On Radius-Incorporated Multiple Kernel Learning

Xinwang Liu, Jianping Yin, and Jun Long

School of Computer, National University of Defense Technology, Changsha, Hunan, 410073, China
{1022xinwang.liu,jdragonnudt}@gmail.com, jpyin@nudt.edu.cn

Abstract. In this paper, we review existing radius-incorporated Multiple Kernel Learning (MKL) algorithms, trying to explore the similarities and differences, and provide a deep understanding of them. Our analysis and discussion uncover that traditional margin based MKL algorithms also take an approximate radius into consideration implicitly by base kernel normalization. We perform experiments to systematically compare a number of recently developed MKL algorithms, including radius-incorporated, margin based and discriminative variants, on four MKL benchmark data sets including Protein Subcellular Localization, Protein Fold Prediction, Oxford Flower17 and Caltech101 in terms of both the classification performance, measured by classification accuracy and mean average precision. We see that overall, radius-incorporated MKL algorithms achieve significant improvement over other counterparts in terms of classification performance.

Keywords: Multiple Kernel Learning, Support Vector Machines, Radius Margin Bound, Minimum Enclosing Ball, Kernel Methods.

1 Introduction

Kernel methods have achieved great successes in machine learning community and have been widely adopted. As well known, their performance heavily depends on the choice of kernels. Many efforts have been devoted to address this issue by designing data-dependent optimal kernel algorithms [10,1,4], so-called "learning kernels from data". Among these algorithms, Multiple Kernel Learning (MKL) algorithms have been paid intensive attention since they are not only capable of adaptively tuning an optimal kernel for a specific learning task, but also provide an elegant framework to integrate multiple heterogenous source data.

The idea of MKL can be applied to both margin and class separability maximization criteria, leading to margin-based [1,5,4] and discriminative MKL algorithms [14], respectively. In this paper, we mainly focus on margin based MKL algorithms due to the popularity of margin maximization framework. There are several research trends in existing margin based MKL algorithms. The first direction focuses on designing computationally efficient MKL algorithms [1,11,13]. The second one aims to develop non-sparse and non-linear combination MKL algorithms [13], which usually achieve superior performance compared with sparse

V. Torra et al. (Eds.): MDAI 2014, LNAI 8825, pp. 227–240, 2014.

counterparts. By arguing that the generalization error bound of SVMs is dependent on both radius and margin, the last direction simultaneously takes the margin and the radius of the minimum hyper-sphere which encloses all training samples in the multi-kernel induced feature space into consideration [3,4,6,7].

Our work in this paper follows the last direction by proposing a radius-incorporated MKL framework. Using this framework as a toolbox, we instantiate three different radius-incorporated MKL algorithms by approximating the radius of Minimum Enclosing Ball (MEB) with the trace of each base kernel, the trace of total scatter matrix, and the radiuses induced by each base kernel, respectively. We further theoretically show that the above three radius-incorporated MKL algorithms can be rewritten as the traditional MKL formulation with only one difference being that different linearly weighted equality constraints on the kernel combination coefficients are employed. Specifically, the trace of base kernels, the trace of total scatter matrix of base kernels, and the base radiuses of each base kernel are respectively applied to linearly weight the coefficients of each base kernel in the above radius-incorporated MKL algorithms. Moreover, we uncover the relationship between the radius-incorporated MKL algorithms with kernel normalization which is still an open issue in existing MKL literature. Though different kernel normalization manners have been used [5], there is still lack of a principled way to explain why this normalization should be employed and which normalization usually works well in real work applications. We answer these questions by pointing out that different normalization manners in essence correspond to different radius-incorporation manners, which further correspond to different criteria in minimizing the generalization error of SVMs. From this perspective, our proposed radius-incorporated framework builds a bridge between kernel normalization approaches and the generalization error criteria. The contributions of this paper are highlighted as follows:

- We propose a radius-incorporated MKL framework which learns the base kernel combination coefficients by simultaneously maximizing the margin between classes and minimizing the radius of MEB. Furthermore, three radius-incorporated MKL algorithms instantiated from the framework are proposed by calculating the radius of MEB with different approaches.
- We uncover the tight connection between kernel normalization and radius incorporation, which provides a potential explanation for different kernel normalization approaches in existing MKL algorithms.
- We systematically compare a number of recently developed radius-incorporated MKL algorithms in terms of classification accuracy, which paves a way for designing excellent radius-incorporated MKL algorithms.

Comprehensive experiments have been conducted on Protein Subcellular Localization, Protein Fold Prediction, Oxford Flower17, Caltech101 and Alzheimer's Disease data sets to compare the proposed radius-incorporated MKL algorithms with state-of-the-art MKL algorithms in terms of classification performance. As the experimental results indicate, our proposed radius-incorporated MKL algorithms achieve better or comparable performance compared to many state-of-the-art MKL algorithms, which validates the effectiveness of the proposed radius-incorporated MKL framework.

2 Related Work

In this section, we first review some margin based MKL algorithms, and then focus on the MKL algorithms in [3,4,6,7] which optimizes both radius of MEB and margin. Let $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ be a given training set, where $\mathbf{x}_i$ and $y_i \in \{-1, +1\}$ represent i-th training sample and its corresponding label, respectively. Let $\{\phi_p\}_{p=1}^m$ be a group of feature mappings where ϕ_p induces a kernel function k_p. One can define $\mathbf{K}_p$ as the kernel matrix computed with k_p on the training set $\{\mathbf{x}_i\}_{i=1}^n$. In existing MKL literature, each sample $\mathbf{x}_i$ is mapped onto m feature spaces by $\phi(\mathbf{x};\boldsymbol{\gamma}) \triangleq [\sqrt{\gamma_1}\phi_1(\mathbf{x}), \cdots, \sqrt{\gamma_m}\phi_m(\mathbf{x})]^\top$, where γ_p is the coefficient of the p-th base kernel. Correspondingly, the induced kernel function can be expressed as a linear combination of p base kernels, $k(\boldsymbol{\gamma}) = \sum_{p=1}^m \gamma_p k_p$ and $\mathbf{K}(\boldsymbol{\gamma}) = \sum_{p=1}^m \gamma_p \mathbf{K}_p$.

