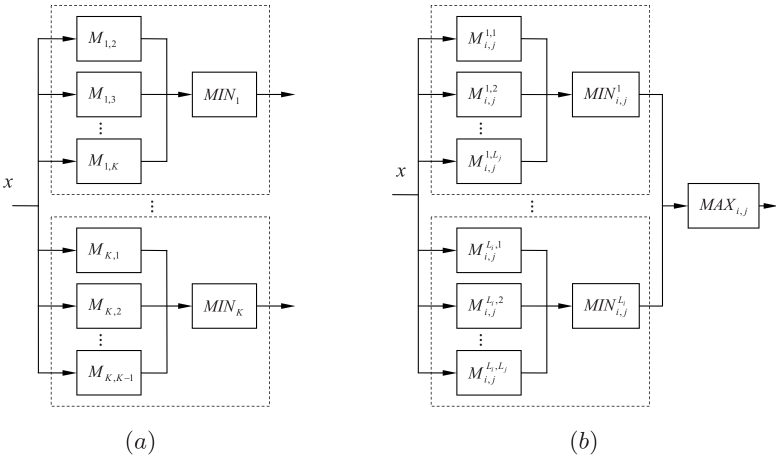

Fig. 1. Structure of M^3-GZC network. (a) A K-class problem; (b)Further decomposition of a two-class problem.

The output of M^3-GZC network is defined as follows.

$$g_i(x) = \begin{cases} 1 & \text{if } y_i(x) > \theta^+ \\ Unknown & \text{if } \theta^- \leq y_i(x) \leq \theta^+ \\ -1 & \text{if } y_i(x) < \theta^- \end{cases} \tag{3}$$

where θ^+ and θ^- are the upper and lower threshold limits, and y_i denotes the transfer function of the M^3 network for class C_i, which discriminates the pattern of the M^3 network for class C_i from those of the rest of the classes.

The structure of M^3-GZC network is shown in Fig.1. It is clear that the total number of modules in a M^3-GZC network is

$$\sum_{i=1}^{K} \sum_{j=1,j\neq i}^{K} L_i \times L_j, \tag{4}$$

which means quadratic complexity in storage space and response time.

3 Properties of Receptive Field in M^3-GZC Network

The receptive field in a M^3-GZC network is defined as the input space that can be classified to one class.

$$RF = \{x | x \epsilon R^n, \exists i, g_i(x) = 1\}. \tag{5}$$

Suppose there are only two samples c_i and c_j, and we only concentrate on the receptive field around c_i. According to the axiom of norm, the following equation is satisfied.

$$\|c_i - c_j\| - \|x - c_i\| \leq \|x - c_j\| \leq \|c_i - c_j\| + \|x - c_i\|. \tag{6}$$

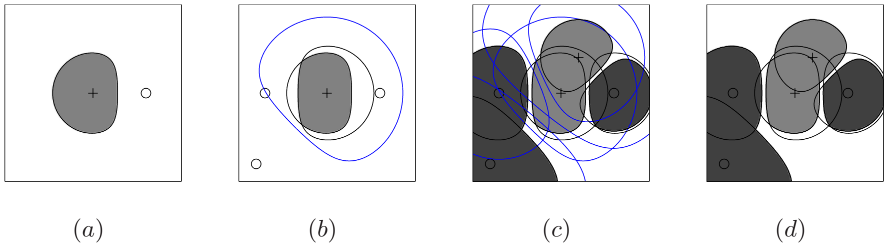

Fig. 2. An illustration of structure pruning. (a) and (b) Receptive fields of a MIN unit; (c) Modules and final decision boundaries in a M^3-GZC network before pruning; (d)Modules and final decision boundaries in a M^3-GZC network after pruning.

So the shortest receptive field radius r_{min} can be obtained when $\|x - c_j\| = \|c_i - c_j\| - \|x - c_i\|$, while the longest receptive field radius r_{max} can be achieved when $\|x - c_j\| = \|c_i - c_j\| + \|x - c_i\|$, as depicted in Fig.2 (a).

