International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 22, No. 1 (2008) 109–120

FAST FUZZY MULTICATEGORY SVM BASED ON SUPPORT VECTOR DOMAIN DESCRIPTION

XI-ZHAO WANG*, SHU-XIA LU† and JUN-HAI ZHAI

Faculty of Mathematics and Computer Science
Hebei University, Baoding, Hebei, China
**wangxz@hbu.cn*
†*mclsx@hbu.cn*

This paper proposes a fast fuzzy classifier of multicategory support vector machines (FMSVM) based on support vector domain description (SVDD). The main idea is that the proposed FMSVM is obtained by directly considering all data in one optimization formulation, using a fuzzy membership to each input point. The fuzzy membership is determined by support vector domain description (SVDD). For making support vector machine (SVM) more practical, we use an implement of the modified sequential minimal optimization (SMO) that can quickly solve SVM quadratic programming (QP) problems without any extra matrix storage or the use of numerical QP optimization steps at all. Compared with the existing SVMs, the newly proposed FMSVM that uses the L_2-norm in the objective function shows improvement with regards to accuracy of classification and reduction of the effects of noises and outliers. The experiment also shows the efficiency of the modified SMO for expediting the training of SVM.

Keywords: Fuzzy membership; support vector machines; multicategory classification; modified sequential minimal optimization; support vector domain description.

1. Introduction

Support vector machines (SVMs) proposed by Vapnik[18] are trained by solving a quadratic optimization problem. SVMs were originally designed for binary classification. Since many real-world applications are problems of multicategory classification, how to effectively extend two-class SVMs to a multicategory SVM is still an ongoing research issue. Currently, there are two types of approaches for multicategory SVM. One is by constructing and combining several binary classifiers: *one-against-all* algorithm transforms a k-class problem into k two-class problems where one class is separated from the remaining ones; *one-against-one* (pair-wise) algorithm converts the k-class problem into $k(k-1)/2$ two-class problems where pairwise optimal hyperplanes for each pair of classes are constructed and max-voting strategy is used to predict their classes; and DAGSVM is the same as the one-against-one method in the training phase, however, in the testing phase, it uses a rooted binary directed acyclic graph which has $k(k-1)/2$ internal nodes and k leaves;[7,13,18] the other is by directly considering all data in one optimization formulation based on *multiclass support vector machines* (MSVM).[4,20] We propose a new classifier of fuzzy multicategory Support Vector Machines, which extends the method of multiclass support vector machines (MSVM) by using a fuzzy membership to each input point. Our formulation that uses the L_2-norm in the objective function has some additional advantages such as strong convexity. It aims to enhance the generation capability of the FMSVM. Fuzzy Support Vector Machines (FSVMs) can be applied to reduce the effects of noises and outliers.[8,9,17] The fuzzy membership is determined by the method of support vector domain description (SVDD). The fuzzy membership is defined according to the position of samples in sphere space.[11,15,16]

For speeding up the training of SVM, we use an implement of the modified SMO that can quickly solve the SVM (support vector machine) QP (quadratic programming) problems without any extra matrix storage or the use of numerical QP optimization steps at all.[6,14] SMO decomposes the overall QP problem into QP sub-problems by using Osuna's theorem to ensure convergence. For making SVM more practical, special algorithms are developed, such as Vapnik's chunking, Osuna's decomposition, Platt's SMO[12] and Joachims's $\mathrm{SVM}^{\mathrm{light}}$.[5] They make the training of SVM possible by breaking the large QP problem into a series of smaller QP problems and optimizing only a subset of training data patterns at each step. These approaches are categorized as the working set methods. Later, Keerthi *et al.*[6,14] ascertained inefficiency associated with Platt's SMO and suggested two modified versions of SMO that are much more efficient than Platt's original SMO. The second modification is particularly good and used in popular SVM packages such as a library for support vector machines (LIBSVM).[3] We use the second modified SMO algorithm.

Compared with the existing SVM algorithms, the newly proposed FMSVM show improvement in aspects of classification accuracy and reduction of the effects of noises and outliers. Numerical simulations show the feasibility and effectiveness of this algorithm.

This paper consists of the following sections. Section 2 gives an overview of the modified SMO. Section 3 proposes the new classifier of the fuzzy multicategory support vector machines (FMSVM). Section 4 provides some simulations to demonstrate the feasibility and effectiveness of the FMSVM. And the last section concludes this paper.

