

Automatic Construction Algorithm for Multi-class Support Vector Machines with Binary Tree Architecture

Gexiang Zhang, and Weidong Jin

School of Electrical Engineering, Southwest Jiaotong University, Chengdu, 610-031 China

Summary

Multi-class support vector machines with binary tree architecture (SVM-BTA) have the fastest decision-making speed in the existing multi-class SVMs. But SVM-BTA usually has bad classification capability. According to internal characteristics of feature samples, this paper uses resemblance coefficient method to construct automatically binary tree to incorporate multiple binary SVMs. The multi-class SVMs with constructed binary tree have good classification capability and fast decision-making speed. Experimental results of yeast protein localization site prediction and radar emitter signal recognition show that the introduced multi-class SVMs with binary tree architectures are superior to several popular multi-class SVMs including one-against-all, one-against-one, directed acyclic graph, bottom-up binary tree and several classification methods in the recent literatures.

Key words:

Support vector machines, Resemblance coefficient, Radar emitter, Binary tree architecture

1. Introduction

Depending on a large number of observations, even depending on infinite observations, traditional statistical methods have good performances in theory. Different from traditional statistical methods, support vector machines (SVMs) can achieve ideal results when the number of given observations is small [1-3]. SVM is a novel and small-size sample machine learning technique. From the nature of logic, SVMs use efficient transduction inference from training samples to predicting samples, which avoids the traditional thinking from induction to deduction [1,3]. So SVMs simplify the problems of conventional classification and regression greatly. Because the decision functions are only related to support vectors, SVMs have good robustness and generalization [4]. As a valid method for dealing with nonlinear classification and nonlinear regression, SVMs have become very popular in many domains, such as audio classification and retrieval [5], visual object recognition [6], target recognition [7], radar emitter signal recognition [8], handwritten Chinese character recognition [9], wind speed prediction [10] and financial time series forecasting [11].

However, SVM was designed for binary classification originally and it is not a straightforward issue to extend binary SVM to multi-class problem [12-15]. Constructing multi-class SVMs is still an on-going research issue [13-21]. The combination of multiple binary SVMs is a popular technique to solve multi-class classification problems. A great number of experimental results and engineering applications verify that the combination method of binary SVMs is a valid and practical way for solving multi-class classification problems [5-9,13-21]. In the existing literatures, there are mainly 5 combination approaches presented to incorporate binary-class SVMs. They are respectively one-against-all (OAA) [14,16,17], one-against-one (OAO) [14,19], directed acyclic graph (DAG) [20], bottom-up binary tree (BUBT) [5] and binary trees architecture (BTA) [6,13-16,21,22].

For an N -class classification problem, BTA need only test $\log_2 N$ binary SVMs for classification decision, while the other 4 methods need make at least $N-1$ binary decisions. So BTA has the fastest decision speed among the 5 combination methods. The recognition speed is a very vital performance index in some real-time applications such as radar emitter signal recognition. In the procedure of constructing a binary tree, how to choose the root node of every layer is an important issue. In [16], k -means clustering method was used to construct the binary tree. In [15], a kernel-based self-organizing map was used to convert the multi-class problem into binary hierarchies. The conversion employed two methods including human drawing and automatic clustering that maximizes a scattering measure calculated from the data. In [21], the minimum spanning tree algorithm was used as a tool for finding binary partition of classes in a multi-class learning problem. But k -means clustering and automatic clustering [15] can not process effectively the cases that there are overlaps among multiple or all classes. Human drawing is suitable for the classification problem with only 2-dimensional or 3-dimensional features because high dimensional features vector can not plot intuitively in feature space. Furthermore, the conversion in [15] requires exhaustive search for all possible combinations. The minimum spanning tree algorithm will product multiple binary tree architectures [21], that is to say, the algorithm

will result in multiple solutions, which brings another problem of choosing the best binary tree.

This paper proposes a novel method for constructing multi-class SVM classifiers with binary tree architectures. The method called Resemblance Coefficient multi-class SVM classifiers with Binary Tree Architectures (RCBTA) uses resemblance coefficient construction algorithm (RCA) to convert the multi-class problem into several binary-class problems. Utilizing the distribution of samples, RCA finds the optimal binary partition of classes in every layer of the binary tree. Both testing experiments and an application example of radar emitter signal recognition verify that the introduced method is superior to OAA, OAO, DAG, BUBT and several methods in [22,24-26] in recognition speed and classification capability.

The remainder of the paper is organized as follows. Section 2 introduces RCA. Results of simulation experiments are analyzed in section 3. Conclusions are listed in Section 4.

