Diversified SVM Ensembles for Large Data Sets

Ivor W. Tsang[1], Andras Kocsor[2], and James T. Kwok[1]

[1] Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong
{ivor, jamesk}@cse.ust.hk
[2] Research Group on Artificial Intelligence
Hungarian Academy of Sciences and University of Szeged
H-6720 Szeged, Aradi vrt. 1., Hungary
kocsor@inf.u-szeged.hu

Abstract. Recently, the core vector machine (CVM) has shown significant speedups on classification and regression problems with massive data sets. Its performance is also almost as accurate as other state-of-the-art SVM implementations. By incorporating the orthogonality constraints to diversify the CVM ensembles, this turns out to speed up the maximum margin discriminant analysis (MMDA) algorithm. Extensive comparisons with the MMDA ensemble along with bagging on a number of large data sets show that the proposed diversified CVM ensemble can improve classification performance, and is also faster than the original MMDA algorithm by more than an order of magnitude.

1 Introduction

Support vector machines (SVMs) have been highly successful in many machine learning problems. Recently, the core vector machines (CVM) [1] is proposed for scaling up SVM. The main idea is to formulate the learning problem as a minimum enclosing ball (MEB) problem, and then apply an $(1+\epsilon)$-approximation algorithm. It has a provably asymptotic time complexity that is *linear* in m and a space complexity that is *independent* of m. Experiments on large classification [1] and regression [2] data sets demonstrate that the CVM is much faster and can handle much larger data sets than existing scale-up methods.

However, while a single SVM is often good in most cases, it is not always perfect. In particular, when there are many noisy patterns, they may corrupt the optimal decision boundary of a single SVM hyperplane. To address this problem, several ensemble methods, such as bagging, boosting and nonlinear ensemble approaches [3,4], have been proposed to improve SVM performance by combining multiple SVMs. However, these SVM ensemble methods require having many SVMs as base classifiers [4].

On the other hand, AdaBoost [5] has achieved good generalization performance by constructing weak classifier ensembles. The key idea is to update the probability distribution d_i's over the training set subject to the corrective

J. Fürnkranz, T. Scheffer, and M. Spiliopoulou (Eds.): ECML 2006, LNAI 4212, pp. 792–800, 2006.

constraint that the new distribution is orthogonal to the vector of the margin errors $-y_i f_t(\mathbf{x}_i)$. Consider the following weak classifier that is a variant of the Parzen window classifier, with the patterns weighted by d_i's: $f_t(\mathbf{x}) = \sum_{i=1}^m d_i^t y_i k(\mathbf{x}_i, \mathbf{x}) = \mathbf{w}_t' \varphi(\mathbf{x})$, where φ is the feature map associated with the kernel k, and $\mathbf{w}_t$ is the current weight vector. Then, the constraint for the new d_i^{t+1} distribution is $\sum_{i=1}^m d_i^{t+1} y_i f_t(\mathbf{x}_i) = 0$ or $\mathbf{w}_{t+1}' \mathbf{w}_t = 0$. This implies that the weight vector of the two consecutive weak classifiers are orthogonal. Moreover, Kivinen *et al.* [5] suggested finding the new distribution subject to the totally corrective constraints, i.e., the new distribution is orthogonal to the vectors of margin errors of all existing classifiers ($\mathbf{w}_{t+1}' \mathbf{w}_r = 0$ for $r = 1, \ldots, t$). Thus, usually only a few weak classifiers are required in constructing an ensemble with good classification performance.

The diversity of the base classifiers can improve the performance of the ensembles [4,6]. Intuitively, the orthogonality constraints can also be exploited to diversify the base SVM classifiers. By adding orthogonality constraints to the CVM ensemble, we will show in this paper that this can be seen as integrating maximum margin discriminant analysis MMDA [7] with the CVM. However, in order to apply the CVM algorithm, the QP problem corresponding to the kernel method of interest has to take a particular form. This, however, is not met by the MMDA, as the original CVM does not allow orthogonality constraints on the weight vectors. Thus, we propose an extension of the MEB problem by placing orthogonality constraints on the center of the MEB. We can then obtain orthogonal CVM ensembles on large data sets efficiently.

The rest of this paper is organized as follows. Section 2 first reviews MMDA. Section 3 then describes the proposed extension of the MEB problem, the modified CVM algorithm, and other variants of MMDA. Experimental results are presented in Section 4, followed by some concluding remarks in the last section.

