# Multi-Class Classification with Maximum Margin Multiple Kernel

**Corinna Cortes** CORTES@GOOGLE.COM
Google Research, 76 Ninth Avenue, New York, NY 10011

**Mehryar Mohri** MOHRI@CIMS.NYU.EDU
Courant Institute and Google Research, 251 Mercer Street, New York, NY 10012

**Afshin Rostamizadeh** ROSTAMI@GOOGLE.COM
Google Research, 76 Ninth Avenue, New York, NY 10011

## Abstract

We present a new algorithm for multi-class classification with multiple kernels. Our algorithm is based on a natural notion of the *multi-class margin of a kernel*. We show that larger values of this quantity guarantee the existence of an accurate multi-class predictor and also define a family of multiple kernel algorithms based on the maximization of the multi-class margin of a kernel ($M^3K$). We present an extensive theoretical analysis in support of our algorithm, including novel multi-class Rademacher complexity margin bounds. Finally, we also report the results of a series of experiments with several data sets, including comparisons where we improve upon the performance of state-of-the-art algorithms both in binary and multi-class classification with multiple kernels.

## 1. Introduction

The problem of learning with multiple kernels has attracted much attention from the machine learning community in the last few years (see e.g. (Lanckriet et al., 2004; Bach et al., 2004; Kloft et al., 2011) and the vast list of references in (Cortes et al., 2011)). Unlike the standard use of kernel methods where the critical step of selecting a suitable kernel for a task is left to the user, multiple kernel algorithms instead require the user only to supply a family of kernels. The algorithm then uses the training data to both select the appropriate kernel out of that family and to determine a good hypothesis based on that kernel.

*Proceedings of the $30^{th}$ International Conference on Machine Learning*, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. 

Much of the literature deals with the problem of learning kernels in the binary classification case or regression setting, while the focus of this paper is on learning with multiple kernels in the multi-class classification setting. Improvements in multi-class classification performance has emerged as one of the success stories in multiple kernel learning. While it has proven surprisingly difficult to outperform the simple uniform combination of base kernels for binary classification and regression problems, multi-class classification has benefited from a number of improvements due to multiple kernel learning. Zien & Ong (2007) present a one-stage multi-class multiple kernel learning (MCMKL) algorithm as a generalization of the multi-class loss function (Crammer & Singer, 2001; Tsochantaridis et al., 2004). The kernel and the classifiers are trained as a joint semi-infinite linear program (SILP) problem. The optimization over the kernel combination is carried out with an $L_1$ regularization that enforces sparsity in the kernel domain. They report significant performance improvements for this algorithm over the state-of-the-art in terms of AUC, Matthews Correlation Coefficient, and F1-score on a number of real-world datasets from cell biology.

In (Orabona et al., 2010) and (Orabona & Jie, 2011), stochastic gradient decent methods (named OBSCURE and UFO-MKL, respectively) are used to optimize primal versions of equivalent problems that select linear combinations of kernels with $L_p$-norm or mixed-norm regularization terms. The mixed regularization is selected specifically to allow for a strongly convex objective function, which can be optimized efficiently using a mirror descent-based algorithm. Since the problem is solved in the primal, general loss functions including the multi-class loss function can be used. In (Orabona & Jie, 2011), the OBSCURE and UFO-MKL algorithms are compared against MCMKL

and performance improvements in terms of misclassification accuracy are reported for a multi-class image classification problem. The OBSCURE algorithm is also shown to perform comparably to the state-of-the-art LP-$\beta$ algorithm of Gehler et al. (Gehler & Nowozin, 2009). LP-$\beta$ is a two-stage ensemble-based algorithm where multi-class classifiers are first trained independently for each kernel, then the resulting classifiers are combined by solving an LP problem. Most recently, Kumar et al. (2012) modeled kernel selection as a binary classification problem and introduced a multi-class kernel learning algorithm, BinaryMKL, which learns non-positive kernel weights that aim to maximize the distance between points of differing classes. There are several technical issues with the paper (Kumar et al., 2012), regarding both the theory and the algorithm, some of which we mention specifically later in this paper.

We present a new algorithm for multi-class classification with multiple kernels. Our algorithm is based on a natural notion of the *multi-class margin of a kernel*. We show that large values of this quantity guarantee the existence of an accurate multi-class predictor in the Hilbert space associated to the kernel. This leads us to the definition of a family of multiple kernel algorithms ($\mathrm{M^3K}$) based on the maximization of the multi-class margin of a kernel or its corresponding regularization. We present an extensive theoretical analysis in support of our algorithm, including novel multi-class Rademacher complexity margin bounds. We also report the results of experiments with several data sets, including comparisons where we improve upon the performance of state-of-the-art both in binary and multi-class classification with multiple kernels.

## 2. Preliminaries

We consider a standard multi-class classification supervised learning problem with $c \geq 2$ classes. Let $\mathcal{X}$ denote the input space and let $\mathcal{Y} = \{1, \ldots, c\}$ the set of classes. We assume that the learner receives a labeled sample $S = ((x_1, y_1), \ldots, (x_m, y_m)) \in \mathcal{X} \times \mathcal{Y}$ of size $m$ drawn i.i.d. according to an unknown distribution $D$ over $\mathcal{X} \times \mathcal{Y}$.

Consider a family $H$ of hypotheses mapping from $\mathcal{X} \times \mathcal{Y}$ to $\mathbb{R}$. In multi-class classification, the label predicted by $h \in H$ for point $x$ is chosen as $\operatorname{argmax}_{y \in \mathcal{Y}} h(x, y)$. For any hypothesis $h \in H$, $\rho_h(x, y)$ denotes its multi-class margin for the pair $(x, y)$:

$$\rho_h(x, y) = h(x, y) - \max_{y' \neq y} h(x, y'). \tag{1}$$

We will say that $h$ misclassifies point $x$ when $\rho_h(x, y) \leq 0$ for a labeled example $(x, y)$. The generalization error of $h$ is denoted by $R(h)$ and defined by $R(h) = \mathrm{E}_{(x,y)\sim D}[1_{\rho_h(x,y)\leq 0}]$. We will denote by $\widehat{D}$ the empirical distribution defined by the sample $S$. The empirical error of $h \in H$ is then defined by $\widehat{R}(h) = \mathrm{E}_{(x,y)\sim \widehat{D}}[1_{\rho_h(x,y)\leq 0}]$.

We assume that $p \geq 1$ positive semi-definite (PSD) base kernels over $\mathcal{X} \times \mathcal{X}$ are given and we consider a hypothesis set based on a kernel $K$ of the form $K = \sum_{k=1}^{p} \mu_k K_k$ where $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)^\top$ is chosen from $\Delta_q = \{\boldsymbol{\mu} \colon \boldsymbol{\mu} \geq 0, \|\boldsymbol{\mu}\|_q = 1\}$ with $q \geq 1$. We typically consider the case $q = 1$, but much of our analysis holds for $q > 1$. The hypothesis set we consider is based on the kernel property introduced in the next section.

## 3. Multi-class kernel margin

We first introduce a natural measure of the quality of a PSD kernel in the multi-class setting.