The objective of MKL algorithms is to learn the base kernel coefficients $\boldsymbol{\gamma}$ and the structural parameters $(\boldsymbol{\omega}, b)$ jointly. To achieve this goal, most of MKL algorithms [1,11,5] propose to minimize the following optimization problem,

$$\min_{\boldsymbol{\gamma},\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum\nolimits_{i=1}^n \xi_i \ s.t. \ y_i(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma}) + b) \geq 1 - \xi_i, \ \xi_i \geq 0, \forall i, \ \|\boldsymbol{\gamma}\|_q = 1, \ \boldsymbol{\gamma} \succeq 0, \tag{1}$$

where $\boldsymbol{\omega}$ is the normal of the separating hyperplane, b the bias term, $\boldsymbol{\xi} = [\xi_1, \cdots, \xi_n]^\top$ is the vector of slack variables, and $\boldsymbol{\gamma}$ is the base kernel coefficients. Another important issue in Eq. (1) is that $q > 1$ will induce non-sparse kernel coefficients (called non-sparse MKL) while $q = 1$ will lead to sparse kernel combination (called sparse MKL).

Several recent research on MKL has gradually realized the importance of radius of MEB in MKL and successfully incorporated this radius into the traditional MKL formulation, achieving better kernel learning performance [3,4,6]. The theoretical justification for the radius incorporation lies at that the generalization error bound of SVMs is dependent on both the margin and the radius of the MEB of training data [10]. Furthermore, as pointed out in [4], only maximizing the margin with respect to $\boldsymbol{\gamma}$ will cause scaling and initialization issues. A larger margin could be arbitrarily achieved by scaling $\boldsymbol{\gamma}$ to $\tau\boldsymbol{\gamma}$ ($\tau > 1$), and this will affect the convergency of the optimization problem. Usually, a norm-constraint is imposed on $\boldsymbol{\gamma}$ to address this issue. Nevertheless, identifying an appropriate norm-constraint for a given kernel learning task remain an open issue itself [5]. Moreover, even if a norm-constraint is imposed, a good kernel could still be misjudged as a poor one by simply down-scaling the corresponding kernel weight [4]. These issues can be removed or mitigated by the incorporation of radius information. In the following, we review the radius-incorporated MKL algorithms in literature.

The pioneering work on radius-incorporated MKL in [3] proposes to minimize the optimization problem in Eq (2).

$$\min_{\boldsymbol{\gamma},\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}R^2(\boldsymbol{\gamma})\|\boldsymbol{\omega}\|^2 + \frac{C}{2}\sum_{i=1}^{n}\xi_i^2 \ s.t.\ y_i(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b) \geq 1-\xi_i, \forall i, \sum_{p=1}^{m}\gamma_p = 1, \boldsymbol{\gamma}\succeq 0, \tag{2}$$

where R^2 is the squared radius of the MEB in the multi-kernel induced space and can be calculated as

$$R^2(\boldsymbol{\gamma}) \ = \ \left\{\max_{\boldsymbol{\beta}} \boldsymbol{\beta}^\top \mathrm{diag}(\mathbf{K}(\boldsymbol{\gamma})) - \boldsymbol{\beta}^\top\mathbf{K}(\boldsymbol{\gamma})\boldsymbol{\beta} \ \ s.t.\ \boldsymbol{\beta}^\top\mathbf{1} = 1,\ 0 \leq \beta_i,\ \forall i\right\}. \tag{3}$$

Like the margin, R^2 is also a function of $\boldsymbol{\gamma}$. Instead of solving the optimization problem in Eq. (2) directly, the authors turn to minimize the following upper bounding convex optimization problem:

$$\min_{\boldsymbol{\gamma},\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + \frac{C}{2\sum_{p=1}^{m}\gamma_p R_p^2}\sum_{i=1}^{n}\xi_i^2 \ s.t.\ y_i(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b) \geq 1-\xi_i, \forall i, \sum_{p=1}^{m}\gamma_p = 1, \boldsymbol{\gamma}\succeq 0, \tag{4}$$

where R_p^2 is the squared radius of the MEB in the p-th base kernel induced space and can be calculated as

$$R_p^2 \ = \ \left\{\max_{\boldsymbol{\beta}} \boldsymbol{\beta}^\top \mathrm{diag}(\mathbf{K}_p) - \boldsymbol{\beta}^\top\mathbf{K}_p\boldsymbol{\beta} \ \ s.t.\ \boldsymbol{\beta}^\top\mathbf{1} = 1,\ 0 \leq \beta_i,\ \forall i\right\}. \tag{5}$$

The work in [3] focuses on how to approximate the optimization problem in Eq. (3) with a convex one in Eq. (4), and does not address the scaling issue mentioned above. Differently, the work in [4] directly solves the optimization in Eq. (6) and carefully discusses how the scaling issue can be addressed.

$$\min_{\boldsymbol{\gamma},\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}R^2(\boldsymbol{\gamma})\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \ s.t.\ y_i(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b) \geq 1-\xi_i, \forall i,\ \xi_i \geq 0,\ \ \boldsymbol{\gamma}\succeq 0. \tag{6}$$