2. The Modified SMO

Given l data points $(x_1, y_1), \ldots, (x_l, y_l)$, where $x_i \in R^N$ and $y_i \in \{-1, 1\}$. Training an SVM in classification can be formulated as

$$\begin{aligned} &\min_{w,b,\xi} \ \psi(w,b,\xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i \\ &\text{s.t. } \ y_i((w \cdot x_i) - b) + \xi_i \geq 1, \quad \xi_i \geq 0, \quad i = 1, \ldots, l. \end{aligned} \tag{1}$$

The Wolfe dual problem of (1) is

$$\begin{aligned} &\max \ \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i\alpha_j y_i y_j K(x_i, x_j) \\ &\text{s.t.} \ \sum_{i=1}^{l} \alpha_i y_i = 0, \quad 0 \le \alpha_i \le C, \quad i = 1, \ldots, l \end{aligned} \tag{2}$$

where $K(x_i, x_j)$ is the kernel function. α_i is the Lagrange multiplier to be optimized. C is the penalty parameter. After solving the problem (2), the decision function

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{l} \alpha_i y_i K(x_i, x) - b\right).$$

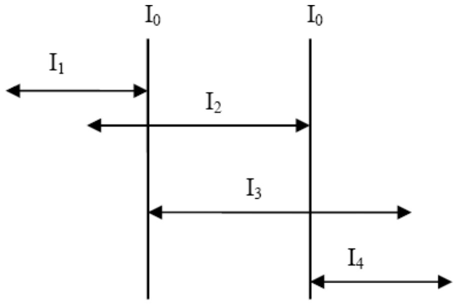


Fig. 1. The index of training data patterns.

Let

$$I_0 \equiv \{i : 0 < \alpha_i < C\} \quad I_1 \equiv \{i : y_i = +1, \alpha_i = 0\} \quad I_2 \equiv \{i : y_i = -1, \alpha_i = C\}$$

$$I_3 \equiv \{i : y_i = +1, \alpha_i = C\} \quad I_4 \equiv \{i : y_i = -1, \alpha_i = 0\}$$

$$F_i \equiv \sum_{j=1}^{l} \alpha_j y_j K(x_j, x_i) - y_i, \quad b_{\text{up}} = \min\{F_i : i \in I_0 \cup I_1 \cup I_2\},$$

$$b_{\text{low}} = \max\{F_i : i \in I_0 \cup I_3 \cup I_4\}$$

$$I_{\text{up}} = \arg\min_i F_i, \quad I_{\text{low}} = \arg\max_i F_i, \quad \tau = 10^{-6}.$$

The idea of the modified SMO is to optimize the two α_i associated with b_{up} and b_{low} at each step. Their associated indexes are I_{up} and I_{low}

$$\alpha_2^{\text{new}} = \alpha_2^{\text{old}} - \frac{y_2(F_1^{\text{old}} - F_2^{\text{old}})}{\eta}, \quad \alpha_1^{\text{new}} = \alpha_1^{\text{old}} + s(\alpha_2^{\text{old}} - \alpha_2^{\text{new}}) \tag{3}$$

where $s = y_1 y_2$, $\eta = 2K(x_1 \cdot x_2) - K(x_1 \cdot x_1) - K(x_2 \cdot x_2)$, $0 \le \alpha_1^{\text{new}}, \alpha_2^{\text{new}} \le C$.

After optimizing α_1 and α_2, F_i denoting the error on the ith training data pattern, is updated according to the following:

$$F_i^{\text{new}} = F_i^{\text{old}} + (\alpha_1^{\text{new}} - \alpha_1^{\text{old}}) y_1 K(x_1, x_i) + (\alpha_2^{\text{new}} - \alpha_2^{\text{old}}) y_2 K(x_2, x_i). \tag{4}$$

Based on the updated values of F_i, b_{up} and b_{low}, the associated index I_{up} and I_{low} are updated again according to their definitions. The updated values are then used to choose another two new α_i to optimize at the next step.

