BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

L_2-norm multiple kernel learning and its application to biomedical data fusion

Shi Yu[1*], Tillmann Falck[2], Anneleen Daemen[1], Leon-Charles Tranchevent[1], Johan AK Suykens[2], Bart De Moor[1], Yves Moreau[1]

Abstract

Background: This paper introduces the notion of optimizing different norms in the dual problem of support vector machines with multiple kernels. The selection of norms yields different extensions of multiple kernel learning (MKL) such as L_∞, L_1, and L_2 MKL. In particular, L_2 MKL is a novel method that leads to non-sparse optimal kernel coefficients, which is different from the sparse kernel coefficients optimized by the existing L_∞ MKL method. In real biomedical applications, L_2 MKL may have more advantages over sparse integration method for thoroughly combining complementary information in heterogeneous data sources.

Results: We provide a theoretical analysis of the relationship between the L_2 optimization of kernels in the dual problem with the L_2 coefficient regularization in the primal problem. Understanding the dual L_2 problem grants a unified view on MKL and enables us to extend the L_2 method to a wide range of machine learning problems. We implement L_2 MKL for ranking and classification problems and compare its performance with the sparse L_∞ and the averaging L_1 MKL methods. The experiments are carried out on six real biomedical data sets and two large scale UCI data sets. L_2 MKL yields better performance on most of the benchmark data sets. In particular, we propose a novel L_2 MKL least squares support vector machine (LSSVM) algorithm, which is shown to be an efficient and promising classifier for large scale data sets processing.

Conclusions: This paper extends the statistical framework of genomic data fusion based on MKL. Allowing non-sparse weights on the data sources is an attractive option in settings where we believe most data sources to be relevant to the problem at hand and want to avoid a "winner-takes-all" effect seen in L_∞ MKL, which can be detrimental to the performance in prospective studies. The notion of optimizing L_2 kernels can be straightforwardly extended to ranking, classification, regression, and clustering algorithms. To tackle the computational burden of MKL, this paper proposes several novel LSSVM based MKL algorithms. Systematic comparison on real data sets shows that LSSVM MKL has comparable performance as the conventional SVM MKL algorithms. Moreover, large scale numerical experiments indicate that when cast as semi-infinite programming, LSSVM MKL can be solved more efficiently than SVM MKL.

Availability: The MATLAB code of algorithms implemented in this paper is downloadable from http://homes.esat.kuleuven.be/~sistawww/bioi/syu/l2lssvm.html.

Background

In the era of information overflow, data mining and machine learning are indispensable tools to retrieve information and knowledge from data. The idea of incorporating several data sources in analysis may be beneficial by reducing the noise, as well as by improving statistical significance and leveraging the interactions and correlations between data sources to obtain more refined and higher-level information [1], which is known as *data fusion*. In bioinformatics, considerable effort has been devoted to *genomic data fusion*, which is an emerging topic pertaining to a lot of applications. At present, terabytes of data are generated by high-throughput techniques at an increasing rate. In data fusion, these terabytes are further multiplied by the number of data sources or the number of species. A statistical model

* Correspondence: shee.yu@gmail.com
[1] Bioinformatics Group, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, Heverlee B-3001, Belgium

describing this data is therefore not an easy matter. To tackle this challenge, it is rather effective to consider the data as being generated by a complex and unknown black box with the goal of finding a function or an algorithm that operates on an input to predict the output. About 15 years ago, Vapnik [2] introduced the support vector method which makes use of kernel functions. This method has offered plenty of opportunities to solve complicated problems but also brought lots of interdisciplinary challenges in statistics, optimization theory, and the applications therein [3].

Multiple kernel learning (MKL) has been pioneered by Lanckriet *et al.* [4] and Bach *et al.* [5] as an additive extension of single kernel SVM to incorporate multiple kernels in classification. It has also been applied as a statistical learning framework for genomic data fusion [6] and many other applications [7]. The essence of MKL, which is the additive extension of the dual problem, relies only on the kernel representation (kernel trick) while the heterogeneities of data sources are resolved by transforming different data structures (i.e., vectors, strings, trees, graphs, etc.) into kernel matrices. In the dual problem, these kernels are combined into a single kernel, moreover, the coefficients of the kernels are leveraged adaptively to optimize the algorithmic objective, known as *kernel fusion*. The notion of kernel fusion was originally proposed to solve classification problems in computational biology, but recent efforts have lead to analogous solutions for one class [7] and unsupervised learning problems (Yu *et al.*: Optimized data fusion for kernel K-means clustering, submitted). Currently, most of the existing MKL methods are based on the formulation proposed by Lanckriet *et al.* [4], which is clarified in our paper as the optimization of the infinity norm (L_∞) of kernel fusion. Optimizing L_∞ MKL in the dual problem corresponds to posing L_1 regularization on the kernel coefficients in the primal problem. As known, L_1 regularization is characterized by the sparseness of the kernel coefficients [8]. Thus, the solution obtained by L_∞ MKL is also sparse, which assigns dominant coefficients to only one or two kernels. The sparseness is useful to distinguish relevant sources from a large number of irrelevant data sources. However, in biomedical applications, there are usually a small number of sources and most of these data sources are carefully selected and preprocessed. They thus often are directly relevant to the problem. In these cases, a sparse solution may be too selective to thoroughly combine the complementary information in the data sources. While the performance on benchmark data may be good, the selected sources may not be as strong on truly novel problems where the quality of the information is much lower. We may thus expect the performance of such solutions to degrade significantly on actual real-world applications. To address this problem, we propose a new kernel fusion scheme by optimizing the L_2-norm of multiple kernels. The L_2 MKL yields a non-sparse solution, which smoothly distributes the coefficients on multiple kernels and, at the same time, leverages the effects of kernels in the objective optimization. Empirical results show that the L_2-norm kernel fusion can lead to a better performance in biomedical data fusion.

Methods

Acronyms

The symbols and notations used in this paper are defined in Table 1 (in the order of appearance).

Formal definition of the problem

We consider the problem of minimizing a quadratic cost of a real vector in function of $\vec{\alpha}$ and a real positive semi-definite (PSD) matrix Q, given by

$$\begin{aligned} \underset{\vec{\alpha}}{\text{minimize}} \quad & \vec{\alpha}^T Q \vec{\alpha} \\ \text{subject to} \quad & \vec{\alpha} \in \mathcal{C}, \end{aligned} \tag{1}$$

where $\mathcal{C}$ denotes a convex set. Also, PSD implies that $\forall \vec{\alpha}, \vec{\alpha}^T Q \vec{\alpha} \geq 0$. We will show that many machine learning problems can be cast in form (1) with additional constraints on $\vec{\alpha}$. In particular, if we restrict $\vec{\alpha}^T \vec{\alpha} = 1$, the problem in (1) becomes a Rayleigh quotient and leads to the eigenvalue problem. Now we consider a convex parametric linear combination of a set of p PSD matrices Q_j, given by:

$$\Omega \left\{ \sum_{j=1}^{p} \theta_j Q_j \,\middle|\, \forall j,\ \theta_j \geq 0, Q_j \succeq 0 \right\}. \tag{2}$$

To bound the coefficients θ_j, we restrict that, for example, $||\theta_j||_1 = 1$, and (1) can be equivalently rewritten as a min-max problem, given by

$$\begin{aligned} \underset{\vec{\alpha}}{\text{minimize}} \quad \underset{\vec{\theta}}{\text{maximize}} \quad & \vec{\alpha}^T \left(\sum_{j=1}^{p} \theta_j Q_j \right) \vec{\alpha} \\ \text{subject to} \quad & Q_j \succeq 0, \quad j = 1, \ldots, p \\ & \vec{\alpha} \in \mathcal{C}, \\ & \theta_j \geq 0, \quad j = 1, \ldots, p \\ & \sum_{j=1}^{p} \theta_j = 1. \end{aligned} \tag{3}$$

To solve (3), we denote $t = \vec{\alpha}^T \left(\sum_{j=1}^{p} \theta_j Q_j \right) \vec{\alpha}$, the min-max problem can be formulated in a form of quadraticly constrained linear program (QCLP), given by

Table 1 Acronyms

$\vec{\alpha}$	$\mathbb{R}^{N}$	**the dual variable of SVM**
Q	$\mathbb{R}^{N \times N}$	a semi-positive definite matrix
C	$\mathbb{R}^{N}$	a convex set
Ω	$\mathbb{R}^{N \times N}$	a combination of multiple semi-positive definite matrices
j	$\mathbb{N}$	the index of kernel matrices
p	$\mathbb{N}$	the number of kernel matrices
θ	[0, 1]	coefficients of kernel matrices
t	[0, + ∞)	dummy variable in optimization problem
$\vec{s}$	$\mathbb{R}^{p}$	$\vec{s} = \left\{\vec{\alpha}^T Q_1 \vec{\alpha}, \ldots, \vec{\alpha}^T Q_p \vec{\alpha}\right\}^T$
$\vec{v}$	$\mathbb{R}^{p}$	$\vec{v} = \left\{\vec{\alpha}^T K_1 \vec{\alpha}, \ldots, \vec{\alpha}^T K_p \vec{\alpha}\right\}^T$
$\vec{w}$	$\mathbb{R}^{D}$ or $\mathbb{R}^{\Phi}$	the norm vector of the separating hyperplane
$\phi(\cdot)$	$\mathbb{R}^{D} \rightarrow \mathbb{R}^{\Phi}$	the feature map
i	$\mathbb{N}$	the index of training samples
$\vec{x}_i$	$\mathbb{R}^{D}$	the vector of the *i*-th training sample
ρ	$\mathbb{R}$	bias term in 1-SVM
ν	$\mathbb{R}^{+}$	regularization term of 1-SVM
ξ_i	$\mathbb{R}$	slack variable for the *i*-th training sample
K	$\mathbb{R}^{N \times N}$	kernel matrix
$k\left(\vec{x}_i, \vec{x}_j\right)$	$\mathbb{R}^{D} \times \mathbb{R}^{D} \rightarrow \mathbb{R}$	kernel function, $K_{ij} = k\left(\vec{x}_i, \vec{x}_j\right)$
$\vec{z}$	$\mathbb{R}^{D}$	the vector of a test data sample
y_i	-1 or +1	the class label of the *i*-th training sample
Y	$\mathbb{R}^{N \times N}$	the diagonal matrix of class labels $Y = diag(y_1, ..., y_N)$
C	$\mathbb{R}^{+}$	the box constraint on dual variables of SVM
b	$\mathbb{R}^{+}$	the bias term in SVM and LSSVM
$\vec{\gamma}$	$\mathbb{R}^{p}$	$\vec{\gamma} = \left\{\vec{\alpha}^T Y K_1 Y \vec{\alpha}, \ldots, \vec{\alpha}^T Y K_p Y \vec{\alpha}\right\}^T$
k	$\mathbb{N}$	the number of classes
$\vec{\eta}$	$\mathbb{R}^{p}$	$\vec{\eta} = \left\{\sum_{q=1}^{k}\left(\vec{\alpha}_q^T Y_q K_1 Y_q \vec{\alpha}_q\right), \ldots, \sum_{q=1}^{k}\left(\vec{\alpha}_q^T Y_q K_1 Y_q \vec{\alpha}_q\right)\right\}^T$
$\vec{\delta}$	$\mathbb{R}^{p}$	variable vector in SIP problem
u	$\mathbb{R}$	dummy variable in SIP problem
q	$\mathbb{N}$	the index of class number in classification problem, $q = 1, ..., k$
A	$\mathbb{R}^{N \times N}$	$A_j = \sum_{q=1}^{k}\left(\vec{\alpha}_q^T Y_q K_j Y_q \vec{\alpha}_q\right)$
λ	$\mathbb{R}^{+}$	the regularization parameter in LSSVM
e_i	$\mathbb{R}$	the error term of the *i*-th sample in LSSVM
$\vec{\beta}$	$\mathbb{R}^{N}$	the dual variable of LSSVM, $\vec{\beta} = Y\vec{\alpha}$
ϵ	$\mathbb{R}^{+}$	precision value as the stopping criterion of SIP iteration
τ	$\mathbb{N}$	index parameter of SIP iterations
$\vec{g}$	$\mathbb{R}^{p}$	$\vec{g} = \left\{\vec{\beta}^T K_1 \vec{\beta}, \ldots, \vec{\beta}^T K_p \vec{\beta}\right\}^T$

$$
\begin{aligned}
\underset{\vec{\alpha},t}{\text{minimize}} \quad & t \\
\text{subject to} \quad & Q_j \succeq 0, \ j = 1, \ldots, p \\
& \vec{\alpha} \in \mathcal{C}, \\
& t \geq \vec{\alpha}^T Q_j \vec{\alpha}, \ j = 1, \ldots, p.
\end{aligned}
\tag{4}
$$

The optimal solution $\vec{\theta}^*$ in (3) is obtained from the dual variable corresponding to the quadratic constraints in (4). The optimal t^* is equivalent to the *Chebyshev* or L_∞-norm of the vector of quadratic terms, given by:

$$
t^* = \left\| \vec{\alpha}^T Q_j \vec{\alpha} \right\|_\infty = \max\left\{\alpha^T Q_1 \vec{\alpha}, \ldots, \alpha^T Q_p \vec{\alpha}\right\}. \tag{5}
$$

The L_∞-norm is the upper bound w.r.t. the constraint $\sum_{j=1}^{p} \theta_j = 1$ because

$$
\vec{\alpha}^T \left(\sum_{j=1}^{p} \theta_j Q_j \right) \vec{\alpha} \leq t^*. \tag{6}
$$

Apparently, suppose the optimal $\vec{\alpha}^*$ is given, optimizing the L_∞-norm in (5) will pick the single term with the maximal value, and the optimal solution of the coefficients is more likely to be sparse. An alternative solution to (3) is to introduce a different constraint on the coefficients, for example, $||\theta_j||_2 = 1$. We thus propose a new extension of the problem in (1), given by

$$\begin{aligned}\underset{\vec{\alpha}}{\text{minimize}}\ \underset{\vec{\theta}}{\text{maximize}}\ & \vec{\alpha}^T\left(\sum_{j=1}^{p}\theta_j Q_j\right)\vec{\alpha}\\ \text{subject to}\ & Q_j \succeq 0,\ \ j=1,\dots,p\\ & \vec{\alpha}\in C,\\ & \theta_j \geq 0,\ \ j=1,\dots,p\\ & \left\|\theta_j\right\|_2 = 1.\end{aligned} \tag{7}$$

This new extension is analogously solved as a QCLP problem with modified constraints, given by

$$\begin{aligned}\underset{\vec{\alpha},\eta}{\text{minimize}}\ & \eta\\ \text{subject to}\ & Q_j \succeq 0,\ \ j=1,\dots p\\ & \vec{\alpha}\in \mathcal{C},\\ & \eta \geq \left\|\vec{s}\right\|_2,\ \ \ j=1,\dots,p,\end{aligned} \tag{8}$$

where $\vec{s}=\left\{\vec{\alpha}^T Q_1\vec{\alpha},\dots,\vec{\alpha}^T Q_p\vec{\alpha}\right\}^T$. The proof that (8) is the solution of (7) is given in the following theorem.