2. Resemblance Coefficient Construction Algorithm

The existence of confusion classes is one of the most important problems in multi-class pattern recognition problems [16]. It is also the main reason that results in misclassifications of different classes. How to deal with the overlapping samples of different classes is the key technique that converts multi-class classification problems into multiple binary-class classification problems [16]. So when binary-class SVMs are used to incorporate a binary tree, it is critical technique for making a coarse discrimination between confusion classes and then a finer discrimination within the confusion classes to determine how to partition multiple classes in every binary-class SVM in the tree. This section gives an algorithm to accomplish this task. In this section, the definition and the property of resemblance coefficient are firstly given. Then, a criterion function is presented to judge the degree of confusion of different classes. Finally, the detailed algorithm for constructing the binary tree is introduced

Definition 1 Suppose that one-dimensional functions $f(x)$ and $g(x)$ are continuous, positive and real, i.e.

$$f(x) \geq 0, \quad g(x) \geq 0. \quad (1)$$

Resemblance coefficient of function $f(x)$ and $g(x)$ is defined as

$$C_r = \frac{\int f(x)g(x)dx}{\sqrt{\int f^2(x)dx} \cdot \sqrt{\int g^2(x)dx}}. \quad (2)$$

In Eq. (2), the integral domains of $f(x)$ and $g(x)$ are their definable domains of the variable x . Moreover, when x is within its definable domain, the value of function $f(x)$ or $g(x)$ cannot be always equal to 0 [27].

Property 1 The value domain of resemblance coefficient C_r is

$$0 \leq C_r \leq 1. \quad (3)$$

Because $f(x)$ and $g(x)$ are positive functions, according to the famous *Cauchy Schwartz* inequality, we can obtain

$$0 \leq \int f(x)g(x)dx \leq \sqrt{\int f^2(x)dx} \cdot \sqrt{\int g^2(x)dx} \quad (4)$$

$$0 \leq \frac{\int f(x)g(x)dx}{\sqrt{\int f^2(x)dx} \cdot \sqrt{\int g^2(x)dx}} \leq 1 \quad (5)$$

Obviously, we can infer $0 \leq C_r \leq 1$. According to the conditions of *Cauchy Schwartz* inequality, if $f(x)$ equals to $g(x)$, resemblance coefficient C_r of $f(x)$ and $g(x)$ gets the maximal value 1. In fact, if and only if the $f(x)$ -to- $g(x)$ ratio in every point is constant, resemblance coefficient C_r equals to 1. If and only if the integral of product of $f(x)$ and $g(x)$ is zero, i.e. for arbitrary x , $f(x)=0$ or $g(x)=0$, resemblance coefficient C_r equals to the minimal value 0 [27].

From Def. 1, computing resemblance coefficient of two functions corresponds to computing the correlation of the two functions. The value of resemblance coefficient mainly depends on the characteristics of two functions. If $f(x)$ is in proportion to $g(x)$, i.e. $f(x)=kg(x)$, $k>0$, the value of resemblance coefficient C_r equals to 1, which indicates function $f(x)$ resembles $g(x)$ completely. As the overlapping of the two functions decreases gradually, resemblance coefficient C_r will increase gradually, which indicates that $f(x)$ and $g(x)$ are resemblant partly. When $f(x)$ and $g(x)$ are completely separable, C_r gets to the minimal value 0, which implies $f(x)$ does not resemble $g(x)$ at all.

To compute the degree of sample overlapping of two classes, we introduce the following criterion function:

$$J = 1 - \frac{\int f(x)g(x)dx}{\sqrt{\int f^2(x)dx} \cdot \sqrt{\int g^2(x)dx}}. \quad (6)$$

According to the definition and property of resemblance coefficient, the value of J is always equal to or more than zero. For any x , if $f(x) \neq 0$ and $g(x)=0$ or if $g(x) \neq 0$ and $f(x)=0$, J arrives at the maximal value. If $f(x)$ is the same as $g(x)$, $J=0$. So the criterion function J given in Eq.(6) satisfies the three class separability conditions that class

separability criterion based on probability distribution must satisfy [23]. i.e. (i) the criterion function value is not negative; (ii) if there is not overlapping part of distribution functions of two classes, the criterion function value gets to the maximal value; (iii) if distribution functions of two classes are identical, the criterion function value is 0.

When the two functions $f(x)$ and $g(x)$ in Eq.(6) are regarded respectively as probability distribution functions of feature samples of two classes A and B, several separability cases of A and B are shown in Fig. 1. For all x , if one of $f(x)$ and $g(x)$ is zero at least, which is shown in Fig.1(a), A and B are completely separable and the criterion function J arrives at the maximal value 1. If there are some points of x that make $f(x)$ and $g(x)$ not equal to 0 simultaneously, which is shown in Fig.1(b), A and B are partly separable and the criterion function J lies in the range between 0 and 1. For all x , if $f(x)=kg(x)$, $k \in R^+$, which is shown in Fig.1(c), $k=2$, A and B are not completely separable and the criterion function J arrives at the minimal value 0.

