Learning the Kernel Matrix in Discriminant Analysis via Quadratically Constrained Quadratic Programming

Jieping Ye
Arizona State University
Tempe, AZ 85287
jieping.ye@asu.edu

Shuiwang Ji
Arizona State University
Tempe, AZ 85287
shuiwang.ji@asu.edu

Jianhui Chen
Arizona State University
Tempe, AZ 85287
jianhui.chen@asu.edu

ABSTRACT

The kernel function plays a central role in kernel methods. In this paper, we consider the automated learning of the kernel matrix over a convex combination of pre-specified kernel matrices in Regularized Kernel Discriminant Analysis (RKDA), which performs linear discriminant analysis in the feature space via the kernel trick. Previous studies have shown that this kernel learning problem can be formulated as a semidefinite program (SDP), which is however computationally expensive, even with the recent advances in interior point methods. Based on the equivalence relationship between RKDA and least square problems in the binary-class case, we propose a Quadratically Constrained Quadratic Programming (QCQP) formulation for the kernel learning problem, which can be solved more efficiently than SDP. While most existing work on kernel learning deal with binary-class problems only, we show that our QCQP formulation can be extended naturally to the multi-class case. Experimental results on both binary-class and multi-class benchmark data sets show the efficacy of the proposed QCQP formulations.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications - Data Mining

General Terms

Algorithms

Keywords

Quadratically constrained quadratic programming, kernel learning, model selection, kernel discriminant analysis, convex optimization

1. INTRODUCTION

Kernel methods [21, 22] have been applied successfully in various machine learning tasks, such as dimensionality reduction, clustering, classification, and regression. They work by embedding the data into some high-dimensional feature space through the so-called *kernel function*. The key fact underlying the success of kernel methods is that the embedding into feature space can be uniquely determined by specifying the kernel function that computes the dot products between data points in the feature space. In other words, the kernel function implicitly defines the nonlinear mapping to the feature space and expensive computations in the high-dimensional space can be avoided by evaluating the kernel function. Therefore, one of the central issues in kernel methods is the problem of kernel selection.

The problem of automated kernel learning has been an active area of research recently. Lanckriet *et al.* [12] pioneered the work of learning a linear combination of pre-specified kernels for Support Vector Machines (SVM) [5, 26] using convex programming and it has been improved in [3] using the Sequential Minimal Optimization (SMO) algorithm [20]. Reformulation of this problem as semi-infinite linear program has been proposed in [23] along with extensions to regression and one-class classification. While most approaches produce stationary combinations, Lewis *et al.* [13] proposed to use different combinations for different inputs. Argyriou *et al.* [2] proposed to combine potentially an infinite number of kernels, which was formulated as a difference of convex (DC) program. In general, approaches based on learning a convex combination of kernels facilitate heterogeneous data integration from multiple data sources and it has been applied for integrating various types of biological data, *e.g.*, amino acid sequences, hydropathy profiles, and gene expression data for enhanced biological inference [11]. Jebara [9] considered the kernel selection problem in the context of multi-task learning. Zien and Ong [30] recently studied multi-class multiple kernel learning.

This paper addresses the issue of kernel learning for regularized kernel discriminant analysis [16, 17]. The proposed algorithms learn an optimal kernel matrix as a convex combination of a set of pre-specified kernel matrices. In this sense, our proposed methods are similar to those proposed in [10, 12] and can thus be used for heterogeneous data integration. The problem of kernel learning for discriminant analysis has been addressed originally in [6]. Kim *et al.* [10] formulated this problem as a semidefinite program (SDP), and hence the globally optimal solution can be obtained. The algorithms in [6, 10] deal with binary-class problems only. They were extended to the multi-class case in [28]. One limitation of the SDP formulation is its high computational cost, even with the recent advances in interior point

KDD'07, August 12–15, 2007, San Jose, California, USA.

methods. In this case, the overhead of model (kernel) selection may dominate the learning process. We address this issue in this paper. Key contributions of this paper include:

- We propose a Quadratically Constrained Quadratic Programming (QCQP) formulation for discriminant kernel learning in the binary-class case. It is known that interior point algorithms for solving QCQP problems are much more efficient than those of their SDP counterparts.
- We extend the above formulation to the multi-class setting by decomposing the multi-class RKDA kernel learning problem into a set of binary-class kernel learning problems that are constrained to share a common kernel. It is worth noting that the optimal kernel function is the same for the original and the decomposed formulations, though the optimal transformation matrices may not coincide. That is, the decomposed form is equivalent to the original one for the purpose of kernel learning.
- We conduct extensive experiments to compare the proposed algorithms with several existing approaches for kernel learning, as well as cross-validation based approaches under a common experimental setup. Our experimental results validate the effectiveness of the proposed QCQP formulations.

The rest of this paper is organized as follows: Section 2 provides background information on kernel methods, as well as the SDP formulation for kernel learning. Section 3 derives the QCQP formulation for the binary-class case. Section 4 extends the QCQP formulation to the multi-class case. Section 5 presents our experimental study and this paper concludes with discussion and conclusion in Section 6.

2. BACKGROUND

Let $\mathcal{X}$ denote the input or instance space, which is a subspace of $\mathbb{R}^d$, and $\mathcal{Y} = \{-1, +1\}$ denote the output or class label set. An example, i.e., an input-output pair (x, y), where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, is called positive (negative) if its class label y is $+1$ (-1). We assume that the examples are drawn randomly and independently from a fixed, but unknown, underlying probability distribution over $\mathcal{X} \times \mathcal{Y}$.

A symmetric function $K : \mathcal{X} \times \mathcal{X} \to R$ is called a kernel function [21] if it satisfies the finitely positive semidefinite property: for any $x_1, \cdots, x_m \in \mathcal{X}$, the *Gram* matrix $G \in \mathbb{R}^{m \times m}$, defined by $G_{ij} = K(x_i, x_j)$ is positive semidefinite. Any kernel function K implicitly maps the input set $\mathcal{X}$ to a high-dimensional (possibly infinite) Hilbert space $\mathcal{H}_K$ equipped with the inner product $(\cdot, \cdot)_{\mathcal{H}_K}$ through a mapping $\phi_K : \mathcal{X} \to \mathcal{H}_K$:

$$K(x, z) = (\phi_K(x), \phi_K(z))_{\mathcal{H}_K}. \quad (1)$$

In kernel-based classification, the algorithms learn a classifier $f : \mathcal{X} \to \{-1, +1\}$ whose decision boundary between the two classes is affine in the feature space $\mathcal{H}_K$:

$$f(x) = \mathrm{sgn}(w^T \phi_K(x) + b), \quad (2)$$

where $w \in \mathcal{H}_K$ is the vector of feature weights, $b \in \mathbb{R}$ is the intercept, and $\mathrm{sgn}(u) = +1$, if $u > 0$, and -1 otherwise.