2 Maximum Margin Discriminant Analysis (MMDA)

Given a training set $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, with $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \pm 1$. Consider the following variant of the Lagrangian SVM [8], where the weight $\mathbf{w}$ is orthogonal to $\mathbf{u}_q = \mathbf{w}_q / \|\mathbf{w}_q\|$ for $q = 1, \ldots, s$:

$$\min \|\mathbf{w}\|^2 + b^2 + C \sum_{i=1}^m \xi_i^2 \quad : \quad y_i(\mathbf{w}' \varphi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \;\; \mathbf{u}_q' \mathbf{w} = 0. \tag{1}$$

Here, φ is the nonlinear feature map associated with kernel k, ξ_i's are slack variables and C is a regularization parameter. Introducing Lagrangian multipliers $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_m]'$ and $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_s]'$ for the inequality and equality constraints, we obtain the dual:

$$\max 2\boldsymbol{\alpha}' \mathbf{1} - \boldsymbol{\alpha}' \hat{\mathbf{K}} \boldsymbol{\alpha} - 2\boldsymbol{\alpha}' \mathbf{Y} \boldsymbol{\Phi}' \mathbf{U} \boldsymbol{\gamma} - \boldsymbol{\gamma}' \mathbf{U}' \mathbf{U} \boldsymbol{\gamma} \quad : \quad \boldsymbol{\alpha} \geq \mathbf{0}, \tag{2}$$

where $\mathbf{0}, \mathbf{1} \in \mathbb{R}^m$ are vectors of zeros and ones, $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_s]$, $\mathbf{K} = \boldsymbol{\Phi}' \boldsymbol{\Phi}$ (where $\boldsymbol{\Phi} = [\varphi(\mathbf{x}_1), \ldots, \varphi(\mathbf{x}_m)]$) is the kernel matrix, $\mathbf{Y} = \mathrm{diag}(y_1, \ldots, y_m)$, and

$$\hat{\mathbf{K}} = \mathbf{Y}\left(\mathbf{K} + \mathbf{1}\mathbf{1}' + \mathbf{I}/C\right)\mathbf{Y}, \tag{3}$$

is the transformed "kernel" matrix. By using the Karush-Kuhn-Tucker (KKT) conditions, the primal variables $\mathbf{w}, b$ can be recovered from the optimal $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and $\mathbf{u}_q$. Using (1), MMDA then extracts the weights ($\mathbf{w}$'s) one by one, and each of these can be expressed as a linear combination of $\varphi(\mathbf{x}_i)$'s. Note, however, that this MMDA formulation does not fit the existing MEB models in [1,2].

3 Core Vector Machine Ensembles

3.1 MEB with Multiple Projection Constraints on the Center

The center-constrained MEB problem in [2] constrains the center $\mathbf{c}$ to lie on the hyperplane $[\mathbf{0}'\ 1]\mathbf{c} = 0$. Here, we instead confine $\mathbf{c}$ to lie on multiple hyperplanes defined by $\tilde{\mathbf{u}}_1, \tilde{\mathbf{u}}_2, \ldots, \tilde{\mathbf{u}}_s$:

$$\min R^2 \quad : \quad \|\mathbf{c} - \tilde{\varphi}(\mathbf{x}_i)\|^2 \le R^2, \ \ \tilde{\mathbf{u}}_q'\mathbf{c} = v_q. \tag{4}$$

Introducing Lagrangian multipliers $\tilde{\boldsymbol{\alpha}} = [\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m]'$ and $\tilde{\boldsymbol{\gamma}} = [\tilde{\gamma}_1, \ldots, \tilde{\gamma}_s]'$ for the inequality and equality constraints, we obtain the dual:

$$\max \tilde{\boldsymbol{\alpha}}'\mathrm{diag}(\tilde{\mathbf{K}}) + \tilde{\boldsymbol{\gamma}}'\mathbf{v} - \tilde{\boldsymbol{\alpha}}'\tilde{\mathbf{K}}\tilde{\boldsymbol{\alpha}} - 2\tilde{\boldsymbol{\alpha}}'\tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} - \tilde{\boldsymbol{\gamma}}'\tilde{\mathbf{U}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} \quad : \quad \tilde{\boldsymbol{\alpha}} \ge \mathbf{0}, \ \ \tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1, \tag{5}$$

where $\mathbf{v} = [v_1, \ldots, v_s]'$, $\tilde{\mathbf{U}} = [\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_s]$, $\tilde{\mathbf{K}} = \tilde{\boldsymbol{\Phi}}'\tilde{\boldsymbol{\Phi}}$ and $\tilde{\boldsymbol{\Phi}} = [\tilde{\varphi}(\mathbf{x}_1), \ldots, \tilde{\varphi}(\mathbf{x}_m)]$. Assume that for any pattern $\mathbf{x}$, $\tilde{k}$ satisfies