In pattern recognition, the extracted features usually order a certain laws. In general, the features vary in the neighboring area of expectation value because of plenty of noise and measurement errors. If occurrences of all feature samples are computed in statistical way, a feature probability distribution function can be obtained. The function can be considered approximately as a Gaussian distribution function with the parameters of expectation and variance of feature samples. Thus, functions $f(x)$ and $g(x)$ in Eq.(6) can be considered as feature distribution functions of different classes. Fig.1(b) and Eq. (6) indicate that the more serious the overlapping of two classes is, the larger the criterion function is. In other words, when functions $f(x)$ and $g(x)$ in Eq.(6) stand for feature distribution functions of two classes, the criterion function value denotes the confusion of two classes. Thus, we can use the criterion function in (6) to classify multiple classes into binary classes gradually. An additional explanation is that any function satisfied the conditions in definition 1 can be used as $f(x)$ or $g(x)$ in Eq.(6).

According to the above criterion function, the detailed algorithm for constructing binary tree architecture to combine multiple binary-class SVMs is given as follows.

Step 1 Initialization: deciding the number N of classes, the number M of features and the number L of samples in training set.

Step 2 Computing the mean value a_{ij} ($i = 1, 2, \dots, N, j = 1, 2, \dots, M$) and the variance value v_{ij} ($i = 1, 2, \dots, N, j = 1, 2, \dots, M$) of L samples of the j th ($j = 1, 2, \dots, M$) feature of the i th ($i = 1, 2, \dots, N$) class.

Step 3 Let $j=j+1$, repeating step 2 until $j=M$.

Step 4 Let $i=i+1$, repeating step 2 and step 3 until $i=N$.

Step 5 For the j th ($j = 1, 2, \dots, M$) feature (in the beginning, $j = 1$), the mean values $a_{1j}, a_{2j}, \dots, a_{Nj}$ of N classes are sorted from small to the large.

The sorted results are represented with $b_{1j}, b_{2j}, \dots, b_{Nj}$.

Step 6 Let $j=j+1$, repeating step 5 until $j=M$.

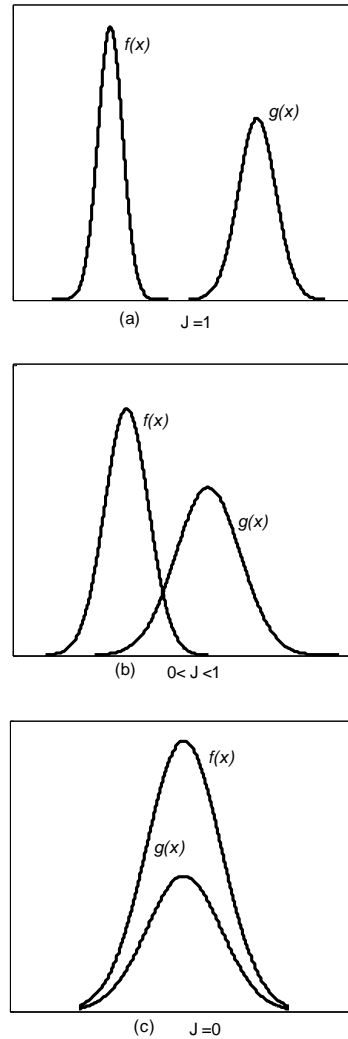


Fig. 1 Three separability cases of functions $f(x)$ and $g(x)$.

Step 7 Choosing a real number T_h ($0 \leq T_h \leq 1$) as the adjusting parameter. When $T_h = 1$, the neighboring two classes have no overlapping samples nearby. When $T_h = 0$, the samples of the neighboring two classes are overlapping seriously. As T_h decreases

from 1 to 0, the confusion samples increase. Through changing the adjusting parameter, we can gradually convert multi-class classification problems into binary-class classification problems.

Step 8 The position, where the smallest mean value b_{1j} of the j th feature is, is encoded to zero (i.e. Code=0) to be the initial value.

Step 9 Starting with the smallest value b_{1j} of the j th feature, the criterion function value $J_k (k=2,3,\dots,N)$ of b_{1j} and $b_{1(j+1)}$ is computed by the sorted order $b_{1j}, b_{2j}, \dots, b_{Nj}$ using (6). In the procedure of computing, the function $f(x)$ is considered as a Gaussian function of the mean value b_{1j} and its corresponding variance value and the function $g(x)$ is considered as a Gaussian function of the mean value $b_{1(j+1)}$ and its corresponding variance value. If $J_k \geq T_h$, the encoded value of the corresponding position of the mean value $b_{1(j+1)}$ adds 1, i.e. Code=Code+1. Otherwise, if $J_k < T_h$, the encoded value of the corresponding position of the mean value $b_{1(j+1)}$ keeps unchanging, i.e. Code=0.

Step 10 Let $k=k+1$, repeating step 9 till $k=N$.

