Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

A multi-class large margin classifier*

Liang TANG†, Qi XUAN, Rong XIONG†‡, Tie-jun WU, Jian CHU
(*Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China*)
†E-mail: tang@iipc.zju.edu.cn; rxiong@iipc.zju.edu.cn

Received Feb. 20, 2008; Revision accepted June 19, 2008; Crosschecked Dec. 26, 2008

Abstract: Currently there are two approaches for a multi-class support vector classifier (SVC). One is to construct and combine several binary classifiers while the other is to directly consider all classes of data in one optimization formulation. For a *K*-class problem (*K*>2), the first approach has to construct at least *K* classifiers, and the second approach has to solve a much larger optimization problem proportional to *K* by the algorithms developed so far. In this paper, following the second approach, we present a novel multi-class large margin classifier (MLMC). This new machine can solve *K*-class problems in one optimization formulation without increasing the size of the quadratic programming (QP) problem proportional to *K*. This property allows us to construct just one classifier with as few variables in the QP problem as possible to classify multi-class data, and we can gain the advantage of speed from it especially when *K* is large. Our experiments indicate that MLMC almost works as well as (sometimes better than) many other multi-class SVCs for some benchmark data classification problems, and obtains a reasonable performance in face recognition application on the AR face database.

Key words: Multi-classification, Support vector machine (SVM), Quadratic programming (QP) problem, Large margin
doi:10.1631/jzus.A0820122 **Document code:** A **CLC number:** TN911.7

INTRODUCTION

Support vector classifier (SVC) is specially developed for binary-class problems and achieves great success in practice (Vapnik, 2000). There are also many multi-class problems in practice, so effectively extending binary SVC for multi-class problems is of much importance.

The standard method for *K*-class SVC is training *K* SVCs: the examples in the *i*th class with a positive label, and all the other examples with negative labels. SVC trained in this way is referred to as one-versus-rest (1-v-r) SVC (Cortes and Vapnik, 1995; Vapnik, 2000). Another general method constructs all the $K(K-1)/2$ possible binary SVCs; each SVC is trained on only two out of all the *K* classes. This approach is denoted as one-versus-one (1-v-1) SVC, such as DAGSVM (Platt *et al.*, 2000) and *K*-SVCR (Angulo *et al.*, 2003). Both above decomposing-reconstruction architecture approaches have to construct more than one classifier and associate with the problem of how to combine the sub-classifiers to get the best properties (Moreira and Mayoraz, 1998; Mayoraz and Alpaydin, 1999). Additionally, the idea of error correcting output codes (ECOC) (Dietterich and Bakiri, 1995) was applied in some combinatorial distribution of the training patterns to add the redundancy in pattern information. Nevertheless, many more sub-classifiers should be constructed and no algorithm exists to make the best choice between them.

Different to the decomposing-reconstruction scheme, over the last few years some researchers have proposed a new SVC methodology to solve *K*-class problems in one optimization formulation considering all the classes at once. However, the associated quadratic programming (QP) problem of a size proportional to *K* has to be solved in this way, so the optimization scale is usually very large. In this paper, inspired by the idea of a reformulation of the standard

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (No. 60675049), the National Creative Research Groups Science Foundation of China (No. 60721062), and the Natural Science Foundation of Zhejiang Province, China (No. Y106414)

binary SVC (Keerthi *et al.*, 2000; Kowalczyk, 2000), we present a novel multi-class SVC of a single QP problem, named as a multi-class large margin classifier (MLMC). MLMC simplifies the multi-class problem to find a representative point for each class in feature space, and classifies the new data to the class whose representative point is the nearest. This new machine solves *K*-class problems in one optimization formulation without increasing the size of the QP problem proportional to *K*. Finally our experiments indicate that MLMC is almost as good as (sometimes better than) many other multi-class SVCs for practical data classification.

The rest of this paper is organized as follows. In Section 2 we show some related work, where a brief review of multi-class SVCs for a single optimization problem is presented and a reformulation to standard binary SVC is introduced. In Section 3, as a theoretical development of the reformulated binary SVC in Section 2, we propose the new multi-class classifier, MLMC. Experiments in MLMC classification are depicted in Section 4 and results are discussed there. Finally, we provide a conclusion to the paper in Section 5.

RELATED WORK

Let

$$\Gamma=\{(\boldsymbol{x}_i, y_i)\,|\,i=1,2,\cdots,l\}\subset X\times Y \tag{1}$$

be a training dataset containing independent and identically distributed (i.i.d.) samples of an unknown probability density function. The task of multi-class classification from examples is to find a decision function $f(\boldsymbol{x}, \boldsymbol{w})$ approximation of the unknown function, defined from an input space $X\subseteq\mathbb{R}^N$ into a set of K classes $Y=\{\theta_1, \theta_2, \ldots, \theta_K\}$ having the smaller discrepancy with the real system answer (Angulo *et al.*, 2003). As an example, for a binary SVC problem, a decision function has the form $f(\boldsymbol{x}, \boldsymbol{w})=\mathrm{sgn}(h(\boldsymbol{x}, \boldsymbol{w}))$ with the outputs $\{\pm1\}$, where

$$h(\boldsymbol{x},\boldsymbol{w})=\langle\boldsymbol{w},\varphi(\boldsymbol{x})\rangle+b \tag{2}$$

is a separating hyperplane in some feature space F, with $\boldsymbol{w}\in F$, $b\in\mathbb{R}$, φ: $X\to F$ being a nonlinear mapping from the original input space to a usually high-dimensional space. The space F is dotted with an inner product $k(\boldsymbol{x},\boldsymbol{x}')\triangleq\langle\varphi(\boldsymbol{x}),\varphi(\boldsymbol{x}')\rangle$ accomplishing Mercer's theory (Vapnik, 2000).

K-class SVCs of a single QP problem

In this subsection, we conclude several approaches considering all the classes at once as multi-class SVCs that have been developed over the last few years. For simplification we do not use a kernel here and default all the slack variables $\xi_i\geq0$.