**Definition 1** (multi-class kernel margin)**.** *For any PSD kernel $K$, we define the* multi-class kernel margin of $K$ for a labeled instance $(x, y) \in \mathcal{X} \times \mathcal{Y}$ *as the minimum difference between the average $K$-similarity of $x$ to points belonging to its class and its similarity to points in any other class and denote this quantity by $\gamma_K(x, y)$:*

$$\gamma_K(x, y) = \mathop{\mathrm{E}}_{(x',y')\sim D}[K(x, x')|y' = y] - \max_{y' \neq y} \mathop{\mathrm{E}}_{(x'',y'')\sim D}[K(x, x'')|y'' = y']. \tag{2}$$

*We define the* multi-class kernel margin of $K$ *as* $\overline{\gamma}_K = \mathrm{E}_{(x,y)\sim D}[\gamma_K(x, y)]$.

Our notion of kernel margin is distinct from the one maximized by BinaryMKL (Kumar et al., 2012) which, for every pair of points $(x, x')$, creates an instance $(K_1(x, x'), \ldots, K_p(x, x'))$ with binary label $\mathbf{1}_{y=y'}$ and then learns a weight vector $\boldsymbol{\mu}$ using a linear SVM objective with a non-negativity constraint $\boldsymbol{\mu} \geq 0$: the BinaryMKL objective aims to maximize the difference between any two distinct classes, while $\gamma_K$ is defined based upon the difference of class $y$ and only the *closest* distinct class $y'$. Our choice closely matches the margin quantity relevant in the multi-class setting (1) and is further supported by the following proposition, which shows that for a kernel $K$ with a large multi-class margin, there exists a hypothesis $h^* \colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ defined by $h^*(x, y) = \mathrm{E}_{(x',y')\sim D}[K(x, x')|y' = y]$ admitting a small generalization error. We also point out that the theoretical guarantees provided in Kumar et al. (2012) do not appear to match the suggested algorithm: both the fact that the constructed training examples $(x, x')$ are no longer i.i.d. and the fact that

the learned SVM weights are constrained to be positive are not addressed in their analyses.

**Proposition 1.** *Let $K$ be a PSD kernel with $K(x,x) \leq 1$ for all $x \in \mathcal{X}$. Then, the following upper bound holds for the multi-class generalization error of $h^*$:*

$$R(h^*) \leq 1 - \frac{\overline{\gamma}_K}{\gamma_{K_{\max}}} \leq 1 - \frac{\overline{\gamma}_K}{2}. \tag{3}$$

*Proof.* By definition of $R(h^*)$, we can write

$$\begin{aligned} 1 - R(h^*) &= \mathop{\mathrm{E}}_{(x,y)\sim D}[1_{\rho_{h^*}(x,y)>0}] \\ &= \mathop{\mathrm{E}}_{(x,y)\sim D}[1_{h^*(x,y) - \max_{y' \neq y} h^*(x,y') > 0}] \\ &= \mathop{\mathrm{E}}_{(x,y)\sim D}[1_{\gamma_K(x,y)>0}]. \end{aligned}$$

For any $(x,y)$, we can write $1_{\gamma_K(x,y)>0} \geq \gamma_K(x,y)/\gamma_{K_{\max}}$. Therefore, $1 - R(h^*) \geq \mathrm{E}_{(x,y)\sim D}[\gamma_K(x,y)/\gamma_{K_{\max}}] = \overline{\gamma}_K/\gamma_{K_{\max}}$. Since $K(x,x) \leq 1$ for all $x \in \mathcal{X}$, by the Cauchy-Schwarz inequality, the inequality $|K(x,x')| \leq \sqrt{K(x,x)K(x',x')} \leq 1$ holds for all $x, x' \in \mathcal{X}$, which implies that $\gamma_K(x,y) \leq 2$ for all $(x,y)$ and completes the proof. □

This result further justifies the notion of margin introduced and motivates the algorithms described next.

## 4. Algorithms

### 4.1. Multi-class kernel margin maximization

In view of the definition and results of the previous section, a natural kernel learning algorithm consists of selecting $\boldsymbol{\mu}$ to maximize the empirical multi-class margin of the combination kernel $K_{\boldsymbol{\mu}} = \sum_{k=1}^p \mu_k K_k$. Let $C(y)$ denote the set of sample points in $S$ labeled with $y$: $C(y) = \{x_i \colon y_i = y, i \in [1,m]\}$. Then, the optimization problem can be written as follows:

$$\begin{aligned} \max_{\boldsymbol{\mu}\in\Delta_q} \frac{1}{m}\sum_{i=1}^m \min_{y\neq y_i} \Big\{ &\frac{1}{|C(y_i)|} \sum_{x'\in C(y_i)} K_{\boldsymbol{\mu}}(x_i,x') \\ &- \frac{1}{|C(y)|} \sum_{x'\in C(y)} K_{\boldsymbol{\mu}}(x_i,x') \Big\} \end{aligned} \tag{4}$$

For any $k \in [1,p]$, $i \in [1,m]$, and $y \in \mathcal{Y}$, we define

$$\eta_k(x_i,y_i,y) = \frac{\sum_{x'\in C(y_i)} K_k(x_i,x')}{|C(y_i)|} - \frac{\sum_{x'\in C(y)} K_k(x_i,x')}{|C(y)|},$$

and denote by $\boldsymbol{\eta}(x_i,y_i,y) \in \mathbb{R}^p$ the vector whose $k$th component is $\eta_k(x_i,y_i,y)$. Then, the optimization problem can be equivalently written as

$$\max_{\boldsymbol{\mu}\in\Delta_q} \sum_{i=1}^m \min_{y\neq y_i} \boldsymbol{\mu}\cdot\boldsymbol{\eta}(x_i,y_i,y).$$

Note that the coefficients $\boldsymbol{\eta}(x_i,y_i,y)$ are independent of $\boldsymbol{\mu}$ and can be precomputed for a given sample and set of kernels. Introducing new variables denoting the minima, the optimization problem can then be equivalently written as the following convex optimization problem which is a linear programming (LP) problem in the case $q = 1$:

$$\max_{\substack{\boldsymbol{\mu}\in\Delta_q,\\ \boldsymbol{\gamma}}} \sum_{i=1}^m \gamma_i \text{ s.t.} \forall i \in [1,m], \forall y \neq y_i, \boldsymbol{\mu}\cdot\boldsymbol{\eta}(x_i,y_i,y) \geq \gamma_i. \tag{5}$$

An alternative idea consists of maximizing the minimum kernel margin: $\max_{\boldsymbol{\mu}\in\Delta_q} \min_{x_i,y\neq y_i} \boldsymbol{\mu}\cdot\boldsymbol{\eta}(x_i,y_i,y)$. However, this does not directly match the requirement for the existence of the good multi-class solution discussed in the previous section and may be too strong a condition. We have in fact verified that it typically leads to a poor performance.

