Tighter and Convex Maximum Margin Clustering

Yu-Feng Li[1] **Ivor W. Tsang**[2] **James T. Kwok**[3] **Zhi-Hua Zhou**[1]
[1] National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China
[2] School of Computer Engineering, Nanyang Technological University, Singapore 639798
[3] Department of Computer Science and Engineering, Hong Kong University of Science & Technology, Hong Kong
{liyf, zhouzh}@lamda.nju.edu.cn IvorTsang@ntu.edu.sg jamesk@cse.ust.hk

Abstract

Maximum margin principle has been successfully applied to many supervised and semi-supervised problems in machine learning. Recently, this principle was extended for clustering, referred to as Maximum Margin Clustering (MMC) and achieved promising performance in recent studies. To avoid the problem of local minima, MMC can be solved globally via convex semi-definite programming (SDP) relaxation. Although many efficient approaches have been proposed to alleviate the computational burden of SDP, convex MMCs are still not scalable for medium data sets. In this paper, we propose a novel convex optimization method, LG-MMC, which maximizes the margin of opposite clusters via "Label Generation". It can be shown that LG-MMC is much more scalable than existing convex approaches. Moreover, we show that our convex relaxation is tighter than state-of-art convex MMCs. Experiments on seventeen UCI datasets and MNIST dataset show significant improvement over existing MMC algorithms.

1 INTRODUCTION

Clustering is an important research area in machine learning, data mining and pattern recognition (Jain & Dubes, 1988). It aims at discovering the underlying structure (or concepts) of the data, and grouping similar instances. Clustering not only supplies valuable data analysis in practice, but is also widely used in various domains including information retrieval, computer version, bioinformatics and so on.

Over past decades, many clustering algorithms have been proposed such as k-means, spectral clustering and mixture model. Recently, inspired by the success of large margin criterion in support vector machine (SVM), Xu et al. (2005) proposed the use of maximum margin principle for clustering, referred to as Maximum Margin Clustering (MMC), which simultaneously learns the optimal hyperplane and cluster labels. However, the resultant optimization problem involves integer variables for cluster labels, and suffers from local minima. Xu et al. (2005) relaxed this optimization problem as a convex semi-definite programming (SDP) problem, in which a symmetric and real-valued label relation matrix approximates the outer-product of label vectors, can be solved globally (Boyd & Vandenberghe, 2004). Experimental results (Xu et al., 2005) show that MMC achieves the state-of-the-art performance in many clustering problems. Moreover, their formulation can be naturally extended to semi-supervised learning (Xu & Schuurmans, 2005).

Unlike quadratic programming (QP) used in kernel methods, the worst-case time complexity of the SDP used in MMC ($O(n^{6.5})$) is much higher than QP solvers ($O(n^3)$) (Boyd & Vandenberghe, 2004; Zhang et al., 2009), where n is the data set size. Although generalized maximum margin clustering (GMMC) (Valizadegan & Jin, 2007) reduces the variables from $O(n^2)$ to $O(n)$ and has speeded-up MMC by 100 times, GMMC still cannot handle medium datasets with more than one thousand instances. Recently, several researchers proposed efficient algorithms for MMC at the cost of losing the convexity. For instances, Zhang et al. (2007); Zhang et al. (2009) proposed an alternative descend method which preformed clustering by sequentially solving various support vector regression (SVR) problems; Zhao et al. (2008) proposed an efficient cutting-plane method for linear MMC by using a series of constrained convex-concave procedure (CCCP)

Appearing in Proceedings of the 12^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2009, Clearwater Beach, Florida, USA. Volume 5 of JMLR: W&CP 5.

to relax the nonconvex constraint. All these methods are non-convex optimization strategies which still get stuck in local minima. Therefore, it is desirable to investigate a scalable algorithm that still achieves the globally optimal solution for MMC.

In this paper, we perform MMC via a so-called "label generation" strategy, referred to as Label-Generating MMC (LG-MMC). Instead of solving the label relation matrix in MMC via SDP (Xu et al., 2005), LG-MMC maximizes the margin by generating the most violated label vectors iteratively, and then combines them via efficient multiple kernel learning. The overall procedure can be formulated as a relaxed convex optimization of MMC problem. Furthermore, it can be shown that the learned linear combination of the outer-product of the label vectors is in a convex-hull of the label space. It is known that convex-hull is the smallest convex set that contains certain non-convex set (Boyd & Vandenberghe, 2004), and thus our formulation achieves a tighter relaxation than the convex relaxation of MMC proposed by Xu et al. (2005). Moreover, LG-MMC involves a series of SVM subproblems which can be solved in a scalable and efficient manner via various state-of-the-art SVM softwares such as SVM-*perf* (Joachims, 2006), LIBLINEAR (Hsieh et al., 2008) and CVM (Tsang et al., 2006), and LG-MMC scales much better than existing convex approaches.

The rest of this paper is organized as follows. Section 2 briefly introduces maximum margin clustering. Section 3 describes the propose LG-MMC algorithm. Experimental results are shown in Section 4. The last section gives the conclusive remarks.

2 MAXIMAL MARGIN CLUSTERING

In the sequel, $\mathbf{M} \succ 0$ (resp. $\mathbf{M} \succeq 0$) means that the matrix $\mathbf{M}$ is symmetric and positive definite (pd) (resp. positive semidefinite (psd)). Moreover, the transpose of vector / matrix (in both the input and feature spaces) will be denoted by the superscript $'$, and $\mathbf{0}, \mathbf{1} \in \mathbb{R}^n$ denote the zero vector and the vector of all ones, respectively. The inequality $\mathbf{v} = [v_1, \cdots, v_k]^\top \geq \mathbf{0}$ means that $v_i \geq 0$ for $i = 1, \cdots, k$.