In detail, a tri-level optimization problem is proposed in that work,

$$\min_{\boldsymbol{\gamma}} \ \widehat{\mathcal{J}}(\boldsymbol{\gamma}) \quad s.t.\ \ \gamma_p \geq 0,\ \ \forall p. \tag{7}$$

where

$$\widehat{\mathcal{J}}(\boldsymbol{\gamma}) = \left\{\max_{\boldsymbol{\alpha}} \boldsymbol{\alpha}^\top\mathbf{1} - \frac{1}{2R^2(\boldsymbol{\gamma})}(\boldsymbol{\alpha}\circ\mathbf{y})^\top\mathbf{K}(\boldsymbol{\gamma})(\boldsymbol{\alpha}\circ\mathbf{y}) \ \ s.t. \quad \boldsymbol{\alpha}^\top\mathbf{y} = 0,\ 0 \leq \alpha_i \leq C,\ \forall i\right\} \tag{8}$$

and $R^2(\boldsymbol{\gamma})$ is calculated by Eq. (3). To solve the optimization problem, a tri-level optimization structure is developed accordingly. Specifically, in the first step, R^2 is computed by solving the Quadratic Programming (QP) in Eq. (3) with a given $\boldsymbol{\gamma}$. Then, the obtained R^2 is taken into Eq. (8) to solve another QP to calculate $\widehat{\mathcal{J}}(\boldsymbol{\gamma})$. The last step is to update the base kernel coefficients $\boldsymbol{\gamma}$. The above procedure is repeated until a stopping criterion is satisfied. Compared with traditional MKL algorithms, an extra QP is introduced and solved at each iteration. This can considerably increase the computation cost of SVMs based

MKL, especially when the size of training set is large. Moreover, the performance of MKL could be adversely affected by the notorious sensitivity of this radius to outliers. In [6], instead of directly incorporating the radius of MEB, the authors propose to incorporate its close relative, the trace of data scattering matrix, to avoid the above problems. Specifically, their optimization problems is as follows in Eq. (9),

$$\min_{\boldsymbol{\gamma},\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}\mathrm{tr}\left(\mathbf{St}(\boldsymbol{\gamma})\right)\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \;\; s.t.\; y_i(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b) \geq 1-\xi_i, \forall i,\; \xi_i \geq 0,\;\; \boldsymbol{\gamma} \succeq 0, \tag{9}$$

where

$$\mathrm{tr}\left(\mathbf{St}(\boldsymbol{\gamma})\right) = \mathrm{tr}(\mathbf{K}(\boldsymbol{\gamma})) - \frac{1}{n}\mathbf{1}^\top\mathbf{K}(\boldsymbol{\gamma})\mathbf{1} \tag{10}$$

with **1** is a column vector with all elements one. Though usually demonstrating superior performance from the experimental perspective, it is criticized from the theoretic perspective since it may not be a upper bound of generalization error bound such as Radius Margin Bound [10].

In the following, we propose a radius-incorporated MKL framework where different radius variants could be integrated. Then we theoretically show that radius-margin based framework can be equivalently addressed by solving a traditional margin based MKL algorithms, with a difference being that a weighted constraint on the base kernel coefficients encoding the radius information. Furthermore, we formally, for the first time, uncover the connection between radius incorporation and kernel normalization.

3 Radius-Incorporated MKL Framework

3.1 The Proposed Framework

The radius-incorporated MKL framework in this paper is presented as follows,

$$\min_{\boldsymbol{\gamma}} \mathcal{J}(\boldsymbol{\gamma}),\; s.t.\; \gamma_p \geq 0,\; \forall p. \tag{11}$$

where

$$\mathcal{J}(\boldsymbol{\gamma}) = \left\{ \min_{\boldsymbol{\omega},b} \frac{1}{2}R^2(\boldsymbol{\gamma})\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \;\; s.t.\; y_i\big(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b\big) \geq 1-\xi_i,\; \xi_i \geq 0, \forall i \right\} \tag{12}$$

Proposition 1. *$\mathcal{J}(\tau\boldsymbol{\gamma}) = \mathcal{J}(\boldsymbol{\gamma})$, where $\tau > 0$ is any positive scalar. And the SVM decision function using the combined kernel is not affected by τ.*

Proof. The proof is elaborated in our earlier publication [7].

Proposition 1 indicates that our formulation in Eq. (11) is invariant when the kernel combination weights are uniformly scaled up by a positive scalar τ. In this case, the optimal value of $\boldsymbol{\omega}$ will correspondingly be down scaled by $1/\tau$, leaving the SVMs decision function unchanged. Based on Proposition 1, the following

Theorem 1 demonstrates that our objective function can be rewritten as the common form used by the existing margin based MKL algorithms, with only one difference that a constraint is imposed on the kernel coefficients encoding the radius information.

Theorem 1. *The optimal solution of optimization problem in Eq. (11), denoted as $\boldsymbol{\gamma}^\star$, can be written as $\boldsymbol{\gamma}^\star = R^2(\boldsymbol{\gamma})\boldsymbol{\eta}^\star$, where $\boldsymbol{\eta}^\star$ is the optimal solution of the following optimization problem in Eq. (13),*

$$\min_{\boldsymbol{\eta}} \mathcal{J}(\boldsymbol{\eta}) \;\; s.t.\; R^2(\boldsymbol{\eta}) = 1,\, \eta_p \geq 0, \;\; \forall p. \tag{13}$$

where

$$\mathcal{J}(\boldsymbol{\eta}) = \Big\{ \min_{\boldsymbol{\omega},b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \;s.t.\; y_i\big(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\eta}) + b\big) \geq 1-\xi_i,\, \xi_i \geq 0, \forall i\Big\} \tag{14}$$

Proof. The proof is elaborated in our earlier publication [7].

Theorem 1 indicates our proposed radius-incorporated MKL framework in Eq. (11) can be reformulated as a traditional margin based one, with only one difference being that an additional constraint on the kernel combination coefficients encoding radius information, as shown in Eq. (13).