Step 11 Let $j=j+1$, repeating step 8, step 9 and step 10 until $j=M$.

Step 12 An encoded table is finally obtained. An example of the encoded table is shown in Table 1.

Table 1: An example of encoded table (2 rows denote two features and 10 column denote 10 classes)

Classes	1	2	3	4	5	6	7	8	9	10
Feature1	1	1	1	0	0	3	2	1	1	0
Feature2	0	0	0	0	0	0	0	0	0	0

Step 13 According to the encoded table, we can easily classify multiple classes into two groups. The detailed classification steps are described in the following description. The classes with the same lines are considered as a group. Take Table 1 for example, column 1,2,3,8 and 9 are the same and column 4,5 and 10 are the same. So we can first group classes 1,2,3,8 and 9. The rest classes 4,5,6,7 and 10 are grouped. Of course, classes 6 and 7 may belong to any group because the two classes have different code from any of 10 classes. To balance the binary-class SVM, we here group classes 6 and 7 with classes 4, 5 and 10.

Step 14 Decreasing the adjusting parameter, repeating step 7 to step 13. When the adjusting parameter decreases, we obtain Table 2. In Table 2, classes 1,2 and 8 have the same code. So we classify the former group into two subgroups: one is composed of

classes 1,2 and 8; the other is composed of classes 3 and 9. Similarly, the latter group is also classified two subgroups: one is composed of classes 4 and 5; the other is composed of classes 6, 7 and 10.

Step 15 The rest may be deduced by analogy. Finally, a whole binary tree, which is shown in Fig.2, is constructed.

Table 2: The encoded table obtained when the adjusting parameter decreases

Classes	1	2	3	4	5	6	7	8	9	10
Feature1	1	1	2	0	0	4	3	1	2	0
Feature2	0	0	0	2	2	0	1	0	2	0

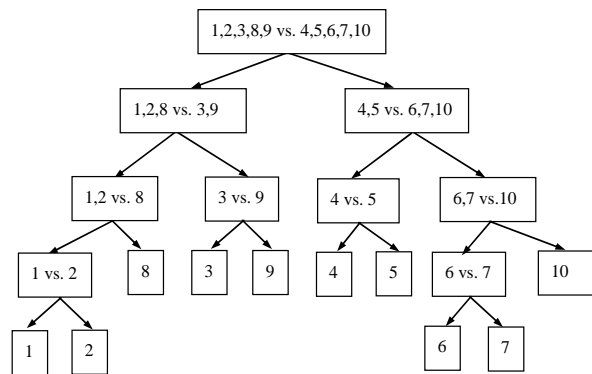


Fig. 2 An example of binary tree architecture constructed.

3. Experiments

In this section, two experiments are made to verify the performance of the proposed algorithm. One is prediction of yeast protein cellular localization sites, which is a general benchmark and a very difficult classification problem [22, 24-26]. The other is an application example of radar emitter signal recognition, which is a very important and difficult problem in modern electronic warfare [27-30].

3.1 Testing Experiments

The dataset of yeast protein localization sites was donated by Horton and Nakai in 1996. The detailed information can be found in [24,25]. The dataset can be obtained in this web [31]. In this dataset, the class is the localization site and there are 10 classes and 8 features for prediction. The number of samples for predicting yeast protein localizations is 1484. The 10 classes are respectively cytosolic or cytoskeletal (CYT), nuclear (NUC), mitochondrial (MIT), membrane protein with no N-terminal signal (ME3), membrane protein with uncleaved signal (ME2), membrane protein with cleaved

signal (ME1), extracellular (EXC), vacuolar (VAC), peroxisomal (POX) and endoplasmic reticulum lumen (ERL). The 10 classes are labeled 1, 2, 3, 4, 5, 6, 7, 8, 9 and 10, respectively. The description of this dataset is shown in Table 3.

According to the samples of 10 classes in this dataset, we use the proposed algorithm to construct binary tree architecture, which is shown in Fig.3.

Table 3: Yeast protein dataset description (NoS is abbreviation of number of samples)

classes	CYT	NUC	MIT	ME3	ME2
NoS	463	429	244	163	51
classes	ME1	EXC	VAC	POX	ERL
NoS	44	35	30	20	5

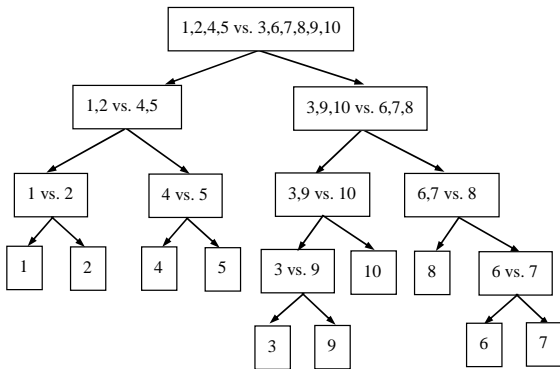


Fig. 3 Binary tree architecture for predicting yeast protein localization sites.