First, the two-class SVC is the following optimization problem:

$$\begin{aligned}&\min\left(\frac{1}{2}\|\boldsymbol{w}\|^2+C\sum_{i=1}^{l}\xi_i\right),\\&\text{s.t. } y_i(\langle\boldsymbol{w},\boldsymbol{x}_i\rangle+b)\geq1-\xi_i,\ i=1,2,\ldots,l.\end{aligned} \tag{3}$$

Weston and Watkins (1998) generalized the optimization problem of Eq.(3) to solve the *K*-class problem in a natural way:

$$\begin{aligned}&\min\left(\frac{1}{2}\sum_{p=1}^{K}\|\boldsymbol{w}^{(p)}\|^2+C\sum_{i=1}^{l}\sum_{y\neq y_i}\xi_{i,y}\right),\\&\text{s.t. } (\langle\boldsymbol{w}^{(y_i)},\boldsymbol{x}_i\rangle+b^{(y_i)})-(\langle\boldsymbol{w}^{(y)},\boldsymbol{x}_i\rangle+b^{(y)})\geq1-\xi_{i,y},\\&\forall i,\ y\neq y_i.\end{aligned} \tag{4}$$

Crammer and Singer (2002) substituted the maximal value for the sum of empirical loss for each sample, giving a modification to Eq.(4):

$$\begin{aligned}&\min\left(\frac{1}{2}\sum_{p=1}^{K}\|\boldsymbol{w}^{(p)}\|^2+C\sum_{i=1}^{l}\xi_i\right),\\&\text{s.t. } (\langle\boldsymbol{w}^{(y_i)},\boldsymbol{x}_i\rangle+b^{(y_i)})-\max_{y\neq y_i}(\langle\boldsymbol{w}^{(y)},\boldsymbol{x}_i\rangle+b^{(y)})\geq1-\xi_i,\ \forall i.\end{aligned} \tag{5}$$

Eqs.(4) and (5) are an intuitive extension to the Dichotomy structure of SVC, and can be seen as a direct generalization of a standard binary-class SVC algorithm. Based on a uniform convergence result derived for *K*-class discriminant models, Guermeur *et al.*(2000) proposed their multi-class SVC and claimed that they implemented the structural risk minimization (SRM) inductive principle in this way

$$\begin{aligned}&\min\left(\frac{1}{2}\sum_{p=1}^{K-1}\sum_{q=p+1}^{K}\|\boldsymbol{w}^{(p)}-\boldsymbol{w}^{(q)}\|^2+C\sum_{i=1}^{l}\sum_{y\neq y_i}\xi_{i,y}\right),\\&\text{s.t. } (\langle\boldsymbol{w}^{(y_i)},\boldsymbol{x}_i\rangle+b^{(y_i)})-(\langle\boldsymbol{w}^{(y)},\boldsymbol{x}_i\rangle+b^{(y)})\geq1-\xi_{i,y},\\&\forall i,\ y\neq y_i.\end{aligned} \tag{6}$$

Another multi-class SVC (M-SVM) was proposed in (Bredensteiner and Bennett, 1999), which is similar to Eq.(6).

Aforementioned methods under different forms allow people to regard the multi-class problem as a whole, but they will all associate with the large QP problem whose complexity tends to increase with the increase of K. Usually, a QP problem of the dual system for any above primal form has to deal with $l(K-1)$ variables considering at least $2l(K-1)$ inequalities and K equalities (Angulo *et al.*, 2003).

Alternative solution of classical binary SVC

A large margin classifier (LMC), which is both the basis of the following MLMC and a reformulation to standard binary SVC presented in a maximal margin perceptron (MMP) (Kowalczyk, 2000) should first be introduced here. This LMC is based on solving the original (primal) problem rather than the dual problem via satisfying KKT conditions. $I^{(+1)}$, $I^{(-1)}$ denote the subscript collections of the two classes, and let $\boldsymbol{w}^{(+1)} \triangleq \sum_{i\in I^{(+1)}} \alpha_i \varphi(\boldsymbol{x}_i)$, $\boldsymbol{w}^{(-1)} \triangleq \sum_{i\in I^{(-1)}} \alpha_i \varphi(\boldsymbol{x}_i)$. Then the alternative QP problem of two-class SVC is formulated as (Keerthi *et al.*, 2000; Kowalczyk, 2000)

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{w}^{(+1)} - \boldsymbol{w}^{(-1)}\|^2,$$
$$\text{s.t.} \sum_{i\in I^{(+1)}} \alpha_i = \sum_{i\in I^{(-1)}} \alpha_i = 1,\ \alpha_i \geq 0,\ i = 1, 2, ..., l. \tag{7}$$

Vectors $\boldsymbol{w}^{(+1)}$ and $\boldsymbol{w}^{(-1)}$ are named as 'support centers' of the hyperplane for the binary decision. The hyperplane is orthogonal to the vector $\boldsymbol{w}^{(+1)}-\boldsymbol{w}^{(-1)}$ and passes through the center $(\boldsymbol{w}^{(+1)}+\boldsymbol{w}^{(-1)})/2$ of the segment joining $\boldsymbol{w}^{(+1)}$ to $\boldsymbol{w}^{(-1)}$. The underlying idea behind this method is to generate a solution by approximating the closest points between two convex polytopes (formed by convex hulls of the data points separated according to their class label). Minimal $\|\boldsymbol{w}^{(+1)}-\boldsymbol{w}^{(-1)}\|$ represents the closest distance between two convex polytopes, such that the hyperplane offers a maximal margin between two classes. If we denote $\boldsymbol{w}=\boldsymbol{w}^{(+1)}-\boldsymbol{w}^{(-1)}$, $b=(\|\boldsymbol{w}^{(-1)}\|^2-\|\boldsymbol{w}^{(+1)}\|^2)/2$, then the corresponding decision function will be $f(\boldsymbol{x}, \boldsymbol{w})=\operatorname{sgn}(\langle \boldsymbol{w}, \varphi(\boldsymbol{x})\rangle + b)$.