Theorem 0.1 *The QCLP problem in (8) equivalently solves the problem in (7).*

Proof Given two vectors $\{x_1, \dots, x_p\}$, $\{y_1, \dots, y_p\}$, $x_j, y_j \in \mathbb{R}$, $j = 1, \dots, p$, the Cauchy-Schwarz inequality states that

$$0 \leq \left(\sum_{j=1}^{p} x_j y_j\right)^2 \leq \sum_{j=1}^{p} x_j^2 \sum_{j=1}^{p} y_j^2, \tag{9}$$

with as equivalent form:

$$0 \leq \left[\left(\sum_{j=1}^{p} x_j y_j\right)^2\right]^{\frac{1}{2}} \leq \left[\sum_{j=1}^{p} x_j^2 \sum_{j=1}^{p} y_j^2\right]^{\frac{1}{2}}. \tag{10}$$

Let us denote $x_j = \theta_j$ and $y_j = \vec{\alpha}^T Q_j \vec{\alpha}$, (10) becomes

$$0 \leq \sum_{j=1}^{p}\left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha}\right) \leq \left[\sum_{j=1}^{p}\theta_j^2 \sum_{j=1}^{p}\left(\vec{\alpha}^T Q_j \vec{\alpha}\right)^2\right]^{\frac{1}{2}}. \tag{11}$$

Since $||\theta_j||_2 = 1$, (11) is equivalent to

$$0 \leq \sum_{j=1}^{p}\left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha}\right) \leq \left[\sum_{j=1}^{p}\left(\vec{\alpha}^T Q_j \vec{\alpha}\right)^2\right]^{\frac{1}{2}}. \tag{12}$$

Therefore, given $\vec{s}=\left\{\vec{\alpha}^T Q_1\vec{\alpha},\dots,\vec{\alpha}^T Q_p\vec{\alpha}\right\}^T$, the additive term $\sum_{j=1}^{p}\left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha}\right)$ is bounded by the L_2-norm $||\,\vec{s}\,||_2$.

Moreover, it is easy to prove that when $\theta_j^* = \vec{\alpha}^T Q_j \vec{\alpha} \,/\, \|\vec{s}\|_2$, the parametric combination reaches the upperbound and the equality holds. Optimizing this L_2-norm results in a non-sparse solution in θ_j. In order to distinguish this from the solution obtained by (3) and (4), we denote it as the L_2-norm approach. It can also easily be seen (not shown here) that the L_1-norm approach is simply averaging the quadratic terms with uniform coefficients.

The L_2-norm bound is also generalizable to any positive real number $n \geq 1$, defined as L_n-norm MKL. Recently, the similar topic is also investigated by [9] and a solution is proposed to solve the primal MKL problem. In this paper, we will show that our primal-dual interpretation of MKL is also extendable to the n-norm. Let us assume that $\vec{\theta}$ is regularized by the L_m-norm as $||\,\vec{\theta}\,||_m = 1$, then the L_m-norm extension of equation (7) is given by

$$\begin{aligned}\underset{\vec{\alpha}}{\text{minimize}}\ \underset{\vec{\theta}}{\text{maximize}}\ & \vec{\alpha}^T\left(\sum_{j=1}^{p}\theta_j Q_j\right)\vec{\alpha}\\ \text{subject to}\ & Q_j \succeq 0,\ j=1,\dots p\\ & \vec{\alpha}\in \mathcal{C},\\ & \theta_j \geq 0,\ j=1,\dots,p\\ & \left\|\vec{\theta}\right\|_m = 1.\end{aligned} \tag{13}$$

In the following theorem, we prove that (13) can be equivalently solved as a QCLP problem, given by

$$\begin{aligned}\underset{\vec{\alpha},\eta}{\text{minimize}}\ & \eta\\ \text{subject to}\ & Q_j \succeq 0,\ \ j=1,\dots,p\\ & \vec{\alpha}\in \mathcal{C},\\ & \eta \geq \left\|\vec{s}\right\|_{n,}\end{aligned} \tag{14}$$

where $\vec{s}=\left\{\vec{\alpha}^T Q_1\vec{\alpha},\dots,\vec{\alpha}^T Q_p\vec{\alpha}\right\}^T$ and the constraint is in L_n-norm, moreover, $n=\frac{m}{m-1}$. The problem in (14) is convex and can be solved by cvx toolbox [10,11].

Theorem 0.2 *If the coefficient vector $\vec{\theta}$ is regularized by a L_m-norm in (13), the problem can be solved as a convex programming problem in (14) with L_n-norm constraint. Moreover, $n=\frac{m}{m-1}$.*

Proof We generalize the Cauchy-Schwarz inequality to Hölder's inequality. Let m, $n > 1$ be two numbers that satisfy $\frac{1}{m}+\frac{1}{n}=1$. Then

$$0 \leq \sum_{j=1}^{p} x_j y_j \leq \left(\sum_{j=1}^{p} x_j^m\right)^{\frac{1}{m}}\left(\sum_{j=1}^{p} y_j^n\right)^{\frac{1}{n}}. \tag{15}$$

Let us denote $x_j = \theta_j$ and $\gamma_j = \vec{\alpha}^T Q_j \vec{\alpha}$, (2) becomes

$$0 \le \sum_{j=1}^{p} \left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha} \right) \le \left(\sum_{j=1}^{p} \theta_j^m \right)^{\frac{1}{m}} \left[\sum_{j=1}^{p} \left(\vec{\alpha}^T Q_j \vec{\alpha} \right)^n \right]^{\frac{1}{n}}. \tag{16}$$

Since $|| \vec{\theta} ||_m = 1$, therefore the term $\left(\sum_{j=1}^{p} \theta_j^m \right)^{\frac{1}{m}}$ can be omitted in the equation, so (3) is equivalent to

$$0 \ge \sum_{j=1}^{p} \left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha} \right) \le \left[\sum_{j=1}^{p} \left(\vec{\alpha}^T Q_j \vec{\alpha} \right)^n \right]^{\frac{1}{n}}. \tag{17}$$

Due to the condition that $\frac{1}{m} + \frac{1}{n} = 1$, so $n = \frac{m}{m-1}$, we prove that with the L_m-norm constraint posed on $\vec{\theta}$, the additive multiple kernel term $\sum_{j=1}^{p} \left(\theta_j \vec{\alpha}^T Q_j \vec{\alpha} \right)$ is bounded by the L_n-norm of the vector $\left\{ \vec{\alpha}^T Q_1 \vec{\alpha}, \dots, \vec{\alpha}^T Q_n \vec{\alpha} \right\}^T$. Moreover, we have $n = \frac{m}{m-1}$.

In this section, we have explained the L_∞, L_1, L_2, and L_n-norm approaches to extend the basic problem in (1) to multiple matrices Q_j. These approaches differed mainly on the constraints applied on the coefficients. To clarify the difference of notations used in this paper with the common interpretations of L_1 and L_2 regularization on $\vec{\theta}$, we illustrate the mapping of our L_∞, L_1, L_2, and L_n notations between the common interpretations of coefficient regularization. As shown in Table 2, the notations used in this paper are interpreted in the dual space and are equivalent to regularization of kernel coefficients in the primal space. The advantage of dual space interpretation is that we can easily extend the analogue solution to various machine learning algorithms. To keep the discussion concise, we will from now on mainly focus on comparing the L_∞, L_1 and L_2 in the dual problems and present the solutions in the dual space.

Table 2 The notation used in this paper is based on the dual problem and can be linked to a equivalent notation in the primal problem

	primal problem	dual problem
variable	θ_j	$\vec{\alpha}^T K_j \vec{\alpha}$
L_∞	$\left\| \vec{\theta} \right\| = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_\infty$
L_1	$\theta_j = \bar{\theta}$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_1$
L_2	$\left\| \vec{\theta} \right\|_2 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_2$
$L_{1.5}$	$\left\| \vec{\theta} \right\|_3 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_{1.5}$
$L_{1.3333}$	$\left\| \vec{\theta} \right\|_4 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_{1.3333}$
$L_{1.25}$	$\left\| \vec{\theta} \right\|_5 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_{1.25}$
$L_{1.2}$	$\left\| \vec{\theta} \right\|_6 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_{1.2}$
$L_{1.1667}$	$\left\| \vec{\theta} \right\|_7 = 1$	$\max \left\| \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \dots, \vec{\alpha}^T K_j \vec{\alpha} \right\} \right\|_{1.1667}$

Next, we will investigate several concrete kernel fusion algorithms and will propose the corresponding L_2 solutions.

One class SVM kernel fusion for ranking

The primal problem of one class SVM (1-SVM) is defined by Tax and Duin [12] and Schölkopf *et al.* [13] as

$$\boxed{\text{P:}} \quad \begin{aligned} & \underset{\vec{w},\xi,\rho}{\text{minimize}} \ \frac{1}{2}\vec{w}^T\vec{w} - \frac{1}{\nu l}\sum_{i-1}^{l}\xi_i - \rho \\ & \text{subject to } \vec{w}^T\phi(\vec{x}_i) \ge \rho - \xi_i, \ i = 1, \dots, N \\ & \xi_i \ge 0, i = 1, \dots, N \end{aligned} \tag{18}$$

where $\vec{w}$ is the norm vector of the separating hyperplane, $\vec{x}_i$ are the training samples, ν is the regularization constant penalizing outliers in the training samples, $\phi(\cdot)$ denotes the feature map, ρ is a bias term, ξ_i are slack variables, and N is the number of training samples. Taking the conditions for optimality from the Lagrangian, one obtains the dual problem, given by:

$$\boxed{\text{D:}} \quad \begin{aligned} & \underset{\vec{\alpha}}{\text{minimize}} \, \vec{\alpha}^T K \vec{\alpha} \\ & \text{subject to } 0 \le \alpha_i \le \frac{1}{\nu N}, \ i = 1, \dots, N, \\ & \sum_{i=1}^{N} \alpha_i = 1, \end{aligned} \tag{19}$$

where α_i are dual variables, K represents the kernel matrix obtained by the inner product between any pair of samples specified by a kernel function $k\left(\vec{x}_i, \vec{x}_j\right) = \phi\left(\vec{x}_i\right)^T \phi\left(\vec{x}_j\right)$, $i, j = 1, \dots, N$. To incorporate multiple kernels in (19), De Bie *et al.* proposed a solution [7] with the dual problem formulated as

$$\boxed{\text{D:}} \quad \begin{aligned} & \underset{\vec{\alpha}}{\text{minimize}} \ t \\ & \text{subject to } t \ge \vec{\alpha}^T K_j \vec{\alpha}, \qquad j = 1, \dots, p \\ & 0 \le \alpha_i \le \frac{1}{\nu N}, \qquad i = 1, \dots, N \\ & \sum_{i=1}^{N} \alpha_i = 1, \end{aligned} \tag{20}$$

where p is the number of data sources and K_j is the j-th kernel matrix. The formulation exactly corresponds to the L_∞ solution of the problem defined in the previous section (the PSD constraint is implied in the kernel matrix) with additional constraints imposed on $\vec{\alpha}$. The optimal coefficients θ_j are used to combine multiple kernels as

$$\Omega = \left\{ \sum_{j=1}^{p} \theta_j K_j \middle| \sum_{j=1}^{p} \theta_j = 1, \forall_{j=1}, \theta_j \geq 0 \right\}, \tag{21}$$

and the ranking function is given by

$$f(\vec{z}) \frac{1}{\sqrt{\vec{\alpha}^T \Omega_N \vec{\alpha}}} \sum_{i=1}^{N} \alpha_i \Omega(\vec{z}, \vec{x}_i), \tag{22}$$

where Ω_N is the combined kernel of training data $\vec{x}_i$, $i = 1, ..., N$, $\vec{z}$ is the test data point to be ranked, $\Omega(\vec{x}, \vec{x}_i)$ is the kernel function applied on test data and training data, $\vec{\alpha}$ is the dual variable solved as (20). De Bie *et al.* applied the method in the application of disease gene prioritization, where multiple genomic data sources are combined to rank a large set of test genes using the 1-SVM model trained from a small set of training genes which are known to be relevant for certain diseases. The L_∞ formulation in their approach yields a sparse solution when integrating genomic data sources (see Figure 2 of [7]). To avoid this disadvantage, they proposed a regularization method by restricting the minimal boundary on the kernel coefficients, notated as θ_{min}, to ensure the minimal contribution of each genomic data source to be θ_{min}/p. According to their experiments, the regularized solution performed best, being significantly better than the sparse integration and the average combination of kernels.

Instead of setting the ad hoc parameter θ_{min}, one can also straightforwardly propose an L_2-norm approach to solve the identical problem, given by

$$\begin{aligned} \boxed{\text{D}:} \quad & \underset{\alpha}{\text{minimize}}\, t \\ & \text{subject to}\, t \geq \|\vec{v}\|_2, \\ & 0 \leq \alpha_i \leq \frac{1}{\nu N}, i = 1, \ldots, N \\ & \sum_{i=1}^{N} \alpha_i = 1, \end{aligned} \tag{23}$$

where $\vec{v} = \left\{ \vec{\alpha}^T K_1 \vec{\alpha}, \ldots, \vec{\alpha}^T K_p \vec{\alpha} \right\}^T, \vec{v} \in \mathbb{R}^p$. The problem above is a QCLP problem and can be solved by conic optimization solvers such as Sedumi [14]. In (23), the first constraint represents a Lorentz cone and the second constraint corresponds to p number of rotated Lorentz cones (R cones). The optimal kernel coefficients θ_j correspond to the dual variables of the R cones with $||\theta||_2 = 1$. In this L_2-norm approach, the integrated kernel Ω is combined by different θ_j^* and the same scoring function as in (22) is applied on the different solutions of $\vec{\alpha}$ and Ω.