The value of (2), represented by Dual, is updated at each step

$$\text{Dual}^{\text{new}} = \text{Dual}^{\text{old}} - \frac{\alpha_1^{\text{new}} - \alpha_1^{\text{old}}}{y_1}(F_1^{\text{old}} - F_2^{\text{old}}) + \frac{1}{2}\eta \left(\frac{\alpha_1^{\text{new}} - \alpha_1^{\text{old}}}{y_1}\right)^2. \tag{5}$$

DualityGap, representing the difference between the primal and the dual objective functions in SVM, is calculated by

$$\text{DualityGap} = \sum_{i=0}^{l} \alpha_i y_i F_i + \sum_{i=0}^{l} \varepsilon_i \tag{6}$$

where

$$\varepsilon_i = \begin{cases} C \max(0, b - F_i), & \text{if } y_i = 1 \\ C \max(0, -b + F_i), & \text{if } y_i = -1. \end{cases}$$

Dual and DualityGap are used for checking the convergence of the program. A simple description of the modified SMO in the sequential form can be summarized as

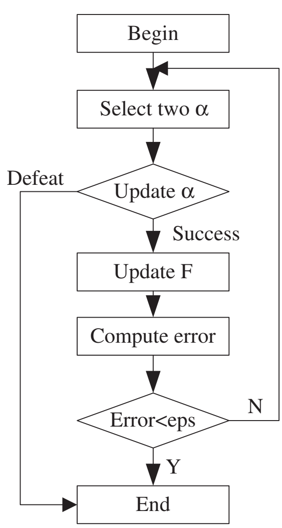


Fig. 2. The heuristics for picking two α_i's for optimization in the modified SMO.

***Modified SMO Algorithm*:**

Initialize $\alpha_i = 0$, $F_i = -y_i$, Dual $= 0$, $i = 1, \ldots, l$. Calculate b_{up}, I_{up}, b_{low}, I_{low} and DualityGap until DualityGap $\leq \tau|\text{Dual}|$

(1) Optimize $\alpha_{I_{\text{up}}}, \alpha_{I_{\text{low}}}$.
(2) Update $F_i, i = 1, \ldots, l$ based on Eq. (4).
(3) Calculate b_{up}, I_{up}, b_{low}, I_{low}, DualityGap and update Dual.

Repeat.

3. A New Fuzzy Multicategory Support Vector Machines

3.1. *The fuzzy membership*

A method defining the affinity among samples is considered here by using a sphere with minimum volume while containing all (or most of) the samples. When one or a few very remote objects are in the training set, we may obtain a very large sphere that will not represent the data well. Therefore, we allow for some data points outside the sphere and introduce slack variable. Suppose we are given a sequence of training points $\{x_i, i = 1, \ldots, n\}$. A sphere with minimum volume is obtained by solving optimal hyperplane problem

$$\begin{aligned} &\min F(R, a, \xi_i) = R^2 + D\sum_{i=1}^{n}\xi_i \\ &\text{s.t. } \|x_i - a\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, n \end{aligned} \tag{7}$$

where ξ_i is slack variable; R is the radius of the minimum sphere; a is the center of the minimum sphere; $D > 0$ is the penalty parameter.

To solve this optimization we construct the Lagrangian

$$L(R, a, \beta, \gamma, \xi) = R^2 + D\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\beta_i\{R^2 + \xi_i - \|x_i - a\|^2\} - \sum_{i=1}^{n}\gamma_i\xi_i \tag{8}$$

where β_i, γ_i are Lagrangian non-negative multipliers.

Finding the saddle point of $L(R, a, \beta, \gamma, \xi)$, which has to be maximized with respect to β, γ and minimized with respect to R, a, ξ, and setting the partial derivatives with respect to R, a, ξ to zero, we obtain $\sum_{i=1}^{n}\beta_i = 1$, $a = \sum_{i=1}^{n}\beta_i x_i$, $D - \beta_i - \gamma_i = 0$, $i = 1, \ldots, n$.

Applying these equations into the Lagrangian (8), the primal problem (7) can be transformed into the following equivalent dual problem

$$\begin{aligned} &\max\ Q(\beta) = \sum_{i=1}^{n}\beta_i(x_i \cdot x_i) - \sum_{i,j=1}^{n}\beta_i\beta_j(x_i \cdot x_j) \\ &\text{s.t. } \sum_{i=1}^{n}\beta_i = 1, \quad 0 \leq \beta_i \leq D, \quad i = 1, \ldots, n. \end{aligned} \tag{9}$$

The radius of the minimum sphere $R = \|x_i - a\|$, where the point x_i is a support vector, determined by β_i which is randomly selected in $(0, D)$. The objects that have nonzero coefficients β_i are called support objects. Objects with $\beta_i = D$ are outside the sphere. These support vectors are considered to be outliers or noises. Because of the constraint $\sum_{i=1}^{n} \beta_i = 1$ and $\beta_i \geq 0$, we only choose the parameter D as $\frac{1}{n} \leq D \leq 1$.