Let $\{x_1^+, \cdots, x_{m_+}^+\}$ and $\{x_1^-, \cdots, x_{m_-}^-\}$ denote the collections of data points from the positive and negative classes, respectively. let $X = [x_1^+, \cdots, x_{m_+}^+, x_1^-, \cdots, x_{m_-}^-]$ be the data matrix. The total number of data points in the training set is $m = m_+ + m_-$. For a given kernel function K, the basic idea of RKDA in the binary-class case is to find a direction in the feature space $\mathcal{H}_K$ onto which the projections of the two sets $\{\phi(x_i^+)\}_{i=1}^{m_+}$ and $\{\phi(x_i^-)\}_{i=1}^{m_-}$ are well separated. Define the centroids of the two classes as follows:

$$\mu_K^+ = \frac{1}{m_+} \sum_{i=1}^{m_+} \phi_K(x_i^+), \qquad \mu_K^- = \frac{1}{m_-} \sum_{i=1}^{m_-} \phi_K(x_i^-), \quad (3)$$

and the two sample class covariance matrices as follows:

$$S_K^+ = \frac{1}{m_+} \sum_{i=1}^{m_+} (\phi_K(x_i^+) - \mu_K^+)(\phi_K(x_i^+) - \mu_K^+)^T, \quad (4)$$

$$S_K^- = \frac{1}{m_-} \sum_{i=1}^{m_-} (\phi_K(x_i^-) - \mu_K^-)(\phi_K(x_i^-) - \mu_K^-)^T. \quad (5)$$

Specifically, in RKDA the separation between two classes is measured by the ratio of the variance $(w^T(\mu_K^+ - \mu_K^-))^2$ between the classes to $w^T \left(m_+/m S_K^+ + m_-/m S_K^-\right) w$, the variance within the classes. Thus, the following criterion is commonly maximized in RKDA:

$$F_1(w, K) = \frac{(w^T(\mu_K^+ - \mu_K^-))^2}{w^T \left(m_+/m S_K^+ + m_-/m S_K^- + \lambda I\right) w}, \quad (6)$$

where $\lambda > 0$ is a regularization parameter. For a fixed kernel function K and regularization parameter λ, the optimal weight vector $w^* \equiv \mathrm{argmax}_w \{F_1(w, K)\}$, that maximizes the objective in Eq. (6) is given by

$$w^* = (m_+/m S_K^+ + m_-/m S_K^- + \lambda I)^{-1}(\mu_K^+ - \mu_K^-). \quad (7)$$

The maximum value $F_1^*(K) \equiv \max_w \{F_1(w, K)\}$ of the objective function in Eq. (6) achieved by the optimal weight vector w^* is given by

$$F_1^*(K) = (\mu_K^+ - \mu_K^-)^T \left(\frac{m_+}{m} S_K^+ + \frac{m_-}{m} S_K^- + \lambda I\right)^{-1} (\mu_K^+ - \mu_K^-). \quad (8)$$

It can be shown [10] that the optimal value $F_1^*(K)$ in Eq. (8) can be simplified to the following:

$$F_1^*(K) = \frac{1}{\lambda} a^T G(I - J(\lambda I + JGJ)^{-1} JG)a, \quad (9)$$

where I is the identity matrix, a is defined as

$$a = [1/m_+, \cdots, 1/m_+, -1/m_-, \cdots, -1/m_-]^T \in \mathbb{R}^m, \quad (10)$$

the matrix J is defined as:

$$J = \begin{pmatrix} \frac{1}{\sqrt{m_+}}(I - \frac{1}{m_+} e_{m_+} e_{m_+}^T) & 0 \\ 0 & \frac{1}{\sqrt{m_-}}(I - \frac{1}{m_-} e_{m_-} e_{m_-}^T) \end{pmatrix},$$

and e_{m_+} and e_{m_-} are vectors of all ones of length m_+ and m_-, respectively.

In [10], G is restricted to be a linear combination of the p given kernel matrices $G_1, \cdots, G_p$ as

$$G \in \mathcal{G} = \left\{ G \,\middle|\, G = \sum_{i=1}^{p} \theta_i G_i, \;\; \sum_{i=1}^{p} \theta_i = 1, \;\; \theta_i \geq 0 \;\; \forall i \right\}. \quad (11)$$

It was shown in [10] that, for a fixed regularization parameter λ, the optimal Gram matrix G computed from the kernel function K that maximizes $F_1^*(K)$ given in Eq. (9) can be

obtained by solving a SDP [4, 25]. General-purpose optimization packages such as SeDuMi [24] use the interior-point methods [18] to solve SDP. However, it is known that SDP is expensive to solve practically, and thus for problems of moderate size, this overhead of optimal kernel learning is large. In order to alleviate this computational problem, we propose in the next section a convex QCQP formulation for this kernel learning problem. Interior point methods for solving QCQP is more efficient than its SDP counterpart. Furthermore, the proposed QCQP formulation can be extended naturally to the multi-class case.

3. PROPOSED QCQP FORMULATION FOR BINARY-CLASS PROBLEMS

In the following discussion, we work on the centered version of kernel matrices. This is equivalent to a data centering process. More precisely, given a set of p kernel matrices $G_1, \cdots, G_p$, the proposed algorithms learn an optimal kernel matrix $\tilde{G} \in \tilde{\mathcal{G}}$, where

$$\tilde{\mathcal{G}} = \left\{ \tilde{G} \,\middle|\, \tilde{G} = \sum_{i=1}^{p} \theta_i \tilde{G}_i, \;\; \sum_{i=1}^{p} \theta_i \, r_i = 1, \;\; \theta_i \geq 0 \right\}, \quad (12)$$

$\tilde{G}_i = PG_iP$, $r_i = \mathrm{trace}(\tilde{G}_i)$, and $P \in \mathbb{R}^{m \times m}$ is the centering matrix defined as

$$P = I - \frac{1}{m} e_m e_m^T, \quad (13)$$

and e_m is the vector of all ones of size m.

Consider the maximization of the following objective function [28]:

$$F_2(w, K) = \frac{(w^T(\mu_K^+ - \mu_K^-))^2}{w^T(\Sigma_K + \lambda I)w}, \quad (14)$$

where Σ_K is defined as follows:

$$\Sigma_K = \frac{1}{m} \phi_K(X) \, P \, \phi_K(X)^T, \quad (15)$$

$\phi_K(X)$ is the data matrix in the feature space given by

$$\phi_K(X) = \left[\phi_K(x_1^+), \cdots, \phi_K(x_{m_+}^+), \phi_K(x_1^-), \cdots, \phi_K(x_{m_-}^-) \right],$$

P is defined in Eq. (13), and

$$\mu_K = \frac{1}{m} \left(\sum_{i=1}^{m_+} \phi_K(x_i^+) + \sum_{i=1}^{m_-} \phi_K(x_i^-) \right) \quad (16)$$

is the global centroid of the data in the feature space. It is easy to verify that for fixed K and λ, Eqs. (6) and (14) are equivalent for the computation of the optimal w.

Consider the regularized least squares problem, which minimizes the following objective function:

$$F_3(w, K) = ||(\phi_K(X)P)^T w - a||^2 + \lambda ||w||^2. \quad (17)$$

It is known that discriminant analysis and least square problems are equivalent in terms of the computation of the optimal w for fixed K and λ in the binary-class case [15]. We show in the following lemma that discriminant analysis and least square problems are also equivalent in terms of the computation of the optimal kernel function K.

LEMMA 3.1. *Let w^*, K^*, $\tilde{w}^*$, and $\tilde{K}^*$ solve the following optimization problems:*

$$(w^*, K^*) = \max_K \max_w F_2(w, K), \quad (18)$$

$$(\tilde{w}^*, \tilde{K}^*) = \min_K \min_w F_3(w, K). \quad (19)$$

Then $w^* = \tilde{w}^*$ and $K^* = \tilde{K}^*$.