Support vector machine MKL for classification

The notion of MKL is originally proposed in a binary SVM classification, where the primal objective is given by

$$\begin{aligned} \boxed{\text{P}:} \quad & \underset{\vec{w},b,\xi}{\text{minimize}} && \frac{1}{2}\vec{w}^T\vec{w} + C\sum_{i=1}^{N}\xi_i \\ & \text{subject to} && y_i\left[\vec{w}^T\phi(\vec{x}_i) + b\right] \geq 1 - \xi_i, \\ & && i = 1, \ldots, N \\ & && \xi_i \geq 0, \quad i = 1, \ldots, N, \end{aligned} \tag{24}$$

where $\vec{x}_i$ are data samples, $\varphi(\cdot)$ is the feature map, y_i are class labels, $C > 0$ is a positive regularization parameter, ξ_i are slack variables, $\vec{w}$ is the norm vector of the separating hyperplane, and b is the bias. This problem is convex and can be solved as a dual problem, given by

$$\begin{aligned} \boxed{\text{D}:} \quad & \underset{\vec{\alpha}}{\text{minimize}} && \frac{1}{2}\vec{\alpha}^T YKY\vec{\alpha} - \vec{\alpha}^T\vec{1} \\ & \text{subject to} && (Y\vec{\alpha})^T\vec{1} = 0 \\ & && 0 \leq \alpha_i \leq C, i = 1, \ldots, N, \end{aligned} \tag{25}$$

where $\vec{\alpha}$ are the dual variables, $Y = diag(y_1, ..., y_N)$, K is the kernel matrix, and C is the upperbound of the box constraint on the dual variables. To incorporate multiple kernels in (25), Lanckriet *et al.* [6,4] and Bach *et al.* [5] proposed a multiple kernel learning (MKL) problem as follows:

$$\begin{aligned} \boxed{\text{D}:} \quad & \underset{t,\vec{\alpha}}{\text{minimize}} && \frac{1}{2}t - \vec{\alpha}^T\vec{1} \\ & \text{subject to} && (Y\vec{\alpha})^T\vec{1} = 0 \\ & && 0 \leq \alpha_i \leq C, i = 1, \ldots, N \\ & && t \geq \vec{\alpha}^T YK_jY\vec{\alpha},\ j = 1, \ldots, p, \end{aligned} \tag{26}$$

where p is the number of kernels. (26) optimizes the L_∞-norm of the set of kernel quadratic terms. Based on the previous discussions, the L_2-norm solution is analogously given by

$$\begin{aligned} \boxed{\mathrm{D:}}\quad & \underset{t,\vec{\alpha}}{\text{minimize}} && \frac{1}{2}t - \vec{\alpha}^{T}\vec{1} \\ & \text{subject to} && \left(Y\vec{\alpha}\right)^{T}\vec{1} = 0 \\ &&& 0 \le \alpha_i \le C,\ i = 1,\dots,N \\ &&& t \ge \|\vec{\gamma}\|_2, \end{aligned} \tag{27}$$

where $\vec{\gamma} = \left\{\vec{\alpha}^{T}YK_1Y\vec{\alpha},\dots,\vec{\alpha}^{T}YK_pY\vec{\alpha}\right\}^{T}, \vec{\gamma} \in \mathbb{R}^{p}$. Both formulations in (26) and (27) can be efficiently solved as second order cone programming (SOCP) problems by a conic optimization solver (i.e., Sedumi [14]) or as QCQP problems by a general QP solver (i.e., MOSEK [15]). It is also known that a binary MKL problem can also be formulated as Semi-definite Programming (SDP), as proposed by Lanckriet *et al.* [4] and Kim *et al.* [16]. However, in a multi-class problem, SDP problems are computationally prohibitive due to the presence of PSD constraints and can only be solved approximately by relaxation [17]. On the contrary, the QCLP and QCQP formulations of binary classification problems can be easily extended to a multi-class setting using the one-versus-all (1vsA) coding, i.e., solving the problem of k classes as k number of binary problems. Therefore, the L_∞ multi-class SVM MKL is then formulated as

$$\begin{aligned} \boxed{\mathrm{D:}}\quad & \underset{t,\vec{\alpha}}{\text{minimize}} && \frac{1}{2}t - \sum_{q=1}^{k}\vec{\alpha}_q^{T}\vec{1} \\ & \text{subject to} && \left(Y_q\vec{\alpha}_q\right)^{T}\vec{1} = 0, \quad q = 1,\dots,k \\ &&& 0 \le \alpha_{iq} \le C, \qquad i = 1,\dots,N, \\ &&& q = 1,\dots,k \\ &&& t \ge \sum_{q=1}^{k}\left(\vec{\alpha}_q^{T}Y_qK_jY_q\vec{\alpha}_q\right), \\ &&& j = 1,\dots,p. \end{aligned} \tag{28}$$

The L_2 multi-class SVM MKL is given by

$$\begin{aligned} \boxed{\mathrm{D:}}\quad & \underset{t,\vec{\alpha}}{\text{minimize}} && \frac{1}{2}t - \sum_{q=1}^{k}\vec{\alpha}_q^{T}\vec{1} \\ & \text{subject to} && \left(Y_q\vec{\alpha}_q\right)^{T}\vec{1} = 0, q = 1,\dots,k \\ &&& 0 \le \alpha_{iq} \le C,\ i = 1,\dots,N, \\ &&& q = 1,\dots,k \\ &&& t \ge \|\vec{\eta}\|_2, \end{aligned} \tag{29}$$

where

$$\vec{\eta} = \left\{\sum\nolimits_{q=1}^{k}\left(\vec{\alpha}_q^{T}Y_qK_1Y_q\vec{\alpha}_q\right),\dots,\sum\nolimits_{q=1}^{k}\left(\vec{\alpha}_q^{T}Y_qK_pY_q\vec{\alpha}_q\right)\right\}^{T}, \vec{\eta} \in \mathbb{R}^{p}.$$

SIP formulation for SVM MKL on larger scale data

Unfortunately, the kernel fusion problem becomes challenging on large scale data because it may scale up in three dimensions: the number of data points, the number of classes, and the number of kernels. When these dimensions are all large, memory issues may arise as the kernel matrices need to be stored in memory. Though it is feasible to approximate the kernel matrices by a low rank decomposition (i.e., incomplete Cholesky decomposition) and to reduce the computational burden of conic optimization using these low rank matrices, conic problems involve a large amount of variables and constraints and it is usually less efficient than QCQP. Moreover, the precision of the low rank approximation relies on the assumption that the eigenvalues of kernel matrices decay rapidly, which may not always be true when the intrinsic dimensions of the kernels are large. To tackle the computational burden of MKL, Sonnenburg *et al.* reformulated the QP problem as semi-infinite programming (SIP) and approximated the QP solution using a bi-level strategy (wrapper method) [18]. The standard form of SIP is given by

$$\begin{aligned} & \underset{\vec{\delta}}{\text{maximize}} && \vec{c}^{T}\vec{\delta} \\ & \text{subject to} && f_t\left(\vec{\delta}\right) \le 0, \forall t \in \Upsilon, \end{aligned} \tag{30}$$

where the constraint functions in $f_t(\vec{\delta})$ can be either linear or quadratic and there are infinite number of them in $\forall t \in \Upsilon$. To solve it, a *discretization* method is usually applied, which is briefly summarized as follows [19-21]:

1. Choose a finite subset $\mathcal{N} \subset \Upsilon$.
2. Solve the convex programming problem

$$\underset{\vec{\delta}}{\text{maximize}} \quad \vec{c}^{T}\vec{\delta} \tag{31}$$

$$\text{subject to} \quad f_t(\vec{\delta}) \le 0, \quad t \in \mathcal{N}. \tag{32}$$

3. If the solution of 2 is not satisfactorily close to the original problem then choose a larger, but still finite subset $\mathcal{N}$ and repeat from Step 2.

The convergence of SIP and the accuracy of the discretization method have been extensively described (see [19-21]). As proposed by Sonnenburg *et al.* [18], the multi-class SVM MKL objective in (26) can be formulated as a SIP problem, given by

$$\begin{aligned}
&\underset{\vec{\theta}}{\text{maximize}} && u\\
&\text{subject to} && \vec{\theta}_j \geq 0, \quad j = 1, \cdots, p\\
& && \sum_{j=1}^{p} \theta_j = 1,\\
& && \sum_{j=1}^{p} \theta_j f_j\left(\vec{\alpha}_q\right) \geq u, \forall \vec{\alpha}_q, \quad q = 1, \ldots, k \qquad (33)\\
& && f_j\left(\vec{\alpha}_q\right) = \sum_{q=1}^{k}\left(\frac{1}{2}\vec{\alpha}_q^T Y_q K_j Y_q \vec{\alpha}_q - \vec{\alpha}_q^T \vec{1}\right)\\
& && 0 \leq \alpha_{iq} \leq C, \ i = 1, \cdots, N, \ q = 1, \cdots, k\\
& && \left(Y_q \vec{\alpha}_1\right)^T \vec{1}\, q = 1, \cdots, k.
\end{aligned}$$

The SIP problem above is solved as a bi-level algorithm for which the pseudo code is presented in Algorithm 1 in the Appendix. In each loop τ, Step 1 optimizes $\vec{\theta}^{(\tau)}$ and $u^{(\tau)}$ for a restricted subset of constraints as a linear programming. Step 3 is an SVM problem with a single kernel and generates a new $\vec{\alpha}^{(\tau)}$. If $\vec{\alpha}^{(\tau)}$ is not satisfied by the current $\vec{\theta}^{(\tau)}$ and $u^{(\tau)}$, it will be added successively to step 1 until all constraints are satisfied. The starting points $\vec{\alpha}_q^{(0)}$ are randomly initialized and SIP always converges to a identical result.

Algorithm 1 is also applicable to the L_2-norm situation of SVM MKL, whereas the non-convex constraint $\left\|\vec{\theta}\right\|_2 = 1$ in Step 1 needs to be relaxed as $\left\|\vec{\theta}\right\|_2 \leq 1$, and the $f_j(\vec{\alpha})$ term in (32) is modified as only containing the quadratic term. The SIP formulation for L_2-norm SVM MKL is given by

$$\begin{aligned}
&\underset{\vec{\theta}, u}{\text{maximize}} && u\\
&\text{subject to} && \vec{\theta}_j \geq 0, j = 1, \cdots, p,\\
& && \left\|\vec{\theta}\right\|_2 \leq 1,\\
& && \sum_{j=1}^{p} \theta_j f_j\left(\vec{\alpha}_q\right) - \sum_{q=1}^{k} \alpha_q^T \vec{1} \geq u, \qquad (34)\\
& && \forall \vec{\alpha}_q, \ q = 1, \ldots, k\\
& && f_j\left(\vec{\alpha}_q\right) = \frac{1}{2}\sum_{q=1}^{k}\left(\vec{\alpha}_q^T Y_q K_j Y_q \vec{\alpha}_q\right), \ j = 1, \ldots, p\\
& && 0 \leq \alpha_{iq} \leq C, \ i = 1, \ldots, N, \ q = 1, \ldots, k\\
& && \left(Y_q \vec{\alpha}_q\right)^T \vec{1} = 0, \ q = 1, \ldots, k.
\end{aligned}$$

With these modifications, Step 1 of Algorithm 1 becomes a QCLP problem given by

$$\begin{aligned}
&\underset{\theta, u}{\max imize} && u\\
&\textit{subject to} && \frac{1}{2}\sum_{j=1}^{p} \theta_j A_j - \vec{\alpha}^T \vec{1} \geq u, \qquad (35)\\
& && 1 \geq \theta_1^2 + \ldots + \theta_p^2,
\end{aligned}$$

where $A_j = \sum_{q=1}^{k}\left(\vec{\alpha}_q^T Y_q K_j Y_q \vec{\alpha}_q\right)$ and $\vec{\alpha}$ is a given value. Moreover, the PSD property of kernel matrices ensures that $A_j \geq 0$, thus the optimal solution always satisfies $\left\|\vec{\theta}\right\|_2 = 1$.

In the SIP formulation, the SVM MKL is solved iteratively as two components. The first component is a single kernel SVM, which is solved more efficiently when the data scale is larger then thousands of data points (and smaller than ten thousands) and, requires much less memory than the QP formulation. The second component is a small scale problem, which is a linear problem in L_∞ case and a QCLP problem in the L_2 approach. As shown, the complexity of the SIP based SVM MKL is mainly determined by the burden of a single kernel SVM multiplied by the number of iterations. This has inspired us to adopt more efficient single SVM learning algorithms to further improve the efficiency. The least squares support vector machines (LSSVM) [22] is known for its simple differentiable cost function, the equality constraints in the separating hyperplane and its solution based on linear equations, which is preferable for large scaler problems. Next, we will investigate the MKL solutions issue using LSSVM formulations.

Least squares SVM MKL for classification

In LSSVM, the primal problem is given by [22]

$$\begin{aligned}
&\boxed{\text{P}:} \quad \underset{\vec{w}, b, \vec{e}}{\text{minimize}} && \frac{1}{2}\vec{w}^T \vec{w} + \frac{1}{2}»\text{e}^T \vec{e}\\
&\qquad\quad \text{subject to} && y_{\text{i}}\left[\vec{w}^T \phi\left(\vec{x}_i + b\right)\right] = 1 - e_{i,} \qquad (36)\\
& && i = 1, \ldots, N,
\end{aligned}$$

where most of the variables are defined in a similar way as in (24). The main difference is that the nonnegative slack variable ξ is replaced by a squared error term $\vec{e}^T\vec{e}$ and the inequality constraints are modified as equality ones. Taking the conditions for optimality from the Lagrangian, eliminating $\vec{w}, \vec{e}$, defining $\vec{y} = \left[y_{1,\ldots} y_N\right]^T = [y_1, \ldots, y_N]$ and $Y = diag(y_1, \ldots, y_N)$, one obtains the following linear system [22]:

$$\boxed{\mathrm{D:}}\left[\begin{array}{c|c} 0 & \vec{y}^{T} \\ \hline \vec{y} & Y\,KY+I/\lambda \end{array}\right]\left[\begin{array}{c} b \\ \hline \vec{\alpha} \end{array}\right]=\left[\begin{array}{c} 0 \\ \hline \vec{1} \end{array}\right], \tag{37}$$

where $\vec{\alpha}$ are unconstrained dual variables. Without the loss of generality, we denote $\vec{\beta}=Y\vec{\alpha}$ and rewrite (37) as

$$\boxed{\mathrm{D\,:}}\left[\begin{array}{c|c} 0 & \vec{1}^{T} \\ \hline \vec{1} & K+Y^{-2}/\lambda \end{array}\right]\left[\begin{array}{c} b \\ \hline \vec{\beta} \end{array}\right]=\left[\begin{array}{c} 0 \\ \hline Y^{-1}{}_{1} \end{array}\right]. \tag{38}$$

In (38), we add an additional constraint as $Y^{-2} = I$ then the coefficient becomes a static value in the multi-class case. In 1vsA coding, (37) requires to solve k number of linear problems whereas in (38), the coefficient matrix is only factorized once such that the solution of $\vec{\beta}_q$ w.r.t. the multi-class label vectors $\vec{y}_q$ is very efficient to obtain. The constraint $Y^{-2} = I$ can be simply satisfied by assuming the class labels to be -1 and +1. Thus, from now on, we assume $Y^{-2} = I$ in the following discussion.

To incorporate multiple kernels in LSSVM classification, the L_∞-norm approach is a QP problem, given by (assuming $Y^{-2} = I$)

$$\begin{aligned} \underset{\vec{\alpha},t}{\text{minimize}} \quad & \frac{1}{2}t+\frac{1}{2\lambda}\vec{\beta}^T\vec{\beta}-\vec{\beta}^T Y^{-1}\vec{1} \\ \text{subject to} \quad & \sum_{i=1}^{N}\beta_i=0, \\ & t\geq\vec{\beta}^T K_j\vec{\beta},\ j=1,\dots,p. \end{aligned} \tag{39}$$

The L_2-norm approach is analogously formulated as

$$\begin{aligned} \underset{\vec{\alpha},t}{\text{minimize}} \quad & \frac{1}{2}t+\frac{1}{2\lambda}\vec{\beta}^T\vec{\beta}-\vec{\beta}^T Y^{-1}\vec{1} \\ \text{subject to} \quad & \sum_{i=1}^{N}\beta_i=0, \\ & t\geq\|\vec{g}\|_2,\ j=1,\dots,p, \end{aligned} \tag{40}$$

where $\vec{g}=\left\{\beta^T K_1\vec{\beta},\dots,\beta^T K_p\vec{\beta}\right\}^T,\vec{g}\in\mathbb{R}^p$. The λ parameter regularizes the squared error term in the primal objective in (36) and the quadratic term $\vec{\beta}^T\vec{\beta}$ in the dual problem. Usually, the optimal λ needs to be selected empirically by cross-validation. In the kernel fusion of LSSVM, we can alternatively transform the effect of regularization as an identity kernel matrix in $\frac{1}{2}\vec{\beta}^T\left(\sum_{j=1}^{p}K_j+\theta_{p+1}I\right)\vec{\beta}$, where $\theta_{p+1} = 1/\lambda$. Then the MKL problem of combining p kernels is equivalent to combining $p + 1$ kernels where the last kernel is an identity matrix with the optimal coefficient corresponding to the λ value. This method has been mentioned by Lanckriet *et al.* to tackle the estimation of the regularization parameter in the soft margin SVM [4]. It has also been used by Ye *et al.* to jointly estimate the optimal kernel for discriminant analysis [17]. Saving the effort of validating λ may significantly reduce the model selection cost in complicated learning problems. By this transformation, the objective of LSSVM MKL becomes similar to that of SVM MKL with the main difference that the dual variables are unconstrained. Though (39) and (40) can in principle both be solved as QP problems by a conic solver or a QP solver, the efficiency of a linear solution of the LSSVM is lost. Fortunately, in a SIP formulation, the LSSVM MKL can be decomposed into iterations of the master problem of single kernel LSSVM learning, which is an unconstrained QP problem, and a coefficient optimization problem with very small scale.