The fuzzy membership $s(x)$ is defined according to the position of samples in sphere space

$$s(x_i) = \begin{cases} 0.6 \times \left(\dfrac{1 - \frac{d(x_i)}{R}}{1 + \frac{d(x_i)}{R}} \right) + 0.4, & d(x_i) \leq R \\ 0.4 \times \left(\dfrac{1}{1 + (d(x_i) - R)} \right), & d(x_i) > R \end{cases} \tag{10}$$

where $d(x_i) = \|x_i - a\|,\ i = 1, \ldots, n$.

3.2. *Reformulate FMSVM*

Suppose we are given a set S of labeled training points with associated fuzzy membership $(x_1, y_1, s_1), \ldots, (x_l, y_l, s_l)$, each training point $x_i \in R^N$ is given a label $y_i \in \{1, 2, \ldots, k\}$ and a fuzzy membership $\sigma \leq s_i \leq 1$ $(i = 1, \ldots, l)$, where σ is a sufficiently small positive number. Let $z = \varphi(x)$ denote the corresponding feature space vector with a mapping φ from R^N to a feature space Z. The fuzzy membership s_i is the attitude of the corresponding point x_i toward one class and the parameter ξ_i is a measure of error in the SVM, the term $s_i\xi_i$ is a measure of error with different weight.

The optimal hyperplane problem is regarded as the solution to

$$\begin{aligned} \min_{(w_m, b_m, \xi^m)} \ & \frac{1}{2} \sum_{m=1}^{k} ((w_m \cdot w_m) + b_m^2) + \frac{C}{2} \sum_{i=1}^{l} \sum_{m \neq y_i} s_i^m (\xi_i^m)^2 \\ \text{s.t.} \ & ((w_{y_i} \cdot x_i) + b_{y_i}) \geq (w_m \cdot x_i) + b_m + 2 - \xi_i^m, \\ & i = 1, \ldots, l; \quad m, y_i \in \{1, \ldots, k\}; \quad m \neq y_i, \end{aligned} \tag{11}$$

where C is a constant. A smaller s_i reduces the effect of the parameter ξ_i in problem (11) such that the corresponding point x_i is treated as less important. Note that no explicit non-negative constraint is needed on ξ_i^m, because if any component ξ_i^m is negative, the objective function can be decreased by setting that $\xi_i^m = 0$ while still satisfying the corresponding inequality constraint. Note further that the L_2-norm of the error vector ξ_i^m is minimized instead of the L_1-norm, and the margin between the bounding planes is maximized with respect to both the orientation w_m and the relative location of samples to the origin b_m. This formulation has some additional advantages such as strong convexity of the objective function.

To solve this optimization we construct the Lagrangian

$$L(w,b,\xi,\alpha) = \frac{1}{2}\sum_{m=1}^{k}((w_m \cdot w_m) + b_m^2) + \frac{C}{2}\sum_{i=1}^{l}\sum_{m=1}^{k} s_i^m (\xi_i^m)^2 \\ - \sum_{i=1}^{l}\sum_{m=1}^{k} \alpha_i^m [((w_{y_i} - w_m)\cdot x_i) + b_{y_i} - b_m - 2 + \xi_i^m] \qquad (12)$$

where α_i^m is the non-negative Lagrangian multipliers, with

$$\alpha_i^{y_i} = 0, \quad \xi_i^{y_i} = 2, \quad s_i^{y_i} = 1, \qquad i = 1, \dots, l$$

and constraints $\alpha_i^m \geq 0$, $0 < \sigma \leq s_i^m \leq 1$, $i = 1, \dots, l$, $m \in \{1, \dots, k\} \backslash y_i$.