PROOF. The optimal vector $w^* \equiv \mathrm{argmax}_w \{F_2(w, K)\}$ for a fixed K is given by

$$w^* = (\Sigma_K + \lambda I)^{-1}(\mu_K^+ - \mu_K^-). \quad (20)$$

The maximum value of the objective function in Eq. (14) achieved by w^* is given by

$$F_2^*(K) \equiv F_2(w^*, K) = (\mu_K^+ - \mu_K^-)^T (\Sigma_K + \lambda I)^{-1} (\mu_K^+ - \mu_K^-).$$

It follows from the Sherman-Woodbury-Morrison formula [7] that

$$\begin{aligned} w^* &= (\Sigma_K + \lambda I)^{-1}(\mu_K^+ - \mu_K^-) \\ &= \left(\phi_K(X) \, P \, \phi_K(X)^T + \lambda I \right)^{-1} \phi_K(X) a \\ &= \left((\phi_K(X)P)(\phi_K(X)P)^T + \lambda I \right)^{-1} \phi_K(X) a \\ &= \frac{1}{\lambda} \phi_K(X) \left(I - P \left(\lambda I + PGP \right)^{-1} PG \right) a. \end{aligned} \quad (21)$$

and thus

$$\begin{aligned} F_2^*(K) &= (\mu_K^+ - \mu_K^-)^T w^* = a^T \phi_K(X)^T w^* \\ &= \frac{1}{\lambda} a^T \left(G - GP \left(\lambda I + PGP \right)^{-1} PG \right) a. \end{aligned} \quad (22)$$

Since the vector a defined in Eq. (10) is of zero mean, i.e., $Pa = a$, we have

$$\begin{aligned} F_2^*(K) &= \frac{1}{\lambda} a^T P \left(G - GP(\lambda I + PGP)^{-1} PG \right) Pa \\ &= \frac{1}{\lambda} a^T \left(\tilde{G} - \tilde{G}(\lambda I + \tilde{G})^{-1} \tilde{G} \right) a, \end{aligned} \quad (23)$$

where $\tilde{G} = PGP$ is derived from G with both rows and columns centered. Since

$$\begin{aligned} \tilde{G} - \tilde{G}(\lambda I + \tilde{G})^{-1} \tilde{G} &= \tilde{G} - \tilde{G}(\lambda I + \tilde{G})^{-1}(\tilde{G} + \lambda I - \lambda I) \\ &= \lambda \tilde{G} (\lambda I + \tilde{G})^{-1} \\ &= \lambda (\tilde{G} + \lambda I - \lambda I)(\lambda I + \tilde{G})^{-1} \\ &= \lambda I - \lambda^2 (\lambda I + \tilde{G})^{-1}, \end{aligned} \quad (24)$$

the optimal value $F_2^*(K)$ in Eq. (23) can be simplified as

$$F_2^*(K) = a^T a - \lambda a^T (\lambda I + \tilde{G})^{-1} a. \quad (25)$$

For a fixed K and λ, the optimal $\tilde{w}^*$ that minimizes the objective function in Eq. (17) is given by

$$\begin{aligned} \tilde{w}^* &= \left(\lambda I + \phi_K(X) P \phi_K(X)^T \right)^{-1} \phi_K(X) Pa \\ &= \frac{1}{\lambda} \phi_K(X) \left(I - P \left(\lambda I + PGP \right)^{-1} PG \right) a = w^*. \end{aligned} \quad (26)$$

The optimal value of the objective function in Eq. (17) is therefore given by

$$F_3^*(K) = \lambda a^T (\lambda I + \tilde{G})^{-1} a. \quad (27)$$

Thus, the maximization of $F_2^*(K)$ in Eq. (25) is equivalent to the minimization of $F_3^*(K)$ in Eq. (27). This completes the proof. □

Based on this equivalence result, we can formulate the kernel learning problem for RKDA as a convex QCQP problem, as summarized in the following theorem.

THEOREM 3.1. *Given a set of p centered kernel matrices $\tilde{G}_1, \cdots, \tilde{G}_p$, the optimal kernel matrix, in the form of linear combination of the given p kernel matrices, that optimizes the criterion in Eq. (17) can be found by solving the following convex QCQP problem:*

$$\begin{aligned} \max_{\beta,t} \quad & -\frac{1}{4}\beta^T\beta + \beta^T a - \frac{1}{4\lambda}t \\ \textit{subject to} \quad & t \geq \frac{1}{r_i}\beta^T\tilde{G}_i\beta,\ i = 1, \cdots, p, \end{aligned} \tag{28}$$

where $r_i = trace(\tilde{G}_i)$ and $r = [r_1, r_2, \cdots, r_p]$.

PROOF. We consider the dual formulation of the minimization of $F_3(w, K)$ in terms of w. Denote

$$\eta = (\phi_K(X)P)^T w - a. \tag{29}$$

It follows that

$$F_3(w, K) = ||\eta||^2 + \lambda||w||^2. \tag{30}$$

This leads to the following optimization problem:

$$\begin{aligned} \min_{w,\eta} \quad & F_3(w, K) = ||\eta||^2 + \lambda||w||^2 \\ \text{subject to} \quad & \eta = (\phi_K(X)P)^T w - a \end{aligned} \tag{31}$$

Define its Lagrangian function as follows:

$$L(\eta, w, \beta) = ||\eta||^2 + \lambda||w||^2 - \beta^T((\phi_K(X)P)^T w - a - \eta), \tag{32}$$

where β is the vector of Lagrangian dual variables. Taking the derivative of $L(\eta, w, \beta)$ with respect to η and w and setting them equal to zero, we get

$$\frac{\partial L(\eta, w, \beta)}{\partial \eta} = 2\eta + \beta = 0, \tag{33}$$

$$\frac{\partial L(\eta, w, \beta)}{\partial w} = 2\lambda w - \phi_K(X)P\beta = 0. \tag{34}$$

It follows that

$$\eta = -\frac{\beta}{2}, \quad \text{and} \quad w = \frac{\phi_K(X)P\beta}{2\lambda}. \tag{35}$$

Thus we obtain the following Lagrangian dual function:

$$g(\beta) = \min_{w,\eta} L(\eta, w, \beta) = -\frac{1}{4}\beta^T\left(I + \frac{1}{\lambda}PGP\right)\beta + \beta^T a.$$

The optimal β^* is computed by maximizing $g(\beta)$ as

$$\beta^* = \text{argmax}_\beta \left\{-\frac{1}{4}\beta^T\left(I + \frac{1}{\lambda}PGP\right)\beta + \beta^T a\right\}. \tag{36}$$

Since strong duality holds, the optimal kernel is given by solving the following optimization problem:

$$\min_{\tilde{G}\in\tilde{\mathcal{G}}} \max_\beta \left\{-\frac{1}{4}\beta^T\left(I + \frac{1}{\lambda}\tilde{G}\right)\beta + \beta^T a\right\}. \tag{37}$$

We can rewrite the above optimization problem as

$$\begin{aligned} & \min_{\theta:\,\theta\geq 0,\,\theta^T r=1} \max_\beta \left\{-\frac{1}{4}\beta^T\left(I + \frac{1}{\lambda}\sum_{i=1}^p \theta_i\tilde{G}_i\right)\beta + \beta^T a\right\} \\ = & \max_\beta \min_{\theta:\,\theta\geq 0,\,\theta^T r=1} \left\{-\frac{1}{4}\beta^T\left(I + \frac{1}{\lambda}\sum_{i=1}^p \theta_i\tilde{G}_i\right)\beta + \beta^T a\right\} \\ = & \max_\beta \min_{\theta:\,\theta\geq 0,\,\theta^T r=1} \left\{-\frac{1}{4\lambda}\sum_{i=1}^p \theta_i\beta^T\tilde{G}_i\beta - \frac{1}{4}\beta^T\beta + \beta^T a\right\} \\ = & \max_\beta \left\{-\frac{1}{4}\beta^T\beta + \beta^T a - \frac{1}{4\lambda}\max_{\theta:\,\theta\geq 0,\,\theta^T r=1}\left(\sum_{i=1}^p \theta_i\beta^T\tilde{G}_i\beta\right)\right\} \\ = & \max_\beta \left\{-\frac{1}{4}\beta^T\beta + \beta^T a - \frac{1}{4\lambda}\max_i\left(\frac{1}{r_i}\theta_i\beta^T\tilde{G}_i\beta\right)\right\}. \end{aligned} \tag{38}$$