In order to bring into comparison with the methods in [24-26], we use 5-fold cross-validation methodology to test the introduced algorithm. That is to say, the samples of every class are divided into 5 subsets of approximately equal size. Gaussian function is chosen as kernel function of SVMs. To decrease the effect of changing parameters, 63 combinations of constant $C = [10^0, 10^1, 10^2, 10^3, 10^4, 10^5, 10^6]$ and kernel parameter $\sigma = [0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10]$ are used to test respectively the multi-class SVMs with the binary tree architecture shown in Fig.3. The testing results of 5 subsets are given in Table 4. In this Table, we give the highest recognition rate among 63 experimental results for every cross-validation. Also, the parameters corresponding to the best result is given. The final experimental result obtained using the multi-class SVM classifiers with the binary tree architecture shown in Fig.3 is given in Table 5. In [24, 25, 26], multiple methods were used to predict yeast protein localization sites. The methods are respectively probabilistic classification system (PCS) [25], k NN [24], decision tree (DT) [24], Naïve Bayes (NB) [24], HN [24],

growing cell structures (GCS) [26], feed forward neural networks (FN) [26], genetic algorithm (GA) [26] and ERR [26]. The experimental results in [24, 25, 26] are also shown in Table 5. Additional explanation is that the number of samples of CYT and NUC has a little difference. In [24], CYT and NUC have 444 samples and 426 samples respectively, while in this paper, they are 463 and 429, respectively.

Table 4: Experimental results of 5 subsets (SS) in 5-fold cross-validation tests (%)

Classes	SS ₁	SS ₂	SS ₃	SS ₄	SS ₅
1	70.33	58.06	82.80	61.29	72.04
2	62.35	50.00	52.33	53.49	43.02
3	41.67	46.94	42.86	51.14	69.39
4	77.42	66.67	84.85	87.88	87.88
5	9.09	20.00	40.00	50.00	50.00
6	75.00	88.89	77.78	66.67	77.78
7	28.57	85.71	57.14	42.86	57.14
8	0	0	0	16.67	0
9	50	25.00	25.00	0	50.00
10	100	0	0	100	100
C	10^2	10^3	10^2	10^3	10^6
σ	0.5	1	0.5	0.5	5

The baseline correct recognition rate, which is the error rate for a classifier that always predicts the class with most instances, is 54.50% . From Table 4 and Table 5, the introduced method achieves 59.37% correct recognition rate, which is much higher than PCS, DT, NB, HN, GCS, FN, GA and ERR. Though, the correct recognition rate is a little lower than that of PCS. The reason is that the number of samples of CYT and NUC in [24] is not identical with that in this paper. The experimental results indicate the validity and superiority of the introduced method.

In [22], only 9 classes were used to make 10-fold cross-validation experiments. To compare RCBTA with several methods in [22], we also use the 9 classes to test the performance of the proposed method. In this experiment, 1484 samples are divided into 10 subsets. In every test, 9 subsets are employed to train SVMs and the rest one subset is used to test trained SVMs. The testing results of 10 subsets are given in Table 6. In this Table, we give the highest recognition rate among 63 experimental results for every cross-validation. The average correct recognition rate of RCBTA is shown in Table 7. There are 13 methods in [22] and they are 1AA (one-against-all), AAA (all- against-all), DDAG (decision directed acyclic graph), ECOC (error correcting output codes), 1, 2, 3, 4, 5, 6, 7, 8 and DT, respectively. 1AA, AAA, DDAG and ECOC are 4 multi-class SVM methods. 1AA, AAA, DDAG are respectively identical with OAA, OAO and DAG in this paper. Experimental results of the 13 methods in [22] are also shown in Table 7.

Table 6 and Table 7 show that the introduced method is superior to the other 13 methods in [22] in

Table 5: Comparisons of RCBTA and the methods in [24, 25, 26] (%) (The blank cells in this table have no values in [26])

Class	RCBTA	PCS	kNN	DT	NB	HN	GCS	FN	GA	ERR
1	68.90	74.3	55.78	55.10	53.74	55.10				
2	52.21	35.7	59.18	51.02	57.82	55.78				
3	51.64	47.1	60.96	56.16	56.16	58.22				
4	80.98	85.3	65.75	58.22	58.22	55.48				
5	33.33	15.7	48.63	50.00	45.21	47.95				
6	77.07	63.6	62.33	57.53	54.11	53.42				
7	54.29	45.7	68.49	65.75	60.27	67.81				
8	3.33	10.0	58.90	57.53	61.64	56.16				
9	30.00	0	56.85	56.85	56.16	55.48				
10	60.00	60.0	58.22	57.53	59.59	57.53				
Ave.	59.37	54.9	59.51	56.57	56.29	56.29	55	57	55	56