SIP formulation for LSSVM SVM MKL on larger scale data

The L_∞-norm approach of multi-class LSSVM MKL is formulated as

$$\begin{aligned} \underset{\vec{\theta},\mathrm{u}}{\text{maximize}} \quad & u \\ \text{subject to} \quad & \theta_j\geq 0,\ j=1,\dots,p+1 \\ & \sum_{j=1}^{p+1}\theta_j=1, \\ & \sum_{j=1}^{p+1}\theta_j f_j\left(\vec{\beta}_q\right)\geq u,\forall\vec{\beta}_q,\quad q=1,\dots,k \\ & f_j\left(\vec{\beta}_q\right)=\sum_{q=1}^{k}\left(\frac{1}{2}\vec{\beta}_q^T K_j\vec{\beta}_q-\vec{\beta}_q^T Y_q{}^{-1}\vec{1}\right), \\ & j=1,\dots,p+1,\quad q=1,\dots,k. \end{aligned} \tag{41}$$

In the formulation above, K_j represents the j −th kernel matrix in a set of $p + 1$ kernels with the $p + 1$-th kernel being the identity matrix. The L_2-norm LSSVM MKL is formulated as

$$
\begin{aligned}
\underset{\vec{\theta},\mathrm{u}}{\text{maximize}} \quad & u \\
\text{subject to} \quad & \theta_j \geq 0,\ j = 1, \ldots, p+1 \\
& \sum_{j=1}^{p+1} \theta_j^2 \leq 1, \\
& \sum_{j=1}^{p+1} \theta_j f_j\left(\vec{\beta}_q\right) - \sum_{q=1}^{k} \vec{\beta}_q^T Y_q^{-1} \vec{1} \geq u, \\
& \forall \vec{\beta}_q,\ q = 1, \ldots, k \\
& f_j\left(\vec{\beta}_q\right) = \sum_{q=1}^{k} \left(\frac{1}{2} \vec{\beta}_q^T K_j \vec{\beta}_q \right) \\
& j = 1, \ldots, p+1,\ q = 1, \ldots, k.
\end{aligned}
\tag{42}
$$

The pseudocode of L_∞ -norm and L_2-norm LSSVM MKL is presented in Algorithm 2 in the Appendix. In L_∞ approach, Step 1 optimizes $\vec{\theta}$ as a linear programming. In L_2 approach, Step 1 optimizes $\vec{\theta}$ as a QCLP problem. Since the regularization coefficient is automatically estimated as θ_{p+1}, Step 3 simplifies to a linear problem as

$$
\left[\begin{array}{c|c} 0 & \vec{1}^T \\ \hline \vec{1} & \Omega^{(\tau)} \end{array}\right]
\left[\begin{array}{c} b^{(\tau)} \\ \hline \vec{\beta}^{(\tau)} \end{array}\right]
= \left[\begin{array}{c} 0 \\ \hline Y^{-1}\vec{1} \end{array}\right],
\tag{43}
$$

where $\Omega^{(\tau)} = \sum_{j=1}^{p+1} \theta_j^{(\tau)} K_j$.

Summary of algorithms

As discussed, the dual L_2 MKL solution can be extended to many machine learning problems. In principle, all MKL algorithms can be formulated in L_∞, L_1, and L_2 forms and lead to different solutions. To validate the proposed approach, we implemented and compared 20 algorithms on various data sets. The summary of all implemented algorithms is presented in Table 3. These algorithms combine L_∞, L_1, and L_2 MKL with 1-SVM, SVM, and LSSVM. Moreover, to cope with imbalanced data in classification, we also extended Weighted SVM [23,24] and Weighted LSSVM [25,26] to their MKL formulations (presented in Additional file 1). Though we mainly focus on L_∞, L_1, and L_2 MKL methods, we also implement the L_n-norm MKL for 1-SVM, SVM, LS-SVM and Weighted SVM. These algorithms are applied on the four biomedical experimental data sets and the performance is reported in section 8 of Additional file 1. Moreover, the L_n-norm algorithms are also available on the website of this paper.

Experimental setup and data sets

The performance of the proposed L_2 MKL method was systematically evaluated and compared on six real benchmark data sets. The computational efficiency was compared on two UCI data sets. On each data set, we compared the L_2 method with the L_∞, L_1 and regularized L_∞ MKL method. In the regularized L_∞, we set the minimal boundary of kernel coefficients θ_{min} to 0.5, denoted as L_∞ (0.5). We also compared the three different optimization formulations SOCP, QCQP and SIP on the UCI data sets. The experiments were categorized in five groups as summarized in Table 4.

Experiment 1

In the first experiment, we demonstrated a disease gene prioritization application to compare the performance of optimizing different norms in MKL. The computational definition of gene prioritization is mentioned in our earlier work [7,27,28]. In this paper, we applied four 1-SVM MKL algorithms to combine kernels derived from 9 heterogeneous genomic sources (shown in section 1 of Additional file 1) to prioritize 620 genes that are annotated to be relevant for 29 diseases in OMIM. The performance was evaluated by leave-one-out (LOO) validation: for each disease which contains K relevant genes, one gene, termed the "defector" gene, was removed from the set of training genes and added to 99 randomly selected test genes (test set). We used the remaining K - 1 genes (training set) to build our prioritization model. Then, we prioritized the test set of 100 genes with the trained model and determined the rank of that defector gene in test data. The prioritization function in (22) scored the relevant genes higher and others lower, thus, by labeling the "defector" gene as class "+1" and the random candidate genes as class "-1", we plotted the Receiver Operating Characteristic (ROC) curves to compare different models using the error of AUC (one minus the area under the ROC curve).

The kernels of data sources were all constructed using linear functions except the sequence data that was transformed into a kernel using a 2-mer string kernel function [29] (details in section 1 of Additional file 1). In total 9 kernels were combined in this experiment. The regularization parameter ν in 1-SVM was set to 0.5 for all comparing algorithms. Since there was no hyperparameter needed to be tuned in LOO validation, we reported the LOO results as the performance of generalization. For each disease relevant gene, the 99 test genes were randomly selected in each LOO validation run from the whole human protein-coding genome. We repeated the experiment 20 times and the mean value and standard deviation were used for comparison.

Experiment 2

In the second experiment we used the same data sources and kernel matrices as in the previous experiment to prioritize 9 prostate cancer genes recently discovered by Eeles *et al.* [30], Thomas *et al.* [31] and Gudmundsson *et al.* [32]. A training set of 14 known prostate cancer genes

Table 3 Summary of algorithms implemented in the paper

Algorithm Nr.	Formulation Nr.	Name	References	Formulation	Equations
1	1-A	1-SVM L_∞ MKL	[7]	SOCP	(20)
1	1-B	1-SVM L_∞ MKL	[7]	QCQP	(20)
2	2-A	1-SVM L_∞ (0.5) MKL	[7]	SOCP	(20)
2	2-B	1-SVM L_∞ (0.5) MKL	[7]	QCQP	(20)
3	3-A	1-SVM L_1 MKL	[12,13]	SOCP	(19)
3	3-B	1-SVM L_1 MKL	[12,13]	QCQP	(19)
4	4-A	1-SVM L_2 MKL	novel	SOCP	(23)
5	5-B	SVM L_∞ MKL	[4,6,5]	QCQP	(26)
5	5-C	SVM L_∞ MKL	[18]	SIP	(33)
6	6-B	SVM L_∞ (0.5) MKL	novel	QCQP	(26)
7	7-A	SVM L_1 MKL	[2]	SOCP	(25)
7	7-B	SVM L_1 MKL	[4]	QCQP	(25)
8	8-A	SVM L_2 MKL	novel	SOCP	(27)
8	8-C	SVM L_2 MKL	[40]	SIP	(34)
9	9-B	Weighted SVM L_∞ MKL	novel	QCQP	Suppl. (3)
10	10-B	Weighted SVM L_∞ (0.5) MKL	novel	QCQP	Suppl. (3)
11	11-B	Weighted SVM L_1 MKL	[25]	QCQP	Suppl. (2)
12	12-A	Weighted SVM L_2 MKL	novel	SOCP	Suppl. (4)
13	13-B	LSSVM L_∞ MKL	[17]	QCQP	(39)
13	13-C	LSSVM L_∞ MKL	[17]	SIP	(41)
14	14-B	LSSVM L_∞ (0.5) MKL	novel	QCQP	(39)
15	15-D	LSSVM L_1 MKL	[22]	linear	(38)
16	16-B	LSSVM L_2 MKL	novel	SOCP	(40)
16	16-C	LSSVM L_2 MKL	novel	SIP	(42)
17	17-B	Weighted LSSVM L_∞ MKL	novel	QCQP	Suppl. (8)
18	18-B	Weighted LSSVM L_∞ (0.5) MKL	novel	QCQP	Suppl. (8)
19	19-D	Weighted LSSVM L_1 MKL	[25]	linear	Suppl. (6)
20	20-A	Weighted LSSVM L_2 MKL	novel	SOCP	Suppl. (9)

Summary of algorithms implemented in the paper. Because a same algorithm can be solved via different formulations. The different formulation numbers correspond to a same algorithm number and represent multiple formulations. In total 20 different algorithms were implemented, which were solved through 28 different formulations. For an algorithm with different formulations, the solutions are identical and only differ by computational efficiency. Some algorithms have already been proposed in the literature as shown in the reference column. The novel algorithms and formulations proposed in this paper are labeled as "novel".

was compiled from the reference database OMIM including only the discoveries prior to January 2008. This training set was then used to train the prioritization model. For each novel prostate cancer gene, the test set contained the newly discovered gene plus its 99 closest neighbors on the chromosome. Besides the error of AUC, we also compared the ranking position of the novel prostate cancer gene among its 99 closet neighboring genes. Moreover, we compared the MKL results with the ones obtained via the Endeavour application.

Experiment 3

The third experiment is taken from the work of Daemen *et al.* about the kernel-based integration of genome-wide data for clinical decision support in cancer diagnosis [33]. Thirty-six patients with rectal cancer were treated by combination of cetuximab, capecitabine and external

Table 4 Summary of data sets and algorithms used in five experiments

Nr.	Data Set	Problem	Samples	Classes	Algorihtms	Evaluation
1	disease relevant genes	ranking	620	1	1-4	LOO AUC
2	prostate cancer genes	ranking	9	1	1-4	AUC
3	rectal cancer patients	classification	36	2	5-8,13-16	LOO AUC
4	endometrial disease	classification	339	2	5-8,13-16	3-fold AUC
	miscarriage	classification	2356	2	5-8,13-16	3-fold AUC
	pregnancy	classification	856	2	9-12,17-20	3-fold AUC
5	UCI pen digit and optical digit	classification	1000-3000	10	1A,1B,5B,5C,13B,13C	CPU time

beam radiotherapy and their tissue and plasma samples were gathered at three time points: before treatment (T_0); at the early therapy treatment (T_1) and at the moment of surgery (T_2). The tissue samples were hybridized to gene chip arrays and after processing, the expression was reduced to 6,913 genes. Ninety-six proteins known to be involved in cancer were measured in the plasma samples, and the ones that had absolute values above the detection limit in less than 20% of the samples were excluded for each time point separately. This resulted in the exclusion of six proteins at T_0 and four at T_1. “Responders” were distinguished from “non-responders” according to the pathologic lymph node stage at surgery (pN-STAGE). The “responder” class contains 22 patients with no lymph node found at surgery whereas the “non-responder” class contains 14 patients with at least 1 regional lymph node. Only the two array-expression data sets (MA) measured at T_0 and T_1 and the two proteomics data sets (PT) measured at T_0 and T_1 were used to predict the outcome of cancer at surgery.

Similar to the original method applied on the data [33], we used R BioConductor DEDS as feature selection techniques for microarray data and the Wilcoxon rank sum test for proteomics data. The statistical feature selection procedure was independent to the classification procedure, however, the performance varied widely with the number of selected genes and proteins. We considered the relevance of features (genes and proteins) as prior knowledge and systematically evaluated the performance using multiple numbers of genes and proteins. According to the ranking of statistical feature selection, we gradually increased the number of genes and proteins from 11 to 36, and combined the linear kernels constructed by these features. The performance was evaluated by LOO method, where the reason was two folded: firstly, the number of samples was small (36 patients); secondly, the kernels were all constructed with a linear function. Moreover, in LSSVM classification we proposed the strategy to estimate the regularization parameter λ in kernel fusion. Therefore, no hyperparameter was needed to be tuned so we reported the LOO validation result as the performance of generalization.

Experiment 4

Our fourth experiment considered three clinical data sets. These three data sets were derived from different clinical studies and were used by Daemen and De Moor [34] as validation data for clinical kernel function development. Data set I contains clinical information on 402 patients with an endometrial disease who underwent an echographic examination and color Droppler [35]. The patients are divided into two groups according to their histology: malignant (hyperplasia, polyp, myoma, and carcinoma) versus benign (proliferative endometrium, secretory endometrium, atrophia). After excluding patients with incomplete data, the data contains 339 patients of which 163 malignant and 176 benign. Data set II comes from a prospective observational study of 1828 women undergoing transvaginal sonography before 12 weeks gestation, resulting in data for 2356 pregnancies of which 1458 normal at week 12 and 898 miscarriages during the first trimester [36]. Data set III contains data on 1003 pregnancies of unknown location (PUL) [37]. Within the PUL group, there are four clinical outcomes: a failing PUL, an intrauterine pregnancy (IUP), an ectopic pregnancy (EP) or a persisting PUL. Because persisting PULs are rare (18 cases in the data set), they were excluded, as well as pregnancies with missing data. The final data set consists of 856 PULs among which 460 failing PULs, 330 IUPs, and 66 EPs. As the most important diagnostic problem is the correct classification of the EPs versus non-EPs [38], the data was divided as 790 non-EPs and 66 EPs. To simulate a problem of combining multiple sources, for each data we created eight kernels and combined them using MKL algorithms for classification. The eight kernels included one linear kernel, three RBF kernels, three polynomial kernels and a clinical kernel. The kernel width of the first RBF kernel is selected by empirical rules as four times the average covariance of all the samples, the second and the third kernel widths were respectively six and eight times the average covariance. The degrees of the three polynomial kernels were set to 2, 3, and 4 respectively. The bias term of polynomial kernels was set to 1. The clinical kernels were constructed as proposed by Daemen and De Moor [33]. All the kernel functions are explained in section 3 of Additional file 1. We noticed that the class labels of the pregnancy data were quite imbalanced (790 non-EPs and 66 EPs). In literature, the class imbalanced problem can be tackled by modifying the cost of different classes in the objective function of SVM. Therefore, we applied weighted SVM MKL and weighted LSSVM MKL on the imbalanced pregnancy data. For the other two data sets, we compared the performance of SVM MKL and LSSVM MKL with different norms.