MULTI-CLASS LARGE MARGIN CLASSIFIER

Proposition of MLMC

Naturally, Eq.(7) can be generalized for K-class problems, and the corresponding QP problem can be written as

$$\min_{\boldsymbol{\alpha}} \left(\frac{1}{2} \sum_{p,q=1;p\neq q}^{K} \| \boldsymbol{w}^{(p)} - \boldsymbol{w}^{(q)} \|^2 \right),$$
$$\text{s.t.} \sum_{i\in I^{(p)}} \alpha_i = 1,\ p = 1, 2, ..., K,\ \alpha_i \geq 0,\ i = 1, 2, ..., l, \tag{8}$$

where $I^{(p)}$ denotes the subscript collection of the pth class and support center $\boldsymbol{w}^{(p)} \triangleq \sum_{i\in I^{(p)}} \alpha_i \varphi(\boldsymbol{x}_i)$ for all p=1, 2, …, K. To be consistent with the SVM's conventional terminology, we use the term 'support vectors' to refer to those training samples for which the coefficients α_i's are non-zero. So the support center $\boldsymbol{w}^{(p)}$ is the combination of the support vectors belonging to the pth class. Let $b^{(p)} \triangleq -\| \boldsymbol{w}^{(p)} \|^2/2$, $g^{(p)}(\boldsymbol{x}) = \langle \boldsymbol{w}^{(p)}, \varphi(\boldsymbol{x})\rangle + b^{(p)}$ for all p=1, 2, …, K, then the K-class label decision function can be described as $f(\boldsymbol{x}, \boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, ..., \boldsymbol{w}^{(K)}) = \arg\max_p g^{(p)}(\boldsymbol{x})$.

To simplify the problem, replace $\boldsymbol{w}^{(p)}$ in Eq.(8) by $\sum_{i\in I^{(p)}} \alpha_i \varphi(\boldsymbol{x}_i)$, and then a simpler notation for formulation of the K-class QP problem can be written as

$$\min_{\boldsymbol{\alpha}} \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{H} \boldsymbol{\alpha},$$
$$\text{s.t.} \sum_{i\in I^{(p)}} \alpha_i = 1,\ p = 1, 2, ..., K,\ \alpha_i \geq 0,\ i = 1, 2, ..., l, \tag{9}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_l]^{\mathrm{T}}$ is an $l\times 1$ vector and

$$H_{ij} = \begin{cases} (K-1)k(\boldsymbol{x}_i, \boldsymbol{x}_j), & \text{if } \boldsymbol{x}_i, \boldsymbol{x}_j \text{ are in the same class,} \\ -k(\boldsymbol{x}_i, \boldsymbol{x}_j), & \text{otherwise.} \end{cases} \tag{10}$$

In a 2-norm soft margin SVC algorithm, the only change for introducing slack is the addition of $1/C$ to the diagonal of the inner product matrix associated with the training set, where C is a specific constant named as 'penalization level' (Vapnik, 2000). We can therefore simply view it as a change of kernel:

replacing $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ by $k(\boldsymbol{x}_i, \boldsymbol{x}_j)+\delta_{ij}/C$, where δ_{ij} is a delta function whose value is 1 when $i=j$, and 0 otherwise. Obviously it can be generalized to a soft margin algorithm in MLMC by transforming Eq.(10) to

$$H_{ij}=\begin{cases}(K-1)(k(\boldsymbol{x}_i,\boldsymbol{x}_j)+\delta_{ij}/C), \\ \quad \text{if } \boldsymbol{x}_i,\boldsymbol{x}_j \text{ are in the same class}, \\ -(k(\boldsymbol{x}_i,\boldsymbol{x}_j)+\delta_{ij}/C), \quad \text{otherwise.}\end{cases} \tag{11}$$

We briefly conclude the steps for constructing MLMC discrimination as follows:

Step 1: Calculate $\boldsymbol{H}$ through Eq.(11) and obtain $\boldsymbol{\alpha}$ by solving the QP problem Eq.(9).

Step 2: Calculate $g^{(p)}(\boldsymbol{x})$ for all p=1, 2, …, K by

$$\begin{aligned} g^{(p)}(\boldsymbol{x}) &\triangleq \langle \boldsymbol{w}^{(p)}, \varphi(\boldsymbol{x})\rangle + b^{(p)} \\ &= \sum_{i\in I^{(p)}} \alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}) - \frac{1}{2}\|\boldsymbol{w}^{(p)}\|^2 \\ &= \sum_{i\in I^{(p)}} \alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}) - \frac{1}{2}\sum_{i,j\in I^{(p)}} \alpha_i\alpha_j k(\boldsymbol{x}_i,\boldsymbol{x}_j). \end{aligned} \tag{12}$$

Step 3: Classify new $\boldsymbol{x}$ to class t, where

$$t=\arg\max_p (g^{(p)}(\boldsymbol{x})). \tag{13}$$

Analysis for MLMC

If we denote

$$\begin{aligned} \phi^{(p)}(\boldsymbol{x}) &\triangleq \frac{1}{2}\|\varphi(\boldsymbol{x})\|^2 - g^{(p)}(\boldsymbol{x}) \\ &= \frac{1}{2}\|\varphi(\boldsymbol{x})\|^2 - \langle \boldsymbol{w}^{(p)}, \varphi(\boldsymbol{x})\rangle + \frac{1}{2}\|\boldsymbol{w}^{(p)}\|^2 \\ &= \frac{1}{2}\|\varphi(\boldsymbol{x}) - \boldsymbol{w}^{(p)}\|^2, \end{aligned} \tag{14}$$

since $\|\varphi(\boldsymbol{x})\|^2/2$ is independent of p, then the decision function can also be written as

$$f(\boldsymbol{x}, \boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, ..., \boldsymbol{w}^{(K)}) = \arg\min_p (\phi^{(p)}(\boldsymbol{x})). \tag{15}$$

Eqs.(14) and (15) mean that the underlying idea behind this method is to find a support center $\boldsymbol{w}^{(p)}$ for each class p in the feature space, more intuitively denominated as 'representative point' in this paper. Then we can classify the new $\boldsymbol{x}$ in the class whose representative point is the nearest in the feature space. Geometrically, each representative point locates within the polytope that constitutes convex hulls of corresponding class data points in a certain feature space. As we demonstrate in Eqs.(14) and (15), the decision using Eq.(13) is equivalent to implicitly establishing the separation of all classes as Voronoi Diagram (Aurenhammer and Klein, 2000) according to representative points in the feature space.