### 4.2. Maximum margin multiple kernel (M$^3$K) algorithm

Given a training sample, we can define the *empirical multi-class kernel margin* as follows:

$$\widehat{\gamma}_{K_{\boldsymbol{\mu}}} = \frac{1}{m}\sum_{i=1}^m \min_{y\neq y_i} \boldsymbol{\mu}\cdot\boldsymbol{\eta}(x_i,y_i,y), \tag{6}$$

which can then be used to define the data-dependent set $\widehat{\mathcal{M}}_q = \{\boldsymbol{\mu} \colon \boldsymbol{\mu} \in \Delta_q, \widehat{\gamma}_{K_{\boldsymbol{\mu}}} \geq \gamma_0\}$. This set can be incorporated as an additional form of regularization into a kernel learning optimization problem based on multi-class SVM (Weston & Watkins, 1999; Crammer & Singer, 2001):

$$\begin{aligned} \min_{\boldsymbol{\mu}\in\widehat{\mathcal{M}}_q,\mathbf{w},\boldsymbol{\xi}} \ &\frac{1}{2}\sum_{y=1}^c \sum_{k=1}^p \frac{\|\mathbf{w}_{y,k}\|^2}{\mu_k} + C\sum_{i=1}^m \xi_i \\ \text{subject to: } &\forall i \in [1,m],\ \xi_i \geq 0,\ \forall y \neq y_i, \\ &\xi_i \geq 1 - \left(\mathbf{w}_{y_i}\cdot\Phi(x_i) - \mathbf{w}_y\cdot\Phi(x_i)\right), \end{aligned} \tag{7}$$

where $C \geq 0$ is a regularization parameter. Here, we have defined for any class $y \in \mathcal{Y}$ the associated hypothesis $\mathbf{w}_y = (\mathbf{w}_{y,1},\ldots,\mathbf{w}_{y,p})^\top$ and let $\Phi(x) = (\Phi_{K_1}(x),\ldots,\Phi_{K_p}(x))^\top$, where $\Phi_K$ denote a feature mapping associated to the kernel $K$. We refer to the algorithm based on optimization (7) as the multi-class *maximum margin multiple kernel* (M$^3$K) algorithm.

The additional constraint $\widehat{\gamma}_K \geq \gamma_0$ in $\widehat{\mathcal{M}}_q$ ensures that $\boldsymbol{\mu}$ is selected such that the average empirical kernel margin is at least $\gamma_0$. It is important to note that if $\gamma_0$ is chosen to be too large, then the optimization problem becomes infeasible. There are in fact two extremes in choosing $\gamma_0$: setting it equal to the maximum feasible value will guarantee that the selected $\boldsymbol{\mu}$ is also

a solution to the optimization problem in (5), while setting it equal to $-\infty$ in the case $q = 1$ will reduce the algorithm to the MCMKL algorithm presented by (Zien & Ong, 2007).

The dual formulation of M$^3$K is written as follows:

$$\min_{\boldsymbol{\mu}\in\widehat{\mathcal{M}}_q} \max_{\boldsymbol{\alpha}\in\mathbb{R}^{m\times c}} \sum_{i=1}^{m} \boldsymbol{\alpha}_i\cdot\mathbf{e}_{y_i} - \frac{C}{2}\sum_{i,j=1}^{m}(\boldsymbol{\alpha}_i\cdot\boldsymbol{\alpha}_j)\sum_{k=1}^{p}\mu_k K_k(x_i,x_j)$$
$$\text{subject to: } \forall i\in[1,m],\ \boldsymbol{\alpha}_i \le \mathbf{e}_{y_i} \wedge \boldsymbol{\alpha}_i\cdot\mathbf{1} = 0.$$

Here, $\boldsymbol{\alpha}\in\mathbb{R}^{m\times c}$ is a matrix, $\boldsymbol{\alpha}_i$ is its $i$th row, and $\mathbf{e}_l$ the $l$th unit vector in $\mathbb{R}^c$, $l\in[1,c]$. In the case $q=1$, this min-max problem can be solved using a standard reduction to a SILP problem as in (Sonnenburg et al., 2006):

$$\min_{\boldsymbol{\mu}\in\widehat{\mathcal{M}}_1,\theta} \theta \quad \text{subject to:}$$
$$\forall\boldsymbol{\alpha}\in\left\{\boldsymbol{\alpha}\colon \forall i\in[1,m],\ \ \boldsymbol{\alpha}_i\le\mathbf{e}_{y_i}\wedge\boldsymbol{\alpha}_i\cdot\mathbf{1}=0\right\}$$
$$\theta \ge \sum_{i=1}^{m}\boldsymbol{\alpha}_i\cdot\mathbf{e}_{y_i} - \frac{C}{2}\sum_{i=1}^{m}(\boldsymbol{\alpha}_i\cdot\boldsymbol{\alpha}_j)\sum_{k=1}^{p}\mu_k K_k(x_i,x_j)\,.$$

This SILP problem is solved using a cutting-plane type algorithm, which considers only a finite subset of the constraints over $\boldsymbol{\alpha}$. Initially, we consider no $\boldsymbol{\alpha}$-based constraint and only find a feasible $\boldsymbol{\mu}\in\widehat{\mathcal{M}}_1$. We then find a most violated constraint $\boldsymbol{\alpha} = \max_{\boldsymbol{\alpha}}\sum_{i=1}^{m}\boldsymbol{\alpha}_i\cdot\mathbf{e}_{y_i} - \frac{C}{2}\sum_{i=1}^{m}(\boldsymbol{\alpha}_i\cdot\boldsymbol{\alpha}_j)\sum_{k=1}^{p}\mu_k K_k(x_i,x_j)$ and the constraint defined by this $\boldsymbol{\alpha}$ is added to the optimization problem. An optimal $\boldsymbol{\mu}$ that obeys all constraints added up to this point is then found and a new most violated $\boldsymbol{\alpha}$ is added to the optimization. These iterations continue until either a violating constraint cannot be found or the difference in successive choices of $\boldsymbol{\mu}$ is insignificant. At each iteration we solve an LP to find the current best choice of $\boldsymbol{\mu}$ and a quadratic program (QP) to find a most violated constraint $\boldsymbol{\alpha}$. Although in this paper we focus on $L_1$-regularized choices of $\boldsymbol{\mu}$, we note that it is also possible to solve the problem for other $L_q$ regularization ($q > 1$), or even using group norms over $\boldsymbol{\mu}$, with different optimization techniques.

## 5. Generalization bounds

In this section, we present generalization bounds for learning kernels in the multi-class setting for a hypothesis set based on our notion of multi-class margin. We start with a general margin-bound for multi-class classification, then analyze the empirical Rademacher complexity of the hypothesis set we consider to derive margin-based guarantees for multiple kernel learning in the multi-class setting.

### 5.1. General multi-class margin bounds

Let $H$ be a set of hypotheses mapping from $\mathcal{X}$ to $\mathbb{R}$. We will denote by $\widehat{\mathfrak{R}}_S(H)$ the empirical Rademacher complexity of the set $H$ for a sample $S$: $\widehat{\mathfrak{R}}_S(H) = \mathrm{E}_{\boldsymbol{\sigma}}\left[\sup_{h\in H}\frac{1}{m}\sum_{i=1}^{m}\sigma_i h(x_i)\right]$, where the $\sigma_i$s are independent uniform random variables taking values in $\{-1,+1\}$.