$$\tilde{k}(\mathbf{x}, \mathbf{x}) = \tilde{\kappa}, \tag{6}$$

a constant. Using the constraint $\tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1$, we obtain $\tilde{\boldsymbol{\alpha}}'\mathrm{diag}(\tilde{\mathbf{K}}) = \tilde{\kappa}$. Dropping this constant from the objective in (5), we obtain a simpler QP:

$$\max \tilde{\boldsymbol{\gamma}}'\mathbf{v} - \tilde{\boldsymbol{\alpha}}'\tilde{\mathbf{K}}\tilde{\boldsymbol{\alpha}} - 2\tilde{\boldsymbol{\alpha}}'\tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} - \tilde{\boldsymbol{\gamma}}'\tilde{\mathbf{U}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} \quad : \quad \tilde{\boldsymbol{\alpha}} \ge \mathbf{0}, \ \ \tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1. \tag{7}$$

The radius $R = \sqrt{\tilde{\boldsymbol{\alpha}}'\mathrm{diag}(\tilde{\mathbf{K}}) + \tilde{\boldsymbol{\gamma}}'\mathbf{v} - \tilde{\boldsymbol{\alpha}}'\tilde{\mathbf{K}}\tilde{\boldsymbol{\alpha}} - 2\tilde{\boldsymbol{\alpha}}'\tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} - \tilde{\boldsymbol{\gamma}}'\tilde{\mathbf{U}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}}}$ and the center $\mathbf{c} = \sum_{i=1}^{m} \tilde{\alpha}_i\tilde{\varphi}(\mathbf{x}_i) + \sum_{q=1}^{s} \tilde{\gamma}_q\tilde{\mathbf{u}}_q$ are recovered from the optimal $\tilde{\boldsymbol{\alpha}}$ and $\tilde{\boldsymbol{\gamma}}$. Conversely, any QP in the form of (7) can be regarded as a MEB problem.

Once we have a MEB problem, one can apply the core-set approximation and probabilistic speedup techniques in CVM [1,2] to obtain an approximate solution of the MEB problem efficiently. The CVM procedure can be easily adapted to cater for this center $\mathbf{c}$. Each iteration then becomes the solving of the subproblem MEB($\mathcal{S}_t$) defined on the core-set $\mathcal{S}_t$.

Notice that finding MEB($\mathcal{S}_t$) still involves a QP. Instead of solving a QP with the equality constraint in (7), we follow the trick in [9] and remove the constraints by introducing Lagrangian multipliers $\tilde{\mu}_i$'s (where $\tilde{\mu}_i \ge 0$) for the nonnegative constraints $\tilde{\alpha}_i \ge 0$ and β for the equality constraint $\tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1$ in (7). Then the Lagrangian becomes $\tilde{L}(\tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{\mu}}, \beta) = \tilde{\boldsymbol{\gamma}}'\mathbf{v} - \tilde{\boldsymbol{\alpha}}'\tilde{\mathbf{K}}\tilde{\boldsymbol{\alpha}} - 2\tilde{\boldsymbol{\alpha}}'\tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} - \tilde{\boldsymbol{\gamma}}'\tilde{\mathbf{U}}'\tilde{\mathbf{U}}\tilde{\boldsymbol{\gamma}} + 2\tilde{\boldsymbol{\alpha}}'\tilde{\boldsymbol{\mu}} + 2\beta(\tilde{\boldsymbol{\alpha}}'\mathbf{1} - 1)$, where $\tilde{\boldsymbol{\mu}} = [\tilde{\mu}_1, \ldots, \tilde{\mu}_m]'$. We set its derivatives w.r.t. $\tilde{\boldsymbol{\alpha}}$ and $\tilde{\boldsymbol{\gamma}}$ to zero. Since $\tilde{\mathbf{K}} \succeq 0$ is pd and $\tilde{\mathbf{u}}_q$'s are independent, $\tilde{\mathbf{U}}'\tilde{\mathbf{U}} \succ 0$, and so $\tilde{\mathbf{G}} = \begin{bmatrix} \tilde{\mathbf{K}} & \tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}} \\ \tilde{\mathbf{U}}'\tilde{\boldsymbol{\Phi}} & \tilde{\mathbf{U}}'\tilde{\mathbf{U}} \end{bmatrix} \succ 0$. Hence, the optimal solution is:

$$[\tilde{\boldsymbol{\alpha}}'\ \tilde{\boldsymbol{\gamma}}']' = \tilde{\mathbf{G}}^{-1}[(\beta\mathbf{1} + \tilde{\boldsymbol{\mu}})'\ \mathbf{v}']', \tag{8}$$

where $\tilde{\boldsymbol{\mu}}$ and β are such that $\tilde{\boldsymbol{\alpha}} \geq \mathbf{0}, \tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1, \tilde{\boldsymbol{\alpha}} \odot \tilde{\boldsymbol{\mu}} = \mathbf{0}$ and $\tilde{\boldsymbol{\mu}} \geq \mathbf{0}$ (here, $\tilde{\boldsymbol{\alpha}} \odot \tilde{\boldsymbol{\mu}}$ is the elementwise product of $\tilde{\boldsymbol{\alpha}}$ and $\tilde{\boldsymbol{\mu}}$).

3.2 Connection to MMDA

We now return to the QP problem associated with MMDA in (2). Introduce Lagrangian multipliers $\mu_i \geq 0$'s for the nonnegative constraints $\alpha_i \geq 0$ in (2), then the Lagrangian is $L(\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\mu}) = 2\boldsymbol{\alpha}'\mathbf{1} - \boldsymbol{\alpha}'\hat{\mathbf{K}}\boldsymbol{\alpha} - 2\boldsymbol{\alpha}'\mathbf{Y}\boldsymbol{\Phi}'\mathbf{U}\boldsymbol{\gamma} - \boldsymbol{\gamma}'\mathbf{U}'\mathbf{U}\boldsymbol{\gamma} + 2\boldsymbol{\alpha}'\boldsymbol{\mu}$, where $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_m]'$. Since $\mathbf{K} \succeq 0$, $\hat{\mathbf{K}}$ in (3) is pd and $\mathbf{u}_q$'s are independent, $\mathbf{U}'\mathbf{U} \succ 0$, and so $\mathbf{G} = \begin{bmatrix} \hat{\mathbf{K}} & \mathbf{Y}\boldsymbol{\Phi}'\mathbf{U} \\ \mathbf{U}'\boldsymbol{\Phi}\mathbf{Y} & \mathbf{U}'\mathbf{U} \end{bmatrix} \succ 0$. Analogous to (8), an optimal solution is obtained as:

$$[\boldsymbol{\alpha}'\ \boldsymbol{\gamma}']' = \mathbf{G}^{-1}[(\mathbf{1} + \boldsymbol{\mu})'\ \mathbf{0}']', \tag{9}$$

where $\boldsymbol{\mu} \geq \mathbf{0}$, $\boldsymbol{\alpha} \geq \mathbf{0}$ and $\boldsymbol{\alpha} \odot \boldsymbol{\mu} = \mathbf{0}$. Alternatively, the optimal values for $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ can be solved by using the trick in [8]: $\mathbf{0} \leq \mathbf{a} \perp \mathbf{b} \geq \mathbf{0} \Leftrightarrow \mathbf{a} = (\mathbf{a} - \tau\mathbf{b})_+$ for $\tau > 0$, then $\boldsymbol{\mu} = ((\hat{\mathbf{K}}\boldsymbol{\alpha} + \mathbf{Y}\boldsymbol{\Phi}'\mathbf{U}\boldsymbol{\gamma} - \mathbf{1}) - \tau\boldsymbol{\alpha})_+$ by choosing a learning rate $\tau = 1.9/C$ as suggested in [8] (here, $\mathbf{a} \perp \mathbf{b}$ means $\mathbf{a}$ and $\mathbf{b}$ are perpendicular).