3.2 Radius-Incorporated MKL Variants

In the following, we instantiate the calculation of $R^2(\boldsymbol{\gamma})$ by three different approaches: $\mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma}))$, $\mathrm{Tr}(\mathbf{S}_\mathrm{t}(\boldsymbol{\gamma}))$ and $\sum_{p=1}^m \gamma_p R_p^2$.

TrK-margin MKL By substituting $R^2(\boldsymbol{\gamma})$ in Eq. (13) with $\mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma}))$, we obtain the objective of Tr**K**-margin MKL as follows in Eq. (15),

$$\min_{\boldsymbol{\gamma}} \min_{\boldsymbol{\omega},b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \;\; s.t. \;\;\; y_i\big(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma}) + b\big) \geq 1-\xi_i,\, \xi_i \geq 0,\; \mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma})) = 1,\; \gamma_p \geq 0. \tag{15}$$

where $\mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma})) = \sum_{p=1}^m \gamma_p \mathrm{Tr}(\mathbf{K}_p)$.

TrS$_\mathbf{t}$-margin MKL By substituting $R^2(\boldsymbol{\gamma})$ with $\mathrm{Tr}(\mathbf{S}_\mathrm{t}(\boldsymbol{\gamma}))$, we obtain the objective of Tr$\mathbf{S}_\mathrm{t}$-margin MKL as follows,

$$\min_{\boldsymbol{\gamma}} \min_{\boldsymbol{\omega},b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \;\; s.t.\; y_i\big(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma}) + b\big) \geq 1-\xi_i,\, \xi_i \geq 0,\; \mathrm{Tr}(\mathbf{S}_\mathrm{t}(\boldsymbol{\gamma})) = 1,\; \gamma_p \geq 0. \tag{16}$$

where $\mathrm{Tr}(\mathbf{S}_\mathrm{t}(\boldsymbol{\gamma})) = \mathrm{Tr}\left(\mathbf{K}(\boldsymbol{\gamma})\right) - \frac{1}{n}\mathbf{1}^\top\mathbf{K}(\boldsymbol{\gamma})\mathbf{1} = \sum_{p=1}^m \gamma_p \left(\mathrm{Tr}\left(\mathbf{K}_p\right) - \frac{1}{n}\mathbf{1}^\top\mathbf{K}_p\mathbf{1}\right)$.

Base Radiuses-margin MKL By substituting $R^2(\boldsymbol{\gamma})$ with $\sum_{p=1}^{m}\gamma_p R_p^2$, we obtain the objective of Base Radiuses margin (BR-margin) MKL as follows,

$$\min_{\boldsymbol{\gamma}} \min_{\boldsymbol{\omega},b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \ s.t. \ y_i\big(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma})+b\big) \geq 1-\xi_i, \ \sum_{p=1}^{m}\gamma_p R_p^2 = 1, \ \gamma_p \geq 0. \tag{17}$$

where $R_p^2\,(p=1,\cdots,m)$ is calculated by Eq. (5).

3.3 Algorithm

We propose an efficient algorithm to solve our proposed radius-incorporated MKL algorithms. We take the Tr**K**-margin MKL algorithm as an example to show how it can be efficiently solved while this derivation can be directly applied to Tr$\mathbf{S}_\mathrm{t}$-margin MKL and Base Radiuses-margin MKL algorithms.

By defining $\widetilde{\boldsymbol{\omega}_p} = \sqrt{\gamma_p}\boldsymbol{\omega}_p$, $(p=1,\cdots,m)$, Eq. (15) can be rewritten as

$$\min_{\boldsymbol{\gamma}} \min_{\widetilde{\boldsymbol{\omega}},b} \frac{1}{2}\sum_{p=1}^{m}\frac{\|\widetilde{\boldsymbol{\omega}_p}\|^2}{\gamma_p} + C\sum_{i=1}^{n}\xi_i \ s.t. \ y_i(\sum_{p=1}^{m}\widetilde{\boldsymbol{\omega}_p}^\top\phi_p(\mathbf{x}_i)+b) \geq 1-\xi_i, \ \xi_i \geq 0, \ \mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma})) = 1, \ \gamma_p \geq 0. \tag{18}$$

The Lagrange function of Eq. (23) with respect to $\boldsymbol{\gamma}$ is

$$L(\boldsymbol{\gamma};\tau) = \frac{1}{2}\sum_{p=1}^{m}\frac{\|\widetilde{\boldsymbol{\omega}_p}\|^2}{\gamma_p} + C\sum_{i=1}^{n}\xi_i + \tau\left(\sum_{p=1}^{m}\gamma_p\mathrm{Tr}(\mathbf{K}_p)-1\right). \tag{19}$$

By letting the derivative of Eq. (19) with respect to $\gamma_p\,(p=1,\cdots,m)$ be zero, we obtain,

$$\frac{\partial L(\boldsymbol{\gamma};\tau)}{\partial\gamma_p} = -\frac{1}{2}\frac{\|\widetilde{\boldsymbol{\omega}_p}\|^2}{\gamma_p^2} + \tau\mathrm{Tr}(\mathbf{K}_p) = 0. \tag{20}$$

From Eq. (20), the optimal kernel combination weights can be analytically calculated as,

$$\gamma_p = \frac{\|\widetilde{\boldsymbol{\omega}_p}\|}{\sqrt{\mathrm{Tr}\left(\mathbf{K}_p\right)}\left(\sum_{p=1}^{m}\sqrt{\mathrm{Tr}\left(\mathbf{K}_p\right)}\|\widetilde{\boldsymbol{\omega}_p}\|\right)} \tag{21}$$

The overall algorithm for solving the Tr**K**-margin MKL formulation is presented in Algorithm 1.