Fix $\rho > 0$, then, for any hypothesis $h\in H$, the empirical margin loss of $h$ in the multi-class setting can be defined by $\widehat{R}_\rho(h) = \frac{1}{m}\sum_{i=1}^{m} 1_{\rho_h(x_i,y_i)\le\rho}$. Let $H_{\mathcal{X}}$ denote the set of functions defined over $\mathcal{X}$ and derived from $H$ as follows: $H_{\mathcal{X}} = \{x\mapsto h(x,y)\colon y\in\mathcal{Y}, h\in H\}$. Then, the following general margin bound can be given in the multi-class setting. This is a simpler version of a result given by (Koltchinskii & Panchenko, 2002), a full proof is provided in the appendix.

**Theorem 1.** *Let $H\subseteq\mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ be a hypothesis set with $\mathcal{Y} = \{1,\ldots,c\}$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1-\delta$, the following multi-class classification generalization bound holds for all $h\in H$:*

$$R(h) \le \widehat{R}_\rho(h) + \frac{2c^2}{\rho}\widehat{\mathfrak{R}}_S(H_{\mathcal{X}}) + 3\sqrt{\frac{\log\frac{2}{\delta}}{2m}}\,. \tag{8}$$

As for all margin guarantees, the bound expresses a trade-off between margin maximization (larger $\rho$ values) and empirical margin loss minimization (smaller empirical margin loss values, $\widehat{R}_\rho$). The presence of the quadratic term $c^2$ suggests that larger margin values are required in the multi-class setting than in binary classification to achieve good generalization guarantees.

### 5.2. Multi-class margin bounds for multiple kernel learning

To apply this generalization bound in our context, we will analyze the empirical Rademacher complexity of a hypothesis set based on convex combinations of $p$ base kernels and with a lower bounded multi-class kernel margin. For the sake of brevity, our guarantees are presented in the case of an $L_1$ regularization for the mixture weights $\boldsymbol{\mu}$, but much of our analysis can be generalized to other $L_q$ and group-norm regularizations with $q > 1$. Each element of the hypothesis set $H^1$ is defined by $c$ functions $h_1,\ldots,h_c$ belonging to the Hilbert space $\mathbb{H}_{K_{\boldsymbol{\mu}}}$ defined by $K_{\boldsymbol{\mu}}$ where $K_{\boldsymbol{\mu}} = \sum_{k=1}^{p}\mu_k K_k$. Thus, the formal definition of $H^1$ is

$$H^1 = \Big\{(x,y)\in\mathcal{X}\times\mathcal{Y}\mapsto h_y(x)\colon \forall y\in\mathcal{Y}, h_y\in\mathbb{H}_{K_{\boldsymbol{\mu}}},$$
$$\|h_y\|_{K_{\boldsymbol{\mu}}}\le\Lambda, K_{\boldsymbol{\mu}} = \sum_{k=1}^{p}\mu_k K_k, \boldsymbol{\mu}\in\mathcal{M}_1\Big\},$$

where $\gamma_0 \in \mathbb{R}$, $\Lambda \geq 0$, and $\mathcal{M}_1 = \{\boldsymbol{\mu} \colon \boldsymbol{\mu} \in \Delta_1, \overline{\gamma}_{K_{\boldsymbol{\mu}}} \geq \gamma_0\}$. We will assume in what follows that $\gamma_0$ is chosen so that $\mathcal{M}_1 \neq \emptyset$, that is $\gamma_0$ is not above the maximum multi-class margin of $K_{\boldsymbol{\mu}}$ achievable by any $\boldsymbol{\mu} \in \Delta_1$.

The proof of our generalization bound is based on the following series of lemmas and partly makes use of some of the results and techniques given by (Cortes et al., 2010). The proof of the first lemma is given in the appendix.

**Lemma 1.** *For any labeled sample $S$ of size $m$, we have $\widehat{\mathfrak{R}}_S(H^1_{\mathcal{X}}) \leq \frac{\Lambda}{m} \operatorname{E}_{\boldsymbol{\sigma}}\left[\sup_{\boldsymbol{\mu} \in \mathcal{M}_1} \sqrt{\boldsymbol{\mu} \cdot \mathbf{u}_{\boldsymbol{\sigma}}}\right]$ with $\mathbf{u}_{\boldsymbol{\sigma}} = (\boldsymbol{\sigma}^\top \mathbf{K}_1 \boldsymbol{\sigma}, \ldots, \boldsymbol{\sigma}^\top \mathbf{K}_p \boldsymbol{\sigma})^\top$.*

In order to bound the Rademacher complexity, we first analyze and simplify the optimization problem $\sup_{\boldsymbol{\mu} \in \mathcal{M}_1} \sqrt{\boldsymbol{\mu}^\top \mathbf{u}_{\boldsymbol{\sigma}}}$. For any kernel $K$ and $x \in \mathcal{X}$, $y, y' \in \mathcal{Y}$ with $y \neq y'$, we define

$$\gamma_K(x, y, y') = \operatorname*{E}_{(x', y') \sim D}[K(x, x') | y' = y] - \operatorname*{E}_{(x'', y'') \sim D}[K(x, x'') | y'' = y'].$$

Thus, by definition of $\overline{\gamma}_K$, we have $\overline{\gamma}_K = \operatorname{E}_{(x,y) \sim D}[\min_{y' \neq y} \gamma_K(x, y, y')]$.

**Lemma 2.** *For any $k \in [1, p]$, we also define $\widetilde{\gamma}_k = \operatorname{E}_{(x,y) \sim D}[\frac{1}{c-1} \sum_{y' \neq y} \gamma_{K_k}(x, y, y')]$ and denote by $\widetilde{\boldsymbol{\gamma}} \in \mathbb{R}^p$ the vector whose $k$th coordinate is $\widetilde{\gamma}_k$. Then, the following inequality holds:*

$$\max_{\boldsymbol{\mu} \in \mathcal{M}_1} \boldsymbol{\mu}^\top \mathbf{u}_{\boldsymbol{\sigma}} \leq \min_{\lambda \geq 0} \max_{k \in [1,p]} \mathbf{u}_{\boldsymbol{\sigma}, k} + \lambda(\widetilde{\gamma}_k - \gamma_0).$$

*Proof.* By definition of $\gamma_k$s, we can write

$$\begin{aligned}
\overline{\gamma}_{K_{\boldsymbol{\mu}}} &= \operatorname*{E}_{(x,y) \sim D}\left[\min_{y' \neq y} \gamma_{K_{\boldsymbol{\mu}}}(x, y, y')\right] \\
&= \operatorname*{E}_{(x,y) \sim D}\left[\min_{y' \neq y} \sum_{k=1}^{p} \mu_k \gamma_k(x, y, y')\right] \\
&\leq \operatorname*{E}_{(x,y) \sim D}\left[\frac{1}{c-1} \sum_{y' \neq y} \sum_{k=1}^{p} \mu_k \gamma_k(x, y, y')\right] \\
&= \sum_{k=1}^{p} \mu_k \operatorname*{E}_{(x,y) \sim D}\left[\frac{1}{c-1} \sum_{y' \neq y} \gamma_k(x, y, y')\right] = \boldsymbol{\mu} \cdot \widetilde{\boldsymbol{\gamma}}.
\end{aligned}$$