When the kernel k satisfies (6), $\hat{k}$ for the kernel matrix in (3) also satisfies (6), as $\hat{k}(\mathbf{x}, \mathbf{x}) = k(\mathbf{x}, \mathbf{x}) + 1 + 1/C$ is a constant for any $\mathbf{x}$. We set $v_q = 0$, $\tilde{\mathbf{U}} = [\mathbf{U}'\ \mathbf{0}']'$ (where $\mathbf{0}$ is the $s \times (m+1)$ zero matrix), and $\tilde{\varphi}(\mathbf{z}_i) = [y_i\varphi(\mathbf{x}_i)', y_i, y_i/\sqrt{C}\mathbf{e}_i']'$ (where $\mathbf{e}_i$ is the m-dimensional vector which has all zeros except that the ith entry is equal to one). Then $\tilde{\mathbf{K}} = \tilde{\boldsymbol{\Phi}}'\tilde{\boldsymbol{\Phi}} = [\tilde{k}(\mathbf{z}_i, \mathbf{z}_j)]$ with $\tilde{k}(\mathbf{z}_i, \mathbf{z}_j) = y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) + y_i y_j + \delta_{ij} y_i y_j / C$, $\tilde{\boldsymbol{\Phi}}'\tilde{\mathbf{U}} = \mathbf{Y}\boldsymbol{\Phi}'\mathbf{U}$ and $\tilde{\mathbf{U}}'\tilde{\mathbf{U}} = \mathbf{U}'\mathbf{U}$. Multiplying $[\mathbf{1}'\ \mathbf{0}']$ on both sides of (8) and (9): $\tilde{\boldsymbol{\alpha}}'\mathbf{1} - \mathbf{1}'\mathbf{H}\tilde{\boldsymbol{\mu}} = \beta\mathbf{1}'\mathbf{H}\mathbf{1} = \beta(\boldsymbol{\alpha}'\mathbf{1} - \mathbf{1}'\mathbf{H}\boldsymbol{\mu})$, where $\mathbf{H}$ is the left top $m \times m$ submatrix of $\tilde{\mathbf{G}}^{-1}$. Using $\tilde{\boldsymbol{\alpha}}'\mathbf{1} = 1$, and assuming that $\boldsymbol{\alpha}'\mathbf{1} > 0$, we have

$$\beta = \frac{1 - \mathbf{1}'\mathbf{H}\tilde{\boldsymbol{\mu}}}{\boldsymbol{\alpha}'\mathbf{1} - \mathbf{1}'\mathbf{H}\boldsymbol{\mu}} = \frac{1}{\boldsymbol{\alpha}'\mathbf{1}} \frac{\boldsymbol{\alpha}'\mathbf{1} - \mathbf{1}'\mathbf{H}\tilde{\boldsymbol{\mu}}\boldsymbol{\alpha}'\mathbf{1}}{\boldsymbol{\alpha}'\mathbf{1} - \mathbf{1}'\mathbf{H}\boldsymbol{\mu}} = \frac{1}{\boldsymbol{\alpha}'\mathbf{1}}, \tag{10}$$

where $\tilde{\boldsymbol{\mu}} = \frac{\boldsymbol{\mu}}{\boldsymbol{\alpha}'\mathbf{1}} \geq \mathbf{0}$. Furthermore, from (8), (9) and (10), we obtain

$$[\tilde{\boldsymbol{\alpha}}'\ \tilde{\boldsymbol{\gamma}}']' = \beta\tilde{\mathbf{G}}^{-1}[(\mathbf{1} + \boldsymbol{\mu})'\ \mathbf{0}']' = [\boldsymbol{\alpha}'\ \boldsymbol{\gamma}']'/\boldsymbol{\alpha}'\mathbf{1} \tag{11}$$

such that $\tilde{\boldsymbol{\alpha}}'\mathbf{1} = \frac{\boldsymbol{\alpha}'\mathbf{1}}{\boldsymbol{\alpha}'\mathbf{1}} = 1$, $\tilde{\boldsymbol{\alpha}} \odot \tilde{\boldsymbol{\mu}} = \frac{\boldsymbol{\alpha}}{\boldsymbol{\alpha}'\mathbf{1}} \odot \frac{\boldsymbol{\mu}}{\boldsymbol{\alpha}'\mathbf{1}} = \mathbf{0}$, and $\tilde{\boldsymbol{\alpha}} \geq \mathbf{0}$. Hence, using (11), the solutions of $\tilde{\boldsymbol{\alpha}}$ and $\tilde{\boldsymbol{\gamma}}$ in (5) can be recovered from the optimal values for $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ in (2). In other words, the optimization problem associated with MMDA in (1) can now be viewed as a constrained MEB problem in (4), with $\tilde{\varphi}$ being replaced by the new feature map $\hat{\varphi}$ and the associated kernel $\hat{k}$ satisfying (6).