First, we start with a simpler scenario of supervised learning. Given a set of labeled patterns $\{\mathbf{x}_i, y_i\}_{i=1}^n$ where $\mathbf{x}_i \in \mathcal{X}$ is the input and $\hat{y}_i \in \{\pm 1\}$ is the output, and consider finding a decision function $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x})$ that minimizes the structural risk functional:

$$\Omega(\|\mathbf{w}\|_p) + C\sum_{i=1}^n \ell(-y_i \mathbf{w}'\phi(\mathbf{x}_i)),$$

where ϕ is the feature map induced by some kernel function k, Ω is a strictly monotonic increasing function, $\ell(\cdot)$ is a monotonically increasing loss function, and C is a regularization parameter that trades off the empirical risk and the model complexity. In this paper, we focus on $\Omega(\|\mathbf{w}\|_p) = \frac{1}{2}||\mathbf{w}||^2$ and the squared hinge loss:

$$\begin{aligned} \min_{\mathbf{w},\rho,\boldsymbol{\xi}} \quad & \frac{1}{2}||\mathbf{w}||_2^2 - \rho + \frac{C}{2}\sum_{i=1}^n \xi_i^2 \\ \text{s.t} \quad & y_i \mathbf{w}'\phi(\mathbf{x}_i) \geq \rho - \xi_i, \;\; i = 1 \cdots, n. \end{aligned} \tag{1}$$

This is usually solved in its dual:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\frac{1}{2}\sum_{i,j=1}^n \alpha_i \alpha_j \left(y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{C}\delta_{ij} \right) \\ \text{s.t.} \quad & \alpha_i \geq 0, \sum_{i=1}^n \alpha_i = 1, \end{aligned} \tag{2}$$

where α_i is a dual variable for each inequality constraint in (1), and δ_{ij} is the indicator function (i.e., $\delta_{ij} = 1$ if $i = j$; and 0 otherwise). Let $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_n]'$ be the vector of dual variables, and $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)] \in \mathbb{R}^{n \times n}$ be the kernel matrix, and $\mathcal{A} = \{\boldsymbol{\alpha} \mid \boldsymbol{\alpha} \geq \mathbf{0}, \boldsymbol{\alpha}'\mathbf{1} = 1\}$. Then the QP in (2) can be rewritten in matrix form as:

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} \quad -\frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K} \odot \mathbf{y}\mathbf{y}' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}, \tag{3}$$

where $\odot$ denotes the element-wise product. Finally, the decision function can be obtained from the optimal $\boldsymbol{\alpha}$ as $f(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i k(\mathbf{x}_i, \mathbf{x})$.

In maximum margin clustering, the pattern labels are unknown and so also need to be optimized. However, note that one can obtain a trivially "optimal" solution with the infinite margin by assigning all patterns to a single cluster. To prevent such a useless solution, Xu et al. (2005) introduced a class balance constraint

$$-\beta \leq \mathbf{1}'\hat{\mathbf{y}} \leq \beta,$$

where $\hat{\mathbf{y}} = [\hat{y}_1, \cdots, \hat{y}_n]'$ denotes a vector of the unknown pattern labels, and $\beta \geq 0$ is a user-defined constant controlling the class imbalance. The margin is then maximized w.r.t. both the unknown $\hat{\mathbf{y}}$ and the unknown SVM parameter, and (1) is extended to:

$$\begin{aligned} \min_{\hat{\mathbf{y}} \in \mathcal{B}} \min_{\mathbf{w},\rho,\boldsymbol{\xi}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 - \rho + \frac{C}{2}\sum_{i=1}^n \xi_i^2 \\ \text{s.t} \quad & \hat{y}_i \mathbf{w}'\phi(\mathbf{x}_i) \geq \rho - \xi_i, \;\; i = 1 \cdots, n, \end{aligned}$$

where $\mathcal{B} = \{\hat{\mathbf{y}} \mid \hat{y}_i \in \{\pm 1\}, -\beta \leq \mathbf{1}'\hat{\mathbf{y}} \leq \beta\}$. Consequently, (3) becomes:

$$\min_{\hat{\mathbf{y}} \in \mathcal{B}} \max_{\boldsymbol{\alpha} \in \mathcal{A}} \quad -\frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K} \odot \hat{\mathbf{y}}\hat{\mathbf{y}}' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}. \tag{4}$$

This, however, is a mixed integer program.

To make it more tractable, Xu et al. (2005) proposed the MMC algorithm that relaxes the rank-one matrix $\mathbf{M} = \hat{\mathbf{y}}\hat{\mathbf{y}}'$ to a positive semi-definite matrix satisfying $\text{diag}(\mathbf{M}) = \mathbf{1}$. This finally leads to a convex semidefinite program (SDP), which can also be extended to the multi-class setting (Xu & Schuurmans, 2005). However, this involves $O(n^2)$ optimization variables. Recently, Valizadegan and Jin (2007) proposed the general maximum margin clustering (GMMC) algorithm which reduces the number of variables to $O(n)$, and thus results in significantly computational savings. However, because all these are still based on SDPs, MMC and GMMC are limited to small-to-medium data sets.