The exchange of minimization and maximization in deriving the second equation from the first holds since the objective function is convex in θ and concave in β, the minimization problem is strictly feasible in θ and the maximization problem is strictly feasible in β. Therefore, Slater's condition [4] follows and strong duality holds [12, 4]. By simply changing the last term in the above equation to t and moving it to the constraint, we prove this theorem. □

Note that general-purpose optimization software packages like SeDuMi [24] and MOSEK [1] also solve the dual problem. Thus the coefficients $\theta_1, \cdots, \theta_p$ can be obtained from the dual variables by dividing the corresponding traces of centered kernel matrices. The formulation in Eq. (28) is a QCQP, which is a special form of Second Order Cone Program (SOCP) [14] and SDP. Theoretical results on interior point method [18] show that QCQP can be solved more efficiently than SDP and it is therefore more scalable to large-scale problems. The worst-case complexity of this QCQP formulation is $O(pn^3)$. Similar ideas have been used in [12] to formulate the kernel learning problem of SVM as a non-negative linear combination of some given kernel matrices.

Recall that all kernel learning formulations discussed in [10, 12] are constrained to binary-class problems. We show in the next section that our formulation in this section can be extended naturally to deal with multi-class problems.

4. PROPOSED QCQP FORMULATION FOR MULTI-CLASS PROBLEMS

In multi-class classifications, we are given a data set that consists of m samples $\{(x_i, y_i)\}_{i=1}^m$, where $x_i \in \mathbb{R}^d$, and $y_i \in \{1, 2, \cdots, k\}$ is the class label of the i-th sample. Similar to the binary-class case, let $X = [x_1, \cdots, x_m]$ be the data matrix.

In the multi-class RKDA formulation, the maximization of the following objective function is commonly used [27]:

$$F_4(W, K) = \text{trace}\left(\left(W^T(\Sigma_K + \lambda I)W\right)^{-1} W^T B_K W\right), \tag{39}$$

where $W = [w_1, w_2, \cdots, w_\ell]$ $(\ell < k)$ is the transformation matrix, and B_K, the so-called between-class scatter matrix is defined as

$$B_K = \phi_K(X)HH^T\phi_K(X)^T, \tag{40}$$

$H = [h_1, h_2, \cdots, h_k]$, and h_i is a vector whose j-th entry is

given by

$$h_i(j) = \begin{cases} \sqrt{\frac{n}{n_i}} - \sqrt{\frac{n_i}{n}} & \text{if } y_j = i \\ -\sqrt{\frac{n_i}{n}} & \text{otherwise.} \end{cases} \quad (41)$$

The optimal W is given by computing the top eigenvectors of the following matrix:

$$(\Sigma_K + \lambda I)^{-1} B_K.$$

Since the weight vectors are in the span of the images of the data points in the feature space, we can express W as $W = \phi_K(X)A$ for some matrix $A \in \mathbb{R}^{m \times \ell}$, where $A = [\alpha_1, \cdots, \alpha_\ell]$. Then

$$F_4(W, K) = \text{trace}\left(\left(A^T \left(GPG + \lambda G \right) A \right)^{-1} A^T GHH^T GA \right).$$

Define two matrices S_t^K and S_b^K as follows:

$$S_t^K = GPG + \lambda G, \quad (42)$$
$$S_b^K = GHH^T G. \quad (43)$$

Since the null space of S_t^K lies in the null space of S_b^K, there exists a nonsingular matrix Z such that [7]

$$Z^T S_t^K Z = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}, \quad Z^T S_b^K Z = \begin{pmatrix} \Sigma_b & 0 \\ 0 & 0 \end{pmatrix}, \quad (44)$$

where Σ_b is diagonal with the diagonal entries sorted in non-decreasing order. The optimal A^* is given by

$$A^* = Z_q = [z_1, \cdots, z_q], \quad (45)$$

where Z_q consists of the first q columns of Z, and $q = \text{rank}(S_b^K)$. It follows that the optimal value of $F_4(W, K)$ achieved by the optimal A^* is given by

$$F_4^*(K) = \text{trace}(\Sigma_b) = \text{trace}\left(\left(S_t^K \right)^{-1} S_b^K \right). \quad (46)$$

Note that we have assumed $S_t^K = GPG + \lambda G$ is nonsingular in the above discussion. In case it is singular, we could use the pseudo-inverse, and all the following derivations still hold.

Thus, in the multi-class case, the optimal kernel function K can be computed by maximizing $F_4^*(K)$ in Eq. (46), which is however highly nonlinear and difficult to solve. In [28], we present an equivalent formulation as the one in Eq. (46), which is more tractable computationally. The main result is summarized in the following lemma:

LEMMA 4.1. *Let F_4 be defined as in Eq. (39) and*

$$F_5(W, K) = \sum_{i=1}^{k} \frac{(w_i^T \phi_K(X) h_i)^2}{w_i^T (\Sigma_K + \lambda I) w_i}, \quad (47)$$

where $W = [w_1, \cdots, w_k]$, and h_i is defined in Eq. (41). Let W^*, K^*, $\tilde{W}^*$, and $\tilde{K}^*$ solve the following optimization problems:

$$(W^*, K^*) = \max_K \max_W F_4(W, K), \quad (48)$$
$$(\tilde{W}^*, \tilde{K}^*) = \max_K \max_W F_5(W, K). \quad (49)$$

Then $K^* = \tilde{K}^*$.

It is interesting to note that, in general, the optimal W^* and $\tilde{W}^*$ for the optimization problems in Eqs. (48) and (49), respectively, are different. Furthermore, the objective function in Eq. (47) is closely related to its binary counterpart in Eq. (14). Note that a variant of the Fisher Discriminant Ratio (FDR) [10] can be written as:

$$F_2(w, K) = \frac{(w^T \phi_K(X)\, a)^2}{w^T (\Sigma_K + \lambda I) w}. \quad (50)$$

Thus $F_5(W, K)$ in Eq. (47) can be interpreted as the weighted summation of the FDRs between the samples in the i-th class and the rest where $i = 1, \cdots, k$. The weights can be computed from the definition of H in Eq. (41) as follows:

$$h_i = \begin{bmatrix} \vdots \\ \sqrt{\frac{n}{n_i}} - \sqrt{\frac{n_i}{n}} \\ \vdots \\ -\sqrt{\frac{n_i}{n}} \\ \vdots \end{bmatrix} = (n - n_i)\sqrt{\frac{n_i}{n}} \begin{bmatrix} \vdots \\ \frac{1}{n_i} \\ \vdots \\ -\frac{1}{n - n_i} \\ \vdots \end{bmatrix}$$
$$= (n - n_i)\sqrt{\frac{n_i}{n}}\, a^{(i)},$$

where $a^{(i)}$ is obtained from Eq. (10) by taking the samples from the i-th class as positive and the rest as negative. Thus the weight for the i-th binary classification problem is: $(n - n_i)^2 n_i / n$.