3.3 Other Variants of MMDA

Other variants of MMDA that generate a non-orthogonal basis where the data is uncorrelated (but do not use the orthogonality constraints) can also use this new

MEB model. As discussed in [10], the uncorrelated constraints consider the relationship between patterns, and minimize redundancy among the weight vectors in the reduced space. We can replace the orthogonality constraints on $\mathbf{w}$ in (1) by uncorrelated constraints, and the primal becomes: $\min \|\mathbf{w}\|^2 + b^2 + C\sum_{i=1}^{m} \xi_i^2 \ : \ y_i(\mathbf{w}'\varphi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \ \hat{\mathbf{u}}_q'\mathbf{w} = \mathbf{u}_q'\boldsymbol{\Phi}\boldsymbol{\Phi}'\mathbf{w} = 0$. The corresponding dual is $\max 2\boldsymbol{\alpha}'\mathbf{1} - \boldsymbol{\alpha}'\hat{\mathbf{K}}\boldsymbol{\alpha} - 2\boldsymbol{\alpha}'\mathbf{Y}\boldsymbol{\Phi}'\hat{\mathbf{U}}\boldsymbol{\gamma} - \boldsymbol{\gamma}'\hat{\mathbf{U}}'\hat{\mathbf{U}}\boldsymbol{\gamma} \ : \ \boldsymbol{\alpha} \geq \mathbf{0}$, where $\hat{\mathbf{U}} = [\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_s]$. Using the same construction as in Section 3.2, this is also a MEB problem with multiple projection constraints on the center.

4 Experiments

4.1 Experimental Setup

Experiments are performed on a number of real-world data sets[1] (Table 1). All the different base classifier variants are run N_c times using the one-vs-all scheme (where N_c is the number of classes). The following base classifiers are compared: 1) Orthogonal SVM: SVM with orthogonality constraints with all previous SVM classifiers. This is the same as MMDA; 2) Orthogonal CVM: the proposed ensemble; 3) Bagged SVM (the base SVMs are trained by LIBSVM[2]).

As suggested in [3], a double-layer hierarchical combination scheme using nonlinear classifiers can have improved performance. In this experiment, we combine the base SVMs by the following classifiers: 1) SVM; 2) artificial neural network (ANN), with a single layer of 10 hidden units; 3) CVM; 4) Majority voting [3]. To demonstrate the usefulness of the extra orthogonal SVMs, we also compare with the standard SVM and ANN classifiers. The C parameter in (1) is always fixed at 1. We use the Gaussian kernel $\exp(-\|\mathbf{x}-\mathbf{z}\|^2/\beta)$, where $\beta = \frac{1}{m^2}\sum_{i,j=1}^{m} \|\mathbf{x}_i - \mathbf{x}_j\|^2$ is the average squared distance between patterns. Experiments are implemented in MATLAB (except for the bagged SVM which is in C++) and are performed on an AMD Athlon 4400+ PC with 4GB of RAM.

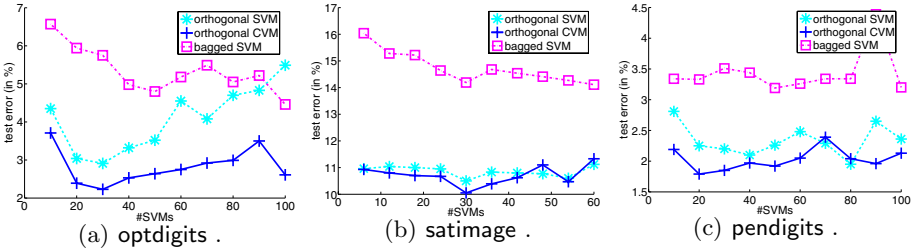
Table 1. Data sets used in the experiments

	optdigits	satimage	pendigits	letters	mnist	usps	face
# classes	10	6	10	26	10	2	2
# attributes	64	36	16	16	780	676	361
# training patterns	3,823	4,435	7,494	16,000	60,000	266,079	346,260
# testing patterns	1,797	2,000	3,498	4,000	10,000	75,383	24,045

The performance of ensemble methods depend critically on the number of base SVMs used, so we first perform some preliminary experiments on this. Figure 1 shows the results on the smaller data sets using the ANN as the final

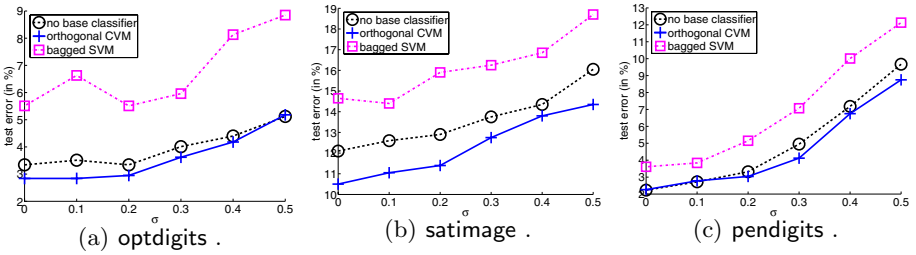
[1] The first five data sets are from the UCI machine learning repository, while the last two are from http://www.cs.ust.hk/∼ivor/cvm.html.