The performance of classification was benchmarked using 3-fold cross validation. Each data set was randomly and equally divided into 3 parts. As introduced in the Methods section, when combining multiple pre-constructed kernels in LSSVM based algorithms, the regularization parameter λ can be jointly estimated as the coefficient of identity matrix. In this case we don’t need to optimize any hyper-parameter in the LSSVM. In the estimation approach of LSSVM and all approaches of SVM, we therefore could use both training and validation data to train the classifier, and test data to evaluate the performance. The evaluation was repeated three times, so each part was used once as test data. The average performance was reported as the evaluation of one repetition. In the standard validation approach of LSSVM, each dataset was partitioned randomly into three parts for training, validation and testing. The classifier was trained on the training data and the hyper-parameter λ was tuned on the validation data. When tuning the λ, its values were sampled uniformly on the log scale from 2^{-10} to 2^{10}. Then, at optimal λ, the classifier was retrained on the combined training and validation set and the resulting model is tested on the testing set. Obviously, the estimation approach is more efficient than the validation approach because the former approach only requires one training process whereas the latter needs to perform 22 times an additional training (21 λ values plus the model retraining). The performance of these two approaches was also investigated in this experiment.

Experiment 5

As introduced in the Methods section, a same MKL problem can be formulated as different optimization problems such as SOCP, QCQP, and SIP. The accuracy of the discretization method for solving SIP is mainly determined by the tolerance value ε predefined in the stopping criterion. In our implementation, ε was set to 5×10^{-4}. These different formulations yield the same result but mainly differ on computational efficiency. In the fifth experiment we compared the efficiency of these optimization techniques on two large scale UCI data sets. The two data sets are digit recognition data for pen based handwriting recognition and optical based digit recognition. Both data sets contain more than 6000 data samples thus they were used as real large scale data sets to evaluate the computational efficiency. In our implementation, the optimization problems were solved by Sedumi [14], MOSEK [15] and the Matlab optimization toolbox. All the numerical experiments were carried on a dual Opteron 250 Unix system with 16 G memory and the computational efficiency was evaluated by the CPU time (in seconds).

Results

Experiment 1: disease relevant gene prioritization by genomic data fusion

In the first experiment, the L_2 1-SVM MKL algorithm performed the best (Error 0.0780). As shown in Table 5, the L_∞ and L_1 approaches all performed significantly worse than the L_2 approach. For example, in the current experiment, when setting the minimal boundary of the kernel coefficients to 0.5, each data source was ensured to have a minimal contribution in integration, thereby improving the L_∞ performance from 0.0923 to 0.0806, although still lower than L_2. In Figure 1 we illustrate the optimal kernel coefficients of different approaches. As shown, the L_∞ method assigned dominant coefficients to Text mining and Gene Ontology data, whereas other data sources were almost discarded from integration. In contrast, the L_2 approach evenly distributed the coefficients over all data sources and thoroughly combined them in integration. When combining multiple kernels, sparse coefficients combine the model only with one or two kernels, making the combined model fragile with respect to the uncertainty and novelty. In real problems, the relevance of a new gene to a certain disease may not have been investigated thus a model solely based on Text and GO annotation is less reliable. L_2 based integration evenly combines multiple genomic data sources. In this experiment, the L_2 approach showed the same effect as the regularized L_∞ by setting some minimal boundaries on kernel coefficients. However, in the regularized L_∞, the minimal boundary θ_{min} usually is predefined according to the "rule of thumb". The main advantage of the L_2 approach is that the θ_{min} values are determined automatically for different kernels and the performance is shown to be better with the manually selected values.

Experiment 2: Prioritization of recently discovered prostate cancer genes by genomic data fusion

In the second experiment, recently discovered prostate cancer genes were prioritized using the same data sources and algorithms as in the first experiment. As shown in Table 6, the L_2 method significantly outperformed other methods on prioritization of gene CDH23, and JAZF1. For 5 other genes (CPNE, EHBP1, MSMB, KLK3, IL16), the performance of the L_2 method was comparable to the best result. In section 4 of Additional file 1, we also presented the optimal kernel coefficients and the prioritization results for individual sources. As shown in Additional file 1, the L_∞ algorithm assigned

Table 5 Results of experiment 1: prioritization of 620 disease relevant genes by genomic data fusion

	Error of AUC (mean)	Error of AUC (std.)	p-value	corr	corr	corr	corr
L_∞	0.0923	0.0035	$2.98 \cdot 10^{-17}$	-	0.94	0.66	0.82
$L_\infty(0.5)$	0.0806	0.0033	$2.66 \cdot 10^{-06}$	0.94	-	0.82	0.92
L_1	0.0908	0.0042	$1.92 \cdot 10^{-16}$	0.66	0.82	-	0.90
L_2	**0.0780**	0.0034	-	0.82	0.92	0.90	-

Results of experiment 1: disease relevant gene prioritization by genomic data fusion. The error of AUC values is evaluated by LOO validation in 20 random repetitions. The best performance (L_2) is shown in bold. The p-values are compared with the best performance using a paired t-test. As shown, the L_2 method is significantly better than other methods. The paired Spearman correlation scores compare similarities of rankings obtained by different approaches when compared with the target rankings (denoted as -). Higher Spearman correlation values mean that the two ranking results are much similar.

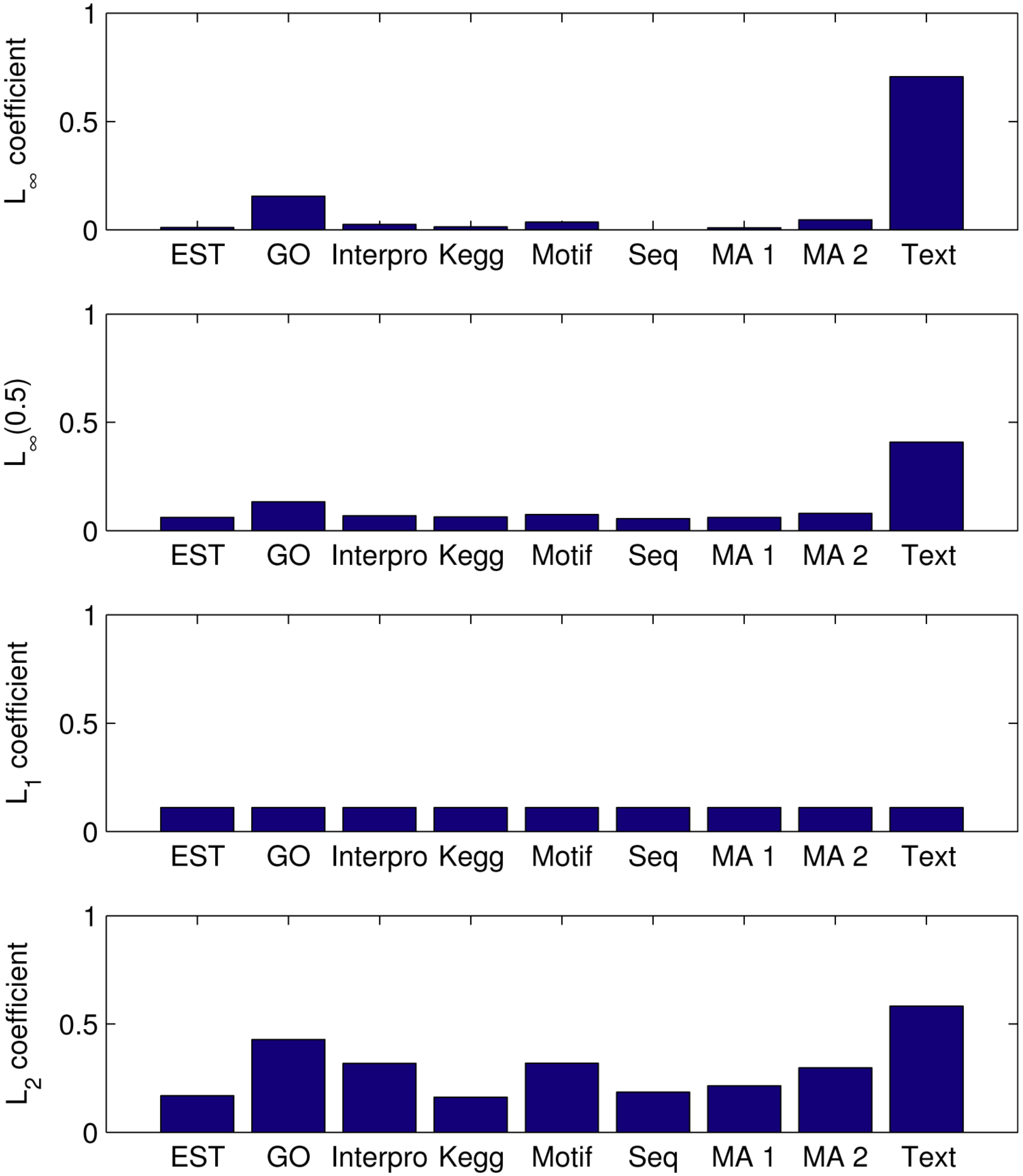

Figure 1 Optimal kernel coefficients for disease gene prioritization. Optimal kernel coefficients assigned on genomic data sources in disease gene prioritization. For each method, the average coefficients of 20 repetitions are shown. The three most important data sources ranked by L_∞ are Text, GO, and Motif. The coefficients on other six sources are almost zero. The L_2 method shows the same ranking on these three best data sources as L_∞, moreover, it also shows ranking for other six sources. Thus, as another advantage of L_2 method, it provides more refined ranking of data sources than L_∞ method in data integration.

Table 6 Results of experiment 2: prioritization of prostate cancer genes by genomic data fusion

Name	Ensemble id	References	L_∞	$L_\infty(0.5)$	L_1	L_2	Endeavour
CPNE	ENSG00000085719	Thomas *et al.*	0.3030	0.2323	**0.1010**	*0.1212*	-
			31/100	24/100	**11/100**	*13/100*	70/100
CDH23	ENSG00000107736	Thomas *et al.*	0.0606	0.0303	*0.0202*	**0.0101**	-
			7/100	4/100	*3/100*	**2/100**	78/100
EHBP1	ENSG00000115504	Gudmundsson *et al.*	0.5354	0.5152	**0.3434**	*0.3939*	-
			54/100	52/100	**35/100**	*40/100*	57/100
MSMB	ENSG00000138294	Eeles *et al.*	**0.0202**	**0.0202**	0.0505	*0.0303*	-
		Thomas *et al.*	**3/100**	**3/100**	6/100	*4/100*	69/100
KLK3	ENSG00000142515	Eeles *et al.*	0.3434	0.3535	*0.2929*	*0.2929*	-
			35/100	36/100	*30/100*	*30/100*	**28/100**
JAZF1	ENSG00000153814	Thomas *et al.*	*0.0505*	**0.0202**	**0.0202**	**0.0202**	-
			6/100	**3/100**	**3/100**	**3/100**	7/100
LMTK2	ENSG00000164715	Eeles *et al.*	*0.3131*	0.4646	0.8081	0.7677	-
			32/100	47/100	81/100	77/100	**31/100**
IL16	ENSG00000172349	Thomas *et al.*	**0**	*0.0101*	0.0303	*0.0101*	-
			1/100	*2/100*	4/100	*2/100*	72/100
CTBP2	ENSG00000175029	Thomas *et al.*	*0.8283*	0.5758	*0.6364*	0.6869	-
			83/100	58/100	*64/100*	69/100	**38/100**

Results of experiment 2: prioritization of prostate cancer genes by genomic data fusion. For each novel prostate cancer gene, the first row shows the error of AUC values and the second row lists the ranking position of the prostate cancer gene among its 99 closet neighboring genes.

most of the coefficients to Text and Microarray data. Text data performs well in the prioritization of known disease genes, however, does not always work the best for newly discovered genes. This experiment demonstrates that when prioritizing novel prostate cancer relevant genes, the L_2 MKL approach evenly optimized the kernel coefficients to combine heterogeneous genomic sources and its performance was significantly better than the L_∞ method. Moreover, we also compared the kernel based data fusion approach with the Endeavour gene prioritization software: for 6 genes the MKL approach performed significantly better than Endeavour.

Experiment 3: Clinical decision support by integrating microarray and proteomics data

One of the main contributions of this paper is that the L_2 MKL notion can be applied on various machine learning problems. The first two experiments demonstrated a ranking problem using 1-SVM MKL to prioritize disease relevant genes. In the third experiment we optimized the L_∞, L_1, and L_2 -norm in SVM MKL and LSSVM MKL classifiers to support the diagnosis of patients according to their lymph node stage in rectal cancer development. The performance of the classifiers greatly depended on the selected features, therefore, for each classifier we compared 25 feature selection results (as a grid of 5 numbers of genes multiplied by 5 numbers of proteins). As shown in Table 7, the best performance was obtained with LSSVM L_1 (error of AUC = 0.0325) using 25 genes and 15 proteins. The L_2 LSSVM MKL classifier was also promising because its performance was comparable to the best result. In particular, for the two compared classifiers (LSSVM and SVM), the L_1 and L_2 approaches significantly outperformed the L_∞ approach. We also tried to regularize the kernel coefficients in L_∞ MKL using different θ_{min} values. Nine different θ_{min} were tried uniformly from 0.1 to 0.9 and the changes in performance is shown in Figure 2. As shown, increasing the θ_{min} value steadily improves the performance of LSSVM MKL and SVM MKL on the rectal cancer data sets. However, determining the optimal θ_{min} was a non-trivial issue. When θ_{min} was smaller than 0.6, the performance of LSSVM MKL L_∞ remained unchanged, meaning that the "rule of thumb" value 0.5 used in experiment 1 is not valid here. In comparison, when using the L_2 based MKL classifiers, there is no need to specify θ_{min} and the performance is still comparable to the best performance obtained with regularized L_∞ MKL.

In LSSVM kernel fusion, we estimated the λ jointly as a coefficient assigned to an identity matrix. Since the number of samples is small in this experiment, the standard cross-validation approach to select the optimal λ on validation data was not tried. To investigate whether the estimated λ value is optimal, we set λ to 51 different values uniformly sampled on the log_2 scale from -10 to 40. We compared the joint estimation result with the optimal classification performance among the sampled λ values. The joint estimation results were found as optimal for most of the results. An example is illustrated in Figure 3 for the integration of four kernels constructed by 27 gene features and 17 protein features. The coefficients estimated by the L_∞-norm were almost 0 thus the λ values were very big. In contrast, the λ values estimated by the non-sparse L_2 method were at reasonable scales.