Most recently, two maximum margin clustering approaches are introduced that are not based on expensive SDP formulations. Zhang et al. (2007) proposed an efficient approach based on iterative kernel regression procedures. Zhao et al. (2008) proposed the cutting plane maximum margin clustering (CPMMC) algorithm which is based on the use of cutting planes (Kelley, 1960) and constrained concave-convex procedure (Smola et al., 2005). However, both methods sacrifice convexity for efficiency. While each iteration only involves the solving of a convex optimization problem, the optimization problem as a whole is still non-convex and so suffers from the problem of local minimum.

3 LG-MMC

In this section, we introduce the proposed Label-Generating MMC (LG-MMC) algorithm.

3.1 THE APPROACH

First, consider interchanging the order of $\max_{\boldsymbol{\alpha}\in\mathcal{A}}$ and $\min_{\hat{\mathbf{y}}\in\mathcal{B}}$ in (4), leading to:

$$\text{LG-MMC}: \quad \max_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\hat{\mathbf{y}}\in\mathcal{B}} -\frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K}\odot\hat{\mathbf{y}}\hat{\mathbf{y}}' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}. \quad (5)$$

According to the minmax theorem (Kim & Boyd, 2008), the optimal objective of (4) is an upper bound of that of (5). This can be further rewritten as:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}}\Big\{ \max_{\theta} -\theta \quad (6)$$
$$\text{s.t. } \theta \geq \frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha},\ \forall\hat{\mathbf{y}}_t\in\mathcal{B}\Big\}.$$

For the inner optimization subproblem, let $\mu_t \geq 0$ be the dual variable for each constraint. Its Lagrangian can be obtained as:

$$-\theta + \sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}} \mu_t\Big(\theta - \frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}\Big).$$

Setting its derivative w.r.t. θ to zero, we have $\sum\mu_t = 1$. Let $\boldsymbol{\mu}$ be the vector of μ_t's, and $\mathcal{M}$ be the simplex $\{\boldsymbol{\mu} \mid \sum\mu_t = 1, \mu_t \geq 0\}$. We can then replace the inner optimization subproblem with its dual and (6) becomes:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}}\min_{\boldsymbol{\mu}\in\mathcal{M}} -\frac{1}{2}\boldsymbol{\alpha}'\Big(\sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}}\mu_t\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}$$
$$= \min_{\boldsymbol{\mu}\in\mathcal{M}}\max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}'\Big(\sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}}\mu_t\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t' + \frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}. \quad (7)$$

Here, we have used the fact that the objective function is concave in $\boldsymbol{\alpha}$ and convex in $\boldsymbol{\mu}$. Moreover, note the similarity with (3), which involves a single kernel matrix $\mathbf{K}\odot\mathbf{y}\mathbf{y}'$. Hence, (7) can be regarded as multiple kernel learning (MKL) (Lanckriet et al., 2004), where the target kernel matrix is a convex combination of $|\mathcal{B}|$ base kernel matrices $\{\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t'\}$, each of which is constructed from a feasible label vector $\hat{\mathbf{y}}_t \in \mathcal{B}$.

Details on how to solve this MKL problem will be described in Sections 3.3 – 3.5. After obtaining $\boldsymbol{\mu}$, the cluster labels can be recovered by eign-decomposing $\mathbf{M} = \sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}}\mu_t\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t'$ (as in (Valizadegan & Jin, 2007; Xu et al., 2005)). In the following, we first show that the formulation in (7) gives a tighter relaxation of the maximum margin clustering problem than MMC.

3.2 TIGHTER RELAXATION

Consider the set $\mathcal{Y}_0$ of all feasible label matrices

$$\mathcal{Y}_0 = \big\{\mathbf{M} \mid \mathbf{M} = \hat{\mathbf{y}}\hat{\mathbf{y}}',\ \forall\hat{\mathbf{y}}\in\mathcal{B}\big\},$$

and the two relaxations

$$\mathcal{Y}_1 = \Big\{\mathbf{M} \mid \mathbf{M} = \sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}}\mu_t\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t',\ \boldsymbol{\mu}\in\mathcal{M}\Big\},$$
$$\mathcal{Y}_2 = \big\{\mathbf{M} \mid \mathbf{M}\succeq 0, \text{diag}(\mathbf{M}) = \mathbf{1}\big\}.$$

Define

$$F(\boldsymbol{\alpha},\mathbf{M}) = -\frac{1}{2C}\boldsymbol{\alpha}'\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K}\odot\mathbf{M}\Big)\boldsymbol{\alpha},$$

then, obviously, the original mixed-integer programming formulation of maximum margin clustering in (4) is the same as

$$\min_{\mathbf{M}\in\mathcal{Y}_0}\max_{\boldsymbol{\alpha}\in\mathcal{A}} F(\boldsymbol{\alpha},\mathbf{M}). \quad (8)$$

Similarly, the proposed formulation in (7) can be written as:

$$\min_{\mathbf{M}\in\mathcal{Y}_1}\max_{\boldsymbol{\alpha}\in\mathcal{A}} F(\boldsymbol{\alpha},\mathbf{M}), \quad (9)$$

and MMC (Xu et al., 2005) as:

$$\min_{\mathbf{M}\in\mathcal{Y}_2}\max_{\boldsymbol{\alpha}\in\mathcal{A}} F(\boldsymbol{\alpha},\mathbf{M}),$$

which are all relaxations of (8).

Note that $\mathcal{Y}_1$ is the convex hull of $\mathcal{Y}_0$, which is the smallest convex set containing $\mathcal{Y}_0$ (Boyd & Vandenberghe, 2004). Therefore, (9) gives the tightest convex relaxation of (8). Moreover, the following proposition shows that $\mathcal{Y}_2$ is more relaxed than $\mathcal{Y}_1$. In other words, MMC is a looser relaxation than the proposed formulation.