Table 6: Experimental results of 10 subsets in 10-fold cross-validation tests (%)

Subsets	CYT	NUC	MIT	ME3	ME2	ME1	EXC	VAC	POX	Ave.
1	79.6	64.3	32.1	73.7	50.0	62.5	50.0	0.0	0.0	61.2
2	63.0	58.1	54.2	81.3	20.0	100	66.7	0.0	100	61.0
3	63.0	53.5	41.7	68.8	20.0	75.0	100	0.0	50.0	55.5
4	58.7	44.2	54.2	75.0	20.0	100	66.7	0.0	0.0	53.4
5	87.0	55.8	20.8	93.8	40.0	75.0	66.7	0.0	0.0	62.3
6	69.1	41.9	54.2	87.5	0.0	100	33.3	0.0	100	63.7
7	69.6	62.8	66.7	75.0	20.0	50.0	33.3	0.0	50.0	63.0
8	69.6	53.5	37.5	93.8	60.0	75.0	100	0.0	50.0	61.0
9	76.1	46.5	75.0	87.5	60.0	100	66.7	0.0	50.0	66.4
10	71.7	53.5	62.5	81.3	40.0	75.0	33.3	0.0	0.0	61.6

Table 7: Comparisons of RCBTA and methods in [22] (%)

	CYT	NUC	MIT	ME3	ME2	ME1	EXC	VAC	POX	Ave.
RCBTA	72.7	53.4	49.6	81.6	33.3	79.5	60.0	0.0	40.0	60.9
IAA	67.7	50.8	57.7	84.8	33.3	81.0	50.0	3.3	45.0	60.1
AAA	72.2	48.2	55.3	78.1	41.3	71.5	61.7	0.0	50.0	59.9
DDAG	70.2	49.9	54.5	78.6	41.3	71.5	61.7	3.3	50.0	59.8
ECOC	69.8	50.1	57.7	81.1	39.3	73.5	61.7	6.7	45.0	60.5
1	71.3	51.3	48.3	75.6	33.7	69.5	47.5	0.0	45.0	58.3
2	56.0	59.4	52.0	78.7	41.3	73.5	61.7	0.0	45.0	57.6
3	70.0	50.4	50.4	78.7	43.3	75.5	52.5	6.7	45.0	59.2
4	74.7	49.4	45.8	71.9	26.0	71.5	65.0	3.3	45.0	58.3
5	71.7	52.0	51.1	71.3	39.0	51.0	40.0	0.0	50.0	58.3
6	69.6	48.0	45.8	83.0	12.0	73.5	65.0	0.0	45.0	57.1
7	61.2	56.4	50.7	79.3	25.0	80.0	52.5	3.3	50.0	57.8
8	71.3	51.3	50.3	75.0	33.7	69.5	55.0	0.0	45.0	58.8
DT	57.5	50.1	52.4	83.6	41.0	74.5	60.0	3.3	30.0	55.8

correct recognition rate. Table 5 and Table 7 indicate that resemblance coefficient algorithm is a valid method to construct binary tree architecture for combining multiple binary-class SVMs to solve multi-class classification problems.

3.2 Application Example

Radar emitter signal recognition is one of the key proce-

sure of signal processing in electronic intelligence systems, electronic support measure systems and radar warning receiver systems in modern electronic warfare [27-30]. In our prior work [28], two entropy features were extracted from 10 advanced radar emitter signals. The 10 signals are represented with x_1, x_2, \dots, x_{10} , respectively. Every radar emitter signal has 500 samples. Thus, there are 5000 samples of 10 signals totally.

Using the samples, RCA is applied to construct binary

tree architecture, which is shown in Fig.2. The encoded tables in the procedure of construction are shown in Table 1 and Table 2. In the experiment, 5-fold cross-validation method is employed to test trained SVM classifiers with the constructed binary tree architecture. That is to say, the total samples are divided into 5 parts of which 4 parts are used as training samples and the remaining one part as the testing samples. Gaussian function is chosen as kernel function of SVM. The parameter σ of Gaussian kernel function varies from 0.001 to 10, i.e. $\sigma = [0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10]$. The constant C varies from 1 to 10^6 , i.e. $C = [10^0, 10^1, 10^2, 10^3, 10^4, 10^5, 10^6]$. Thus, there are 63 combinations of C and σ and consequently 63 tests need fulfill in this experiment. Table 7 shows the lowest error recognition rates (ERR) of the tests of the introduced method.

To compare the classification performances of the proposed multi-class SVMs with those of several popular multi-class SVMs including OAA [14, 16, 17], OAO [14,19], DAG [20], BUBT [5]. Performance evaluation includes ERR and testing time. Each of the 4 methods fulfills 63 tests, of which the lowest error recognition rate is also given in Table 8.