Experiment 4: Clinical decision support by integrating multiple kernels

In the fourth experiment we validated the proposed approach on three clinical data sets containing more samples. On the endometrial and miscarriage data sets, we compared eight MKL algorithms with various norms. For the imbalanced pregnancy data set, we applied eight weighted MKL algorithms. The results are shown in Table 8, 9, and 10. On endometrial data, the difference of performance was rather small. Though the two L_2 methods were not optimal, they were comparable to the best result. On miscarriage data, the L_2 methods performed significantly better than comparing algorithms. On pregnancy data, the weighted L_2 LSSVM MKL and weighted L_1 LSSVM MKL performed significantly better than others. We also regularized the kernel coefficients using different θ_{min} values on LSSVM L_∞ and SVM L_∞ MKL classifiers. The results are presented in Figure 4, Figure 5 and Figure 6. As shown, the optimal θ_{min} value differs across data sets thus the "rule of thumb" value of 0.5 may not work for all the problems. For the endometrial and miscarriage data sets, the optimal θ_{min} for both MKL classifiers is 0.2. For pregnancy data set, the optimal θ_{min} value for LSSVM is 1 and for SVM 0.9. In

Table 7 Results of experiment 3: classification of patients in rectal cancer clinical decision using microarray and proteomics data sets

	LSSVM L_∞					**SVM L_∞**				
	14 p	**15 p**	**16 p**	**17 p**	**18 p**	**14 p**	**15 p**	**16 p**	**17 p**	**18 p**
24 g	0.0584	0.0519	*0.0747*	0.0812	0.0812	0.1331	0.1331	0.1331	0.1331	0.1364
25 g	*0.0390*	*0.0390*	0.0519	0.0617	0.0649	0.1136	0.1104	0.1234	0.1201	0.1234
26 g	0.0487	0.0487	0.0812	0.0844	0.0877	0.1266	0.1136	0.1234	0.1299	0.1364
27 g	0.0617	0.0649	0.0812	0.0877	0.0942	0.1429	0.1364	0.1364	0.1331	0.1461
28 g	0.0552	0.0487	0.0617	0.0747	0.0714	0.1429	0.1331	0.1331	0.1364	0.1396
	LSSVM L_∞ (0.5)					SVM L_∞ (0.5)				
	14 p	15 p	16 p	17 p	18 p	14 p	15 p	16 p	17 p	18 p
24 g	0.0584	0.0519	*0.0747*	0.0812	0.0812	0.1266	0.1006	0.1266	0.1299	0.1331
25 g	*0.0390*	*0.0390*	0.0519	0.0617	0.0649	0.1136	0.1071	0.1234	0.1201	0.1234
26 g	0.0487	0.0487	0.0812	0.0844	0.0877	0.1136	0.1136	0.1201	0.1266	0.1331
27 g	0.0617	0.0649	0.0812	0.0877	0.0942	0.1364	0.1364	0.1364	0.1331	0.1461
28 g	0.0552	0.0487	0.0617	0.0747	0.0714	0.1299	0.1299	0.1299	0.1331	0.1364
	LSSVM L_1					SVM L_1				
	14 p	15 p	16 p	17 p	18 p	14 p	15 p	16 p	17 p	18 p
24 g	**0.0487**	**0.0487**	**0.0682**	**0.0682**	0.0747	0.0747	0.0584	0.0714	**0.0682**	0.0747
25 g	**0.0357**	**0.0325**	**0.0422**	**0.0455**	**0.0455**	0.0584	0.0519	0.0649	0.0714	0.0714
26 g	**0.0357**	**0.0357**	**0.0455**	**0.0455**	**0.0455**	0.0584	0.0519	0.0682	0.0682	0.0682
27 g	**0.0357**	**0.0357**	**0.0455**	**0.0487**	**0.0519**	0.0617	0.0584	0.0714	0.0682	0.0682
28 g	**0.0422**	**0.0325**	**0.0487**	**0.0487**	**0.0519**	0.0584	0.0584	0.0649	0.0649	0.0682
	LSSVM L_2					SVM L_2				
	14 p	15 p	16 p	17 p	18 p	14 p	15 p	16 p	17 p	18 p
24 g	*0.0552*	**0.0487**	*0.0747*	*0.0779*	**0.0714**	0.0909	0.0877	0.0974	0.0942	0.1006
25 g	*0.0390*	*0.0390*	*0.0487*	*0.0552*	*0.0552*	0.0747	0.0649	0.0812	0.0844	0.0844
26 g	*0.0390*	*0.0455*	*0.0552*	*0.0649*	*0.0649*	0.0747	0.0584	0.0812	0.0779	0.0779
27g	*0.0422*	*0.0487*	*0.0552*	*0.0584*	*0.0649*	0.0779	0.0812	0.0844	0.0812	0.0812
28 g	*0.0455*	**0.0325**	**0.0487**	*0.0584*	*0.0552*	0.0812	0.0714	0.0812	0.0779	0.0812

The table shows the error of AUC in patient classification using microarray and proteomics data. In LSSVM L_∞, L_∞ (0.5), and L_2, the regularization parameter λ was estimated jointly as the kernel coefficient of an identity matrix. In LSSVM L_1, λ was set to 1. In all SVM approaches, the *C* parameter of the box constraint was set to 1. In the table, the row and column labels represent the numbers of genes (g) and proteins (p) used to construct the kernels. The genes and proteins were ranked by feature selection techniques (see text). The AUC of LOO validation was evaluated without the bias term *b* (as the implicit bias approach) because its value varied by each left out sample. In this problem, considering the bias term decreased the AUC performance. The performance was compared among eight algorithms for the same number of genes and proteins, where the best values (the smallest Error of AUC) are represented in bold, the second best ones in italic. The best performance of all the feature selection results is underlined. The table presents the 25 best feature selection results of each method. The complete experimental results containing 26 different numbers of genes and 26 numbers of proteins is available at http://homes.esat.kuleuven.be/~sistawww/bioi/syu/l2lssvm.html.

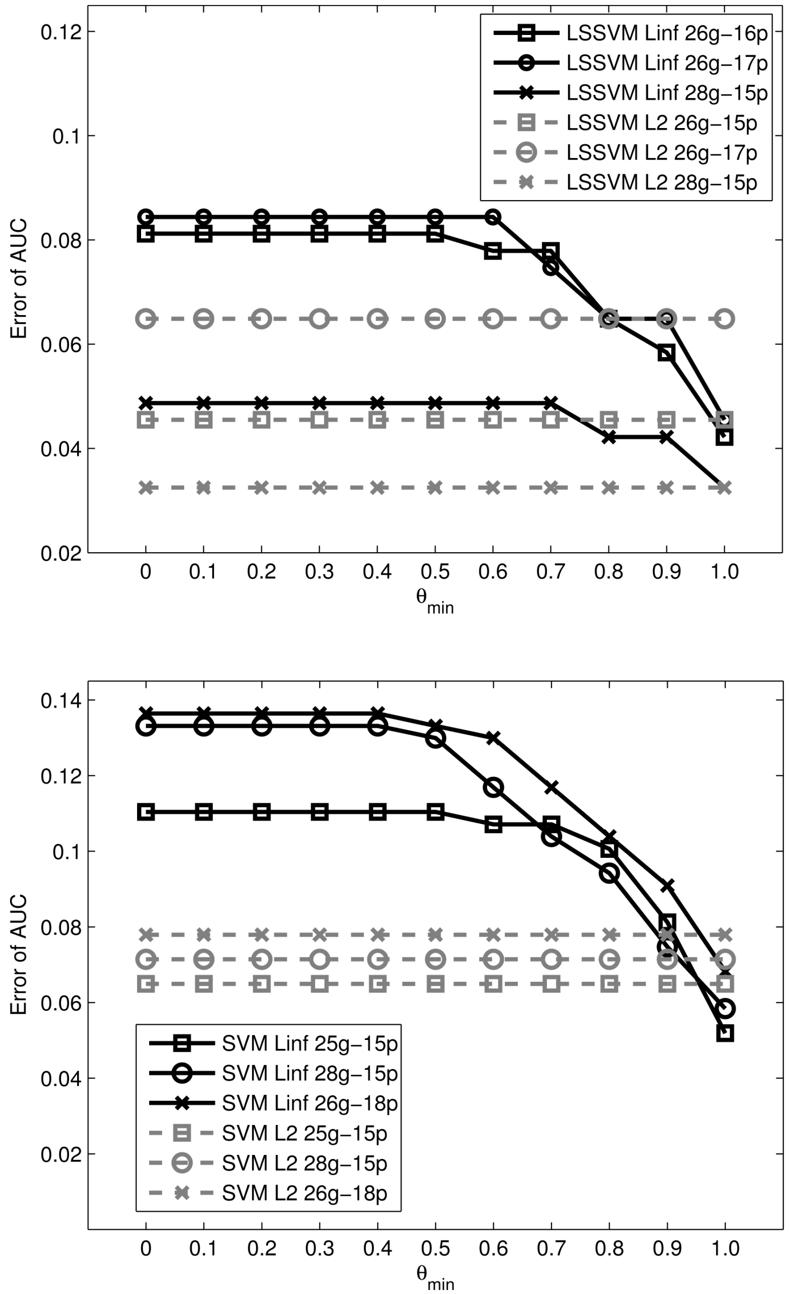

Figure 2 The effect of θ_{min} on LSSVM MKL and SVM MKL classifier in rectal cancer diagnosis. The effect of θ_{min} in LSSVM MKL and SVM MKL classifiers for rectal cancer diagnosis. Figure on the top: the performance of LSSVM MKL. Figure on the bottom: the performance of SVM MKL. In each figure we compare three feature selection results. The performance of L_2 MKL is shown as dashed lines.

comparison, on the miscarriage and pregnancy data set, the performance of the L_2 algorithm is comparable or even much better than the best regularized L_∞ algorithm. For the endometrial data set, though the optimal regularized L_∞ LSSVM and SVM MKL classifiers outperform L_2 classifiers, L_2 methods still perform better than or as equal as the unregularized L_∞ method.

To investigate whether the combination of multiple kernels performs as well as the best individual kernel, we evaluated the performance of all the individual kernels in section 5 of Additional file 1. As shown, the clinical kernel proposed by Daemen and De Moor [33] has better quality than linear, RBF and polynomial kernels on endometrial and pregnancy data sets. For the

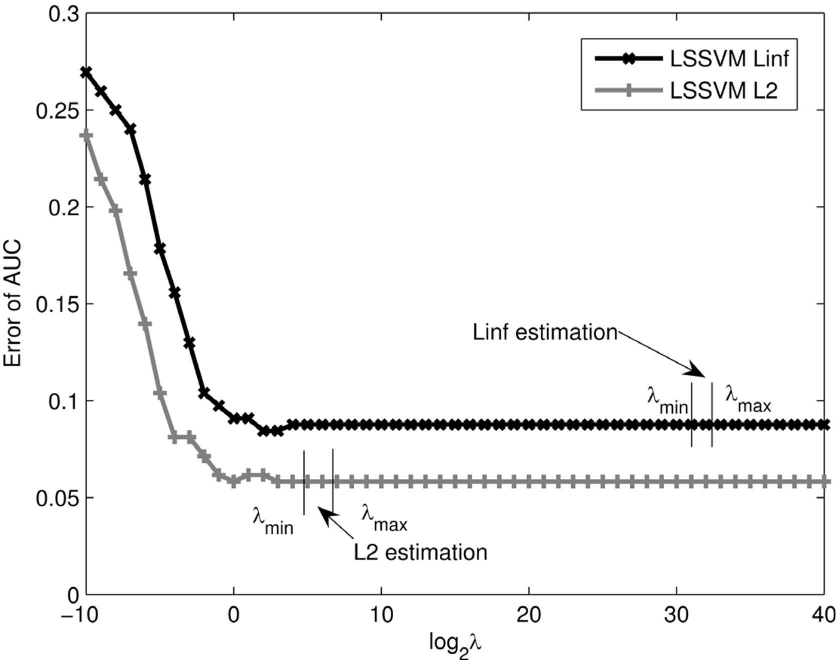

Figure 3 Benchmark of various λ values in LSSVM MKL classifiers in rectal cancer diagnosis. Benchmark of various λ values in LSSVM MKL classifiers for rectal cancer diagnosis. The four kernels were constructed using 27 gene features and 17 protein features (see text). For each fixed λ value, the error of AUC was evaluated by LOO validation. The maximal and minimal estimated λ in L_∞ and L_2 MKL are shown.

miscarriage data set, the first RBF kernel has better quality than the other seven kernels. Despite the difference in individual kernels, the performance of MKL is comparable to the best individual kernel, demonstrating that MKL is also useful to combine candidate kernels derived from a single data set.

The effectiveness of MKL can also be justified by investigating the kernel coefficients optimized on all the data sets and classifiers. As shown in section 6 of Additional file 1, the kernel coefficients optimized by L_∞ MKL algorithms were sparse whereas the L_2 ones were more evenly assigned to different kernels. The best individual kernels of all data sets usually get dominant coefficient, explaining why the performance of MKL algorithms is comparable to the best individual kernels.

Table 8 Results of experiment 4 data set I: classification of endometrial disease patients using multiple kernels derived from clinical data

Classifier	Mean - error of AUC	Std. - error of AUC	pvalue
LSSVM L_∞ (0.5) MKL	**0.2353**	**0.0133**	-
SVM L_∞ (0.5) MKL	**0.2388**	**0.0178**	0.4369
SVM L_∞ MKL	**0.2417**	**0.0165**	0.2483
LSSVM L_2 MKL	0.2456	0.0124	0.0363
SVM L_2 MKL	0.2489	0.0178	0.0130
SVM L_1 MKL	0.2513	0.0144	0.0057
LSSVM L_1 MKL	0.2574	0.0189	$9.98 \cdot 10^{-5}$
LSSVM L_∞ MKL	0.2678	0.0130	$1.53 \cdot 10^{-6}$

Results of experiment 4 data set I: classification of endometrial disease patients using multiple kernels derived from clinical data. The classifier with the best performance is shown in bold. The p-values are compared with the best performance using a paired t-test. The performance of classifiers is sorted from high to low according to the p-values.

Table 9 Results of experiment 4 data set II: classification of miscarriage patients using multiple kernels derived from clinical data

Classifier	Mean - error of AUC	Std. - error of AUC	pvalue
SVM L_2 MKL	**0.1975**	**0.0037**	-
LSSVM L_2 MKL	**0.2002**	**0.0049**	0.0712
LSSVM L_∞ (0.5) MKL	0.2027	0.0045	$9.77 \cdot 10^{-4}$
SVM L_∞ MKL	0.2109	0.0040	$9.55 \cdot 10^{-12}$
SVM L_∞ (0.5) MKL	0.2168	0.0040	$1.79 \cdot 10^{-12}$
LSSVM L_1 MKL	0.2132	0.0029	$1.11 \cdot 10^{-13}$
SVM L_1 MKL	0.2297	0.0038	$1.10 \cdot 10^{-15}$
LSSVM L_∞ MKL	0.2319	0.0015	$3.42 \cdot 10^{-21}$

Results of experiment 4 data set II: classification of miscarriage patients using multiple kernels derived from clinical data. The classifier with the best performance is shown in bold. The p-values are compared with the best performance using a paired t-test. The performance of classifiers is sorted from high to low according to the p-values.

Table 10 Results of experiment 4 data set III: classification of PUL patients using multiple kernels derived from clinical data

Classifier	Mean - error of AUC	Std. - error of AUC	pvalue
Weighted LSSVM L_2 MKL	**0.1165**	**0.0100**	-
Weighted LSSVM L_1 MKL	**0.1243**	**0.0171**	0.0519
Weighted LSSVM L_∞ (0.5) MKL	0.1290	0.0206	0.0169
Weighted SVM L_2 MKL	0.1499	0.0248	$4.79 \cdot 10^{-5}$
Weighted SVM L_∞ MKL	0.1552	0.0210	1.02 · 10-6
Weighted SVM L_∞ (0.5)	0.1551	0.0153	$3.87 \cdot 10^{-6}$
Weighted SVM L_1 MKL	0.1594	0.0162	$2.29 \cdot 10^{-9}$
Weighted LSSVM L_∞ MKL	0.1651	0.0174	$4.41 \cdot 10^{-10}$

Results of experiment 4 data set II: classification of PUL patients using multiple kernels derived from clinical data. The classifier with the best performance is shown in bold. The p-values are compared with the best performance using a paired t-test. The performance of classifiers is sorted from high to low according to the p-values.

In this paper, the regularization parameter λ in LSSVM classifiers was jointly estimated in MKL. Since the clinical data sets contain a sufficient number of samples to select the λ by cross validation, we systematically compared the estimation approach with the standard validation approach to determine the λ values. As shown in Table 11, the estimation approach based on L_∞ performed worse than the validation approach. This is probably because the estimated λ values are either very big or very small when the kernel coefficients were sparse. In contrast, the L_2 based estimation approach yielded comparable performance as the validation approach. We also benchmarked the performance of LSSVM MKL classifiers using 21 different static λ values on the data sets and the results are shown in section 7 of Additional file 1. In real problems, to select the optimal λ value in LSSVM is a non-trivial issue and it is often optimized as a hyper-parameter on validation data. The main advantage of L_2 MKL is that the estimation approach is more computational efficient than cross validation and yields a comparable performance.