Proposition 1. $\mathcal{Y}_1 \subset \mathcal{Y}_2$.

Proof. Let $\mathcal{Y}$ be the set of all binary $\mathbf{y}$'s ($y_i \in \{\pm 1\}$), and

$$\mathcal{Y}_1^+ = \{\mathbf{M} \mid \mathbf{M} = \sum_{\mathbf{y}\in\mathcal{Y}} \mu_{\mathbf{y}} \mathbf{y}\mathbf{y}', \ \ \boldsymbol{\mu} \in \mathcal{M}\}.$$

As $\mathcal{B} \subseteq \mathcal{Y}$, so $\mathcal{Y}_1 \subseteq \mathcal{Y}_1^+$. In the following, we first show that $\mathbf{1}\mathbf{1}' \notin \mathcal{Y}_1$. Assume to the contrary that $\mathbf{1}\mathbf{1}' \in \mathcal{Y}_1$. Then there exists a convex combination $\boldsymbol{\mu} = [\mu_{\mathbf{y}}]$ such that $\mathbf{1}\mathbf{1}' = \sum_{\mathbf{y}\in\mathcal{B}} \mu_{\mathbf{y}} \mathbf{y}\mathbf{y}'$. Note that each entry of $\mathbf{y}\mathbf{y}'$ is in $\{\pm 1\}$. However, if a particular entry of $\mathbf{y}\mathbf{y}'$ (say, $[\mathbf{y}\mathbf{y}']_{ij}$) equals -1, $\mu_{\mathbf{y}}$ must be 0 or else the convex combination $\sum_{\mathbf{y}\in\mathcal{B}} \mu_{\mathbf{y}} [\mathbf{y}\mathbf{y}']_{ij} < 1 \neq [\mathbf{1}\mathbf{1}']_{ij}$. Thus, $\mathbf{y}$'s with positive $\mu_{\mathbf{y}}$'s must be either $\mathbf{1}$ or $-\mathbf{1}$. However, both $\mathbf{1}$ and $-\mathbf{1}$ do not belong to $\mathcal{Y}_1$ because of the balance constraint. Hence, a contradiction and $\mathbf{1}\mathbf{1}' \notin \mathcal{Y}_1$. Also, it is obvious that $\mathbf{1}\mathbf{1}' \in \mathcal{Y}_1^+$ and thus $\mathcal{Y}_1 \subset \mathcal{Y}_1^+$. Finally, it is easy to see that $\mathcal{Y}_1^+ \subseteq \mathcal{Y}_2$. Thus, we have $\mathcal{Y}_1 \subset \mathcal{Y}_1^+ \subseteq \mathcal{Y}_2$. □

3.3 CUTTING PLANE

Recall that (7) can be regarded as a MKL problem. Hence, it appears that existing MKL techniques (Lanckriet et al., 2004; Bach et al., 2004; Rakotomamonjy et al., 2007; Rakotomamonjy et al., 2008; Sonnenburg et al., 2006) can be readily used to solve this problem. However, because of the exponential number of possible labelings $\hat{\mathbf{y}}_t \in \mathcal{B}$, the set of base kernels is also exponential in size and so direct MKL is computationally intractable.

Fortunately, not all the constraints in (6) are active at optimality, and including only a subset of these constraints can usually lead to a very good approximation of the original optimization problem. Therefore, we can apply the cutting-plane method (Kelley, 1960) to handle this exponential number of constraints.

Denote the subset of constraints by $\mathcal{C} \subset \mathcal{B}$. The cutting-plane algorithm is described in Algorithm 1. First, we initialize the vector of Lagrangian multipliers $\boldsymbol{\alpha}$ to $\frac{1}{n}\mathbf{1}$, and set the working set $\mathcal{C} = \{\hat{\mathbf{y}}, -\hat{\mathbf{y}}\}$ so that we have two base kernels to start with. Since the working set $\mathcal{C} \subset \mathcal{B}$ (and thus the number of base kernel matrices is no longer exponential in size), one can perform MKL and obtain $\boldsymbol{\alpha}$ from (7). The most violated label vector $\hat{\mathbf{y}}$ is added to $\mathcal{C}$, and the process repeated until the termination criterion is met. The whole algorithm is summarized in Algorithm 1.

Algorithm 1 Cutting plane algorithm for LG-MMC.

1: Initialize $\boldsymbol{\alpha} = \frac{1}{n}\mathbf{1}$. Find the most violated $\hat{\mathbf{y}}$ and set $\mathcal{C} = \{\hat{\mathbf{y}}, -\hat{\mathbf{y}}\}$
2: Run MKL for the subset of kernel matrices selected in $\mathcal{C}$ and obtain $\boldsymbol{\alpha}$ from (7).
3: Find the most violated $\hat{\mathbf{y}}$ and set $\mathcal{C} = \hat{\mathbf{y}} \bigcup \mathcal{C}$.
4: Repeat steps 2-3 until convergence.

There are two important issues in the cutting plane algorithm. First, how to efficiently solve the MKL optimization problem? Second, how to efficiently find the most violated $\hat{\mathbf{y}}$. These will be addressed in Sections 3.4 and 3.5, respectively.