Penalization level C is not directly related to a fitting error in MLMC, but plays an important role in taking a moderate amount of data into consideration. When $C=\infty$, i.e., there is no slack, the representative points will only occur on the surface (or boundary) of convex hulls of each class data according to the KKT complementarity condition; hence maybe few points on the surface (or boundary) make the decision. In real data, where noise can always be present, this can result in a brittle estimator. When C is a mild positive value, slack is permitted, and more training points can be taken into account, which makes the measure more noise-tolerant.

MLMC will not raise a dual problem, and can almost give an optimal result according to Eq.(9) in any feature space only if the inner product matrix $\boldsymbol{H}$ is positive definite. With the effect of adding $1/C$ to the eigenvalues of $\boldsymbol{H}$, the QP problem gets better conditioned. Furthermore, the corresponding QP problem has only l variables α_i (i=1, 2, …, l) regardless of K, and only regards l inequalities and K equalities as constraints, which makes it a faster training method compared with other present K-class SVCs for a single QP problem.

Architecture complexity comparison

The architecture complexity of SVC is mainly determined by the size of the corresponding QP problem and is directly associated with temporal complexity in training. This study will compare the architecture complexities of several multi-class SVCs in theory. Assuming a five-class problem with 200 patterns in each class, the comparison containing the number of classifiers, variables and constraints (the last two determine the size of the QP problem) (Angulo *et al.*, 2003) is displayed in Table 1.

The top six methods in the table are of the architecture of combining binary classifiers. Each of

Table 1 Comparison of architecture complexities of several multi-class SVCs*

Methods	Number of classifiers	Number of variables**	Number of constraints**
1-v-r	K=5	l=1000	2(V+1)=2002
1-v-1	C_K^2=10	2l/K=400	2(V+1)=802
ECOC (Alpaydin and Mayoraz, 1998)	$\leq 2^{K-1}-1$=15	l=1000	2(V+1)=2002
ECOC (Allwein *et al.*, 2000)	$\leq K(2^{K-1}-1)$=75	$\leq l$=1000	$\leq$2(V+1)=2002
DAGSVM (Platt *et al.*, 2000)	C_K^2=10	2l/K=400	2(V+1)=802
K-SVCR (Angulo *et al.*, 2003)	C_K^2=10	l=1000	2(V+1)=2002
qp-mc-sv (Weston and Watkins, 1998)	1	l(K−1)=4000	2(V+K)=8010
KSVMC (Guermeur *et al.*, 2000)	1	l(K−1)=4000	2(V+K)=8010
M-SVM (Bredensteiner and Bennett, 1999)	1	l(K−1)=4000	2(V+K)=8010
MLMC (This paper)	1	l=1000	V+K=1005

* A five-class problem having 200 patterns in each class has been used for illustration; ** For each method the value of the number of variables is taken as the V for calculating the number of constraints

them needs more than one classifier for a decision. The last four methods qp-mc-sv, KSVMC, M-SVM and our MLMC result in only one ultimate classifier considering all data at once. Standard 1-v-1 and DAGSVM have the least variables and constraints and are maybe the most computationally efficient. However, in (Alpaydin and Mayoraz, 1998) the two main drawbacks for such 1-v-1 decomposition scheme are highlighted: the number of classifiers is high and only two classes are involved for each classifier, so variance is higher and no information is given for the rest of the classes. Apart from these two approaches, MLMC is more efficient than the others, especially compared to other multi-class algorithms for a single QP problem.

EXPERIMENTS

To evaluate the performance of the MLMC algorithm and compare it with other multi-class SVCs, several experiments with artificial data in $\mathbb{R}^2$, four databases from the UCI repository (Blake and Merz, 1998) and two subsets of the United States Postal Service (USPS) handwritten digit set have been employed. At the end of this section, a large-scale discriminant application of face recognition on the AR (Martinez and Benavente, 1998) database is used to further demonstrate the effectiveness of the new algorithm. Our algorithm was implemented on Matlab v7.3 on AMD Athlon 2800+ 1.8 GHz using monqp in the SVM-KM toolbox (Canu *et al.*, 2005) as the QP solver.

Artificial data

An intuitionistic illustration of MLMC's properties and especially of the penalization level C's influence will be made on separable artifical data. In this study, 'separable' means 'piecewise linear separable' defined in (Bredensteiner and Bennett, 1999). Two training sets, Γ_1 and Γ_2, have been generated for the separable case following Gaussian distributions on $\mathbb{R}^2$, each with 150 patterns equally distributed in three classes, as shown in Fig.1 (In the figures throughout the paper, squares, circles, and diamonds each represent a kind of 2D sample). The linear kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$ is used in this separable example.