[2] http://www.csie.ntu.edu.tw/∼cjlin/libsvm/



(a) optdigits . (b) satimage . (c) pendigits .

Fig. 1. Testing error of the different SVM ensembles vs #SVMs



(a) optdigits . (b) satimage . (c) pendigits .

Fig. 2. Testing error of the different SVM ensembles at different noise levels

classifier. We observe that the performance of the bagged SVM first improves as more base SVMs are used, and then becomes more stable or even degraded. The performance using both the orthogonal CVM and SVM ensemble are better than the others when there are around $3N_c$ to $5N_c$ base SVMs. So, in the sequel, $N_c/3N_c/5N_c$ base SVMs are used.

4.2 Experimental Results

First, we show the proposed orthogonal CVM ensemble is more robust than the single SVM classifier and bagged SVMs. We run the orthogonal CVM ensemble and bagged SVM on the first three small data sets in Table 1. The input features are corrupted by zero-mean Gaussian noise at different noise levels (σ). For simplicity, we fix the number of base SVMs at $5N_c$, and the final classifier is a SVM. From Figure 2, we observe that the orthogonal CVM ensemble is more resistant to noise than the single SVM classifier and bagged SVMs.

As can be seen from Table 2, SVM ensembles can improve classification performance. In particular, nonlinear ensemble schemes using orthogonal SVMs outperform a single SVM. Moreover, the orthogonality constraints used in both the SVM and CVM base classifiers lead to lower testing errors than the bagged SVMs when using a few ($3N_c - 5N_c$) base SVMs.

As mentioned in Section 2, each base SVM can be expressed as a linear combination of kernel evaluations. Figure 3 shows the number of kernel evaluations involved in each base SVM. As can be seen, the CVM implementation produces SVMs that are sparser than the original one. As kernel evaluations are relatively

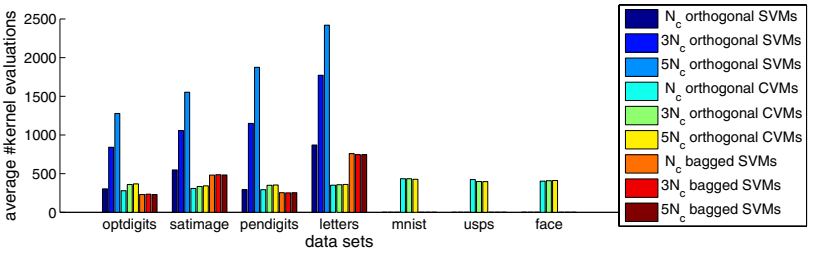


Fig. 3. Average number of kernel evaluations involved in each base SVM

Table 2. Testing errors on the various data sets

base classifier		final classifier	optdigits	satimage	pendigits	letters	mnist	usps	face
no base classifier		SVM	3.34	10.4	3.26	9.05	4.88	–	–
		ANN	5.63	12.6	4.81	29.05	9.61	0.88	2.6
orthogonal SVM	#base=N_c	SVM	3.07	10.55	2.18	7.15	–	–	–
	$3N_c$		3.07	11.45	2.15	6.05	–	–	–
	$5N_c$		3.07	10.35	2.09	5.98	–	–	–
	#base=N_c	ANN	4.35	11.05	2.81	19.8	–	–	–
	$3N_c$		2.91	10.35	2.2	17.98	–	–	–
	$5N_c$		3.52	11.3	2.26	17.35	–	–	–
	#base=N_c	CVM	3.07	10.55	2.18	7.1	–	–	–
	$3N_c$		3.07	10.45	2.12	6.05	–	–	–
	$5N_c$		3.07	10.35	2.09	5.78	–	–	–
orthogonal CVM	#base=N_c	SVM	2.84	10.25	2.29	**5.15**	5.46	–	–
	$3N_c$		2.9	10.5	2.26	5.4	4.27	–	–
	$5N_c$		2.95	10.7	2.21	5.65	**4.08**	–	–
	#base=N_c	ANN	3.73	10.25	2.78	19.03	7.01	0.7	1.72
	$3N_c$		**2.23**	10.7	2.15	17.55	6.72	0.67	**1.61**
	$5N_c$		2.64	**10.05**	**1.92**	17.33	6.66	**0.66**	1.66
	#base=N_c	CVM	2.84	10.25	2.29	5.18	5.46	0.69	1.9
	$3N_c$		2.84	10.5	2.26	5.43	4.28	0.67	1.66
	$5N_c$		2.95	10.7	2.21	5.7	4.09	0.7	1.65
bagged SVM	#base=N_c	SVM	6.35	16.0	3.98	29.6	–	–	–
	$3N_c$		6.07	14.8	3.61	25.8	–	–	–
	$5N_c$		5.63	14.85	3.72	24.83	–	–	–
	#base=N_c	ANN	6.57	16.04	3.34	30.22	–	–	–
	$3N_c$		5.75	15.22	3.51	26.36	–	–	–
	$5N_c$		4.8	15.19	3.19	25.21	–	–	–
	#base=N_c	CVM	6.4	15.75	3.66	29.7	–	–	–
	$3N_c$		5.79	14.6	3.44	25.95	–	–	–
	$5N_c$		5.51	14.65	4.61	25.25	–	–	–
	#base=N_c	voting	7.8	20.5	4.18	32.2	–	–	–
	$3N_c$		6.52	19.3	3.75	28.15	–	–	–
	$5N_c$		5.29	18.45	3.35	26.8	–	–	–