Experiment 5: Computational complexity and numerical experiments on large scale problems

Overview of the convexity and complexity

We concluded the convexity and the time complexity of all proposed methods in Table 12. All problems proposed in this paper are convex or can be transformed to a convex formulation by relaxation. The LSSVM SIP formulation has the lowest time complexity thus it is more preferable for large scale problems.

We verified the efficiency in numerical experiments, which adopts two UCI digit recognition data sets (pendigit and optical digit) to compare the computational time of the proposed algorithms.

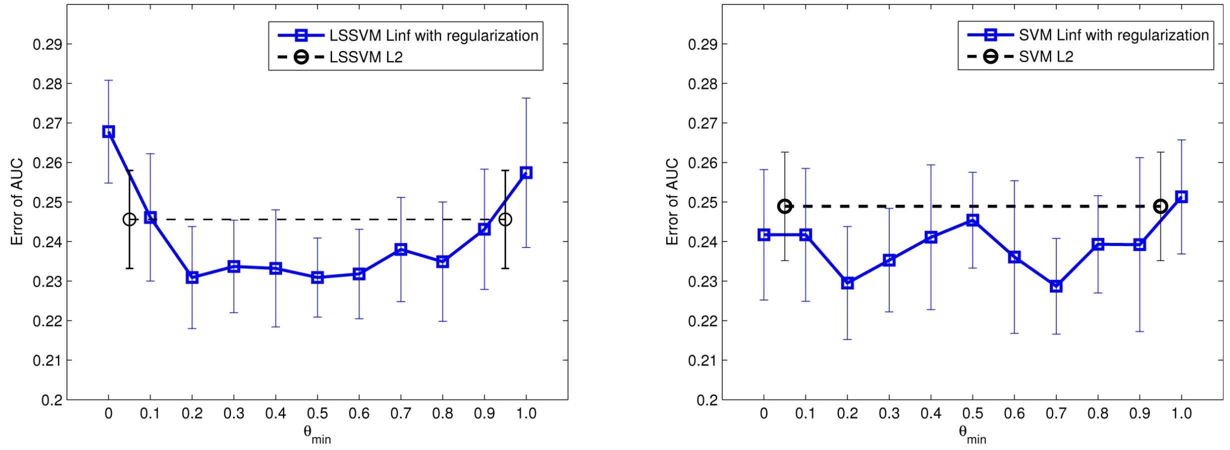

Figure 4 The effect of θ_{min} in LSSVM MKL and SVM MKL classifier on endometrial disease data set. The effect of θ_{min} in LSSVM MKL and SVM MKL classifiers on endometrial disease data set. Figure on the left: performance of the regularized LSSVM L_∞ MKL with various θ_{min} values. Figure on the right: performance of the regularized SVM L_∞ MKL. The black dashed lines represent the performance of the L_2 MKL classifiers. The error bars are standard deviations of 20 repetitions.

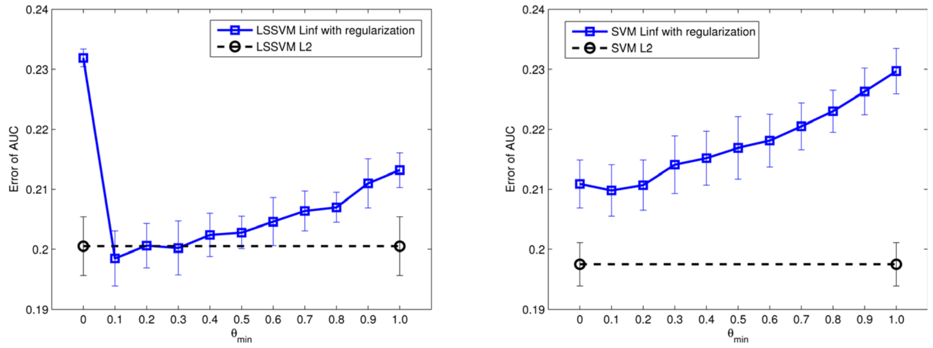

Figure 5 The effect of θ_{min} in LSSVM MKL and SVM MKL classifier on miscarriage data set. The effect of θ_{min} in LSSVM MKL and SVM MKL classifiers on miscarriage data set. Figure on the left: performance of the regularized LSSVM L_∞ MKL with various θ_{min} values. Figure on the right: performance of the regularized SVM L_∞ MKL. The black dashed lines represent the performance of the L_2 MKL classifiers. The error bars are standard deviations of 20 repetitions.

QP formulation is more efficient than SOCP

We investigated the efficiency of various formulations to solve the 1-SVM MKL. As mentioned, the problems presented in (15) can be solved either as QCLP or as SOCP. We applied Sedumi [14] to solve it as SOCP and MOSEK to solve it as QCLP and SOCP. We found that solving the QP by MOSEK was most efficient (142 seconds). In contrast, the MOSEK-SOCP method costed 2608 seconds and the Sedumi-SOCP method took 4500 seconds. This is probably because when transforming a QP to a SOCP, a large number of additional variables and constraints are involved, thus becoming more expensive to solve.

SIP formulation is more efficient than QCQP

To compare the computational time of solving MKL classifiers based on QP and SIP formulations, we scaled up the kernel fusion problem in three dimensions: the number of kernels, the number of classes and the number of samples. As shown in Figure 7, the SIP formulation of LSSVM MKL increases linearly with the number of samples and kernels, and is barely influenced by the number of classes. Solving the SIP based LSSVM MKL is significantly faster than solving SVM MKL because the former optimizes through iterations on a linear systems whereas the latter iterates over quadratic systems. For LSSVM MKL, the SIP formulation is also more preferable than the quadratic formulation. A quadratic system is a memory intensive problem and its complexity increases exponentially with the number of kernels and the number of samples in MKL. In contrast, the SIP formulation separates the problem into a series of linear systems, whose complexity is only determined by the number of samples and less affected by the number of kernels or classes. As shown in step 3 of Algorithm 5.2, the coefficient matrix of the linear system is a combined single kernel matrix and is constant with respect to multiple classes, thus it can be solved very efficiently. We have also compared the CPU time of L_∞

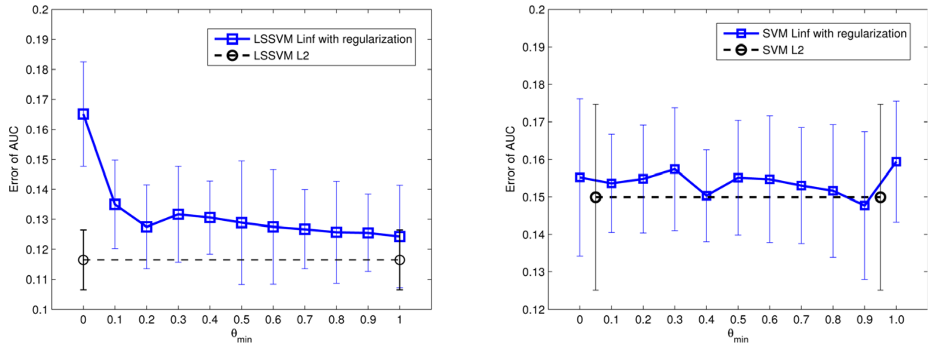

Figure 6 The effect of θ_{min} in weighted LSSVM MKL and weighted SVM MKL classifier on pregnancy data set. The effect of θ_{min} in LSSVM MKL and SVM MKL classifiers on pregnancy data set. Figure on the left: performance of the regularized LSSVM L_∞ MKL with various θ_{min} values. Figure on the right: performance of the regularized SVM L_∞ MKL. The black dashed lines represent the performance of the L_2 MKL classifiers. The error bars are standard deviations of 20 repetitions.

Table 11 Comparison of the performance obtained by joint estimation of λ and standard cross-validation in LSSVM MKL

Data Set	Norm	Validation Approach	Estimation Approach
endometrial disease	L_∞	0.2625 ± 0.0146	0.2678 ± 0.0130
	L_2	0.2584 ± 0.0188	0.2456 ± 0.0124
miscarriage	L_∞	0.1873 ± 0.0100	0.2319 ± 0.0015
	L_2	0.1912 ± 0.0089	0.2002 ± 0.0049
pregnancy	L_∞	0.1321 ± 0.0243	0.1651 ± 0.0173
	L_2	0.1299 ± 0.0172	0.1165 ± 0.0100

Comparison of the performance obtained by joint estimation of λ and standard cross-validation using LSSVM MKL. As shown, the estimation approach based on L_2 MKL is better than L_∞ MKL. This is because when the kernel coefficients are sparse, the estimated regularization parameters λ are either very big or very small, which are usually not optimal values in LSSVM. In contrast, the λ values estimated by L_2 method are at normal scale and often close to the optimal values.

and L_2 LSSVM MKL on large data sets and their efficiency is very similar to each other.

Discussion

In this paper we propose a new L_2 MKL framework as the complement to the existing L_∞ MKL method proposed by Lanckriet *et al.* The L_2 MKL is characterized by the non-sparse integration of multiple kernels to optimize the objective function of machine learning problems. On four real bioinformatics and biomedical applications, we systematically validated the performance through extensive analysis. The motivation for L_2 MKL is as follows. In real biomedical applications, with a small number of sources that are believed to be truly informative, we would usually prefer a nonsparse set of coefficients because we would want to avoid that the dominant source (like text mining or Gene Ontology) gets a coefficient close to 1. The reason to avoid sparse coefficients is that there is a discrepancy between the experimental setup for performance evaluation and "real world" performance. The dominant source will work well on a benchmark because this is a controlled situation with known outcomes. We for example set up a set of already known genes for a given disease and want to demonstrate that our model can capture the available information to discriminate between a gene from this set and randomly selected genes (for example, in a cross-validation setup). Given that these genes are already known to be associated with the disease, this information will be present in sources like text mining or Gene Ontology in the gene prioritization problem. These sources can then identify these known genes with high confidence and should therefore be assigned a high weight. However, when trying to identify truly novel genes for the same disease, the relevance of the information available through such data sources will be much lower and we would like to avoid anyone data source to complete dominate the other. Given that setting up a benchmark requires knowledge of the association between a gene and a disease, this effect is hard to avoid. We can therefore expect that if we have a smoother solution that performs as well as the sparse solution on benchmark data, it is likely to perform better on real discoveries.

Table 12 Convexity and complexity of all methods

Method	convexity	complexity
1-SVM SOCP L_∞, L_2	convex	$O((p+n)^2n^{2.5})$
1-SVM QCQP L_∞	convex	$O(pn^3)$
SVM SOCP L_∞, L_2	convex	$O((p+n)^2(k+n)^{2.5})$
SVM QCQP L_∞	convex	$O(pk^2n^2+k^3n^3)$
SVM SIP L_∞	convex	$O(\tau(kn^3+p^3))$
SVM SIP L_2	relaxation	$O(\tau(kn^3+p^3))$
LSSVM SOCP L_∞, L_2	convex	$O((p+n)^2(k+n)^{2.5})$
LSSVM QCQP L_∞, L_2	convex	$O(pk^2n^2+k^3n^3)$
LSSVM SIP L_∞	convex	$O(\tau(n^2+p^3))$
LSSVM SIP L_2	relaxation	$O(\tau(n^2+p^3))$

Convexity and complexity of all methods. n is the number of samples, p is the number of kernels, k is the number of classes, τ is the number of iterations in SIP. The complexity of LSSVM SIP depends on the algorithms used to solve the linear system. For the conjugate gradient method, the complexity is between $O(n^{1.5})$ and $O(n^2)$ [22].

For the specific problem of gene prioritization, an effective way to address this problem is to setup a benchmark where information is "rolled back" a number of years (e.g., two years) prior to the discovery of the association between a gene and a disease (i.e., older information is used so that the information about the association between the gene and the disease is not yet contained in data sources like text mining or Gene Ontology). Given that the date at which the association was discovered is different for each gene, the setup of such benchmarks is notoriously difficult. In future work, we plan to address this problem by freezing available knowledge at a given data and then collecting novel discoveries and benchmarking against such discoveries in a fashion reminiscent of CASP (Critical Assessment of protein Structure Prediction) [39].

The technical merit of the proposed L_2 MKL lay in the dual form of the learning problems. Though in the literature the issue of using different norms in MKL is recently investigated by Kloft *et al.* [40,9] and Kowalski *et al.* [41], their formulations are based on the primal problems. In our paper, the notion of the proposed L_2 method is discussed in the dual space, which differs from regularizing the norm of coefficients term in the primal space. We have theoretically proven that optimizing the L_2 regularization of kernel coefficients in the primal problem corresponds to solving the L_2-norm of kernel components in the dual problem. Clarifying this

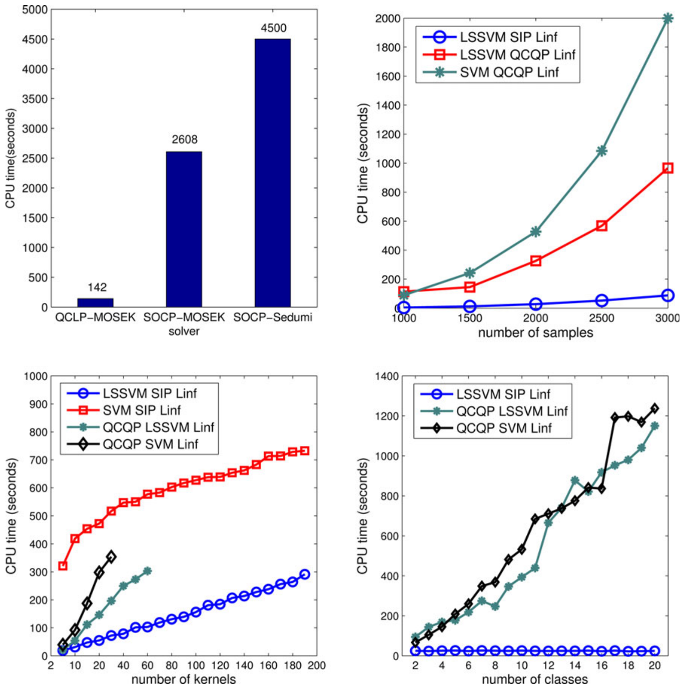

Figure 7 Comparison of QP formulation and SIP formulation on large scale data. Comparison of QP formulation and SIP formulation on large scale data. Figure on the top left: comparison of SOCP and QCQP formulations to solve 1-SVM MKL using two kernels. To simulate the ranking problem in 1-SVM, 3000 digit samples were retrieved as training data. Two kernels were constructed respectively for each data source using RBF kernel functions. The computational time was thus evaluated by combining the two 3000 × 3000 kernel matrices. Figure on the top right: comparison of SVM and LSSVM MKL on problems with increasing number of samples. The benchmark data set was made up of two linear kernels and labels in 10 digit classes. The number of data points was increased from 1000 to 3000. Figure on the bottom left: comparison of SVM and LSSVM MKL on problems with increasing number of kernels. The benchmark data set was constructed by 2000 samples labeled in 2 classes. We used different kernel widths to construct the RBF kernel matrices and increase the number of kernel matrices from 2 to 200. The QCQP formulations had memory issues when the number of kernels was larger than 60. Figure on the bottom right: comparison of SVM and LSSVM on problems with increasing number of classes. The benchmark data was made up of two linear kernel matrices and 2000 samples. The samples were equally and randomly divided into various number of classes. The class number gradually increased from 2 to 20.

dual solution enabled us to directly solve the L_2 problem as a convex SOCP. Moreover, the dual solution can be extended to various other machine learning problems. In this paper we have shown the extensions of 1-SVM, SVM and LSSVM. As a matter of fact, the L_2 dual solution can also be applied in kernel based clustering analysis and regression analysis for a wide range of applications. Another main contribution of our paper is the novel LSSVM L_2 MKL proposed for classification problems. As known, when applying various machine learning techniques to solve real computational biological problems, the performance may depend on the data set and the experimental settings. When the performance evaluations of various methods are comparable, but with one method showing significant computational efficiency over other methods, this would be a “solid” advantage of this method. In this paper, we have shown that the LSSVM MKL classifier based on SIP formulation can be solved more efficiently than SVM MKL. Moreover, the performance of LSSVM L_2 MKL is always comparable to the best performance. The SIP based LSSVM L_2 MKL classifier has two main “solid advantages": the inherent time complexity is small and the regularization parameter λ can be jointly estimated in the experimental setup. Due to these merits, LSSVM L_2 MKL is a very promising technique for problems pertaining to large scale data fusion.