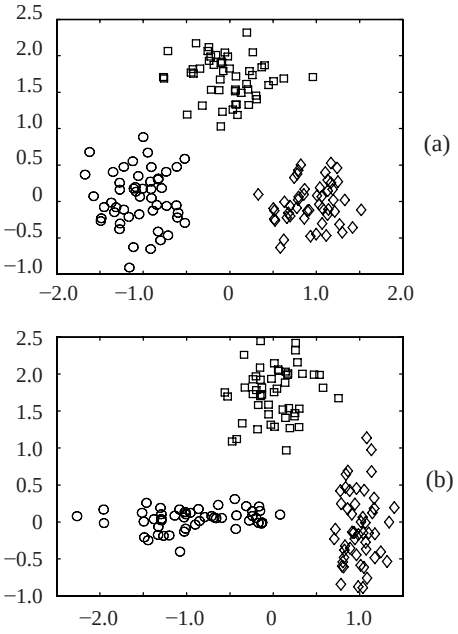

Fig.1 Training sets for the separable case
(a) Training set Γ_1; (b) Training set Γ_2

Fig.2 shows the effect of C on classification. As can be seen from Fig.2, with the decrease of penalization level C the number of support vectors increases. When $C=\infty$ (left), the optimal hyperplane will be very sensitive to noise because it is totally determined by the few nearest points that are obviously on the boundary of convex hulls for each class data. When C=0.5 (middle), more data points of each class are taken into consideration to generate representative points $\boldsymbol{w}^{(p)}$ (p=1, 2, 3). When $C\to 0$ (right), $\boldsymbol{H}\to(K-1)\boldsymbol{I}_{l\times l}/C$ in Eq.(11), and as a result of Eq.(9), representative points will be the means of each class that will lose the margin information. So C can adjust the trade-off between the margin information and the holistic information of data.

For the no separable case evaluation, training sets Γ_3 and Γ_4 have been generated on $\mathbb{R}^2$. Fig.3a shows Γ_3, the 150-pattern dataset of three classes, and Fig.3b displays Γ_4, the nested spirals three-class dataset with a total of 153 patterns.

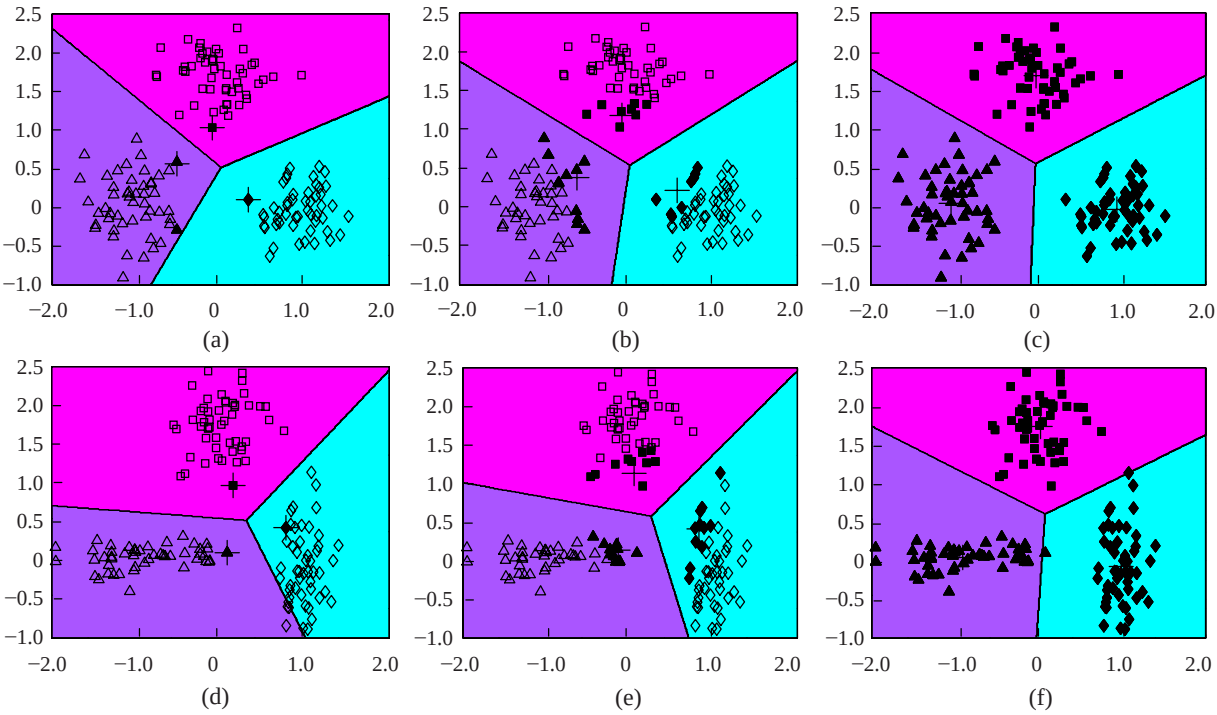

Fig.2 Different penalization levels C on training set Γ_1 ((a), (b), (c)) and Γ_2 ((d), (e), (f))

+: representative points; filled points: support vectors; displayed quantities: penalization level C and number of support vectors, n_{sv}. (a) $C=\infty$, n_{sv}=4; (b) C=0.5, n_{sv}=28; (c) C=0, n_{sv}=150; (d) $C=\infty$, n_{sv}=3; (e) C=0.5, n_{sv}=32; (f) C=0, n_{sv}=150