Next, we propose a QCQP formulation for the multi-class RKDA kernel learning problem. Consider the minimization of the following objective function:

$$F_6(W, K) = \sum_{i=1}^{k} \left(||(\phi_K(X)P)^T w_i - h_i||^2 + \lambda ||w_i||^2 \right), \quad (51)$$

where $W = [w_1, \cdots, w_k]$. It is clear that for a fixed K, the computations of w_i and w_j for $i \neq j$ are independent of each other. By extending the results from Lemma 3.1 and Lemma 4.1, it is easy to show that the optimal kernel function K minimizing the objective function in Eq. (47) coincides with the minimizer of $F_6(W, K)$ in Eq. (51). Motivated by this equivalence result, we derive an efficient QCQP formulation for the multi-class RKDA kernel learning problem, as summarized in the following theorem.

THEOREM 4.1. *Given a set of p centered kernel matrices $\tilde{G}_1, \cdots, \tilde{G}_p$, the optimal kernel matrix, in the form of linear combination of the given p kernel matrices, that optimizes the criterion in Eq. (51) can be found by solving the following convex QCQP problem:*

$$\max_{\beta_1, \cdots, \beta_k, t} \quad \sum_{j=1}^{k} \beta_j^T h_j - \frac{1}{4} \sum_{j=1}^{k} \beta_j^T \beta_j - \frac{1}{4\lambda} t$$
$$\text{subject to} \quad t \geq \frac{1}{r_i} \sum_{j=1}^{k} \beta_j^T \tilde{G}_i \beta_j, \; i = 1, \cdots, p. \quad (52)$$

where $r_i = \text{trace}(\tilde{G}_i)$.

PROOF. We first consider the dual formulation of the minimization of $F_6(W, K)$ in terms of W for a fixed K. Denote

$$\eta_i = (\phi_K(X)P)^T w_i - h_i. \quad (53)$$

It follows that

$$F_6(w,K) = \sum_{i=1}^{k} ||\eta_i||^2 + \lambda \sum_{i=1}^{k} ||w_i||^2. \quad (54)$$

Define the Lagrangian function as follows:

$$\begin{aligned} L(\{\eta_i, w_i, \beta_i\}_{i=1}^k) &= \sum_{i=1}^{k} ||\eta_i||^2 + \lambda \sum_{i=1}^{k} ||w_i||^2 \\ &- \sum_{i=1}^{k} \beta_i^T \left((\phi_K(X)P)^T w_i - h_i - \eta_i \right), \end{aligned}$$

where the β_i's are the vectors of Lagrangian dual variables. Taking the derivative of L with respect to η_i and w_i for all i, and setting them equal to zero, we get

$$\frac{\partial L}{\partial \eta_i} = 2\eta_i + \beta_i = 0, \quad (55)$$

$$\frac{\partial L}{\partial w_i} = 2\lambda w_i - \phi_K(X)P\beta_i = 0. \quad (56)$$

Thus we have

$$\eta_i = -\frac{\beta_i}{2}, \text{ and } w_i = \frac{\phi_K(X)P\beta_i}{2\lambda}, \quad (57)$$

and we obtain the following Lagrangian dual function:

$$\begin{aligned} g(\beta_1, \cdots, \beta_k) &= \min_{w_i, \eta_i, i=1,\cdots,k} L(\{\eta_i, w_i, \beta_i\}_{i=1}^k) \\ &= \sum_{i=1}^{k} \left(-\frac{1}{4}\beta_i^T \left(I + \frac{1}{\lambda} PGP \right) \beta_i + \beta_i^T h_i \right). \end{aligned}$$

The optimal $\beta_1^*, \cdots, \beta_k^*$ can be computed by maximizing $g(\beta_1, \cdots, \beta_k)$ as

$$(\beta_1^*, \cdots, \beta_k^*) = \text{argmax}_{\beta_1,\cdots,\beta_k}\, g(\beta_1, \cdots, \beta_k).$$

Since strong duality holds, the optimal kernel function K is given by solving the following optimization problem:

$$\min_{\tilde{G} \in \tilde{\mathcal{G}}} \max_{\beta_1,\cdots,\beta_k} \left\{ \sum_{i=1}^{k} \left(-\frac{1}{4}\beta_i^T \left(I + \frac{1}{\lambda}\tilde{G} \right) \beta_i + \beta_i^T h_i \right) \right\}. \quad (58)$$

Similar to the binary-class case, the above optimization problem can be written as

$$\min_{\substack{\theta\,:\,\theta \geq 0,\\ \theta^T r = 1}} \max_{\beta_1,\cdots,\beta_k} \left\{ \sum_{j=1}^{k} \left(-\frac{1}{4}\beta_j^T \left(I + \frac{1}{\lambda} \sum_{i=1}^{p} \left(\theta_i \tilde{G}_i \right) \right) \beta_j + \beta_j^T h_j \right) \right\}$$

$$= \max_{\beta_1,\cdots,\beta_k} \min_{\substack{\theta\,:\,\theta \geq 0\\ \theta^T r = 1}} \left\{ \sum_{j=1}^{k} \left(-\frac{1}{4}\beta_j^T \left(I + \frac{1}{\lambda} \sum_{i=1}^{p} \theta_i \tilde{G}_i \right) \beta_j + \beta_j^T h_j \right) \right\}$$

$$= \max_{\beta_1,\cdots,\beta_k} \left\{ \sum_{j=1}^{k} \beta_j^T h_j - \frac{1}{4} \sum_{j=1}^{k} \beta_j^T \beta_j - \frac{1}{4\lambda} \max_i \left(\frac{1}{r_i} \sum_{j=1}^{k} \beta_j^T \tilde{G}_i \beta_j \right) \right\},$$

where the second equality follows since

$$\min_{\substack{\theta\,:\,\theta \geq 0\\ \theta^T r = 1}} \left\{ \sum_{j=1}^{k} \left(-\frac{1}{4}\beta_j^T \left(I + \frac{1}{\lambda} \sum_{i=1}^{p} \theta_i \tilde{G}_i \right) \beta_j + \beta_j^T h_j \right) \right\}$$

$$= \min_{\substack{\theta\,:\,\theta \geq 0\\ \theta^T r = 1}} \left\{ -\frac{1}{4} \sum_{j=1}^{k} \beta_j^T \beta_j - \frac{1}{4\lambda} \sum_{i=1}^{p} \theta_i \left(\sum_{j=1}^{k} \beta_j^T \tilde{G}_i \beta_j \right) + \sum_{j=1}^{k} \beta_j^T h_j \right\}$$

$$= \sum_{j=1}^{k} \beta_j^T h_j - \frac{1}{4} \sum_{j=1}^{k} \beta_j^T \beta_j - \frac{1}{4\lambda} \max_{\substack{\theta\,:\,\theta \geq 0\\ \theta^T r = 1}} \left\{ \sum_{i=1}^{p} \theta_i \left(\sum_{j=1}^{k} \beta_j^T \tilde{G}_i \beta_j \right) \right\}.$$

Adding a variable t, we prove the formulation in Eq. (52). □

5. EXPERIMENTAL EVALUATIONS

In this section, we empirically evaluate the proposed QCQP formulations in comparison with several existing kernel learning approaches. The experimental setup is similar to the one in [10]. We used the MOSEK package [1] to solve the QCQP formulations.

5.1 Experiments on Binary-class Problems

In the binary-class case, we compared our formulations with the SDP formulation proposed in [10], 1-norm soft margin SVM, 2-norm soft margin SVM with and without the regularization parameter C optimized jointly as proposed in [12]. We also employed double cross-validation to choose kernels and regularization parameters for SVM and RKDA. Four data sets were used in the binary case. The *sonar*, *ionosphere*, and *breast cancer* data sets were retrieved from the UCI Machine Learning Repository [19]. The *heart* data set was obtained from the STATLOG project[1]. For each data set, we randomly partitioned the entire data set into training and test sets with 80% of the data in the training set and 20% in the test set. Following [10], we focus on learning a convex combination of ten Gaussian kernels:

$$K(x,z) = \sum_{i=1}^{10} \theta_i \; e^{-||x-z||^2/\sigma_i^2}. \quad (59)$$

The values of σ_i are chosen uniformly on the logarithmic scale over the interval $[10^{-1}, 10^2]$, as in [10]. Then the ten kernels were fed into the optimization packages to obtain the corresponding coefficients for each kernel. Finally, the combined kernel was used for classification.