Algorithm 1. Proposed Radius-incorporated MKL Framework

```
Initialize $\boldsymbol{\gamma}_p^1$.
$i \leftarrow 1$
repeat
    Calculate $\widetilde{\boldsymbol{\omega}_p}^{i+1}$ $(p=1,\cdots,m)$ by a SVMs solver with $\boldsymbol{\gamma}_p^i$.
    Update $\boldsymbol{\gamma}^{i+1}$ with $\widetilde{\boldsymbol{\omega}_p}^{i+1}$ $(p=1,\cdots,m)$ by Eq. (21).
    $i \leftarrow i+1$
until Convergence
```

It is worth noting that Algorithm 1 can be directly applied to solve the Tr$\mathbf{S}_{\mathrm{t}}$-margin MKL and Base Radiuses-margin MKL algorithms with minor modification. In detail, one can achieve this goal by only substituting $\mathrm{Tr}\left(\mathbf{K}_p\right)$ in Eq. (21) with $\left(\mathrm{Tr}\left(\mathbf{K}_p\right) - \frac{1}{n}\mathbf{1}^\top\mathbf{K}_p\mathbf{1}\right)$ in Tr$\mathbf{S}_{\mathrm{t}}$-margin MKL and R_p^2 in Base Radiuses-margin MKL, respectively.

After we obtain the optimal $\widetilde{\boldsymbol{\omega}_p}^\star$, $(p = 1, \cdots, m)$, $b^\star$ and $\boldsymbol{\gamma}^\star$ by Algorithm 1, we can directly write the SVMs decision function as

$$f(\mathbf{x}) = \sum_{p=1}^{m} \gamma_p^\star \left(\widetilde{\boldsymbol{\omega}_p}^\star\right)^\top \phi_p(\mathbf{x}) + b^\star, \tag{22}$$

and it will be used for the prediction the labels of new samples.

3.4 Connections between Radius-Incorporation and Base Kernel Normalization

As mentioned in [5], the base kernel normalization is important for MKL and different normalization approaches will lead to fundamentally different results. However, little systematical analysis on base kernel normalization has been done in existing MKL literature. Moreover, there is also lack of a theoretical explanation for existing base kernel normalization approaches. In the following, we uncover that there is a tight relationship between radius-incorporated MKL algorithms with kernel normalization approaches. This finding builds a bridge between base kernel normalization and MKL optimization criteria.

There are two widely used base kernel normalization approaches: spherical normalization [11] and multiplicative normalization [5], in existing MKL literature. In the following, we show the connections between spherical normalization and Tr$\mathbf{K}$-margin MKL, and multiplicative normalization and Tr$\mathbf{S}_{\mathrm{t}}$-margin MKL, respectively. In detail, by normalizing each base kernel $\mathbf{K}_p$ $(p = 1, \cdots, m)$ to have unit trace as in [11], we obtain the following optimization problem,

$$\min_{\boldsymbol{\gamma}} \min_{\boldsymbol{\omega}, b} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \; s.t. \; y_i\left(\boldsymbol{\omega}^\top\phi(\mathbf{x}_i;\boldsymbol{\gamma}) + b\right) \geq 1 - \xi_i, \; \sum_{p=1}^{m}\gamma_p = 1, \; \gamma_p \geq 0, \tag{23}$$

which is the exact objective function widely adopted by existing MKL algorithms [11,13]. Therefore, we can clearly see that the current margin-based MKL algorithms essentially implicitly incorporate the radius information via Tr($\mathbf{K}$). Similar optimization problem can also be obtained by performing multiplicative normalization on each base kernels.

With the proposed radius-incorporated MKL framework as a tool, we can clearly observe the tight relationship between radius incorporation variants and base kernel normalization alternatives. Furthermore, this framework also establishes the connection between kernel normalization approaches and radius-margin optimization criteria, which potentially provides an explanation for kernel normalization approaches from the perspective of minimizing generalization error theory.

It is worth noting that there exists essential differences among the proposed three radius-incorporated MKL algorithms in terms of generalization error bound. Neither Tr$\mathbf{K}$-margin nor Tr$\mathbf{S}_{\mathrm{t}}$-margin criteria are the upper bound of generalization error due to that $\mathrm{Tr}(\mathbf{K}(\boldsymbol{\gamma}))$ and $\mathrm{Tr}(\mathbf{S}_{\mathrm{t}}(\boldsymbol{\gamma}))$ may not be an upper bound of $R^2(\boldsymbol{\gamma})$, the squared radius of MEB. Differently, the base radiuses-margin criterion is an upper bound of the generalization error since $\sum_{p=1}^{m} \gamma_p R^2$ is an upper bound of $R^2(\boldsymbol{\gamma})$ [3]. With this observation, we can infer that the widely used spherical normalization and multiplicative normalization in existing MKL literature do not strictly follow the generalization error bound. Though having such a deficiency, Tr$\mathbf{S}_{\mathrm{t}}$-margin criterion can usually achieve superior performance, which has been validated in our experiments.

4 Experimental Results

4.1 Experimental Setup

We conduct experiments to compare the proposed radius-incorporated MKL algorithms with many stat-of-the-art MKL algorithms such as SimpleMKL [11], Minimum Ball MKL (MBMKL) [4], Radius MKL (RMKL)[3], non-Sparse MKL (ℓ_p MKL)[5] with $p = 4/3$, 2, 4, Discriminative MKL (MK-FDA) [14], Union Weight MKL (UWMKL), and Single Best SVMs (Single) in terms of classification accuracy. All comparisons have been conducted on protein fold prediction[1], Oxford Flower17[2], Protein Subcellular Localization[3], and Caltech101[4]. When the whole kernel matrix is available, the training set, validation set and test set is partitioned according to $2 : 1 : 1$. For Caltech101, since the training kernel and test kernel are available separately, we randomly partition the original training kernel matrix into new training and validation kernels according to $3 : 2$ while keeping the original test kernels unchanged.