Thus, $\overline{\gamma}_{K_{\boldsymbol{\mu}}} \geq \gamma_0$ implies $\boldsymbol{\mu} \cdot \widetilde{\boldsymbol{\gamma}} \geq \gamma_0$. Therefore, $\max_{\boldsymbol{\mu} \in \mathcal{M}_1} \boldsymbol{\mu}^\top \mathbf{u}_{\boldsymbol{\sigma}}$ is upper bounded by the optimum of the following LP problem:

$$\max_{\boldsymbol{\mu}} \boldsymbol{\mu}^\top \mathbf{u}_{\boldsymbol{\sigma}} \quad \text{subject to } (\boldsymbol{\mu} \in \Delta_1) \wedge (\boldsymbol{\mu}^\top \widetilde{\boldsymbol{\gamma}} \geq \gamma_0).$$

Introducing the dual variables $\boldsymbol{\beta} \in \mathbb{R}^p$, $\boldsymbol{\beta} \geq 0$, $\nu \in \mathbb{R}$, and $\lambda \geq 0$, the Lagrangian $L$ for this problem can be written as

$$L = -\boldsymbol{\mu}^\top \mathbf{u}_{\boldsymbol{\sigma}} - \boldsymbol{\beta}^\top \boldsymbol{\mu} + (-1 + \boldsymbol{\mu}^\top \mathbf{1})\nu + \lambda(\gamma_0 - \boldsymbol{\mu}^\top \widetilde{\boldsymbol{\gamma}}).$$

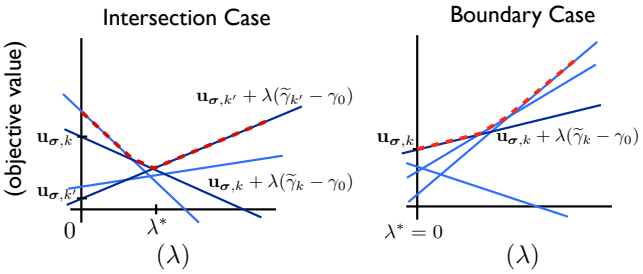


*Figure 1.* Each blue line in the figure above corresponds to a line indexed by $k$, with dependent variable $\lambda$. The dotted red line shows the maximum over these linear functions.

Computing its gradient with respect to $\boldsymbol{\mu}$ and setting it to zero gives

$$\nabla_{\boldsymbol{\mu}} L = 0 \Rightarrow \mathbf{u}_{\boldsymbol{\sigma}} + \boldsymbol{\beta} - \nu \mathbf{1} + \lambda \widetilde{\boldsymbol{\gamma}} = 0.$$

Solving for $\boldsymbol{\beta}$ and plugging in this identity in $L$ leads to the equivalent dual problem:

$$\min_{\nu, \lambda} \ \nu - \lambda \gamma_0 \quad \text{subject to } (\lambda \geq 0) \wedge (\mathbf{u}_{\boldsymbol{\sigma}} + \lambda \widetilde{\boldsymbol{\gamma}} \leq \nu \mathbf{1}).$$

Fixing $\lambda$ and solving for $\nu$ gives the following equivalent convex optimization problem:

$$\begin{aligned}
&\min_{\lambda \geq 0} \max_{k \in [1,p]} (\mathbf{u}_{\boldsymbol{\sigma},k} + \lambda \widetilde{\gamma}_k) - \lambda \gamma_0 \\
&\qquad = \min_{\lambda \geq 0} \max_{k \in [1,p]} \mathbf{u}_{\boldsymbol{\sigma},k} + \lambda(\widetilde{\gamma}_k - \gamma_0).
\end{aligned}$$

□

The optimization problem of lemma 2 is the minimization of a piecewise linear function, where each line segment is indexed by some $k \in [1, p]$. Assuming that the problem is feasible, the optimal solution falls into one of two cases illustrated in figure 1, where each blue line corresponds to a choice of $k$ and the red dotted line is the piecewise linear function that is being minimized over.

In the left panel of the figure, we see the general scenario where the optimal choice of $\lambda$ is described by the intersection of two lines indexed by $k$ and $k'$. Note that in this case, one line must have a non-positive slope, i.e. $\widetilde{\gamma}_k \leq \gamma_0$ and the other must have a non-negative slope, i.e. $\widetilde{\gamma}_{k'} \geq \gamma_0$. Thus, it suffices to consider only the intersection of lines indexed by $k$ and $k'$ that satisfy $\widetilde{\gamma}_k \leq \gamma_0 \leq \widetilde{\gamma}_{k'}$ (with $\widetilde{\gamma}_k \neq \widetilde{\gamma}_{k'}$).

The second case occurs iff for $k_{\max} = \operatorname{argmax}_k \mathbf{u}_{\boldsymbol{\sigma},k}$ we have $\widetilde{\gamma}_{k_{\max}} \geq \gamma_0$. In this case, the optimal choice of $\lambda$ is met at the boundary value 0 and the value of the optimal is simply $\mathbf{u}_{\boldsymbol{\sigma},k_{\max}}$. The following lemma describes these observations, with a formal proof found in the appendix. We first define the following sets

$$I_p = \{k \in [1, p] \colon \widetilde{\gamma}_k \geq \gamma_0\},$$
$$J_p = \{(k, k') \in [1, p]^2 \colon (\widetilde{\gamma}_k \leq \gamma_0 \leq \widetilde{\gamma}_{k'}) \wedge (\widetilde{\gamma}_k \neq \widetilde{\gamma}_{k'})\},$$

used throughout the remainder of the section.

**Lemma 3.** *Given the same definitions as in lemma 2, the following equality holds:*

$$\min_{\lambda \geq 0} \max_{k \in [1,p]} \mathbf{u}_{\boldsymbol{\sigma},k} + \lambda(\widetilde{\gamma}_k - \gamma_0) = \max \begin{Bmatrix} \max_{k \in I_p} \mathbf{u}_{\boldsymbol{\sigma},k}, \\ \max_{(k,k') \in J_p} \alpha_{k,k'} \mathbf{u}_{\boldsymbol{\sigma},k} + (1 - \alpha_{k,k'}) \mathbf{u}_{\boldsymbol{\sigma},k'} \end{Bmatrix},$$

*where* $\alpha_{k,k'} = \frac{\widetilde{\gamma}_{k'} - \gamma_0}{\widetilde{\gamma}_{k'} - \widetilde{\gamma}_k}$.

We now use this bound on the Rademacher complexity to derive our generalization bound.