3.4 MULTIPLE LABEL-KERNEL LEARNING

Several efficient MKL approaches have been developed in recent years. For instance, Lanckriet et al. (2004) first proposed the use of quadratically constrained quadratic programming (QCQP) in MKL. Later, Bach et al. (2004) showed that an approximate solution can be efficiently obtained by using sequential minimization optimization (SMO) (Platt, 1999). Recently, Sonnenburg et al. (2006) proposed a semi-infinite linear programming (SILP) formulation which allows MKL to be iteratively solved with standard SVM solver and linear programming. Rakotomamonjy et al. (2007) and Rakotomamonjy et al. (2008) proposed the related SimpleMKL algorithm. Most recently, Xu et al. (2009) proposed the use of the extended level method to further improve the convergence of MKL.

Unlike standard MKL problems which try to find the optimal kernel function/matrix for a given set of labels, here, we have to find the optimal label kernel matrix. In this paper, we use an adaptation of the SimpleMKL algorithm (Rakotomamonjy et al., 2007; Rakotomamonjy et al., 2008) to solve this multiple label-kernel learning (MLKL) problem. More specifically, suppose that the current working set is $\mathcal{C} = \{\hat{\mathbf{y}}_1, \cdots, \hat{\mathbf{y}}_T\}$. Note that the feature map corresponding to the base kernel matrix $\mathbf{K} \odot \hat{\mathbf{y}}_t \hat{\mathbf{y}}_t'$ is $\hat{y}_{ti}\phi(\mathbf{x}_i)$. The MKL problem in (7) thus corresponds to the following primal optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\mu}\in\mathcal{M},\mathbf{w},\rho,\boldsymbol{\xi}} \quad & \frac{1}{2}\sum_{t=1}^{T}\frac{1}{\mu_t}||\mathbf{w}_t||^2 - \rho + \frac{C}{2}\sum_{i=1}^{n}\xi_i^2 \qquad (10) \\ \text{s.t.} \quad & \sum_{t=1}^{T}\hat{y}_{ti}\mathbf{w}_t'\phi(\mathbf{x}_i) \geq \rho - \xi_i, \forall i = 1, \cdots, n. \end{aligned}$$

It is easy to verify that the dual can be written as

$$\begin{aligned} \max_{\boldsymbol{\alpha}\in\mathcal{A},\theta} \quad & -\theta \\ \text{s.t.} \quad & \theta \geq \frac{1}{2}\boldsymbol{\alpha}'\Big(\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t'+\frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}, \;\; t=1,\ldots,T, \end{aligned}$$

which is the same as (6). Following SimpleMKL, we solve (7) (or, equivalently, (10)) iteratively. First, we fix the mixing coefficients $\boldsymbol{\mu}$ of the base kernel matrices and solve the SVM's dual:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} \quad -\tfrac{1}{2}\boldsymbol{\alpha}'\Big(\textstyle\sum_{t=1}^{T}\mu_t\mathbf{K}\odot\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t'+\frac{1}{C}\mathbf{I}\Big)\boldsymbol{\alpha}.$$

Then, we fix $\boldsymbol{\alpha}$ and use the reduced gradient method for updating $\boldsymbol{\mu}$. These two steps are iterated until convergence.

3.5 FING THE MOST VIOLATED $\hat{\mathbf{y}}$

To find the most violated $\hat{\mathbf{y}}$ in (6), we have to solve the following equivalent optimization problem

$$\max_{\hat{\mathbf{y}}\in\mathcal{B}} \textstyle\sum_{i,j=1}^{n}\alpha_i\alpha_j\hat{y}_i\hat{y}_j\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j). \tag{11}$$

However, this is a concave QP and so cannot be solved efficiently. Note that while the use of the most violated constraint may lead to faster convergence, the cutting plane algorithm only requires the addition of a violated constraint at each iteration. Hence, we propose in the following a simple and efficient method for finding a good approximation of the most violated $\hat{\mathbf{y}}$.

First, note that maximizing (11) is the same as maximizing its square root:

$$\max_{\hat{\mathbf{y}}\in\mathcal{B}} \quad \Big\|\textstyle\sum_{i=1}^{n}\alpha_i\hat{y}_i\phi(\mathbf{x}_i)\Big\|. \tag{12}$$

The key idea is to replace the ℓ_2 norm above with the infinity-norm. For simplicity of notation, let[1] $\phi(\mathbf{x}) = [x^{(1)}, x^{(2)}, \cdots, x^{(d)}]'$, where d is the dimensionality of $\phi(\mathbf{x})$. Then, (12) is replaced by

$$\begin{aligned} &\max_{\hat{\mathbf{y}}\in\mathcal{B}} \Big\|\textstyle\sum_{i=1}^{n}\alpha_i y_i\phi(\mathbf{x}_i)\Big\|_\infty \\ &= \max_{\hat{\mathbf{y}}\in\mathcal{B}}\left(\max_{j=1,\cdots,d}\left|\sum_{i=1}^{n}\alpha_i y_i x_i^{(j)}\right|\right) \\ &= \max_{j=1,\cdots,d}\left(\max_{\hat{\mathbf{y}}\in\mathcal{B}}\left|\sum_{i=1}^{n}\alpha_i y_i x_i^{(j)}\right|\right). \end{aligned} \tag{13}$$

The absolute sign can be removed by writing each inner subproblem as

$$\max\left(\max_{\hat{\mathbf{y}}\in\mathcal{B}}\sum_{i=1}^{n}\alpha_i y_i x_i^{(j)}, \max_{\hat{\mathbf{y}}\in\mathcal{B}} -\sum_{i=1}^{n}\alpha_i\hat{y}_i x_i^{(j)}\right). \tag{14}$$

[1] If $\phi(\mathbf{x})$ has infinite dimensions, we perform singular value decomposition for kernel matrix to get $[x^{(1)}, x^{(2)}, \cdots, x^{(d)}]$.