The codes for SimpleMKL, ℓ_p-MKL, and MK-FDA are respectively downloaded from the authors' websites[5,6,7]. We implement the MBMKL and RMKL based on SimpleMKL toolbox by ourself according to their papers. All source codes, kernel matrix and partitions used in our experiments can be download from the author's website[8]. The optimal regularization parameter C for all MKL algorithms is chosen from $[10^{-2}, 10^{-1}, \cdots, 10^{4}]$ while the regularization parameter λ for MK-FDA [14] is chosen from $[10^{-5}, 10^{-1}, \cdots, 10^{1}]$ on validation set. For the comparison of classification performance, both the classification accuracy (ACC) and maximum a posterior (mAP) criteria are adopted. To conduct

[1] `http://mkl.ucsd.edu/dataset/protein-fold-prediction`

[2] `http://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html`

[3] `http://mkl.ucsd.edu/dataset/protein-subcellular-localization`

[4] `http://mkl.ucsd.edu/dataset/ucsd-mit-caltech-101-mkl-dataset`

[5] `http://asi.insa-rouen.fr/enseignants/~arakoto/`

[6] `http://doc.ml.tu-berlin.de/nonsparse_mkl/`

[7] `http://www.public.asu.edu/~jye02/Software/index.html`

[8] `https://sites.google.com/site/xinwangliunudt/home?previewAsViewer=1`

Table 1. Performance comparison with statistical test on Protein Fold Prediction data set. Boldface means no statistical difference from the best one (p-Val ≥ 0.05). The two rows of each data set represent mean accuracy (mAP) and standard derivation error.

	Proposed			SimpleMKL	MBMKL	RMKL	ℓ_p-MKL [5]			MK-FDA	UWMKL	Single
	TrK	**TrS**$_t$	Radius	[11]	[4]	[3]	$p = 4/3$	$p = 2$	$p = 4$	[14]		
ACC	65.6 ±3.6	**68.4** **±2.7**	**66.9** **±2.2**	65.2 ±3.3	**66.3** **±4.9**	58.1 ±3.6	66.3 ±3.1	63.9 ±4.0	62.5 ±3.6	59.6 ±2.8	60.4 ±3.4	60.2 ±2.6
mAP	70.1 ±3.0	**72.6** **±3.0**	**72.5** **±2.0**	69.7 ±2.0	**70.9** ±5.0	59.3 ±4.1	69.1 ±3.2	66.0 ±3.6	64.0 ±4.2	**71.5** **±2.7**	62.9 ±4.2	66.8 ±1.9

a rigorous comparison, the paired *Student's t-test* is performed. The p-value of the *t-test* represents the probability that two sets of compared results come from distributions with an equal mean. A p-value of 0.05 is considered statistically significant. We repeat the experiments for five times on Caltech101 since there are only five partitions available, while this procedure is repeated ten times on the other data sets. The mean results, standard derivation, and the p-value are reported. The highest accuracy and those whose difference from the highest accuracy are not statistically significant are shown in bold for each data set. All the following experiments are conducted on a high performance cluster server, where each computational node is with 2.3GHz CPU and 16GB memory.

4.2 Experiments on Protein Fold Predication Dataset

As a MKL benchmark data set, Protein Fold Prediction data set has been widely used to evaluate the performance of MKL algorithms [2]. It has 12 different heterogenous data sources, including Amino Acid Composition, Predicted Secondary Structure, Hydrophobicity, Van Der Waals Volume, Polarity, Polarizability, PseAA Pseudo-Amino-Acid Composition at interval 1, 4, 14 and 30, Smith-Waterman scores with the BLOSUM 62 scoring matrix, and Smith-Waterman scores with the PAM 50 scoring matrix. According to [2], 12 base kernels are generated by applying the second order polynomial kernel and inner product (cosine) kernel to the first ten feature sets and the last two feature sets, respectively.

The experimental result on Protein Fold Predication dataset is given in Table 1. From this table, we observe that:

- Radius-incorporated MKL algorithms including **TrS**$_t$-MKL, Radius-MKL and MBMKL [4] significantly outperform other margin based MKL algorithms in terms of both classification accuracy and mAP. In terms of classification accuracy, the proposed **TrS**$_t$-MKL achieves 2.5% improvement over $\ell_{4/3}$-MKL, which is the best margin based MKL algorithm. This amount is enlarged to 2.9% when comparing **TrS**$_t$-MKL with the best margin based MKL algorithm in terms of mAP.
- Different radius-incorporated approaches lead to different classification performance. Compared with **TrK**-MKL, the other proposed **TrS**$_t$-MKL and

Table 2. Performance comparison with statistical test on Protein Subcellular Localization data set