**Theorem 2.** *Fix* $\rho > 0$ *and let* $p' = \mathrm{Card}(I_p) \leq p$ *and* $p'' = \mathrm{Card}(J_p) < p^2$. *Then, for any* $\delta > 0$, *with probability at least* $1 - \delta$ *over the choice of a sample* $S$ *of size* $m$, *the following multi-class classification generalization bound holds for all* $h \in H^1$:

$$R(h) \leq \widehat{R}_\rho(h) + \frac{2c^2\Lambda}{m\rho} \sqrt{\tfrac{23}{22} e \lceil \log(p' + p'') \rceil T_{\gamma_0}} + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}}, \quad (9)$$

*where* $T_{\gamma_0} = \max(\max_{k \in I_p} \mathrm{Tr}[\mathbf{K}_k], \max_{(k,k') \in J_p} \mathrm{Tr}[\mathbf{K}_{k,k'}])$, *with* $\mathbf{K}_{k,k'} = \alpha_{k,k'} \mathbf{K}_k + (1 - \alpha_{k,k'}) \mathbf{K}_{k'}$ *and* $\alpha_{k,k'} = \frac{\widetilde{\gamma}_{k'} - \gamma_0}{\widetilde{\gamma}_{k'} - \widetilde{\gamma}_k}$.

*Proof.* First, define the constant function $K_0 = 0$ and the constant $\widetilde{\gamma}_0 = -\infty$. With this notation, the entire right hand side of the equality in lemma 3 can be simply written as

$$\max_{(k,k') \in M_p} \alpha_{k,k'} \mathbf{u}_{\boldsymbol{\sigma},k} + (1 - \alpha_{k,k'}) \mathbf{u}_{\boldsymbol{\sigma},k'},$$

where $M_p = \big\{(k,k') \in [0,p]^2 \times [1,p] \colon (\widetilde{\gamma}_k \leq \gamma_0 \leq \widetilde{\gamma}_{k'}) \wedge (\widetilde{\gamma}_k \neq \widetilde{\gamma}_{k'})\big\}$. To see this, first note that $J_p$ is a subset of $M_p$. Furthermore, if we fix $k = 0$ then for any $k' \in [1,p]$ we have $\alpha_{0,k'} = 0$, which results in the expression $\max_{(0,k') \in M_p} \alpha_{0,k'} \mathbf{u}_{\boldsymbol{\sigma},0} + (1 - \alpha_{0,k'}) \mathbf{u}_{\boldsymbol{\sigma},k'} = \max_{k' \in I_p} \mathbf{u}_{\boldsymbol{\sigma},k'}$. Thus, the additional elements in $M_p$ account exactly for the elements in $I_p$.

Using this and combining lemma 1, lemma 2, and lemma 3, for any integer $r \geq 1$, we can write

$$\begin{aligned}
&\frac{m}{\Lambda} \widehat{\mathfrak{R}}_S(H^1_{\mathcal{X}}) \\
&\leq \mathop{\mathrm{E}}_{\boldsymbol{\sigma}} \Big[ \Big( \max_{(k,k') \in M_p} \alpha_{k,k'} \boldsymbol{\sigma}^\top \mathbf{K}_k \boldsymbol{\sigma} + (1 - \alpha_{k,k'}) \boldsymbol{\sigma}^\top \mathbf{K}_{k'} \boldsymbol{\sigma} \Big)^{\frac{1}{2}} \Big] \\
&= \mathop{\mathrm{E}}_{\boldsymbol{\sigma}} \Big[ \Big( \max_{(k,k') \in M_p} \boldsymbol{\sigma}^\top [\alpha_{k,k'} \mathbf{K}_k + (1 - \alpha_{k,k'}) \mathbf{K}_{k'}] \boldsymbol{\sigma} \Big)^{\frac{1}{2}} \Big] \\
&\leq \mathop{\mathrm{E}}_{\boldsymbol{\sigma}} \Big[ \Big[ \sum_{\substack{(k,k') \\ \in M_p}} (\boldsymbol{\sigma}^\top \mathbf{K}_{k,k'} \boldsymbol{\sigma})^r \Big]^{\frac{1}{2r}} \Big] \leq \Big[ \sum_{\substack{(k,k') \\ \in M_p}} \mathop{\mathrm{E}}_{\boldsymbol{\sigma}} \Big[ (\boldsymbol{\sigma}^\top \mathbf{K}_{k,k'} \boldsymbol{\sigma})^r \Big] \Big]^{\frac{1}{2r}},
\end{aligned}$$

where we used for the second inequality the fact that $\|\cdot\|_\infty$ is upper bounded by $\|\cdot\|_r$ for any $r \geq 1$ and for the last inequality the concavity of $x \mapsto x^{1/2r}$ and Jensen's inequality. By lemma 1 of (Cortes et al., 2010), the following inequality holds:[1] $\mathrm{E}_{\boldsymbol{\sigma}} \left[ (\boldsymbol{\sigma}^\top \mathbf{K}_{k,k'} \boldsymbol{\sigma})^r \right] \leq \left( \frac{23}{22} r \, \mathrm{Tr}[\mathbf{K}_{k,k'}] \right)^r$. Thus,

$$\begin{aligned}
\widehat{\mathfrak{R}}_S(H^1_{\mathcal{X}}) &\leq \frac{\Lambda}{m} \Big[ \sum_{(k,k') \in M_p} \Big( \tfrac{23}{22} r \, \mathrm{Tr}[\mathbf{K}_{k,k'}] \Big)^r \Big]^{\frac{1}{2r}} \\
&\leq \frac{\Lambda}{m} \sqrt{\tfrac{23}{22} (p' + p'')^{\frac{1}{r}} r \max_{(k,k') \in M_p} \mathrm{Tr}[\mathbf{K}_{k,k'}]}.
\end{aligned}$$

The function $r \mapsto r(p' + p'')^{\frac{1}{r}}$ reaches its minimum at $\log(p' + p'')$, thus this yields the inequality

$$\widehat{\mathfrak{R}}_S(H^1_{\mathcal{X}}) \leq \frac{\Lambda}{m} \sqrt{\tfrac{23}{22} e \lceil \log(p' + p'') \rceil \max_{(k,k') \in M_p} \mathrm{Tr}[\mathbf{K}_{k,k'}]}.$$

Finally, by the definition of $M_p$ we have

$$\max_{(k,k') \in M_p} \mathrm{Tr}[\mathbf{K}_{k,k'}] = \max(\max_{k \in I_p} \mathrm{Tr}[\mathbf{K}_k], \max_{(k,k') \in J_p} \mathrm{Tr}[\mathbf{K}_{k,k'}]).$$

Plugging in this upper bound on the Rademacher complexity of $H^1_{\mathcal{X}}$ in the learning guarantee of theorem 1 concludes the proof. □

The theorem gives a general margin bound for multiple kernel learning based on $\boldsymbol{\mu}$-combinations of $p$ based kernels, with an $L_1$ regularization for $\boldsymbol{\mu}$ augmented with the multi-class kernel margin regularization $\overline{\gamma}_{K_{\boldsymbol{\mu}}} \geq \gamma_0$. The effect of the $\gamma_{K_{\boldsymbol{\mu}}}$-regularization on the complexity term is analyzed by the following lemma.