From equations (2), (3), (5), and (6), we can prove that the relationship between r_{max} and $\|c_i - c_j\|$ can be expressed as

$$r_{max} = k_1\|c_i - c_j\|, \tag{7}$$

where k_1 is only correlated with λ and θ^+.

Proof: Suppose x is on the direction of r_{max} and on the margin of the receptive field, which means $\theta^+ = f_{ij}(x)$. From equations (2) and (7), we get:

$$\begin{aligned}\theta^+ &= exp\left[-\left(\frac{k_1\|c_i - c_j\|}{\lambda\|c_i - c_j\|}\right)^2\right] - exp\left[-\left(\frac{k_1\|c_i - c_j\| + \|c_i - c_j\|}{\lambda\|c_i - c_j\|}\right)^2\right] \\ &= exp\left[-\left(\frac{k_1}{\lambda}\right)^2\right] - exp\left[-\left(\frac{k_1 + 1}{\lambda}\right)^2\right]. \end{aligned} \tag{8}$$

So k_1 is a function of λ and θ^+.

Also, we can prove that the relationship between r_{min} and $\|c_i - c_j\|$ can be expressed as:

$$r_{min} = k_2\|c_i - c_j\|. \tag{9}$$

where k_2 satisfies the following equation.

$$\theta^+ = exp\left[-\left(\frac{k_2}{\lambda}\right)^2\right] - exp\left[-\left(\frac{1 - k_2}{\lambda}\right)^2\right]. \tag{10}$$

4 Redundancy Analysis

When another sample c_j' belonging to class C_j is available, module $M_{i,j'}$ will be established, which determines another receptive field RF_2 around c_i. Then the minimization principle will be used to combine RF_2 and the field RF_1 that was previously determined by c_j and c_i. Since the role of minimization principle is similar to the logical AND [1], only those fields that contained in both RF_1 and RF_2 will be the final receptive field RF, as shown in Fig.2 (b). In other word, if RF_2 includes RF_1, RF will be equal to RF_1. In this case, sample c'_j has no contribution to the final receptive fields around c_i, and module $M_{i,j'}$ is a redundant module.

Now the question of under what circumstances RF_2 will include RF_1 arises. Here we give a sufficient proposition.

Proposition 1: Suppose sample c_j is the nearest sample in class C_j to sample c_i, if sample c'_j in class C_j satisfies equation (11), then module $M_{i,j'}$ is a redundant module.

$$\|c_i - c'_j\| \geq \frac{k_1}{k_2}\|c_i - c_j\| \tag{11}$$

The proof is straightforward. From equation (11) we can get $k_2\|c_i-c_{j'}\| \geq k_1\|c_i-c_j\|$, which means that r_{min} of RF_2 is larger than r_{max} of RF_1, so $RF_1 \subseteq RF_2$, and module $M_{i,j'}$ is a redundant module.

For a k-class classification problem, proposition 1 can be extended to proposition 2 according to the minimization principle in K-class classification problems[1].

Proposition 2: Suppose sample c_j is the nearest sample in class C_j $(1 \leq j \leq K, j \neq i)$to sample c_i, if sample c_k in class C_k $(1 \leq k \leq K, k \neq i)$ satisfies equation (12), then module $M_{i,k}$ is a redundant module.

$$\|c_i - c_k\| \geq \frac{k_1}{k_2}\|c_i - c_j\| \tag{12}$$

5 Pruning Algorithm

For a K-class problem defined in equation (1), according to proposition 2, our pruning algorithm works as below.

1. Calculate k_1 and k_2 according to λ and θ^+;
2. For each sample (x, d) in T,
   (a) Find the nearest neighbor (x', d') in T, $d \neq d'$ and $\|x-x'\| = MIN\{\|x''-x\|\}$, $(x'', d'')\epsilon T$, $d'' \neq d$.
   (b) For each sample (x'', d'') in T $(d'' \neq d)$, if $\|x'' - x\| \geq \frac{k_1}{k_2}\|x' - x\|$, prune the module based on (x, d) and (x'', d'').