Each of these LP subproblems is of the form

$$\max_{\hat{\mathbf{y}}} \; \mathbf{c}'\hat{\mathbf{y}} \;:\; \hat{y}_i\in\{\pm1\}, -\beta\leq\mathbf{1}'\hat{\mathbf{y}}\leq\beta. \tag{15}$$

Moreover, it can be solved without any numeric optimization solver, as is shown by the following proposition.

Proposition 2. *At optimality, $\hat{y}_i \geq \hat{y}_j$ if $c_i > c_j$.*

Proof. Assume, to the contrary, that the optimal $\hat{\mathbf{y}}$ does not have the same sorted order as $\mathbf{c}$. Then, there are two label vectors $\hat{y}_i$ and $\hat{y}_j$, with $c_i > c_j$ but $\hat{y}_i < \hat{y}_j$. Then $c_i\hat{y}_i + c_j\hat{y}_j < c_i\hat{y}_j + c_j\hat{y}_i$ as $(c_i - c_j)(\hat{y}_i - \hat{y}_j) < 0$. Thus, $\hat{y}$ is not optimal, a contradiction. □

Thus, with Proposition 2, we can solve (15) by first sorting c_i's. The label assignment of $\hat{y}_i$'s aligns the sorted values of c_i's. To satisfy the balance constraint $-\beta\leq\mathbf{1}'\hat{\mathbf{y}}\leq\beta$, the first $\frac{n-\beta}{2}$ of $\hat{y}_i$'s are assigned with -1, the last $\frac{n-\beta}{2}$ of them are assigned with 1. The rest are assigned from -1 to 1 such that the objective $\mathbf{c}'\hat{\mathbf{y}}$ is maximized. Therefore, the label assignment in the integer problem (14) can be determined exactly and efficiently by sorting. After solving (14) for each $j = 1, \cdots, d$, these are then put back into (13) and the label vector $\hat{\mathbf{y}}$ corresponding to the maximum is added to the working set $\mathcal{C}$.

3.6 COMPUTATIONAL COMPLEXITY

The time complexities of MMC and GMMC are $O(n^{6.5})$ and $O(n^{4.5})$, respectively (Zhang et al., 2007). These can be expensive even on medium-sized data sets. In contrast, LG-MMC involves solving a sequence of MKL problems and the finding of violated label assignments. Empirically, a maximum of five iterations is good enough for MKL to converge. So the time complexity of MKL is proportional to the complexity of the SVM solver, which usually scales between $O(n)$ and $O(n^{2.3})$ (Platt, 1999). Moreover, computing the gradient in each SimpleMKL iteration takes $O(n^2)$ time. Besides, finding the violated label assignment as described in Section 3.5 takes $O(dn\log n)$ time as sorting n patterns in d dimensions. As will be shown in Section 4, empirically, the number of cutting-plane iterations required is usually small (no more than 20). Finally, as $\mathbf{M} = \sum_{t:\hat{\mathbf{y}}_t\in\mathcal{B}}\mu_t\hat{\mathbf{y}}_t\hat{\mathbf{y}}_t' = \mathbf{Y}\mathrm{diag}(\mu)\mathbf{Y}' = (\mathbf{Y}\mathrm{diag}(\mu)^{0.5})(\mathbf{Y}\mathrm{diag}(\mu)^{0.5})'$ where $\mathbf{Y} = [\hat{\mathbf{y}}_1\cdots\hat{\mathbf{y}}_T]$, the cluster labels can be recovered from the obtained $\mathbf{Y}\mathrm{diag}(\mu)^{0.5}$ by singular value eigen-decomposition, which takes $O(nT^2)$ time. Thus, LG-MMC is computationally efficient.

Table 1: Data sets used in the experiments.

ID	Data	# instances	# Features
1	*Echocardiogram*	132	8
2	*Heart-stalog*	270	13
3	*Haberman*	306	14
4	*LiveDiscorders*	345	6
5	*Spectf*	349	44
6	*Ionosphere*	351	34
7	*House-votes*	435	16
8	*Clean1*	476	166
9	*Breast*	683	9
10	*Australian*	690	42
11	*Diabetes*	768	8
12	*German*	1000	59
13	*LetterAvsB*	1555	16
14	*Satellite1vs2*	2236	36
15	*Krvskp*	3196	36
16	*Sick*	3772	31
17	*Spambase*	4601	57
18	*MNIST3vs8*	13966	784
19	*MNIST1vs7*	15170	784

4 EXPERIMENTS

In this Section, we evaluate LG-MMC using a collection of real-world data sets. Comprehensive evaluation of the clustering performance are performed on 17 UCI data sets, which cover a wide range of properties, and the *MNIST* data set [2] which contains 70,000 instances at all. Information of these data sets are summarized in Table 1. For the *Letter* and *Satellite* data sets, there are multiple classes and we use the first two classes only (Zhang et al., 2007). For the *MNIST* data set, we focus on the most difficult pairs (3 vs 8 and 1 vs 7). Experiments are performed with MATLAB 7.6 on a 2.00GHZ Inter Xeon(R)2 DUO PC running Windows XP with 4GB main memory.

4.1 COMPARED METHODS

We compare LG-MMC with following methods: 1) k-means (KM) method; 2) normalized cut (NC) method (Shi & Malik, 2000); 3) GMMC (Valizadegan & Jin, 2007); 4) IterSVR [3] (Zhang et al., 2007); 5) CPMMC [4] (Zhao et al., 2008).