Table 8 shows that RCBTA achieves the lowest ERR and the shortest testing time among 5 multi-class classification SVM classifiers. The experimental results verify again that the proposed method is valid.

Table 8: Comparisons of RCBTA, OAA, OAO, DAG and BUBT (%)

Signals	RCBTA	OAA	OAO	DAG	BUBT
x_1	3.75	26.75	29.25	23.63	25.38
x_2	46.38	49.75	37.25	40.13	43.88
x_3	0.00	3.50	0.13	1.88	0.25
x_4	0.63	1.38	0.63	0.25	0.25
x_5	1.13	6.50	5.13	2.63	4.00
x_6	0.00	0.00	0.00	0.00	0.00
x_7	0.00	3.25	0.00	0.00	0.00
x_8	13.38	8.25	10.13	11.38	5.25
x_9	0.00	0.38	0.00	0.00	0.25
x_{10}	1.63	4.38	1.38	2.13	0.88
ERR	6.69	10.41	8.38	8.20	8.01
Testing time(Sec.)	92.84	525.41	496.14	103.30	160.75

Although BTA has fastest decision-making speed among several multi-class classification methods using binary-class SVMs, the problem of constructing good binary tree is always ongoing research issue. The examples of prediction of yeast protein localization sites and radar emitter signal recognition indicate that RCA proposed in this paper is a good way to solve this difficult problem. Moreover, Table 5, Table 7 and Table 8 show that if the architecture of binary tree is constructed appropriately,

BTA can achieve better classification performances than other multi-class SVM methods. Thus, BTA has fast decision-making speed and good classification capability simultaneously.

4. Concluding Remarks

This paper proposes a novel preprocessing method for the multi-class classification SVMs with binary tree architecture. The main point of the introduced method is that RCA is used to convert multi-class classification problems into binary-class classification problems in each of layers in binary tree architecture. According to feature distribution, RCA groups preferentially the classes with serious confusion and then groups the classes with a little or no confusion. Therefore, the multi-class SVM classifiers with binary tree architecture constructed by using RCA have good classification capability. Also, the SVM classifier with binary tree architecture has faster decision speed than the existing popular multi-class SVM classifiers including OAA, OAO, DAG and BUBT. So the multi-class classification method in this paper takes advantage of both the efficient computation of BTA and the high classification accuracy of SVMs. Two examples of yeast protein localization site prediction and radar emitter signal recognition verify that the introduced method has high efficiency of decision-making and good classification capability. This paper uses only two examples to compare the performances of RCBTA with other methods. Further study of this paper is that more examples are applied to test the performances of the introduced method. Moreover, RCA is mainly used to process continuous feature values. How to deal with discrete feature values for constructing binary tree architecture is also a problem to study further

References

- [1] V. N. Vapnik, Statistical learning theory, John Wiley and Sons, New York, 1998.
- [2] L. P. Wang (Ed.), Support vector machines: theory and applications, Springer-Verlag, Berlin Heidelberg, New York, 2005.
- [3] V. N. Vapnik, "An overview of statistical learning theory," IEEE Transactions on Neural Networks, vol.10, pp. 988-999, 1999.
- [4] D. Meyer, F. Leisch and K. Hornik, "The support vector machine under test," Neurocomputing, vol.55, pp.169-186, 2003.
- [5] G. D. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," IEEE Transactions on Neural Networks, vol.14, pp.209-215, 2003.
- [6] L. Kahsay, F. Schwenker and G. Palm, "Comparison of multiclass svm decomposition schemes for visual object recognition," W. Kropatsch, R. Sablatnig and A. Hanbury (Eds.), Lecture Notes in Computer Science, vol. 3663, pp. 334-341, 2005.