The final structure of pruned M^3-GZC network is composed of L MIN units, as shown in Fig.3. Each MIN unit is composed of a center sample and some neighbors in different classes around it. When a test sample x is presented, if it is in the receptive field of one MIN unit, then the calculation is completed, and the output is the same as the class of the center sample. If x is rejected by all the MIN units, then the output is '*Unknown*'.

Suppose there are N_i neighbors around one center sample, N_i is determined by the distribution of training samples. The total number of modules in the

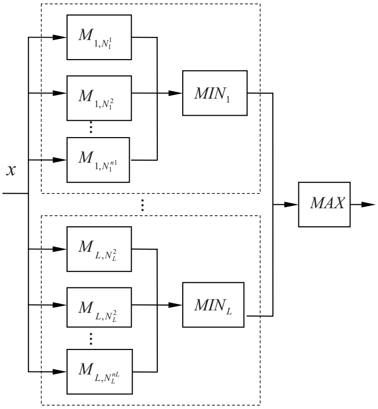

Fig. 3. Structure of pruned M^3-GZC network. N_i^j denotes the jth neighbor around sample i

pruned M^3-GZC network is $\sum_{i=1}^{L} N_i$, which is less than that in the original M^3-GZC network: $\sum_{i=1}^{K}\sum_{j=1,j\neq i}^{K} L_i \times L_j$.

An illustration of our pruning algorithm is depicted in Fig.2 (c) and (d). Each circle line represents a module in M^3-GZC network. The black and grey areas denote the receptive field of each class, while the white area denotes the '*Unknown*' output. Form the results, we can see that the decision boundaries are identical, while 41.7% modules are pruned.

6 Experimental Results

In order to verify our method, we present experiments on three data sets. The first is an artificial problem and the other two are real-world problems. We also do the experiments that integrating our method with sample pruning. All the experiments were performed on a 2.8GHz Pentium 4 PC with 1GB RAM.

6.1 Two-Spiral Problem

We test our structure pruning algorithm on the popular two-spiral benchmark problem firstly. The data include 192 training samples and test samples respectively (non-overlapping). The parameters of the experiment are given as follows: $\lambda = 0.5$; $\theta^{+} = 0.01$; $\theta^{-} = -0.01$. The correspond k_1 and k_2 is 1.073 and 0.497, respectively. Fig.4 (a) shows the original problem, Fig.4 (b) shows the decision boundaries before pruning and Fig.4 (c) shows the decision boundaries after pruning. As we have expected, they are identical, but the number of modules and response time are greatly reduced, as listed in Table 1.

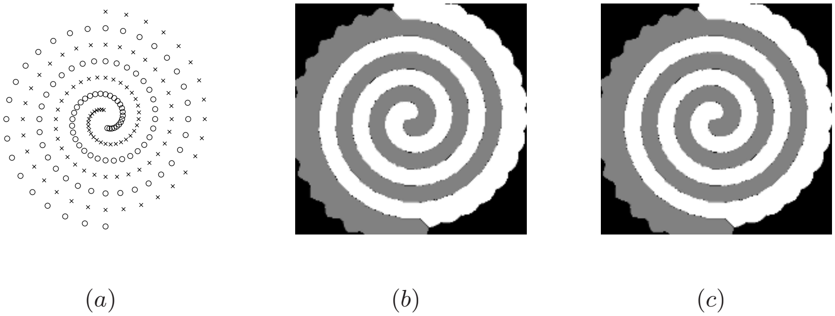

Fig. 4. Results on two-spiral problem. (a) Training samples; (b) Decision boundaries before pruning; (c) Decision boundaries after pruning. Here black area denotes 'Unknown' output.

Table 1. Experimental results. The upper row in each experiment denotes the pruned net while the lower row denotes the original net. The unit of 'Time' is ms.