Conclusions

This paper compared the effect of optimizing different norms in multiple kernel learning in a systematic framework. The obtained results extend and enrich the statistical framework of genomic data fusion proposed by Lanckriet *et al.* [4,6] and Bach *et al.* [5]. According to the optimization of different norms in the dual problem of SVM, we proposed L_∞, L_1, and L_2 MKL, which are respectively corresponding to the L_1 regularization, average combination, and L_2 regularization of kernel coefficients addressed in the primal problem.

Six real biomedical data sets were investigated in this paper, where L_2 MKL approach was shown advantageous over the L_∞ method. We also proposed a novel and efficient LSSVM L_2 MKL classifier to learn the optimal combination of multiple large scale data sets. All the algorithms implemented in this paper are freely accessible on http://homes.esat.kuleuven.be/~sistawww/bioi/syu/l2lssvm.html.

Appendix

```
Algorithm 0.1: SIP-SVM-MKL($K_j$, $Y_q$, $C$, $\varepsilon$)
  Obtain the initial guess $\vec{\alpha}^{(0)} = [\vec{\alpha}_1, \ldots, \vec{\alpha}_k]$
  while ($\Delta u > \varepsilon$)
  do {
      step1 : Fix $\vec{\alpha}$, solve $\vec{\theta}^{(\tau)}$ then obtain $u^{(\tau)}$
      step2 : Compute kernel combination $\Omega^{(\tau)}$
      step3 : Solve single SVM by minimizing
          $f_j(\vec{\alpha}_q)$ and obtain the optimal $\vec{\alpha}_q^{(\tau)}$
      step4 : Compute $f_1(\vec{\alpha}^{(\tau)}), \ldots, f_p(\vec{\alpha}^{(\tau)})$
      step5 : $\triangle u = | 1 - \frac{\Sigma_{j=1}^{p} \theta_i^{(\tau-1)} f_j(\vec{\alpha}^{(\tau)})}{u^{(\tau-1)}}$
  }
  comment: $\tau$ is the indicator of the current loop
  return $(\vec{\theta}^*, \vec{\alpha}^*)$
```

```
Algorithm 0.2: SIP-LSSVM-MKL($K_j$, $Y_q$, $\varepsilon$)
  Obtain the initial guess $\vec{\beta}^{(0)} = [\vec{\beta}_1, \ldots, \vec{\beta}_k]$
  while ($\Delta u > \varepsilon$)
  do {
      step1 : Fix $\vec{\beta}$, solve $\vec{\theta}^{(\tau)}$ then obtain $u^{(\tau)}$
      step2 : Compute kernel combination $\Omega^{(\tau)}$
          step3 : Solve single LSSVM
          and obtain the optimal $\vec{\beta}^{(\tau)}$
      step4 : Compute $f_1(\vec{\beta}^{(\tau)}), \ldots, f_{p+1}(\vec{\beta}^{(\tau)})$
      step5 : $\triangle u = | 1 - \frac{\Sigma_{j=1}^{p+1} \theta_i^{(\tau-1)} f_j(\vec{\beta}^{(\tau)})}{u^{(\tau-1)}} |$
  }
  comment: $\tau$ is the indicator of the current loop
  return $(\vec{\theta}^*, \vec{\beta}^*)$
```

Additional material

Additional file 1: The supplementary material contains (1) Genomic data sources used in experiment 1 and 2; (2) MKL extensions for Weighted SVM and Weighted LSSVM; (3) Kernel functions used in the paper; (4) Optimal kernel coefficients and performance of individual data sources in prostate cancer genes prioritization; (5) Performance of individual kernels in experiment 4; (6) Optimal weights assigned on each individual kernels in Experiment 4; (7) The effect of cost function regularization parameter λ of LSSVM in experiment 4; (8) Experimental results using MKL algorithms based on other norms.

Acknowledgements

The work was supported by Research Council KUL: GOA AMBioRICS, CoE EF/05/007 SymBioSys, PROMETA, several PhD/postdoc and Fellow Grants; FWO: PhD/postdoc Grants, Projects G.0241.04(Functional Genomics), G.0499.04 (Statistics), G.0232.05(Cardiovascular), G.0318.05(subfunctionalization), G.0553.06(VitamineD), G.0302.07(SVM/Kernel), research communities(ICCoS, ANMMM, MLDM); IWT: PhD Grants, GBOU-McKnow-E(Knowledge management algorithms), GBOU-ANA(biosensors), TADBioScope-IT, Silicos; SBO-BioFrame, SBO-MoKa, TBMEndometriosis, TBM-IOTA3, O&O-Dsquare; Belgian Federal Science Policy Office: IUAP P6/25(BioMaGNet, Bioinformatics and Modeling: from Genomes to Networks, 2007-2011); EURTD: ERNSI: European Research Network on System Identification; FP6-NoE Biopattern; FP6-IP e-Tumours, FP6-MC-EST Bioptrain, FP6-STREP Strokemap.

Author details

[1]Bioinformatics Group, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, Heverlee B-3001, Belgium. [2]Systems, Models and Control Group, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, Heverlee B-3001, Belgium.

Authors' contributions

All authors conceived the project and design. SY performed the theoretical analysis, programmed the algorithms, analyzed the data and wrote the paper. TF investigated SIP and implemented SIP formulations for SVM and LSSVM. AD preprocessed the rectal cancer, endometrial, miscarriage and pregnancy data sets. AD also provided the code of clinical kernel construction. LCT provided the data sources, disease relevant benchmark genes and prostate cancer genes for gene prioritization application. LCT also compared the performance of prioritization on Endeavour system. JS is the promoter of TF. BDM is the promoter of AD and SY. YM is the promoter of SY and LCT. All authors read and approved the manuscript. AD is research assistant of the Fund for Scientific Research - Flanders (FWO-Vlaanderen) JS and YM are professor and BDM a full professor at the Katholieke Universiteit Leuven, Belgium. All authors read and approved the manuscript.

Received: 14 January 2010 Accepted: 8 June 2010
Published: 8 June 2010

References

1. Tretyakov K: **Methods of genomic data fusion: An overview.** 2006 [http://ats.cs.ut.ee/u/kt/hw/fusion/fusion.pdf].
2. Vapnik V: **The Nature of Statistical Learning Theory.** Springer-Verlag, New York 1995.
3. Shawe-Taylor J, Cristianini N: **Kernel methods for pattern analysis.** Cambridge: Cambridge University Press 2004.
4. Lanckriet GRG, Cristianini N, Bartlett P, Ghaoui LE, Jordan MI: **Learning the Kernel Matrix with Semidefinite Programming.** *Journal of Machine Learning Research* 2005, **5**:27-72.
5. Bach FR, Lanckriet GRG, Jordan MI: **Multiple kernel learning, conic duality, and the SMO algorithm.** *Proceedings of 21st International Conference of Machine Learning* 2004.
6. Lanckriet GRG, De Bie T, Cristianini N, Jordan MI, Noble WS: **A statistical framework for genomic data fusion.** *Bioinformatics* 2004, **20**:2626-2635.
7. De Bie T, Tranchevent LC, Van Oeffelen L, Moreau Y: **Kernel-based data fusion for gene prioritization.** *Bioinformatics* 2007, **23**:i125-i132.
8. Ng AY: **Feature selection, L1 vs. L2 regularization, and rotational invariance.** *Proceedings of 21st International Conference of Machine Learning* 2004.
9. Kloft M, Brefeld U, Sonnenburg S, Laskov P, Müller K, Zien A: **Efficient and Accurate Lp-norm Multiple Kernel Learning.** *Advances in Neural Information Processing Systems 22* 2009.
10. Grant M, Boyd S: **CVX: Matlab Software for Disciplined Convex Programming, version 1.21.** 2010 [http://cvxr.com/cvx].
11. Grant M, Boyd S: **Graph implementations for nonsmooth convex programs.** *Recent Advances in Learning and Control Lecture Notes in Control*

and Information Sciences Springer-Verlag LimitedBlondel V, Boyd S, Kimura H 2008, 95-110 [http://stanford.edu/~boyd/graph_dcp.html].

12. Tax DMJ, Duin RPW: **Support vector domain description.** *Pattern Recognition Letter* 1999, **20**:1191-1199.
13. Schölkopf B, Platt JC, Shawe-Taylor J, Smola AJ, Williamson RC: **Estimating the support of a high-dimensional distribution.** *Neural Computation* 2001, **13**:1443-1471.
14. Sedumi: [http://sedumi.ie.lehigh.edu/].
15. Andersen ED, Andersen KD: **The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm.** *High Perf Optimization* 2000, 197-232.
16. Kim SJ, Magnani A, Boyd S: **Optimal kernel selection in kernel fisher discriminant analysis.** *Proceeding of 23rd International Conference of Machine Learning* 2006.
17. Ye JP, Ji SH, Chen JH: **Multi-class discriminant kernel learning via convex programming.** *Journal of Machine Learning Research* 2008, **40**:719-758.
18. Sonnenburg S, Rätsch G, Schäfer C, Schölkopf B: **Large scale multiple kernel learning.** *Journal of Machine Learning Research* 2006, **7**:1531-1565.
19. Hettich R, Kortanek KO: **Semi-infinite programming: theory, methods, and applications.** *SIAM Review* 1993, **35(3)**:380-429.
20. Kaliski J, Haglin D, Roos C, Terlaky T: **Logarithmic barrier decomposition methods for semi-infinite programming.** *International Transactions in Operations Research* **4(4)**.
21. Reemtsen R: **Some other approximation methods for semi-infinite optimization problems.** *Jounral of Computational and Applied Mathematics* 1994, **53**:87-108.
22. Suykens JAK, Van Gestel T, Brabanter J, De Moor B, Vandewalle J: **Least Squares Support Vector Machines.** World Scientific Publishing, Singapore 2002.
23. Veropoulos K, N C, C C: **Controlling the sensitivity of support vector machines.** *Proc of the IJCAI 99* 1999, 55-60.
24. Zheng Y, Yang X, Beddoe G: **Reduction of False Positives in Polyp Detection Using Weighted Support Vector Machines.** *Proc. of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* 2007, 4433-4436.
25. Suykens JAK, De Brabanter J, Lukas L, Vandewalle J: **Weighted least squares support vector machines : robustness and sparse approximation.** *Neurocomputing, Special issue on fundamental and information processing aspects of neurocomputing* 2002, **48(1-4)**:85-105.
26. Cawley GC: **Leave-One-Out Cross-Validation Based Model Selection Criteria for Weighted LS-SVMs.** *Proc. of 2006 International Joint Conference on Neural Networks* 2006, 1661-1668.
27. Aerts S, Lambrechts D, Maity S, Van Loo P, Coessens B, De Smet F, Tranchevent LC, De Moor B, Marynen P, Hassan B, Carmeliet P, Moreau Y: **Gene prioritization through genomic data fusion.** *Nature Biotechnology* 2006, **24**:537-544.
28. Yu S, Van Vooren S, Tranchevent LC, De Moor B, Moreau Y: **Comparison of vocabularies, representations and ranking algorithms for gene prioritization by text mining.** *Bioinformatics* 2008, **24**:i119-i125.
29. Leslie C, Eskin E, Weston J, Noble WS: **The spectrum kernel: a string kernel for SVM protein classification.** *Proc. of the Pacific Symposium on Biocomputing 2002* 2002.
30. Eeles RA, Kote-Jarai Z, Giles GG, Olama AAA, Guy M, Jugurnauth SK, Mulholland S, Leongamornlert DA, Edwards SM, Morrison Jea: **Multiple newly identified loci associated with prostate cancer susceptibility.** *Nat Genet* 2008, **40**:316-321.
31. Thomas G, Jacobs KB, Yeager M, Kraft P, Wacholder S, Orr N, Yu K, Chatterjee N, Welch R, Hutchinson Aea: **Multiple loci identified in a genome-wide association study of prostate cancer.** *Nat Genet* 2008, **40**:310-315.
32. Gudmundsson J, Sulem P, Rafnar T, Bergthorsson JT, Manolescu A, Gudbjartsson D, Agnarsson BA, Sigurdsson A, Benediktsdottir KR, Blondal Tea: **Common sequence variants on 2p15 and Xp11.22 confer susceptibility to prostate cancer.** *Nat Genet* 2008, **40**:281-283.
33. Daemen A, Gevaert O, Ojeda F, Debucquoy A, Suykens JAK, Sempous C, Machiels JP, Haustermans K, De Moor B: **A kernel-based integration of genome-wide data for clinical decision support.** *Genome Medicine* 2009, **1**:39.
34. Daemen A, De Moor B: **Development of a kernel function for clinical data.** *Proc. of the 31th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* 2009, 5913-5917.
35. van den Bosch T, Daemen A, Gevaert O, Timmerman D: **Mathematical decision trees versus clinician based algorithms in the diagnosis of endometrial disease.** *Proc. of the 17th World Congress on Ultrasound in Obstetrics and Gynecology (ISUOG)* 2007, 412.
36. Bottomley C, Daemen A, Mukri F, Papageorghiou AT, Kirk E, A P, De Moor B, Timmerman D, Bourne T: **Functional linear discriminant analysis: a new longitudinal approach to the assessment of embryonic growth.** *Human Reproduction* 2007, **24(2)**:278-283.
37. Gevaert O, De Smet F, Timmerman D, Moreau Y, De Moor B: **Predicting the prognosis of breast cancer by integrating clinical and microarray data with Bayesian networks.** *Bioinformatics* 2006, **22(14)**:e184-e190.
38. Condous G, Okaro E, Khalid A, Timmerman D, Lu C, Zhou Y, Van Huffel S, Bourne T: **The use of a new logistic regression model for predicting the outcome of pregnancies of unknown location.** *Human Reproduction* 2004, **21**:278-283.
39. Moult J, Fidelis K, Kryshtafovych A, Rost B, Tramontano A: **Critical assessment of methods of protein structure prediction - Round VIII.** *Proteins: Structure, Function, and Bioinformatics* **77(S9)**.
40. Kloft M, Brefeld U, Laskov P, Sonnenburg S: **Non-sparse multiple kernel learning.** *NIPS 08 workshop: kernel learning automatic selection of optimal kernels* 2008.
41. Kowalski M, Szafranski M, Ralaivola L: **Multiple indefinite kernel learning with mixed norm regularization.** *Proc of the 26th International Conference of Machine Learning* 2009.

doi:10.1186/1471-2105-11-309
Cite this article as: Yu *et al.*: L_2-norm multiple kernel learning and its application to biomedical data fusion. *BMC Bioinformatics* 2010 **11**:309.