Tables 1 and 2 show the corresponding experimental results on the *sonar, heart, ionosphere*, and *breast cancer* data sets. The λ value is fixed to 10^{-8} for the proposed QCQP formulation, as used in [10]. Following [12], we fix C to 1 for SM1 and SM2. We can observe from the tables that the proposed QCQP formulation is comparable to three SVM-based approaches, while they are very competitive with other three approaches, including SDP_{Kim}, $\text{RKDA}_{K,\lambda}$, and $\text{SVM}_{K,C}$. Recall that the first five methods in the tables avoid cross-validation and they can be used for heterogeneous data integration from multiple sources, while $\text{RKDA}_{K,\lambda}$ and $\text{SVM}_{K,C}$ choose the single best kernel.

To understand the relative importance of each kernel when they are used individually, we fix the kernel to each of the ten pre-specified kernels and tune the λ in RKDA and C in SVM using cross-validation and the accuracy of each kernel is recorded (RKDA_λ and SVM_C). We recorded the number of times that a particular kernel has been selected by $\text{RKDA}_{K,\lambda}$ and $\text{SVM}_{K,C}$ in double cross-validation. We expect these quantities to have certain relationship with the coefficients learned by convex optimizations.

For the sonar data, cross-validated RKDA achieves the best performance on kernels corresponding to θ_6 and θ_7 while cross-validated SVM achieves the highest accuracy on θ_6, θ_7, and θ_8. On the other hand, methods using linear combination of kernels seem to favor kernels corresponding to θ_5, θ_6, and θ_7. For the heart data, cross-validated SVM favors kernels corresponding to θ_8, θ_9, and θ_{10} (they were

[1] http://www.liacc.up.pt/ML/old/statlog/datasets.html

Table 1: Comparison of seven methods for the sonar data set. The seven tested methods, listed from top to bottom are: the QCQP formulation as proposed in Theorem 3.1, SDP formulation proposed in [10], 1-norm soft margin SVM, 2-norm soft margin SVM without and with C optimized as proposed in [12], RKDA and SVM with the kernels and regularization parameters selected by double cross-validation. In general, subscripts of names in the first column are used to denote quantities that are optimized. The ten pre-specified kernels are all RBF kernels and the σ values used are 0.10, 0.22, 0.46, 1.00, 2.15, 4.46, 10.00, 21.54, 46.42, 100.00**. The table is partitioned into three (row-wise) sections. In the first section, the columns titled with θ_i are the coefficients learned from the corresponding methods. The column titled with λ/C provides the values of the regularization parameters, whether fixed or learnt, and the test set accuracies are given in the last column. The second section includes RKDA and SVM with kernel and regularization parameter chosen by double cross-validation. We also report the number of times that a particular kernel is selected by cross-validation. The third section shows the accuracies of RKDA and SVM when the kernel is fixed and the regularization parameters chosen by cross-validation. Dashes are used to denote non-applicable items.**

sonar	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	λ^a/C	TSA
QCQP	0	0	0	0	1.673	0.976	0.083	0.214	0.254	0.824	1.0e-5	90.16
SDP_{Kim}	1.552	0.421	1.914	1.200	1.013	3.630	0.271	0	0	0	1.0e-7	88.46
SM1	0	0	0	0.040	3.953	5.514	0.491	0	0	0	1	89.75
SM2	0	0	0	0	2.875	6.765	0.359	0	0	0	1	89.59
SM2_C	0	0.011	0.014	0.084	4.253	6.038	0.570	0.004	0.001	0	5.5e+7	89.84
$\text{RKDA}_{K,\lambda}$ [b]	0	0	0	0	0	9	16	5	0	0	–	88.86
$\text{SVM}_{K,C}$ [c]	0	0	0	0	0	4	16	8	2	0	–	89.27
RKDA_{λ} [d]	53.65	54.95	60.24	73.57	84.95	90.56	89.91	86.99	85.52	84.95	–	–
SVM_{C} [e]	53.65	54.63	59.91	73.41	86.09	89.67	90.65	89.59	86.58	84.22	–	–

[a] We originally assigned $\lambda = 10^{-8}$ for the QCQP formulation. The values shown here were scaled along with the coefficients (θ_1 to θ_{10}) on the same scale for ease of presentation.
[b] The number of times that a kernel is chosen by doubly cross-validated RKDA over 30 randomizations.
[c] The number of times that a kernel is chosen by doubly cross-validated SVM over 30 randomizations.
[d] Accuracies of RKDA when the kernel is fixed to each of the ten pre-specified kernels and λ is chosen by cross-validation.
[e] Accuracies of SVM when the kernel is fixed to each of the ten pre-specified kernels and C is chosen by cross-validation.

chosen 10, 9, 9 times out of 30, respectively) while cross-validated RKDA uses kernels corresponding to θ_6, θ_7, and θ_8 most frequently. Our QCQP formulation gives the kernel corresponds to θ_{10} a large weight while all other methods using linear combination of kernels give this kernel a zero weight. Another interesting observation is that SM1, SM2, and SM2_C give the first kernel a large weight while it performs poorly when used separately. Similar behavior has been observed for the breast cancer data where large weights have been assigned to the second kernel while it is not the best individual kernel. This implies that the best individual kernel may not lead to a large weight when used in combination with others and poorly-performed individual kernel may contain complementary information that is useful when combined with other kernels. Such complementary information can not be incorporated when cross-validation is used to choose a single best kernel. For the ionosphere data, the best three individual kernels chosen by cross-validation are kernels corresponding to θ_5, θ_6, and θ_7. Interestingly, the kernel corresponding to θ_5 has a zero weight for all methods using a linear combination of kernels. Overall, the pattern of the coefficients learned from our QCQP formulation is similar to those of the SVM-based methods.

We compared the running time of the proposed QCQP formulation with two other RKDA-based kernel learning algorithms including the SDP formulation in [10] and doubly cross-validated RKDA ($\text{RKDA}_{K,\lambda}$). Figure 1 shows the running time of 30 different runs of these three methods, where the x-axis denotes the 30 different splits of the data into the training and test sets and the y-axis denotes the running time (in seconds). Table 3 summarizes the average running time over 30 runs. It is clear that the proposed QCQP formulation is much more efficient than the SDP formulation. For example, for the breast cancer data set the average running time over 30 splits for SDP_{Kim} is about $2,000$ seconds while the QCQP formulation takes about 6.4 seconds. Results also show that the QCQP formulation is much more efficient than doubly cross-validated RKDA.

5.2 Experiments on Multi-class Problems

In the multi-class experiment, we compared our formulations with KRDA and SVM with kernels and regularization parameters tuned using double cross-validation. The methods proposed in [10, 12] are restricted to binary-class problems only. We used a total of five data sets for this experiment. The *USPS* handwritten digits database was described in [8]. We choose the first 3, 6, and 8 classes with 100 samples in each class for the experiments. The *wine* and *waveform* data were obtained from UCI Machine Learning Repository and the *satimage* and *segment* were obtained from the STATLOG project. We used the first 3, 5, and 6 classes for the *satimage* data and the first 3 and 4 classes for the *segment* data. For each data set, we randomly partitioned the entire set into two subsets with 60% in the training set and 40% in the test set. Ten RBF kernels, with the σ assigned the same values as in the binary case were constructed from the training set.