Table 3. CPU time (in seconds) required in the ensemble learning of base SVMs

base classifier	optdigits	satimage	pendigits	letters	mnist	usps	face
orthogonal SVM #base=N_c	84	121	127	1,911	–	–	–
$3N_c$	476	421	570	9,646	–	–	–
$5N_c$	1,495	900	1,674	20,860	–	–	–
orthogonal CVM #base=N_c	41	23	20	92	1,610	2,359	105
$3N_c$	181	78	95	301	4,928	6,585	337
$5N_c$	332	136	174	512	8,179	10,630	556

expensive, the orthogonal CVM is generally faster than the original implementation during testing.

Table 3 lists the CPU time needed in the ensemble learning of base SVMs. As can be seen, the proposed method is often faster than the original MMDA by one to two orders of magnitude. In particular, note that the bagged SVM and orthogonal SVM ensembles cannot finish training on the three largest data sets in 24 hours (indicated by "-" in the tables), while the proposed method obtain ensembles for the final classifier in usually less than several thousand seconds.

5 Conclusions

In this paper, we investigate ensemble learning in large scale classification tasks. The use of orthogonality constraints in the SVM ensemble leads to more robust performance than bagging. Moreover, the training time complexity depends only linearly on the training set size. In practice, it is 10-100 times faster than the original SVM ensemble. The proposed method produces sparser base SVMs and with better performance. It also involves fewer kernel evaluations. This in turn allows the combined classifier to be computed much faster during testing. In the future, we will investigate other different constraints on the SVM ensemble.

References

1. Tsang, I.W., Kwok, J.T., Cheung, P.M.: Core vector machines: Fast SVM training on very large data sets. Journal of Machine Learning Research **6** (2005) 363–392
2. Tsang, I.W., Kwok, J.T., Lai, K.T.: Core vector regression for very large regression problems. In: Proceedings of the Twentieth-Second International Conference on Machine Learning, Bonn, Germany (2005) 913–920
3. Kim, H.C., Pang, S., Je, H.M., Kim, D., Bang, S.: Constructing support vector machine ensemble. Pattern Recognition **36** (2003) 2757–2767
4. Valentini, G., Dietterich, T.: Bias-variance analysis of support vector machines for the development of SVM-based ensemble methods. Journal of Machine Learning Research **5** (2004) 725–775
5. Kivinen, J., Warmuth, M.K.: Boosting as entropy projection. In: Proceedings of the twelfth annual conference on Computational learning theory, Santa Cruz, California, United States (1999) 134 – 144

6. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning **51** (2003) 181–207
7. Kocsor, A., Kovács, K., Szepesvári, C.: Margin maximizing discriminant analysis. In: Proceedings of the 15th European Conference on Machine Learning, Pisa, Italy (2004) 227–238
8. Mangasarian, O., Musicant, D.: Lagrangian support vector machines. Journal of Machine Learning Research **1** (2001) 161–177
9. Kienzle, W., Schölkopf, B.: Training support vector machines with multiple equality constraints. In: Proceedings of the European Conference on Machine Learning. (2005)
10. Ye, J., Li, T., Xiong, T., Janardan, R.: Using uncorrelated discriminant analysis for tissue classification with gene expression d. IEEE/ACM Transactions on Computational Biology and Bioinformatics **1** (2004) 181–190