Data set	Accuracy	Unknown	False	Size	Time	Size Ratio	Speed Up
two-spirals	100%	0.0%	0.0%	2516	18	13.7%	129
	100%	0.0%	0.0%	18432	2315		
balance	92.0%	0.0%	8.0%	39377	42	31.6%	137
	92.0%	0.0%	8.0%	124800	5767		
car	57.87%	42.13%	0.0%	126079	1805	37.7%	60
	57.87%	42.13%	0.0%	334006	107878		
image	84.0%	7.33%	8.67%	11280	449	33.0%	66
	84.0%	7.33%	8.67%	34200	29730		
Iris	94.67%	1.33%	4.0%	1843	3	49.1%	84
	94.67%	1.33%	4.0%	3750	252		
optdigits	97.22%	1.45%	1.34%	11454592	10784	89.1%	700
	97.22%	1.45%	1.34%	12862520	7548237		
glass image	86.0%	2.0%	12.0%	1167989	18817	43.7%	125
	86.0%	2.0%	12.0%	2673000	2349796		

6.2 UCI Database

In this experiment, our algorithm is tested on five benchmark data sets from the Machine Learning Database Repository[18]: Balance, Car, Image Segmentation, Iris and Optdigits. The detailed information of each problem is described in Table 2. The parameters of each experiments are same as those in the two-spiral problem, and results are listed in Table 1.

Table 2. Number of class, dimension, training samples and test samples in UCI database

Data Set	Class	Dimension	Training	Test
balance	2	4	500	125
car	4	6	864	864
image	5	19	210	2100
Iris	3	4	75	75
optdigits	9	64	3823	1797

Table 3. Experimental results of integrating sample pruning and structure pruning. The upper row in each experiment denotes the net after sample pruning and structure pruning while the lower row denotes the net only with sample pruning. The unit of 'Time' is ms.

Data set	Accuracy	Unknown	False	Size	Time	Size Ratio	Speed Up
two-spirals	100%	0.0%	0.0%	794	11	4.31%	208.3
	100%	0.0%	0.0%	8192	1268	44.4%	1.82
balance	92.0%	0.0%	8.0%	9878	15	7.92%	384.6
	92.0%	0.0%	8.0%	44676	2510	35.8%	2.30
car	62.15%	34.14%	3.70%	31322	645	9.38%	166.7
	62.15%	34.14%	3.70%	111138	36392	33.3%	2.97
image	82.0%	9.24%	8.76%	3280	478	9.59%	625
	82.0%	9.24%	8.76%	11162	12035	32.6%	2.47
Iris	94.67%	1.33%	4.0%	345	3	9.2%	1.19%
	94.67%	1.33%	4.0%	570	125	15.2%	84.0
optdigits	96.05%	2.62%	1.34%	1137798	3714	8.85%	2000
	96.05%	2.62%	1.34%	1378048	840613	10.7%	9.0
glass image	85.55%	2.59%	11.86%	46397	16049	1.74%	147.1
	85.55%	2.59%	11.86%	176928	151796	6.62%	15.5

6.3 Industry Image Classification

Due to its locally tuned response characteristic and incremental learning ability, M^3-GZC has been used in an industry fault detection project. The purpose of this project is to choose out eligible glass-boards in an industrial product line, which is done by trained workers in practice. It is a boring work; workers are easy to be tired and then make wrong decisions. With the help of M^3-GZC network, workers need only judge the glass-boards that are classified to 'Unknown' by the network. In our experiment, each glass-board image is converted into a 4096 dimension vector, 3420 images are used as training data while 1197 images as test data. The parameters are same as those in the two-spiral problem, and results are listed in Table 1.

From Table 1, several observations can be made. Our pruning method has no influence on the classification accuracy, but the size and response time can be decreased notably, by an average of 42.6% and 0.975%, respectively. The response time is saved much further than the size. This is due to that in the pruned net it need not calculate all the modules to get the answer, if there is a MIN unit accepts it, the calculation can be finished. Only those inputs that the correspond result is '*Unknown*' will calculate all the modules. But in most cases, the '*Unknown*' ratio is very low. So the response time can be cut down greatly.

6.4 Integrated with Sample Pruning

Experiments of integrating sample pruning (Enhanced Threshold Incremental Check)[16] and structure pruning are also conducted on the data sets mentioned above. First we use ETIC to prune redundant samples in each training data set; then we use our structure pruning algorithm to prune redundant models. The results are listed in Table 3. We can see that the size and response time are decreased much further, by an average of 7.28% and 0.49%, respectively.