Figure 2 shows the classification results (measured in error rate) on ten data sets. We can observe from the figure that the proposed QCQP formulation is very competitive with

Table 2: Comparison of seven methods for the heart, ionosphere, and cancer data sets. The result of SDP_{Kim} on the cancer data set is not shown due to the occurrence of numerical problems when running the SDP solver (SeDuMi, `http://sedumi.mcmaster.ca/`). See the caption and footnotes of Table 1 for explanation.

heart	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	λ/C	TSA
QCQP	1.821	0.206	0	0	0	0.001	0	0	0.048	2.579	1.0e-5	82.65
SDP_{Kim}	8.530	0.333	0.004	0	0	0.272	0.860	0	0	0	1.0e-7	81.98
SM1	7.688	0.479	0.001	0.002	0.002	0.024	1.813	0	0	0	1	82.59
SM2	7.317	0.669	0	0	0	0.029	1.994	0	0	0	1	82.71
SM2_C	6.746	0.626	0	0	0	0.036	1.991	0	0	0	4.4e+5	82.53
$\text{RKDA}_{K,\lambda}$	0	0	0	0	0	8	8	7	4	3	–	76.54
$\text{SVM}_{K,C}$	0	0	0	0	0	0	2	10	9	9	–	82.22
RKDA_{λ}	58.64	65.06	69.62	73.33	77.28	79.13	78.70	77.65	76.79	75.92	–	–
SVM_C	57.96	64.75	71.79	76.60	79.93	80.30	81.66	81.54	82.22	82.59	–	–
ionosphere	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	λ/C	TSA
QCQP	6.477	1.315	0.593	2.007	0	3.068	6.895	0	0	0.023	1.0e-4	95.10
SDP_{Kim}	9.452	0	0	0	0	0.401	0.148	0	0	0	1.0e-7	89.14
SM1	3.553	0.672	0.482	0.240	0	4.828	0.221	0	0	0	1	95.28
SM2	2.883	0.682	0.683	0.196	0	5.305	0.248	0	0	0	1	94.81
SM2_C	3.910	0.714	0.561	0.255	0	5.300	0.256	0	0	0	1.4e+7	95.19
$\text{RKDA}_{K,\lambda}$	0	0	0	2	3	11	12	2	0	0	–	93.62
$\text{SVM}_{K,C}$	0	0	0	0	8	14	4	2	0	2	–	93.52
RKDA_{λ}	65.71	76.47	90.33	92.14	93.33	94.28	93.14	91.71	90.61	89.00	–	–
SVM_C	65.38	66.57	89.38	93.00	94.57	95.04	93.80	93.42	92.61	91.95	–	–
cancer	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_9	θ_{10}	λ/C	TSA
QCQP	0.3692	6.356	1.265	0.517	0.135	0.252	3.085	0.281	0	0.612	1.0e-4	97.05
SDP_{Kim}	—	—	—	—	—	—	—	—	—	—	—	—
SM1	1.797	5.706	0.179	0.008	0	2.308	0	0	0	0	1	97.08
SM2	1.483	5.541	0.402	0.023	0.006	2.527	0.013	0	0	0	1	97.15
SM2_C	1.690	4.855	0.546	0.047	0.003	2.521	0.015	0	0	0	1.0e+4	97.01
$\text{RKDA}_{K,\lambda}$	0	0	1	2	4	4	1	1	3	14	–	95.30
$\text{SVM}_{K,C}$	0	0	0	0	0	9	6	6	7	2	–	96.62
RKDA_{λ}	94.54	95.32	96.05	96.15	96.30	95.74	95.59	95.59	95.49	95.64	–	–
SVM_C	92.21	94.93	96.03	96.30	96.81	96.88	96.86	96.66	96.69	96.64	–	–

Table 3: Comparison of running time (in seconds) of three methods averaged over 30 random partitions.

data	QCQP	SDP_{Kim}	$\text{RKDA}_{K,\lambda}$
sonar	0.61	28.45	6.38
heart	1.13	43.06	9.22
ionosphere	1.98	147.00	21.19
cancer	6.40	2040.60	75.20

the other two methods based on cross-validation. Compared with the other two methods, the proposed method learns a convex linear combination of kernels by avoiding the cross-validation. It is expected to perform even better for heterogeneous data integration, when kernels from multiple data sources contain complementary information.

6. DISCUSSION AND CONCLUSION

We addressed the issue of automated kernel learning for discriminant analysis in this paper. We formulate this kernel learning problem as a convex program and thus a globally optimal solution is guaranteed. Practically, some convex optimization problems, such as SDP are computationally expensive and we proposed a QCQP formulation for kernel learning, which is much more efficient to solve than SDP. While most existing work on kernel learning only deal with binary-class problems, we show that our binary formulation can be extended naturally to the multi-class setting.

We conducted extensive experiments to evaluate the proposed algorithms. The proposed formulations are competitive with the SDP-based formulation, SVM-based methods, and doubly cross-validated SVM and RKDA in classification. In terms of running time, the QCQP formulation is much more efficient than its SDP counterpart. When evaluating the relative importance of each kernel (either used separately or in linear combination), we found that the best individual kernel sometimes coincides with the highly-weighted kernels in linear combination and sometimes disagrees considerably. If complementary information exists among different kernels, the proposed formulations can potentially utilize such information.

There are several directions for future work. Most previous formulations for learning SVM kernels are restricted to the binary-class case. The idea from this paper may be useful for kernel learning in multi-class SVM. Lanckriet *et al.* [12] considered the problem of optimizing the kernel and regularization parameter simultaneously. We plan to investigate the effectiveness of incorporating the learning of the