Finding the saddle point of $L(w, b, \xi, \alpha)$ which has to be maximized with respect to α and minimized with respect to w, b, ξ; and introducing the notation

$$c_i^n = \begin{cases} 1 & \text{if } y_i = n \\ 0 & \text{if } y_i \neq n \end{cases}, \quad A_i = \sum_{m=1}^{k} \alpha_i^m. \qquad (13)$$

Computing the partial derivatives with respect to w_n, b_n and ξ_j^n, in the saddle point the solution should satisfy the conditions:

$$\frac{\partial L(w,b,\xi,\alpha)}{\partial w_n} = 0 \Rightarrow w_n = \sum_{i=1}^{l}(c_i^n A_i - \alpha_i^n) x_i$$
$$\frac{\partial L(w,b,\xi,\alpha)}{\partial b_n} = 0 \Rightarrow b_n = \sum_{i=1}^{l} c_i^n A_i - \sum_{i=1}^{l} \alpha_i^n \qquad (14)$$
$$\frac{\partial L(w,b,\xi,\alpha)}{\partial \xi_j^n} = 0 \Rightarrow C s_j^n \xi_j^n = \alpha_j^n \quad \text{and} \quad 0 \leq \alpha_j^n \leq C_1 s_j^n$$

where $C_1 = \max_{i,n}(C\xi_i^n)$.

Applying these conditions (14) into the Lagrangian (12), the problem (11) can be transformed into the following equivalent dual problem

$$\max\ 2\sum_{i,m}\alpha_i^m + \sum_{i,j,m}\left(-\frac{1}{2}c_j^{y_i} A_i A_j + c_i^m A_i \alpha_j^m - \frac{1}{2}\left(1 + \frac{1}{C s_i^m}\right)\alpha_i^m \alpha_j^m\right) \\ + \sum_{i,j,m}\left(-\frac{1}{2}c_j^{y_i} A_i A_j - \frac{1}{2}\alpha_i^m \alpha_j^m + \alpha_i^m \alpha_j^{y_i}\right)(x_i \cdot x_j) \\ \text{s.t. } b_m = \sum_{i=1}^{l} c_i^m A_i - \sum_{i=1}^{l}\alpha_i^m, \quad 0 \leq \alpha_i^m \leq C_1 s_i^m, \quad \alpha_i^{y_i} = 0, \\ i = 1, \dots, l; \quad m, y_i \in \{1, \dots, k\}, \quad m \neq y_i \qquad (15)$$

where $C_1 = \max_{i,m}(C\xi_i^m)$.

This gives the decision function

$$f(x,\alpha) = \arg\max_n \left\{ \sum_{i=1}^{l} (c_i^n A_i - \alpha_i^n)[(x_i \cdot x) + 1] \right\}. \tag{16}$$

As usual, the inner products $(x_i \cdot x_j)$ can be replaced with the convolution of the inner products $K(x_i \cdot x_j)$ and the Kuhn–Tucker conditions are defined as

$$\begin{aligned} &\alpha_i^m[((w_{y_i} - w_m) \cdot x_i) + (b_{y_i} - b_m) - 2 + \xi_i^m] = 0, \\ &i = 1,\ldots,l; \quad y_i, m \in \{1,\ldots,k\}, \quad m \neq y_i. \end{aligned} \tag{17}$$

The point x_i with the corresponding $\alpha_i > 0$ is called a support vector. There are also two types of support vectors. One with corresponding $0 < \alpha_i < C_1 s_i$ lies on the margin of the hyperplane while the other with corresponding $\alpha_i = C_1 s_i$ is misclassified. An important difference between SVM and FSVM is that the points with the same value of α_i may indicate a different type of support vectors in FSVM due to the factor s_i.

The parameter C in SVM controls the tradeoff between the maximization of margin and the amount of misclassifications. In the FMSVM, we can set C to be a sufficiently large value. It is the same as SVM in that the system will get a narrower margin and allow less misclassifications if we set all $s_i = 1$. With different values of s_i, we can control the tradeoff of the respective training point x_i in the system. A small value of s_i makes the corresponding point x_i less important in the training.