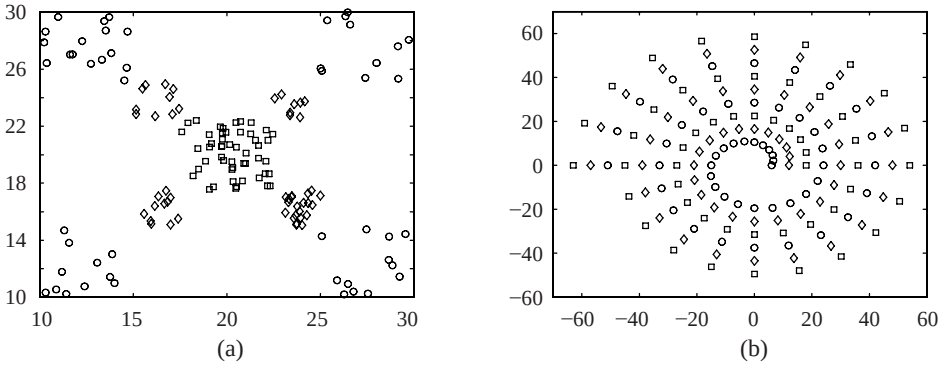

Fig.3 Training sets for the no separable case

(a) Training set Γ_3; (b) Training set Γ_4

MLMC and qp-mc-sv (Weston and Watkins, 1998) have been trained on the dataset Γ_3, which is also a multi-class SVC considering all the classes at once. Polynomial kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle)^d$ has been adopted to compare the performance of the two methods in the finite VC-dimension feature spaces. In the training procedure, the parameter C=30 has been used for both methods. Experimental results are summarized in Fig.4. In this example, qp-mc-sv gives a perfect division when degree d=2, but exhibits bad behavior with other degree choices. This phenomenon indicates that the QP problem (corresponding to Eq.(4)) leading to the qp-mc-sv solution is not correctly solved with such a degree choice, because the high-level constraints are not totally matched in corresponding feature space. Conversely, MLMC gives more stable results in various feature spaces.

As to training set Γ_4, we illustrate MLMC with a radial basis function (RBF) kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\sigma^2)$ while C is set to 30. The excellent generalization performance is clear from the decision boundaries shown on Fig.5. For this three-spiral classification the method gives good results over a wide parameter range of σ.

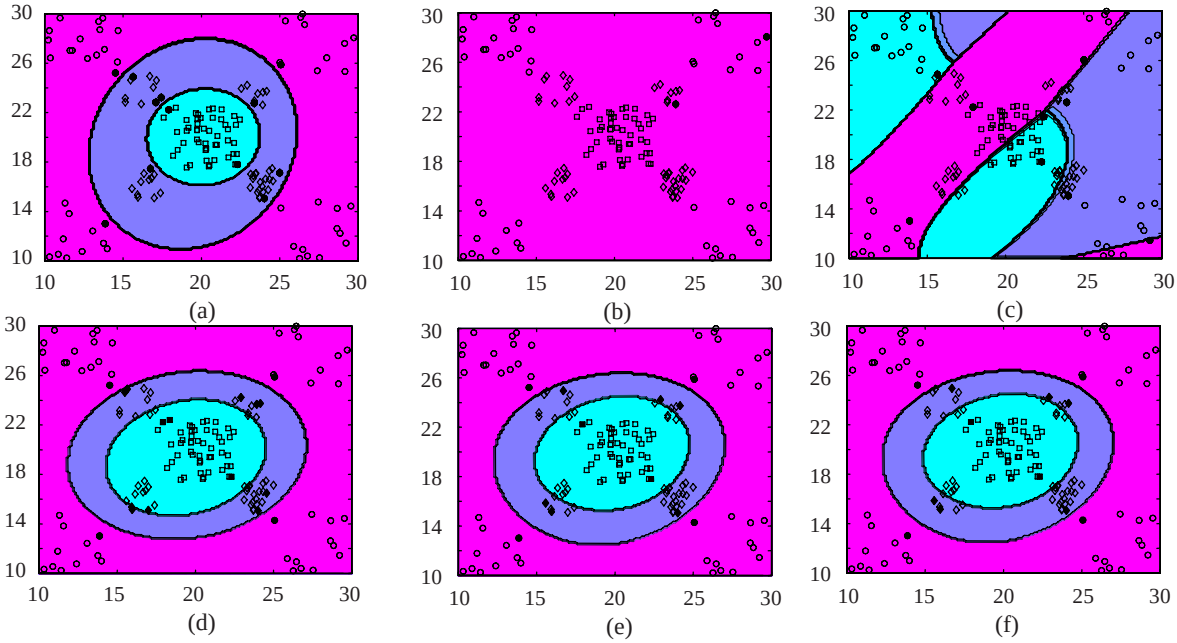

Fig.4 Training results on dataset Γ_3 using polynomial kernels with constant penalization level C=30 and different degrees d
(a), (b), (c): qp-mc-sv trained on Γ_3; (d), (e), (f): MLMC trained on Γ_3. (a), (d): d=2; (b), (e): d=3; (c), (f): d=4

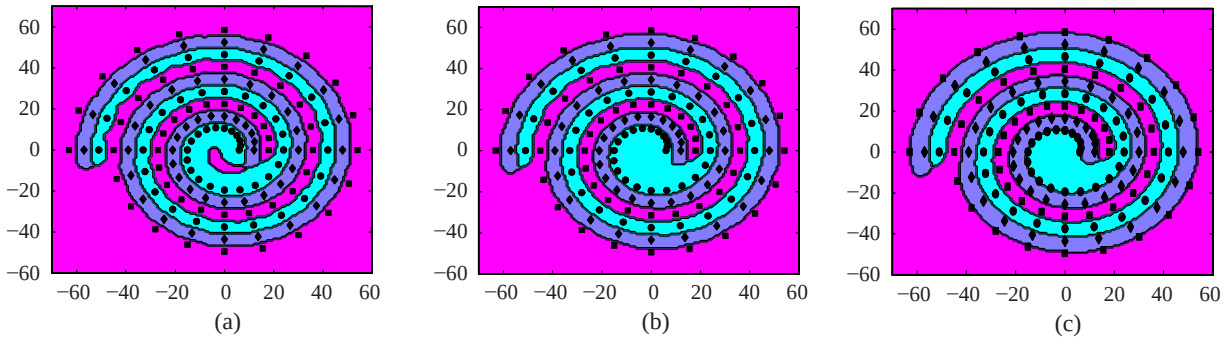

Fig.5 Training results on dataset Γ_4 using radial basis function kernels with constant penalization level C=30 and different parameters σ. (a) σ=5.0; (b) σ=7.0; (c) σ=10.0

Benchmark database

Four popular databases from the UCI repository—Iris, Wine, Glass, Vowel—and two subsets of the USPS have been employed to validate the property of MLMC. This choice has been taken to compare the obtained results with those presented in (Weston and Watkins, 1998; Bredensteiner and Bennett, 1999), the most usual multi-class scheme.