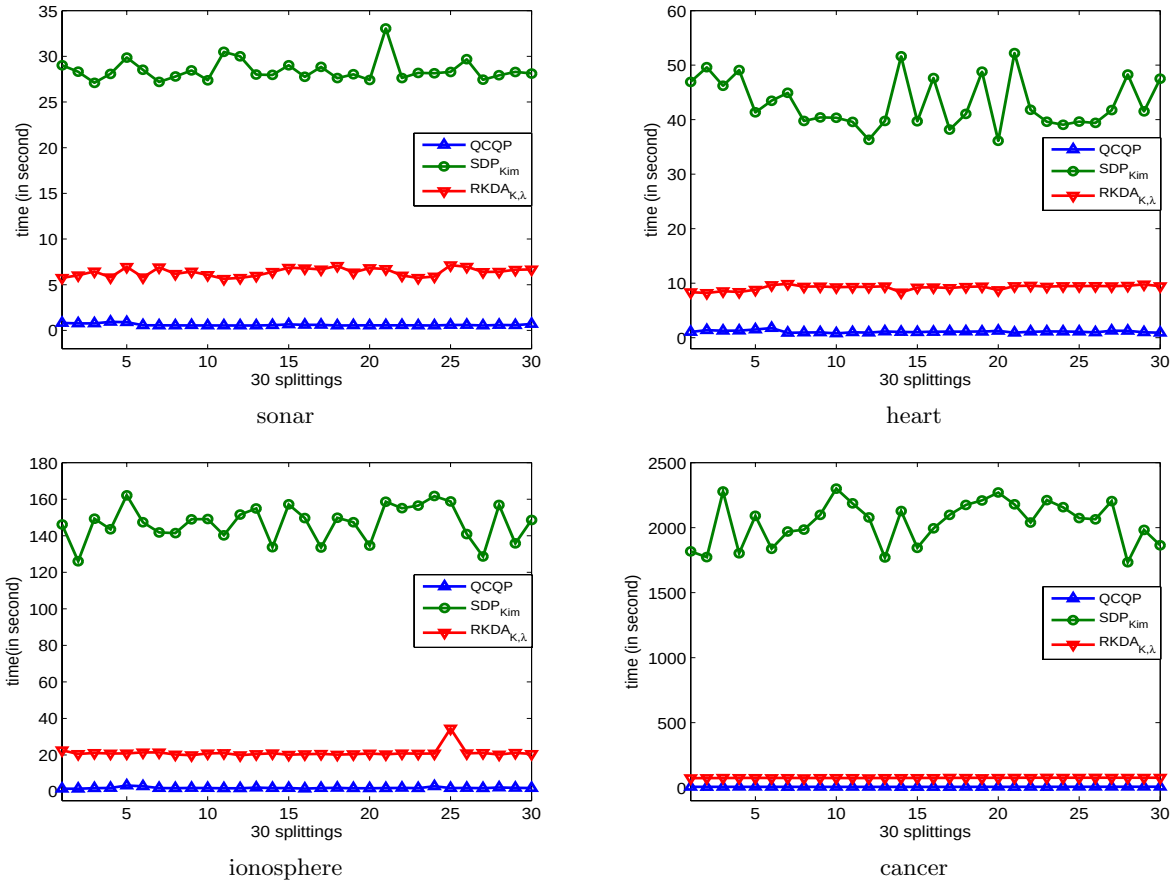

Figure 1: Comparison of running time (in seconds) of three methods on four data sets.

regularization parameter in the framework. The proposed QCQP formulations are still computationally expensive, especially for problems with a large number of samples and a large number of classes. We plan to examine other optimization techniques [23, 30] for efficient multi-class multiple kernel learning in discriminant analysis. The proposed formulations can be applied for heterogeneous data integration from multiple data sources. We plan to apply these approaches to the analysis of biological images [29].

Acknowledgments

This research is sponsored by the Center for Evolutionary Functional Genomics of the Biodesign Institute at Arizona State University and by the National Science Foundation Grant IIS-0612069.

7. REFERENCES

[1] E. D. Andersen and K. D. Andersen. The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In T. T. H. Frenk, K. Roos and S. Zhang, editors, *High Performance Optimization*, pages 197–232. Kluwer Academic Publishers, 2000.

[2] A. Argyriou, R. Hauser, C. Micchelli, and M. Pontil. A DC-programming algorithm for kernel selection. In *Proceedings of the International Conference on Machine Learning*, pages 41–48, 2006.

[3] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. In *Proceedings of the International Conference on Machine Learning*, 2004.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] N. Cristianini and J. Taylor. *An Introduction to Support Vector Machines and other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[6] G. Fung, M. Dundar, J. Bi, and B. Rao. A fast iterative algorithm for Fisher discriminant using heterogeneous kernels. In *Proceedings of the International Conference on Machine Learning*, 2004.

[7] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[8] J. J. Hull. A database for handwritten text recognition research. *IEEE Trans. Pattern Analysis Machine Intelligence*, 16(5):550–554, 1994.

[9] T. Jebara. Multi-task feature and kernel selection for SVMs. In *Proceedings of the International Conference on Machine Learning*, 2004.

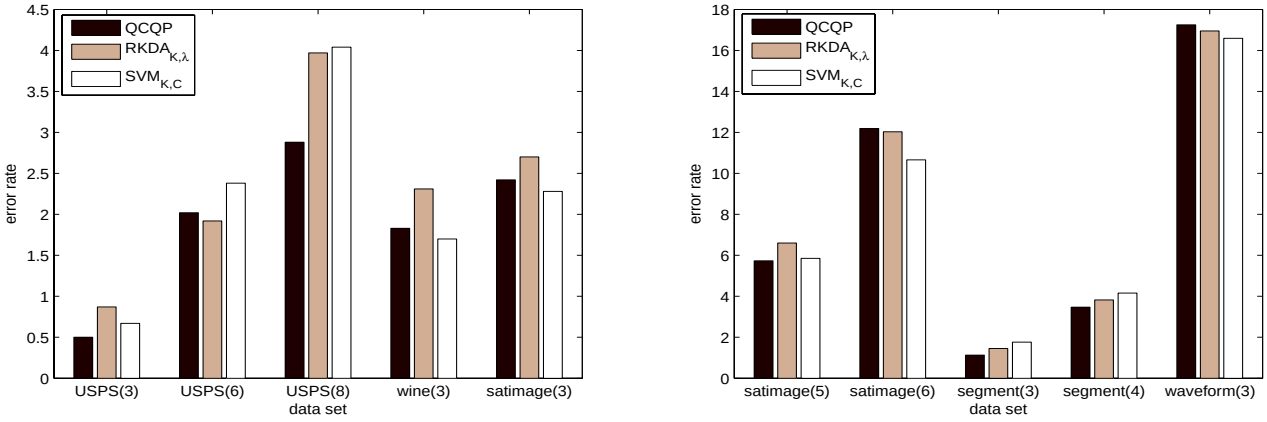

Figure 2: Comparison of the error rate of the proposed multi-class QCQP formulation with doubly cross-validated RKDA and SVM on ten data sets. The numbers in the parenthesis denote the total numbers of classes used in each of the data sets.

[10] S.-J. Kim, A. Magnani, and S. Boyd. Optimal kernel selection in kernel Fisher discriminant analysis. In *Proceedings of the International Conference on Machine Learning*, pages 465–472, 2006.

[11] G. Lanckriet, T. D. Bie, N. Cristianini, M. Jordan, and W. Noble. A statistical framework for genomic data fusion. *Bioinformatics*, 20(16):2626–2635, 2004.

[12] G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[13] D. Lewis, T. Jebara, and W. S. Noble. Nonstationary kernel combination. In *Proceedings of the International Conference on Machine Learning*, pages 553–560, 2006.

[14] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

[15] S. Mika. *Kernel Fisher Discriminants*. PhD thesis, University of Technology, Berlin, Oct. 2002.

[16] S. Mika, G. Rätsch, and K.-R. Müller. A mathematical programming approach to the kernel fisher algorithm. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 591–597. MIT Press, 2001.

[17] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller. Fisher discriminant analysis with kernels. In Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, editors, *Neural Networks for Signal Processing IX*, pages 41–48. IEEE, 1999.

[18] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM Studies in Applied Mathematics. SIAM, 1994.

[19] D. Newman, S. Hettich, C. Blake, and C. Merz. UCI repository of machine learning databases, 1998.

[20] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in kernel methods: support vector learning*, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.

[21] S. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines,Regularization, Optimization and Beyond*. MIT Press, 2002.

[22] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[23] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf. Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, 7:1531–1565, July 2006.

[24] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653, 1999.

[25] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38:49–95, 1996.

[26] V. Vapnik. *Statistical learning theory*. Wiley, New York, 1998.

[27] J. Ye. Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *Journal of Machine Learning Research*, 6:483–502, 2005.

[28] J. Ye, J. Chen, and S. Ji. Discriminant kernel and regularization parameter learning via semidefinite programming. In *Proceedings of the International Conference on Machine Learning*, 2007.

[29] J. Ye, J. Chen, Q. Li, and S. Kumar. Classification of *drosophila* embryonic developmental stage range based on gene expression pattern images. In *Proceedings of the Computational Systems Bioinformatics Conference*, pages 293–298, 2006.

[30] A. Zien and C. Ong. Multiclass multiple kernel learning. In *Proceedings of the International Conference on Machine Learning*, 2007.