***FMSVM Algorithm*:**

(1) Given training data $(x_1, y_1), \ldots, (x_l, y_l)$, parameter D and kernel function, find an optimal solution β by solving the dual problem (9).
(2) Compute the center of sphere $a = \sum_{i=1}^{n} \beta_i x_i$ and the radius of sphere $R = \|x_i - a\|$.
(3) Compute the fuzzy membership $s(x_i)$ according to formula (10).
(4) Given training data $(x_1, y_1, s_1), \ldots (x_l, y_l, s_l)$, parameters C and kernel function etc. find an optimal solution α by solving the dual problem (15).
(5) Determine the decision function $f(x,\alpha) = \arg\max_n[\sum_{i=1}^{l}(c_i^n A_i - \alpha_i^n)((x_i \cdot x) +1)]$.

4. Numerical Examples

To evaluate the performance of the proposed method FMSVM, several experiments have been conducted on the UCI machine learning respository[1] and Statlog database.[10] The description for the six selected data sets is given in Table 1.

Small-scale real-world data: *Iris* is from the UCI machine learning respository.[1]

Large-scale real-world data: *Segment* (image segmentation), *satimage* (classification in a satellite image), *letter* (classification of the English alphabet images), *DNA* (classification between exons and introns in the DNA sequence) and *vehicle* are from Statlog database.[10]

Table 1. The description of the date sets.

Date Set	#Training Data	#Testing Data	#Classes	#Attributes
Iris	150	0	3	4
Vehicle	846	0	4	18
Segment	2310	0	7	19
Satimage	4435	2000	6	36
Letter	15000	5000	26	16
DNA	2000	1186	3	180

The performance of the FMSVM is compared with three used methods: one-against-all SVM (1-a-a), one-against-one SVM (1-a-1) and multiclass support vector machines (MSVM). The criteria for evaluating the performance of these methods are their accurate rate of classification in Table 4, and model building time in Table 5.

We choose the best parameters by performing the model selection. That is, alternative models are constructed on the training data where the test data are assumed unknown and then the parameter set, with the best performance for the test set, is selected for constructing the final model. In the experiment of determining fuzzy membership, the detailed descriptions of parameter values are given in Table 2. In the experiment of four SVM methods, to reduce the search of parameter sets, here we train all datasets only with the RBF kernel $K(x_i, x_j) \equiv e^{\gamma\|x_i - x_j\|^2}$. For each problem, we estimate the generalized accuracy by using different kernel parameters γ and cost parameters C. For datasets *satimage*, *letter* and *DNA* where both training and testing sets are available, for each pair of (C, γ), the validation performance is measured by training 70% of the training set and testing the other 30% of the training set. Then, we train the whole training set by using the pair of (C, γ) that achieves the best validation rate and predict the test set. The resulting accuracy is presented in Table 3. For datasets *iris*, *vehicle* and *segment* where test data may not be available, we simply conduct a ten-fold cross-validation on the whole training data and report the average rate of the ten-fold cross-validation.

Table 2. Parameters in methods.

Kernel	D	Epsilon
Linear	0.3	1e-7

Table 3. Parameters (C, γ) (C: cost parameter, γ: kernel parameter).

Date Set	1-a-1	1-a-a	MSVM	FMSVM
Iris	$(2^{12}, 2^{-9})$	$(2^{9}, 2^{-3})$	$(2^{10}, 2^{-7})$	$(2^{10}, 2^{-7})$
Vehicle	$(2^{9}, 2^{-4})$	$(2^{11}, 2^{-4})$	$(2^{9}, 2^{-4})$	$(2^{9}, 2^{-4})$
Segment	$(2^{6}, 2^{0})$	$(2^{7}, 2^{0})$	$(2^{0}, 2^{3})$	$(2^{0}, 2^{3})$
Satimage	$(2^{4}, 2^{0})$	$(2^{2}, 2^{1})$	$(2^{2}, 2^{2})$	$(2^{2}, 2^{2})$
Letter	$(2^{4}, 2^{2})$	$(2^{2}, 2^{2})$	$(2^{3}, 2^{2})$	$(2^{3}, 2^{2})$
DNA	$(2^{3}, 2^{-6})$	$(2^{2}, 2^{-6})$	$(2^{1}, 2^{-6})$	$(2^{1}, 2^{-6})$

Table 4. Testing accuracy (%).

Date Set	1-a-1	1-a-a	MSVM	FMSVM
Iris	100	100	100	97.333
Vehicle	86.1702	86.5248	86.8794	86.882
Segment	96.8254	96.2963	95.9436	96.98
Satimage	90.35	90.9	90.7	91.35
Letter	97.02	96.92	96.84	97.97
DNA	94.5194	94.688	94.6037	95.76

Table 5. Times (seconds).