Two experiments on these benchmarks were performed. In the first experiment, according to the cross validation procedure established in (Weston and Watkins, 1998; Bredensteiner and Bennett, 1999), Iris, Wine, Glass, Vowel databases were randomly partitioned with a tenth of the data reserved to validate the multi-class SVCs, and this process has been repeated 10 times. The second experiment was conducted on two subsets USPS-1 and USPS-2 of USPS defined in (Bredensteiner and Bennett, 1999), which contain the handwritten integers "3, 5, 8" and "4, 6, 7", respectively. The USPS-1 contains 1657 training samples and 492 testing samples, and the USPS-2 contains 1876 training samples and 517 testing samples.

Features of all benchmark data were normalized to [0, 1], and the RBF kernel was used in both experiments. C=30 was chosen constant and σ was set to 0.1, 0.4, 0.1, 0.4, 4.0 and 4.0 for Iris, Wine, Glass, Vowel, USPS-1 and USPS-2, respectively. Table 2 summarizes the experimental results obtained by MLMC and compares them to those presented in (Weston and Watkins, 1998; Bredensteiner and Bennett, 1999). It also displays the number of training patterns, features, and classes for each database. In particular, we report the average computational training time (in seconds) for qp-mc-sv and MLMC in Table 2. Here the CPU time for qp-mc-sv was obtained by using SVM-KM toolbox with respect to the parameter setting given in (Weston and Watkins, 1998); that is, qp-mc-sv in the experiment has the same QP solver with our MLMC. Noticing again the theoretical analysis of complexity presented in Table 1, qp-mc-sv has an approximately identical scale of the associated QP problem with the other most usual multi-class SVCs of a single QP problem. So it is feasible to evaluate the training speed performance of MLMC through the time comparison between qp-mc-sv and MLMC.

It can be observed that MLMC almost improves the obtained results for all other multi-class architectures in the experiment. In detail, on the Iris set the absolute error is small, and MLMC generalizes surprisingly well and almost achieves a perfect prediction in our test. When the absolute error becomes a bit bigger on the other sets, MLMC still surpasses the accuracies of the rest, including in the large size and high dimensional USPS dataset. As to the Glass and Vowel sets, the absolute errors are the largest; possibly because the data are noisy, the MLMC exhibits comparable behavior to all other multi-class SVCs (superior to 6 of 7 reported results in Table 2). The comparison between the CPU time indicators of qp-mc-sv and MLMC validates that MLMC is a fast training scheme for multi-class SVC of a single QP problem.

The experimental results indicate that even with low demanding computation, MLMC obtains satisfactory generalization compared with others, and behaves more robustly when database classification is relatively simple. Additionally, MLMC gives a moderate number of support vectors, which means that it also has a rapid classification speed in testing among all the SVCs mentioned above.

Table 2 Performance of the MLMC and some multi-class SVCs

Database	Number of points	Number of features	Number of classes	Error percentage (%)[#]				
				1-v-r[*]	1-v-1[*]	qp-mc-sv[*]	M-SVM[**]	MLMC[***]
Iris	150	4	3	1.33 (75)	1.33 (54)	1.33 (31), 1.10 s	–	0.33 (72), 0.56 s
Wine	178	13	3	5.6 (398)	5.6 (268)	3.6 (135), 7.21 s	3.37 (228)	1.18 (114), 0.78 s
Glass	214	9	6	35.2 (308)	36.4 (368)	35.6 (113), 6.72 s	32.71 (1476)	34.1 (142), 1.42 s
Vowel	528	10	11	39.8 (2170)	38.7 (3069)	34.8 (1249)	–	20.7 (266), 6.64 s
USP-1	1657	256	3	–	–	–	7.72 (317)	5.69 (703), 147.38 s
USP-2	1876	256	3	–	–	–	6.00 (180)	3.29 (533), 127.31 s

[#] The total number of support vectors given in the brackets for MLMC and some multi-class SVCs; for qp-mc-sv and MLMC the average computational training time (in seconds) is reported. [*] Proposed by Weston and Watkins (1998); [**] Proposed by Bredensteiner and Bennett (1999); [***] This paper

AR face database

In this subsection we use a subset of the AR face database provided and preprocessed by Martinez and Benavente (1998). This subset contains 2600 face images belonging to 100 people (50 men and 50 women), where each person has 26 different images taken in two sessions (separated by two weeks) under various conditions: variation in expression, different illumination and partial occlusion by glasses or scarves. We resized the images to 33×24 pixels, and the gray level values were rescaled to [0, 1]. So each sample vector has 792 features scaled within [0, 1]. It is a challenging face database for recognition, and we assembled each odd index image of every person as the training sample. In this way we reserved the most variation of presentative information within our training set. Then we obtained a total of l=1300 samples of K=100 classes for training, and the remaining 1300 samples corresponding to even indices act as the testing set.

Obviously, faced with this task, the conventional one-classifier multi-class SVC as Eqs.(4)~(6) has to deal with a QP problem of $l(K-1)$=128 700 variables with $2(l+K)$=2800 constraints, which is an intractable computational demand without a decomposition mechanism. But MLMC just needs to figure out a QP problem of l=1300 variables with $l+K$=1400 constraints.

We used Fisher LDA (Swets and Weng, 1996) as a performance benchmark, since it is an acknowledged effective linear subspace method to extract most discriminant features with nearest neighbor decision, and hence appropriate to be an estimation for the difficulty of separating original data. The LDA's best classification accuracy on testing set is 74.23% after obtaining discriminant features in the training set.