	Proposed			SimpleMKL	MBMKL	RMKL	ℓ_p-MKL [5]			MK-FDA	UWMKL	Single
	TrK	**TrS**$_t$	Radius	[11]	[4]	[3]	$p = 4/3$	$p = 2$	$p = 4$	[14]		
ACC												
psortNeg	**91.1**	**91.1**	90.6	**90.7**	90.4	**90.8**	90.4	89.4	87.6	87.2	87.2	84.0
	±1.2	**±1.6**	±1.5	**±1.2**	±1.7	**±1.5**	±1.4	±1.7	±2.5	±1.8	±2.5	±1.6
psortPos	**86.8**	**86.6**	**86.3**	**86.5**	**85.8**	**86.7**	**87.1**	**86.2**	85.3	84.7	83.5	82.0
	±2.8	**±3.3**	**±2.9**	**±2.6**	**±2.8**	**±2.7**	**±2.8**	**±3.9**	±2.5	±3.1	±3.2	±3.5
plant	**91.5**	**92.0**	90.5	**92.1**	**91.5**	**92.0**	**91.8**	91.1	89.8	83.8	88.1	78.6
	±1.5	**±1.8**	±1.7	**±1.5**	**±1.4**	**±2.1**	**±2.0**	±1.9	±2.2	±3.0	±2.5	±2.2
mAP												
psortNeg	94.8	**95.0**	**94.9**	**94.9**	94.9	**95.1**	94.3	93.1	91.4	**95.0**	90.0	89.6
	±0.7	**±0.9**	**±0.7**	**±0.8**	±0.9	**±0.8**	±0.9	±1.0	±1.1	**±0.7**	±1.3	±1.6
psortPos	**93.6**	**93.3**	92.9	**93.5**	93.1	**93.7**	**93.0**	92.0	90.2	**93.6**	89.7	87.4
	±2.3	**±2.5**	±2.5	**±2.2**	±2.4	**±2.3**	**±2.5**	±3.0	±2.9	**±2.3**	±3.4	±3.2
plant	95.1	**95.2**	94.5	**95.4**	95.0	**95.0**	94.9	93.8	92.8	**95.3**	91.2	80.6
	±1.6	**±1.5**	±1.7	**±1.4**	±1.6	**±1.9**	±1.6	±1.6	±1.5	**±1.5**	±1.5	±1.4

Radius-MKL achieve better classification performance. This result implies that **TrS**$_t$ and Radius normalization is superior to the widely used **TrK** normalization.

4.3 Experiment on Protein Subcellular Localization Dataset

We apply the above MKL algorithms into the protein subcellular localization which places an important role in protein function prediction and protein interactions. Three protein subcellular localization data sets including plant, PsortPos and PsortNeg have been widely used as MKL benchmark data sets [15,5], where 69 base kernels: two kernels on phytogenetic trees, three kernels from BLAST E-values, and 64 sequence motif kernels are constructed.

The experimental results are given in Table 2, from which we observe that

- Though the difference among the compared MKL algorithms is marginal, the proposed **TrS**$_t$-MKL and RMKL [3] achieve the best performance on all three data sets in terms of both classification accuracy and mAP, which validate the necessity of radius incorporation.
- Among the proposed radius-incorporation approaches, the **TrS**$_t$-MKL obtains the best performance, which coincides with the practical consideration in [15,5], where the multiplicative normalization is employed. In essence, our proposed radius-incorporated MKL framework provide an explanation for the effectiveness of multiplicative normalization from the perspective of minimizing the radius-margin bound.

4.4 Experiments on Oxford Flower17 Dataset

We compare the above mentioned MKL algorithms on Oxford Flower17, which has been widely used as a MKL benchmark data set [8]. There are seven heterogeneous

Table 3. Performance comparison with statistical test on Oxford Flower17 data set

	Proposed			SimpleMKL	MBMKL	RMKL	ℓ_p-MKL [5]			MK-FDA	UWMKL	Single
	TrK	**TrS**$_t$	Radius	[11]	[4]	[3]	$p=4/3$	$p=2$	$p=4$	[14]		
ACC	84.7 ±2.2	**86.3** **±1.6**	**85.9** **±1.9**	83.2 ±1.4	**86.3** **±2.0**	84.3 ±2.1	84.6 ±2.0	84.7 ±1.8	84.8 ±1.7	82.4 ±2.1	84.8 ±1.7	70.4 ±3.8
mAP	90.0 ±0.9	**91.5** **±0.9**	91.3 ±0.9	88.9 ±1.0	**91.5** **±1.0**	90.1 ±1.0	90.0 ±1.0	90.0 ±1.0	90.0 ±1.0	90.1 ±1.0	90.0 ±1.1	75.3 ±2.9

data channels available for this data set. For each data channel, four types of kernels are applied: Gaussian kernel (i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma\right)$), Laplacian kernel (i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|/\sqrt{\sigma}\right)$), inverse square distance kernel (i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma + 1}$), and inverse distance kernel (i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|/\sqrt{\sigma} + 1}$), where σ is the kernel parameter. They represent different ways to utilize the dissimilar matrix provided in [8,9]. In our experiments, 3 kernel parameters $2^t\sigma_0$ ($t \in \{-1, 0, 1\}$) are employed for each type of kernel, where σ_0 is set to be the averaged pairwise distance. In this way, we generate 84 ($7 \times 4 \times 3$) base kernels (12 base kernels for each data source), and use them for all the MKL algorithms compared in our experiment.

The results on Oxford Flower17 is given in Table 3, from which we observe that the radius incorporated MKL algorithms including **TrS**$_t$-MKL, Base Radius-MKL and MBMKL [4] significantly outperform other margin based MKL algorithms. Specifically, both **TrS**$_t$-MKL and MBMKL achieve 1.5% achievement over ℓ_4-MKL, which achieves the best results among the margin based MKL algorithms. Similar results can also be observed in terms of mAP.

4.5 Experiments on Caltech101 Dataset

The Caltech101 MKL data set is a group of kernels derived from various visual features computed on the Caltech-101 object recognition task, where 15 training and 15 test examples are available for each object class. It is a MKL benchmark data set and is used here to evaluate the performance of the above MKL algorithms. Twenty-five image descriptors are extracted, including pixels, SIFT, PHOW (Pyramid Histogram Of visual Words), PHOG (Pyramid Histogram Of Gradients), Geometric Blur, the bio-inspired "Sparse Localized Features", V_1-like features, and high-throughput bio-inspired features. This data set includes the kernels computed with the above features for five random splits of training and test sets.