**Lemma 4.** $T_{\gamma_0}$ *is a non-increasing function of* $\gamma_0$.

Due to space constraints we present the proof of the lemma in appendix E within the supplementary section. The lemma implies that the main complexity term that depends on $\gamma_0$ in the bound of theorem 2 becomes smaller as the regularization become more stringent. Also, note that the dependence on $p$ is only logarithmic. This weak dependence strongly encourages the idea of using a large number of base kernels.[2]

Altogether, this analysis provides strong support in favor of an algorithm minimizing the sum of the empirical margin loss of a multi-class hypothesis defined by

[1] This is a vectorial version of a Khintchine-Kahane inequality with an explicit constant more favorable than the best we are aware of for this context (Kwapien & Woyczynski, 1992) .

[2] We have also derived an alternative learning bound with a similar form and a simpler proof not making use of the results of (Cortes et al., 2010) (theorem 3 in appendix).

*Table 1.* Binary classification accuracy of various learning kernel algorithms as well as the performance of the uniform combination of kernels, reported with ±1 standard error on datasets SP (*splice*), SM (*spambase*).

| | UNIF | ALIGNF | MCMKL | $M^3K$ |
|---|---|---|---|---|
| SL | 84.8 ± 2.20 | 86.1 ± 1.28 | 84.9 ± 2.63 | 86.3 ± 0.75 |
| SM | 81.3 ± 2.84 | 82.0 ± 2.34 | 79.2 ± 2.42 | 85.7 ± 1.57 |

$(h_1, \ldots, h_c)$ and a complexity term based on the norm of these $c$ functions, while controlling the $L_1$-norm of $\boldsymbol{\mu}$ and maximizing the multi-class margin $\overline{\gamma}_{K_{\boldsymbol{\mu}}}$. This coincides precisely with our $M^3K$ algorithm modulo the maximization of the empirical multi-class margin $\widehat{\gamma}_{K_{\boldsymbol{\mu}}}$, the quantity computable from a finite sample, instead of $\overline{\gamma}_{K_{\boldsymbol{\mu}}}$. Since with high probability these two quantities are close modulo a term in $O(k/\sqrt{m})$, the bound also provides a strong foundation for our algorithm.

Other results with a similar analysis can be given for a regularization based on the $L_q$-norm of $\boldsymbol{\mu}$. In particular, for an $L_2$-norm regularization, the dependency on the number of kernels is of the form $O((p' + p'')^{1/4})$ instead of a logarithmic dependency.

## 6. Experiments

In this section, we report the results of several experiments with our multi-class multiple kernel $M^3K$ algorithm. First, we show that $M^3K$ performs well in binary classification tasks by comparing with MCMKL (Zien & Ong, 2007) and the best published results (Cortes et al., 2012). Next, in the multi-class setting, we compare against other state-of-the-art algorithms that learn a kernel for multi-class SVM (Crammer & Singer, 2001). These include BinaryMKL (Kumar et al., 2012), OBSCURE (Orabona et al., 2010) and UFO-MKL (Orabona & Jie, 2011). In all the experiments that follow, we consider the case of an $L_1$ regularization of $\boldsymbol{\mu}$ for the $M^3K$ algorithm.

### 6.1. Binary Classification

Table 1 shows the accuracy of several algorithms on the *splice* and *spambase* binary classification datasets. The accuracies are shown with ±1-standard deviation as measured over a 5-fold cross-validation with 1000 examples. We use the same experimental setup as in (Cortes et al., 2012) which uses 7–8 Gaussian kernels with various bandwidths as the set of base kernels (we refer the reader to that reference for further details of the methodology and datasets). For both of these datasets, the parameter $\gamma_0$ of the $M^3K$ algorithm is simply set to the maximum feasible value, which can be found using (5), and $C$ is found via a grid search. We not only find that $M^3K$ outperforms the uniform combination of kernels, which has proven to be a difficult baseline to beat, but also that it either matches or improves over the alignment-based algorithm (*alignf*) presented in (Cortes et al., 2012) and considered state-of-the-art on these datasets. As can be seen, $M^3K$ is also able to significantly outperform MCMKL. More generally, note that $M^3K$ strictly generalizes the MCMKL algorithm of (Zien & Ong, 2007) and thus always performs at least as well as MCMKL. Hence, in the next section, we focus on comparing against other state-of-the-art algorithms for multi-class kernel learning in the multi-class setting.

*Table 2.* Multi-class accuracy with ±1 standard deviation using the datasets P (plant), NP (nonplant), PSP (psortPos), PSN (psortNeg), PR (protein) with training split fraction SP and dataset size $n$.

| | SP | $n$ | UNIF | BINMKL | $M^3K$ |
|---|---|---|---|---|---|
| P | 0.5 | 940 | 86.9 ± 1.7 | 90.1 ± 1.4 | 91.2 ± 1.5 |
| NP | 0.5 | 2732 | 89.3 ± 0.8 | 87.7 ± 0.5 | 91.2 ± 0.9 |
| PSP | 0.8 | 541 | 88.4 ± 2.8 | 90.0 ± 3.0 | 90.8 ± 3.4 |
| PSN | 0.65 | 1444 | 87.9 ± 1.2 | 91.2 ± 0.8 | 91.5 ± 0.9 |
| PR | 0.5 | 694 | 59.2 ± 2.3 | 64.9 ± 2.6 | 67.2 ± 2.5 |

### 6.2. Multi-class Classification

In the multi-class setting, we compare to the uniform combination kernel baseline as well as the BinaryMKL algorithm using the biological datasets (*plant*, *nonplant*, *psortPos*, and *psortNeg*) that are also considered in (Kumar et al., 2012) (and originally in (Zien & Ong, 2007)), which consist of either 3, 4, or 5 classes and use 69 biologically motivated sequence kernels.[3] We also experiment with an additional biological dataset (*proteinFold*) of (Damoulas & Girolami, 2008), which consists of 27 classes and 12 base kernels.[4] Finally, we report the results of experiments with the *caltech101* vision-task dataset with 48 base kernels.[5] For each of the 102 classes, we select 30 examples (for a total of 3060 points) and then split these 30 examples into testing and training folds, which ensures matching training and testing distributions. For this dataset, we additionally compare with the OBSCURE and UFO-MKL algorithms which achieve state-of-the-art performance in this task. The choice of $C \in \{0.5, 1, 2, 4, \ldots\}$ and $\gamma_0$ (in the case of $M^3K$) is optimized via a grid search. For the OBSCURE and UFO-MKL algorithm, we follow the methodology of (Orabona et al., 2010) and (Orabona & Jie, 2011), respectively, for selecting parameters. All kernels are first centered and then scaled so that for all $i$ and $k$ we have $K_k(x_i, x_i) = 1$.

The multi-class accuracy of the uniform combination,

[3] http://raetschlab.org//projects/protsubloc.
[4] http://mkl.ucsd.edu/dataset/protein-fold-prediction.
[5] http://files.is.tue.mpg.de/pgehler/projects/iccv09.