Table 3 contains performance for MLMC on AR using an RBF kernel. The solution that MLMC has found has a remarkably higher testing accuracy than LDA over a wide range of parameters σ and C, especially when σ=5.0 when it achieves the peak classification accuracy. As seen from the results, the situation of overfitting may have occurred when σ is small, that is, a low prediction accuracy but with a perfect recall on training samples. On the other hand, the recall accuracy slightly drops when σ increases to more than 7.0. So a σ between 5.0 and 7.0 is recommended in this application. Notice that, in such a high dimensional case with relatively few samples, changing C does not lead to dramatic variation in classification accuracy. Whereas a larger C consistently finds classification functions using fewer support vectors, which means that a new sample can be classified more quickly. Thus C is a sensitive parameter to choose when classification time is critical. The average CPU time for training under all combinations of parameters above is less than 700 s on our PC, and generally the time consuming trend is that of less training time for smaller σ and smaller C.

Table 3 Performance of the MLMC on AR face database (RBF kernel)

C	Percentage accuracy (%)[#]					
	σ=0.1	1.0	3.0	5.0	7.0	10.0
5	100/1.15 (1300)	100/48.38 (1300)	100/79.15 (1300)	100/85.23 (1300)	99.38/84.62 (1279)	94.77/80.15 (1243)
∞	100/1.15 (1300)	100/40.96 (1300)	100/80.23 (1300)	100/84.08 (1285)	98.69/84.46 (1192)	94.08/80.46 (1063)

[#] Percentage accuracy of the training set/Percentage accuracy of the testing set; figures in the brackets are the total numbers of support vectors

CONCLUSION

MLMC presented in this paper is a novel multi-class SVC. It is a direct generalization of the reformulated binary SVC presented in MMP with a holistic consideration of multi-class information. It solves the K-class problem in one optimization formulation without increasing the size of the QP problem proportional to K. Several experiments on artificial data show the ability of the new classifier to treat multi-class problems and they illustrate the penalization parameter utility. In the meantime, the experiments on the benchmark database show the comparable (sometimes better) properties of the new classifier compared in general with many other multi-class SVCs. A face recognition experiment further validates our method in large-scale applications.

References

Allwein, E.L., Schapire, R.E., Singer, Y., 2000. Reducing multiclass to binary: a unifying approach for margin classifiers. *J. Mach. Learn. Res.*, **1**:113-141.

Alpaydin, E., Mayoraz, E., 1998. Combining Linear Dichomotizers to Construct Nonlinear Polychotomizers. Technical Report IDIAP-RR, Switzerland.

Angulo, C., Parra, X., Catala, A., 2003. K-SVCR. A support vector machine for multi-class classification. *Neurocomputing*, **55**(1-2):57-77. [doi:10.1016/S0925-2312(03)00435-1]

Aurenhammer, F., Klein, R., 2000. Voronoi Diagram. *In*: Sack, J.R., Urrutia, J. (Eds.), Handbook of Computational Geometry. Elsevier Science Publishers, B.V. North-Holland, Amsterdam, p.201-290.

Blake, C.L., Merz, C.J., 1998. UCI Repository of Machine Learning Databases. University of California, Irvine. Http://www.ics.uci.edu/mlearn/MLRepository.html

Bredensteiner, E.J., Bennett, K.P., 1999. Multicategory classification by support vector machines. *Comput. Optim. Appl.*, **12**:53-79. [doi:10.1023/A:1008663629662]

Canu, S., Grandvalet, Y., Guigue, V., Rakotomamonjy, A., 2005. SVM and Kernel Methods Matlab Toolbox. Perception Systèmes et Information, INSA de Rouen, Rouen, France.

Cortes, C., Vapnik, V., 1995. Support vector networks. *Machine Learning*, **20**(3):273-297. [doi:10.1023/A:1022627411411]

Crammer, K., Singer, Y., 2002. On the learnability and design of output codes for multiclass problems. *Machine Learning*, **47**(2-3):201-233. [doi:10.1023/A:1013637720281]

Dietterich, T.G., Bakiri, G., 1995. Solving multiclass learning problem via error-correcting output codes. *J. Artif. Intell. Res.*, **2**:263-286.

Guermeur, Y., Elisseeff, A., Paugam-Moisy, H., 2000. A New Multi-class SVM Based on a Uniform Convergence Result. Proc. IJCNN, Como, Italy, **4**:183-188. [doi:10.1109/IJCNN.2000.860770]

Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R., 2000. A fast iterative nearest point algorithm for support vector machineclassifier design. *IEEE Trans. Neural Networks*, **11**(1):124-136. [doi:10.1109/72.822516]

Kowalczyk, A., 2000. Maximal Margin Perceptron. *In*: Smola, A.J., Bartlett, P.L., Scholkopf, B., Schuurmans, D. (Eds.), Advances in Large Margin Classifiers. MIT Press, Cambrige, MA, p.75-113.

Martinez, A.M., Benavente, R., 1998. The AR Face Database. Technical Report, CVC.

Mayoraz, E., Alpaydin, E., 1999. Support Vector Machines for Multi-class Classification. Proc. IWANN, Alicante, Spain, p.833-842.

Moreira, M., Mayoraz, E., 1998. Improved Pairwise Coupling Classification with Correcting Classifiers. Proc. ECML, Chemnitz, Germany, p.160-171.

Platt, J., Cristianini, N., Shawe-Taylor, J., 2000. Large Margin DAGs for Multiclass Classification. *In*: Solla, S.A., Leen, T.K., Müller, K.R. (Eds.), Advance in Neural Information Process Systems. MIT Press, Cambridge, MA, p.547-553.

Swets, D.L., Weng, J., 1996. Using discriminant eigenfeatures for image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, **18**(8):831-836. [doi:10.1109/34.531802]

Vapnik, V., 2000. The Nature of Statistical Learning Theory. Springer, New York.

Weston, J., Watkins, C., 1998. Multi-class Support Vector Machines. Technical Report, CSD-TR-98-04. Royal Holloway, University of London, Egham, UK.