- [7] D. Casasent and Y. C. Wang, "A hierarchical classifier using new support vector machines for automatic target recognition," *Neural Networks*, vol.18, pp.541-548, 2005.
- [8] G. X. Zhang, Z. X. Cao and Y. J. Gu, et al, "Radar emitter signals recognition based on feature selection and support vector machines," *Lecture Notes in Computer Science*, vol.3644, pp.707-716, 2005.
- [9] J. X. Dong, A. Krzyzak and C. Y. Suen, "An improved handwritten Chinese character recognition system using support vector machine," *Pattern Recognition Letters*, vol.26, pp.1849-1856, 2005.
- [10] M. A. Mohandes, T. O. Halawani, S. Rehman and A. A. Hussain, "Support vector machines for wind speed prediction," *Renewable Energy*, vol.24, pp.939-947, 2004.
- [11] K. J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol.55, pp.307-319, 2003.
- [12] B. Boser, I. Guyon and V. Vapnik, "A training algorithm for optimal margin classifiers," D. Haussler, (Eds.) *Proceedings of the 5th Annual ACM Workshop on COLT*, pp.144-152, 1992.
- [13] H. S. Lei and V. Govindaraju, "Half-against-half multi-class support vector machines," N. C. Oza, et al, (Eds.), *Lecture Notes in Computer Science*, vol.3541, pp.156-164, 2005.
- [14] G. X. Zhang, "Support vector machines with Huffman tree architecture for multi-class classification," *Lecture Notes in Computer Science*, vol.3773, pp.24-33, 2005.
- [15] S. M. Cheong, S. H. Oh and S. Y. Lee, "Support vector machines with binary tree architecture for multi-class classification," *Neural Information Processing-Letters and Reviews*, vol.2, pp.47-51, 2004.
- [16] F. Schwenker and G. Palm, "Tree-structured support vector machines for multi-class pattern recognition," J. Kittler and F. Roli, (Eds.), *Lecture Notes in Computer Science*, vol. 2096, pp.409-417, 2001.
- [17] R. Rifkin and A. Klautau, "In defence of one-vs-all classification," *Journal of Machine Learning Research*, vol.5, pp.101-141, 2004.
- [18] G. X. Zhang, Z. X. Cao and Y. J. Gu, "A hybrid classifier based on rough set theory and support vector machines," *Lecture Notes in Artificial Intelligence*, vol.3613, pp.1287-1296, 2005.
- [19] U. Kreβel, "Pairwise classification and support vector machines," B. Scholkopf, C. J. C. Burges and A. J. Smola (Eds.), *Advances in Kernel Methods-Support Vector Learning*, MIT Press, pp.185-208, 1999.
- [20] J. C. Platt, N. Cristianini and J. Shawe-Taylor, "Large margin DAG's for multiclass classification," *Advances in Neural Information Processing Systems*, Cambridge, MA: MIT Press, vol.12, pp.547-553, 2000.
- [21] A. C. Lorena and A. C. P. L. F. Carvalho, "Minimum spanning trees in hierarchical multiclass support vector machines generation," M. Ali, and F. Esposito, (Eds.), *Lecture Notes in Artificial Intelligence*, vol.3533, pp.422-431, 2005.
- [22] A. C. Lorena and A. C. P. L. F. Carvalho, "Protein cellular localization with multiclass support vector machines and decision trees," *Lecture Notes in Bioinformatics*, vol.3594, pp.42-53, 2005.
- [23] Z. Q. Bian and X. G. Zhang, *Pattern recognition (2nd Edition)*, Tsinghua University Press, Beijing, 2000.
- [24] P. Horton and K. Nakai, "Better prediction of protein cellular localization sites with k-nearest neighbor classifiers," *Proceedings of ISMB*, vol.5, pp.147-152, 1997.
- [25] P. Horton and K. Nakai, "A probabilistic classification system for prediction the cellular localization sites of proteins," *Intelligent Systems in Molecular Biology*, pp.109-115, 1996.
- [26] P. Cairns, C. Huyck, I. Mitchell and W. X. Wu, "A comparison of categorisation algorithms for predicting the localization sites of proteins," *Proceedings of 12th International Workshop on Database and Expert Systems Applications*, pp.296-300, 2001.
- [27] G. X. Zhang, L. Z. Hu and W. D. Jin, "Resemblance coefficient based intrapulse feature extraction approach for radar emitter signals," *Chinese Journal of Electronics*, vol.14, pp.337-341, 2005.
- [28] G. X. Zhang, L. Z. Hu and W. D. Jin, "Radar emitter signal recognition based on entropy features," *Chinese Journal of Radio Science*, vol.20, pp.440-445, 2005.
- [29] J. X. Gong, and Y. P. Gu, "Electromagnetic environment of future warfare and a survey of electronic warfare technology," *Proceedings of Intelligence Research of Electronic Warfare*, pp.20-36, 2001.
- [30] E. Granger, M. A. Rubin and S. Grossberg, et al, "Radar ESM with a what-and where fusion neural network," *Proceedings of the 2001 IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing XI*, North Falmouth, pp.539-548, 2001.
- [31] C. C. Chang and C. J. Lin, LIBSVM: a library <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>.



Gexiang Zhang received the B.S. and M.S degrees in control theory and engineering in 1997 and 2002, respectively, from Southwest University of Science and Technology. He received Ph.D degree in signal and information processing in Southwest Jiaotong University, Chengdu, China. He is a professor at Southwest Jiaotong University. His research interests are in the areas of machine learning, pattern recognition, signal processing and evolutionary computation. He is a member of IEEE, IEEE CS, ISRR, CAAI, and CCF.



Weidong Jin received the B.S., M.S degrees and PhD in 1982, 1990 and 1998, respectively, from Southwest Jiaotong University, Chengdu, P.R.China. He is a professor and the director of Department of Electronics Information in Southwest Jiaotong University. He is a fellow of System Simulation Society of China and Electronics Education Society of China. His research interests are in the areas of intelligent information

processing, optimization theory and method and system simulation.