Date Set	1-a-1	1-a-a	MSVM	FMSVM
Iris	0.5	0.10	0.9	0.91
Vehicle	1.1	2.3	17	19
Segment	1.2	2.4	2.7	2.8
Satimage	2.2	4.1	7.3	7.7
Letter	33	1105	540	675
DNA	1.9	2.8	2.2	3.1

Experimental results are shown in Tables 4 and 5. For making support vector machine (SVM) more practical, we use an implement of the modified sequential minimal optimization (SMO) that can quickly solve SVM quadratic programming (QP) problems. The experimental results demonstrate that by using the proposed method (FMSVM) a better or comparable performance is achieved in terms of accuracy and efficiency for most of the multiclass data sets (except iris data). For the testing error accuracy, the FMSVM is the best (except iris data). The generalization ability of the FMSVM is superior to the other three methods while its training speed is a little slower. The main factor for training/testing change is that we use the fuzzy membership. For the training time, one-against-one SVM is the best. Compared with the methods of Refs. 19 and 20, the FMSVM by using the modified SMO is much faster.[19,20]

5. Conclusions

This paper proposes a fast fuzzy classifier of multicategory support vector machines (FMSVM) based on support vector domain description (SVDD). The main idea is that the proposed FMSVM extends the method of multiclass support vector machines (MSVM) by using a fuzzy membership to each input point. The fuzzy membership is determined by support vector domain description (SVDD). For making support vector machine (SVM) more practical, we use an implement of the modified sequential minimal optimization (SMO). Compared with the existing SVMs, the newly proposed FMSVM shows improvement in classification accuracy and reduction of the effects of noises and outliers. The experiment also shows the efficiency of the SMO for expenditing the training of SVM.

The algorithm is effective and efficient for datasets. Further research involves the parallelization of the algorithm for handling large datasets and selection of proper fuzzy membership function.

Acknowledgments

The author acknowledges the financial support from the National Natural Science Foundation of China (Project No. 60473045), Hebei Hi-tech project (04213533) and the plan of 100 excellent innovative scientists in Hebei Province.

References

1. C. Blake and C. J. Merz, UCI repository of machine learning databases, University of California, Irvine, CA (2002). http://www.ics.uci.edu/∼mlearn/MLRepository.html.
2. L. J. Cao, S. S. Keerthi, C.-J. Ong and J. Q. Zhang, U. Periyathamby, X. J. Fu and H. P. Lee, Parallel sequential minimal optimization for the training of support vector machines, *IEEE Trans. Neural Networks* **17**(4) (2006) 1039–1049.
3. C. C. Chang and C. J. Lin, LIBSVM: a library for support vector machines [online]. Available: http://www.csie.ntu.edu.tw/∼cjlin/libsvm/.
4. C. W. Hsu and C. J. Lin, A comparison of methods for multicategory support vector machines, *IEEE Trans. Neural Networks* **13**(2) (2002) 415–425.
5. T. Joachims, Making large-scale SVM learning practical, *Advances in Kernel Methods — Supprot Vector Learning*, eds. B. Schölkopf, C. J. C. Burges and A. J. Smola (MIT Press, Cambridge, MA, 1998), pp. 169–184.
6. S. S. Keerthi, S. K. Shevade, C. Bhattaacharyya and K. R. K. Murthy, Improvements to Platt's SMO algorithm for SVM classifier design, *Neural Comput.* **13** (2001) 637–649.
7. U. H.-G. Kreßel, Pairwise classification and support vector machines, *Advances in Kernel Methods: Support Vector Learning*, eds. B. Schölkopf, C. J. C. Burges and A. J. Smola (MIT Press, Cambridge, MA, 1999), pp. 255–268.
8. C. F. Lin and S. D. Wang, Fuzzy support vector machines, *IEEE Trans. Neural Networks* **13**(2) (2002) 464–471.
9. C. F. Lin and S. D. Wang, Training algorithm for fuzzy support vector machines with noisy data, *Patt. Recogn. Lett.* **25** (2004) 1647–1656.
10. D. Michie, D. J. Spiegelhalter and C. C. Taylor, *Machine Learning, Neural and Statistical Classification* (Prentice Hall, Englewood Cliffs, NJ, 1994), data available at ftp://ftp.ncc.up.pt/pub/statlog/.
11. A. Navia-Vázquez and E. Parrado-Hernández, Support vector machine interpretation, *Neurocomput. Lett.* **69** (2006) 1754–1759.
12. J. Platt, Fast training of support vector machines using sequential minimal optimization, *Advances in Kernel Methods — Support Vector Learning*, eds. B. Schölkopf, C. J. C. Surges and A. J. Smola (MIT Press, 1999), pp. 185–208.
13. J. C. Platt, N. Cristianini and J. Shawe-Taylor, *Large Margin DAG's for Multicategory Classification*, Advances in Neural Information Processing Systems, Vol. 12 (MIT Press, Cambridge, MA, 2000), pp. 547–553.
14. S. K. Shevade, S. S. Keerthi, C. Bhattacharyya and K. R. K. Murthy, Improvements to the SMO algorithm for SVM regression, *IEEE Trans. Neural Network* **11**(5) (2000) 1188–1193.
15. D. M. J. Tax and R. P. W. Duin, Support vector data description, *Mach. Learn.* **54** (2004) 45–66.