7 Conclusions

M^3-GZC network has the locally tuned response characteristic and emergent incremental learning ability. But it suffers from sample redundancy and module redundancy. In this paper we have presented a novel structure pruning algorithm to reduce the redundant modules based on the properties of receptive field in M^3-GZC network. The decision boundaries of the pruned net are identical with the original network, but the storage and response time requirement decreased significantly. Experiments on structure pruning and integrated with sample pruning verified the effectiveness of our pruning algorithm. We believe that module redundancy reflects sample redundancy, our future work is to investigate the relationship between them and combine them more effectively.

References

1. Lu, B.L. and Ito, M.: Task decomposition based on class relations: a modular neural network architecture for pattern classification. Lecture Notes in Computer Science, Springer vol.1240(1997)330-339
2. Lu, B.L. and Ito, M.: Task Decomposition and Module Combination Based on Class Relations: A Modular Neural Network for Pattern Classification. IEEE Trans. Neural Networks, vol.10 (1999) 1244–1256
3. Lu, B.L., Ma, Q., Ichikawa,M. and Isahara, H. :Efficient Part-of-Speech Tagging with a Min-Max Modular Neural Network Model. Applied Intelligence, vol.19 (2003)65–81

4. Lu, B.L., Shin, J., and Ichikawa, M.: Massively Parallel Classification of Single-Trial EEG Signals Using a Min-Max Modular Neural Network. IEEE Trans. Biomedical Engineering, vol.51, (2004) 551–558
5. Duda, R.O., Hart, P.E. and Stork, D.G.: Pattern Classification, 2nd Ed. John Wiley & Sons, Inc. 2001.
6. Wang, Y.F.: A study of incremental learning method based on Min-Max Module networks (in Chinese). Bachelor Dissertation of SJTU, 2003.
7. Lu, B.L. and Ichikawa, M.: Emergent On-Line Learning with a Gaussian Zero-Crossing Discriminant Function. IJCNN '02, vol.2 (2002) 1263–1268
8. Hart, P.E.: The Condensed Nearest Neighbor Rule. IEEE Trans. Information Theory, vol.14 (1968) 515–516
9. Gates, G.W.: The Reduced Nearest Neighbor Rule. IEEE Trans. Information Theory, vol.18 (1972) 431–433
10. Wilson, D.L.: Asympotic propoties of nearest neighbor rules using edited data. IEEE trans. System, Man, and Cybernetics, vol.2, No.3 (1972) 431–433
11. Aha, D.W., Dennis, K. and Mack, K.A.: Instance-based learning algorithm. Machine Learning, vol.6 (1991) 37–66
12. Zhang, J.P.: Selecting typical instances in instance-based learning. Proceedings of the Ninth International Conference on Machine Learning, (1992) 470–479
13. Wai, L., Keung, C.K. and Liu, D.Y.: Discovering useful concept prototypes for classification based on filtering and abstraction. IEEE Trans. Pattern Analysis and Machine Intelligence, vol.24, No.8 (2002) 1075–1090
14. Skalak, D.B.: Prototype and feature selection by sampling and random mutation hill climbing algorithm. Proceedings of the Eleventh International Conference on Machine Learning, (1994) 293–301
15. Cameron-Jones, R.M.: Instance selection by encoding length heuristic with random mutation hill climbing. Proceddings of the Eightn Australian Joint Conference on Artificial Intelligence, (1995) 99–106
16. Li, J., Lu, B.L. and Ichikawa, M.: Typical Sample Selection and Redundancy Reduction for Min-Max Modular Network with GZC Function. ISNN'05, Lecture Notes in Computer Science, (2005) 467-472
17. Lu, B.L., Wang, K.A., Utiyama, M. and Isahara, H.: A part-versus-part method for massively parallel training of support vector machines. Proceedings of IJCNN'04, Budapast, July25-29 (2004) 735-740.
18. Murphy, P.M. and Aha, D.W.: UCI Repository of Machine Learning Database. Dept. of Information and Computer Science, Univ. of Calif., Irvine, 1994.