We train and test the above 12 MKL algorithms on the pre-defined training and test sets and the experimental results are given in Table 4. From which, we again observe that our proposed **TrS**$_t$-MKL gains 3.5% improvement in terms of classification accuracy over ℓ_2-MKL, which achieves the best results among the margin-based MKL algorithms. Besides, compared with the best margin based MKL algorithm, a 3.7% improvement is achieved in terms of mAP by the

Table 4. Performance comparison with statistical test on Caltech101 data set

	Proposed			SimpleMKL	MBMKL	RMKL	ℓ_p-MKL [5]			MK-FDA	UWMKL	Single
	TrK	**TrS**$_t$	Radius	[11]	[4]	[3]	$p = 4/3$	$p = 2$	$p = 4$	[14]		
ACC	64.0 ±1.3	**68.5** **±1.1**	**67.4** **±1.5**	63.7 ±1.3	**68.3** **±1.2**	64.8 ±1.7	65.0 ±1.4	65.2 ±1.5	65.1 ±1.5	60.4 ±1.1	65.0 ±1.5	60.7 1.5
mAP	66.1 ±0.7	**71.1** **±0.8**	69.2 ±0.9	65.7 ±0.8	**70.3** **±0.6**	66.8 ±0.9	67.4 ±1.0	67.4 ±1.1	67.4 ±1.1	64.3 ±0.6	67.4 ±1.1	64.8 ±1.2

proposed **TrS**$_t$-MKL. All experimental results together demonstrate the effectiveness of the radius-based MKL algorithms.

Based on the experimental results on Protein Fold Prediction, Protein Subcellular Localization, Oxford Flower17, Caltech101 data sets, we have the following remarks:

- It has been validated that the proposed **TrS**$_t$-MKL is usually able to achieve the best classification performance and least computational efficiency. By taking both classification performance and computational efficiency into consideration, it is clearly the best one. Actually, $\frac{\mathbf{TrS}_t}{n}$ is an approximation of the radius of MEB by assigning the treating each training sample equally, which can usually achieve more stable and better performance. More detail relationship between **TrS**$_t$-MKL and the radius of MEB is referred to [12].
- The proposed **TrS**$_t$-MKL usually achieves stable performance than **TrK**-MKL and Radius-MKL. This implies that the multiple normalization on base kernels should be used, other than the commonly used trace normalization in existing MKL literature.
- Among the proposed three radius-incorporated MKL algorithms, only the objective of Radius-MKL is an upper bound of generalization error. However, it does not imply the best results can be obtained by this algorithm. Instead, **TrS**$_t$-MKL is usually achieving better results.

5 Conclusion

In this paper, we propose a radius-incorporated MKL framework in which the margin between classes and the radius of minimum hyper-sphere enclosing all training samples are both considered in the objective functions. We theoretically show the proposed framework can be equivalently rewritten as the existing margin based MKL optimization problem, with only one difference being that a weighted norm constraint is adopted to encode the radius information. This finding connects the radius-incorporation issue and the base kernel normalization issue, which is paid little attention in existing MKL literature. Our framework indeed provides an explanation for existing base kernel normalization approaches, which is a pre-procession step in existing MKL literature, from minimizing generalization error bound perspective. Extensive experiments have been conducted on several benchmark datasets. As experimentally demonstrated, our algorithm gives the overall best classification performance among the compared algorithms.

Acknowledgements. This work is sponsored in part by the National Basic Research Program of China (973) under Grant No. 2014CB340303 and the National Natural Science Foundation of China (Project No. 60970034, 61170287 and 61105050).

References

1. Bach, F.R., Lanckriet, G.R.G., Jordan, M.I.: Multiple kernel learning, conic duality, and the smo algorithm. In: ICML, pp. 649–657 (2004)
2. Damoulas, T., Girolami, M.A.: Probabilistic multi-class multi-kernel learning: on protein fold recognition and remote homology detection. Bioinformatics 24(10), 1264–1270 (2008)
3. Do, H., Kalousis, A., Woznica, A., Hilario, M.: Margin and radius based multile kernel learning. In: ICML, pp. 330–343 (2009)
4. Gai, K., Chen, G., Zhang, C.: Learning kernels with radiuses of minimum enclosing balls. In: NIPS, pp. 649–657 (2010)
5. Kloft, M., Brefeld, U., Sonnenburg, S., Zien, A.: ℓ_p-norm multiple kernel learning. JMLR 12, 953–997 (2011)
6. Liu, X., Wang, L., Yin, J., Liu, L.: Incorporation of radius-info can be simple with simplemkl. Neurocomputing 89, 30–38 (2012)
7. Liu, X., Wang, L., Yin, J., Zhu, E., Zhang, J.: An efficient approach to integrating radius information into multiple kernel learning. IEEE T. Cybernetics 43(2), 557–569 (2013)
8. Nilsback, M.E., Zisserman, A.: A visual vocabulary for flower classification. In: CVPR, vol. 2, pp. 1447–1454 (2006)
9. Nilsback, M.E., Zisserman, A.: Delving deeper into the whorl of flower segmentation. Image Vision Comput. 28(6), 1049–1062 (2010)
10. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. Machine Learning 46, 131–159 (2002)
11. Rakotomamonjy, A., Bach, F., Grandvalet, Y., Canu, S.: Simplemkl. JMLR 9, 2491–2521 (2008)
12. Wang, L.: Feature selection with kernel class separability. IEEE Trans. PAMI 30, 1534–1546 (2008)
13. Xu, Z., Jin, R., Yang, H., King, I., Lyu, M.R.: Simple and efficient multiple kernel learning by group lasso. In: ICML, pp. 1175–1182 (2010)
14. Ye, J., Ji, S., Chen, J.: Multi-class discriminant kernel learning via convex programming. JMLR 9, 719–758 (2008)
15. Zien, A., Ong, C.S.: Multiclass multiple kernel learning. In: ICML, pp. 1191–1198 (2007)