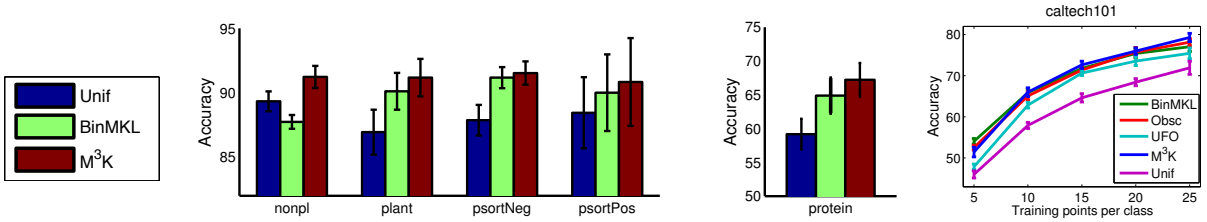


*Figure 2.* Accuracy rates when using a uniform combination of kernels or various multi-class kernel learning algorithms with biological datasets (left) as well as on the *caltech101* dataset with various training set sizes (right). Corresponding numerical values are found in table 2 (and table 4 in appendix G).

*Table 3.* Training time (in seconds) on a 12-core 2.67GHz Intel Xeon CPU with 24GB of RAM for different algorithms using the datasets P (*proteinFold*), C (*caltech101*), PSN (*psortNeg*), NP (*nonplant*).

| | $m$ | $c$ | $p$ | M$^3$K | BINMKL | OBSCURE | UFO |
|---|---|---|---|---|---|---|---|
| P | 347 | 27 | 12 | 51.9 | 3.4 | 11.6 | 0.2 |
| C | 1020 | 102 | 48 | 1048.8 | 24.6 | 330.9 | 18.8 |
| PSN | 936 | 5 | 69 | 190.6 | 238.6 | 66.5 | 2.4 |
| NP | 1366 | 3 | 69 | 821.8 | 1084.6 | 462.9 | 5.35 |

BinaryMKL, and M$^3$K algorithms for the biological datasets is shown in table 2 with one standard deviation as computed over 10 random splits of the data. We observe that for the biological datasets both kernel learning algorithms can perform better than the linear combination, while M$^3$K always performs at least as well as BinaryMKL and sometimes performs significantly better (in particularly *nonplant* and *proteinFold*). The results of figure 2 (right) show that M$^3$K performs comparably to all algorithms in the range of 10-20 training points per class for the *caltech101* dataset and performs even better than state-of-the-art algorithms when training with 25 points per class. Finally, we note that in simple multiclass tasks where overfitting is not an issue we observe that M$^3$K does not always provide a significant improvement over MCMKL. However, for more challenging tasks with more confusable classes, we expect significant improvements, as we found empirically even in some binary classification tasks.

The training time of different algorithms is shown in table 3 for several datasets. BinaryMKL is substantially faster than M$^3$K for the first two datasets of table 3. However, we observe that, when the number of kernels $p$ or the ratio $m/c$ is large, the training time of M$^3$K becomes more favorable than that of BinaryMKL, as in the next two datasets of the table.[6] In all cases, UFO-MKL, which uses a fast stochastic gradient descent method to solve the optimization problem, is significantly faster than all other algorithms. We believe that M$^3$K can benefit from a fast implementation similar to that of UFO-MKL and will actively pursue this question. Let us mention, however, that, as shown in the *caltech101* dataset, the increased speed of UFO-MKL appears to come at some cost in performance. Overall, we find M$^3$K to be a robust algorithm with a competitive performance in all datasets, including significant improvements in several cases.

[6] BinaryMKL subsamples to balance the negative and positive examples in the quadratic binary problem it generates, which can result in an effective training sample of size $2\sum_{j=1}^{c} m_j^2$, that is $2m^2/c$ when the classes sizes $m_j$ are equal, which can become very large depending on $m/c$. We also note that this subsampling is not reflected in the theoretical guarantees provided in (Kumar et al., 2012), which creates a disconnect between their theory and experiments.

## 7. Conclusion

We presented a new analysis of the problem of multiple kernel learning in the multi-class classification setting. We defined the notion of multi-class kernel margin, used it to define a new learning algorithm (M$^3$K), and presented new generalization bounds for hypothesis sets defined in terms of this margin. We also presented a series of empirical results demonstrating the good performance of our algorithm in practice. These results further motivate the search for more efficient solutions to the optimization problems introduced, as well as a finer analysis of alternative algorithms based on the multi-class kernel margin.

## References

Bach, Francis R., Lanckriet, Gert R. G., and Jordan, Michael I. Multiple kernel learning, conic duality, and the smo algorithm. In *ICML*, 2004.

Bartlett, Peter L. and Mendelson, Shahar. Rademacher and Gaussian complexities: Risk bounds and structural results. *JMLR*, 3, 2002.

Cortes, Corinna, Mohri, Mehryar, and Rostamizadeh, Afshin. Generalization bounds for learning kernels. In *ICML*, pp. 247–254, 2010.

Cortes, Corinna, Mohri, Mehryar, and Rostamizadeh, Afshin. Tutorial: Learning kernels. In *ICML*, 2011.

Cortes, Corinna, Mohri, Mehryar, and Rostamizadeh, Afshin. Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning*, 2012.

Crammer, Koby and Singer, Yoram. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2001.

Damoulas, Theodoros and Girolami, Mark A. Probabilistic multi-class multi-kernel learning: on protein fold recognition and remote homology detection. *Bioinformatics*, 24(10):1264–1270, 2008.

Gehler, P. and Nowozin, S. On feature combination for multiclass object classification. In *International Conference on Computer Vision*, pp. 221–228, 2009.

Kloft, M., Brefeld, U., Sonnenburg, S., and Zien, A. $L_p$-norm multiple kernel learning. *Journal of Machine Learning Research*, 12, 2011.

Koltchinskii, Vladmir and Panchenko, Dmitry. Empirical margin distributions and bounding the generalization error of combined classifiers. *Annals of Statistics*, 30, 2002.

Kumar, A., Niculescu-Mizil, A., Kavukcoglu, K., and Daumé III, H. A binary classification framework for two stage kernel learning. In *ICML*, 2012.

Kwapien, Stanislaw and Woyczynski, W Wojbor Andrzej. *Random series and stochastic integrals*. Birkhauser, 1992.

Lanckriet, Gert R. G., Cristianini, Nello, Bartlett, Peter L., Ghaoui, Laurent El, and Jordan, Michael I. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

Ledoux, Michel and Talagrand, Michel. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, New York, 1991.

Orabona, F. and Jie, L. Ultra-fast optimization algorithm for sparse multi kernel learning. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.

Orabona, F., Jie, L., and Caputo, B. Online-batch strongly convex multi kernel learning. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 787–794. IEEE, 2010.

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

Tsochantaridis, Ioannis, Hofmann, Thomas, Joachims, Thorsten, and Altun, Yasemin. Support vector machine learning for interdependent and structured output spaces. In *ICML 2004, Banff, Canada*, 2004.

Weston, Jason and Watkins, Chris. Support vector machines for multi-class pattern recognition. *European Symposium on Artificial Neural Networks*, 4 (6), 1999.

Zien, Alexander and Ong, Cheng Soon. Multiclass multiple kernel learning. In *ICML*, pp. 1191–1198, 2007.