The C parameter is selected in a range of $\{0.1, 0.5, 1, 5, 10, 100\}$ for GMMC, IterSVR, CPMMC and our LG-MMC method. For GMMC, IterSVR CPMMC and LG-MMC, both linear and Gaussian kernels are used. In particular, the width σ of the Gaussian kernel $\exp(-||z||^2/2\sigma^2)$ is picked via $\{0.25\sqrt{\gamma}, 0.5\sqrt{\gamma}, \sqrt{\gamma}, 2\sqrt{\gamma}, 4\sqrt{\gamma}\}$ where γ is the average distance from all pairs of instances. The parameter of normalized cut is picked up from the same range of σ. Since k-means and IterSVR are susceptible to the problem of local minima, these two methods are run 10 times and evaluate the average performance. For the *MNIST* data set, the linear kernel is used for all SVM-type methods (IterSVR, CPMMC and LG-MMC). We use the same setup as (Zhang et al., 2007) for the balance constraint, i.e., β is set as $0.03n$ for balanced data and $0.3n$ for imbalanced data. All the methods are reported with the best parameter setting.

[2] http://yann.lecun.com/exdb/mnist/
[3] http://www.cse.ust.hk/∼twinsen
[4] http://binzhao02.googlepages.com/

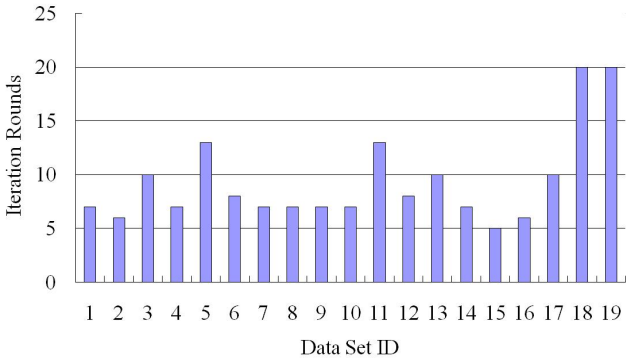

Figure 1: Number of Iterations Require by LG-MMC.

4.2 CLUSTERING ACCURACY

Here, we follow the strategy in (Xu et al., 2005) to evaluate the clustering accuracy. We first remove the labels for all instances, and then predict the clusters via performing clustering algorithms, finally measure the misclassification error according to true label. Results are shown in Table 2. The best performance is listed in bold. The "N/A" in Table 2 indicates that no result can be obtained in reasonable time (5 hours) or numerical problem occurs. As shown in Table 2, LG-MMC achieves improved clustering performance over existing MMC approaches on most data sets; while other MMC approaches are comparable, and are often better than KM.

4.3 SPEED

In this section, we also evaluate the efficiency of different clustering algorithms. Detailed CPU time for all methods are shown in Table 3. As can be seen, LG-MMC scales much better than the GMMC method. On average, LG-MMC is about 10 times faster than GMMC. In general, global optimization methods are still slower than local optimization methods. Figure 1 shows the number of iterations of LG-MMC with the best performance. We can observe that the number of iterations is always no more than 20. Therefore, LG-MMC usually converges in very few iterations.

Table 2: Clustering Error (%) on Various Data Sets

Data	KM	NC	GMMC	IterSVR	CPMMC	LG-MMC
Echocardiogram	24.24	23.48	18.94	21.97	**18.18**	**18.18**
Heart-stalog	25.44	**20.74**	22.59	22.70	44.44	24.81
Haberman	40.03	30.39	38.24	31.44	26.47	**25.82**
LiveDiscorders	45.31	43.19	41.16	43.01	42.03	**34.49**
Spectf	42.35	36.68	**21.20**	34.23	27.22	25.70
Ionosphere	28.77	29.63	**9.77**	23.70	35.90	24.70
House-votes	13.33	14.02	33.56	13.33	38.62	**12.87**
Clean1	45.08	47.48	39.71	40.29	43.49	**31.72**
Breast	3.95	**2.49**	N/A	3.22	34.99	3.51
Australian	45.57	45.22	39.71	32.12	44.49	**14.06**
Diabetes	33.42	34.24	**31.12**	31.51	34.90	32.68
German	43.28	33.60	30.00	34.86	30.00	**29.80**
LetterAvsB	18.24	5.79	N/A	6.63	30.00	**0.00**
Satellite1vs2	4.07	1.80	N/A	3.18	30.00	**0.76**
Krvskp	47.25	43.96	N/A	46.05	47.78	**39.64**
Sick	29.44	15.51	N/A	19.34	**6.12**	**6.12**
Spambase	25.44	41.51	N/A	23.18	39.40	**18.30**
MNIST3vs8	20.04	20.18	N/A	26.65	23.40	**18.12**
MNIST1vs7	4.23	3.52	N/A	4.54	4.24	**2.30**

Table 3: Wall Clock Time (in Seconds)