16. D. M. J. Tax and R. P. W. Duin, Support vector domain description, *Patt. Recogn. Lett.* **20** (1999) 1191–1199.
17. D. Tsujinishi and S. Abe, Fuzzy least squares support vector machines for multicategory problems, *Neural Networks* **16** (2003) 785–792.
18. V. N. Vapnik, *The Nature of Statistical Learning Theory* (Springer, NY, 2000).
19. X. Wang and S. Lu, Improved fuzzy multicategory support vector machines classifier, *Proc. 2006 Int. Conf. Machine Learning and Cybernetics* **6** (2006) 3585–3589.
20. J. Weston and C. Watkins, Multi-class support vector machines, *Proc. ESANN99*, ed. M. Verleysen (Brussels, Belgium, 1999).

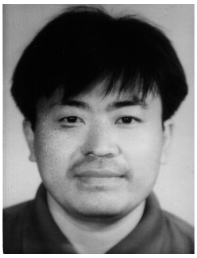
Xi-Zhao Wang is presently the Dean and Professor of the Faculty of Mathematics and Computer Science, Hebei University, China. He received his Ph.D. degree in computer science from Harbin Institute of Technology, Harbin, China, in 1998. From 1998 to 2001, he served as a Research Fellow in the Department of Computing, Hong Kong Polytechnic University, Hong Kong.

He has 150+ publications including 2 books, 5 book chapters, and 52 journal papers in *IEEE Transactions on PAMI/SMC/FS*, *Fuzzy Sets and Systems*, *Pattern Recognition*, etc. He has been the PI/co-PI for 16 research projects supported partially by the National Natural Science Foundation of China and the Research Grant Committee of Hong Kong Government. He is an IEEE Senior Member, the Chair of IEEE SMC Technical Committee on Computational Intelligence, an Associate Editor of *IEEE Transactions on SMC*, Part B, an Associate Editor of *Pattern Recognition and Artificial Intelligence*, an editorial board member of *Information Sciences*. Prof. Wang was the recipient of the IEEE-SMCS Outstanding Contribution Award in 2004 and the recipient of IEEE SMCS Best Associate Editor Award in 2006.

His main research interests include learning from examples with fuzzy representation, fuzzy measures and integrals, neuro-fuzzy systems and genetic algorithms, feature extraction, multi-classifier fusion and applications of machine learning.

Shu-Xia Lu received the B.Sc. and M.Sc. degrees in mathematics from Hebei University, Baoding, China, in 1988 and 1991, respectively, and the Ph.D. degree from Hebei University, Baoding, China, in 2007. Since 1991, she worked as a Lecturer and then an Associate Professor in the Faculty of Mathematics and Computer Science, Hebei University.

Her main research interests include clustering algorithm, SVMs, feature extraction, and applications of machine learning.



Jun-Hai Zhai received his B.S. degree in mathematics and M.S. degree in computing mathematics from Lanzhou University in 1988 and 2000, respectively; He is currently a doctoral candidate in the School of Physics Science and Technology at Hebei University of China.

His primary research interests include machine learning, wavelet analysis, pattern recognition and intelligent computing.