Data	KM	NC	GMMC	IterSVR	CPMMC	LG-MMC
Echocardiogram	0.01	0.09	9.26	0.05	0.06	0.52
Heart-stalog	0.01	0.41	27.0	0.12	0.06	1.36
Haberman	0.01	0.28	32.13	0.10	0.06	4.99
LiveDiscorders	0.01	0.42	42.01	0.25	0.06	4.58
Spectf	0.01	0.44	44.73	0.28	0.06	4.98
Ionosphere	0.01	0.53	13.63	0.18	0.06	1.35
House-votes	0.00	0.71	69.13	0.18	0.06	9.06
Clean1	0.02	1.25	40.81	1.14	0.03	6.44
Breast	0.01	1.66	N/A	0.47	0.03	14.09
Australian	0.01	2.17	353.8	1.24	0.06	25.23
Diabetes	0.01	2.30	326.98	0.23	0.03	83.44
German	0.02	6.26	623.20	2.13	0.04	34.44
LetterAvsB	0.01	25.70	N/A	2.80	0.03	4.81
Satellite1vs2	0.03	70.98	N/A	27.20	0.06	97.69
Krvskp	0.03	150.36	N/A	27.12	0.09	149.42
Sick	0.03	162.89	N/A	19.83	0.09	199.46
Spambase	0.17	497.84	N/A	26.02	0.16	1460.75
Mnist3vs8	17.33	13652.32	N/A	15596.21	106.78	731.22
Mnist1vs7	8.38	2596.21	N/A	2612.81	51.13	566.44

5 CONCLUSION

In this paper, we propose a scalable and global optimization method for maximum margin clustering by maximizing the margin via generating label assignments iteratively. We show that our approach achieves the tighter convex relaxation than the convex relaxation of MMC(Xu et al., 2005). To implement this idea, we employ multiple kernel learning method to efficiently solve the multiple label kernel combination problem. Besides, we propose a simple but effective way to generate new label assignment which will approximatively maximize the margin of opposite clusters. Experimental results in various data sets show that our approach achieves promising clustering performance and scales much better than existing global methods. In future, we will extend the proposed approach for semi-supervised learning.

Acknowledgements

This research was supported by the National Science Foundation of China (60635030, 60721002), the National High Technology Research and Development Program of China (2007AA01Z169), the Jiangsu Science Foundation (BK2008018), the Research Grants

Council of the Hong Kong Special Administrative Region (614508), and the Singapore NTU AcRF Tier-1 Research Grant (RG15/08).

References

Bach, F. R., Lanckriet, G. R. G., & Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the SMO algorithm. *Proceedings of the 21st International Conference on Machine Learning* (pp. 41–48). Banff, Canada.

Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge, UK: Cambridge University Press.

Hsieh, C. J., Chang, K. W., Lin, C. J., Keerthi, S. S., & Sundararajan, S. (2008). A dual coordinate descent method for large-scale linear SVM. *Proceedings of the 25th International Conference on Machine Learning* (pp. 408–415). Helsinki, Finland.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Upper Saddle River, NJ: Prentice Hall.

Joachims, T. (2006). Training linear SVMs in linear time. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data mining* (pp. 217–226). Philadelphia, PA.

Kelley, J. E. (1960). The cutting plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, *8*, 703–712.

Kim, S.-J., & Boyd, S. (2008). A minimax theorem with applications to machine learning, signal processing, and finance. *SIAM Journal on Optimization*, *19*, 1344–1367.

Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E., & Jordan, M. I. (2004). Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, *5*, 27–72.

Platt, J. C. (1999). Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges and A. Smola (Eds.), *Advances in kernel methods - Support vector learning*, 185–208. Cambridge, MA: MIT Press.

Rakotomamonjy, A., Bach, F. R., Canu, S., & Grandvalet, Y. (2007). More efficiency in multiple kernel learning. *Proceedings of the 24th International Conference on Machine Learning* (pp. 775–782). Corvallis, OR.

Rakotomamonjy, A., Bach, F. R., Canu, S., & Grandvalet, Y. (2008). SimpleMKL. *Journal of Machine Learning Research*, *9*, 2491–2521.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 888–905.

Smola, A. J., Vishwanathan, S. V. N., & Hofmann, T. (2005). Kernel methods for missing variables. *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics* (pp. 325–332). Barbados.

Sonnenburg, S., Rätsch, G., Schäfer, C., & Schölkopf, B. (2006). Large scale multiple kernel learning. *Journal of Machine Learning Research*, *7*, 1531–1565.

Tsang, I. W., Kwok, J. T., & Cheung, P. (2006). Core vector machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, *6*, 363–392.

Valizadegan, H., & Jin, R. (2007). Generalized maximum margin clustering and unsupervised kernel learning. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 1417–1424. Cambridge, MA: MIT Press.

Xu, L., Neufeld, J., Larson, B., & Schuurmans, D. (2005). Maximum margin clustering. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*, 1537–1544. Cambridge, MA: MIT Press.

Xu, L., & Schuurmans, D. (2005). Unsupervised and semi-supervised multi-class support vector machines. *Proceedings of the 20th National Conference on Artificial Intelligence* (pp. 904–910). Pittsburgh, PA.

Xu, Z., Jin, R., King, I., & Lyu, M. R. (2009). An extended level method for efficient multiple kernel learning. In D. Koller, D. Schuurmans, Y. Bengio and L. Bottou (Eds.), *Advances in neural information processing systems 21*, 1825–1832. Cambridge, MA: MIT Press.

Zhang, K., Tsang, I. W., & Kwok, J. T. (2007). Maximum margin clustering made practical. *Proceedings of the 24th International Conference on Machine learning* (pp. 1119–1126). Corvallis, OR.

Zhang, K., Tsang, I. W., & Kwok, J. T. (2009). Maximum margin clustering made practical. *To appear in IEEE Transactions on Neural Networks*.

Zhao, B., Wang, F., & Zhang, C. (2008). Efficient maximum margin clustering via cutting plane algorithm. *Proceedings of the 8th SIAM International Conference on Data Mining* (pp. 751–762). Atlanta, GA.