Journal of Machine Learning Research 13 (2012) 607-642 Submitted 3/11; Revised 10/11; Published 3/12

Non-Sparse Multiple Kernel Fisher Discriminant Analysis

Fei Yan F.YAN@SURREY.AC.UK
Josef Kittler J.KITTLER@SURREY.AC.UK
Krystian Mikolajczyk K.MIKOLAJCZYK@SURREY.AC.UK
Atif Tahir M.TAHIR@SURREY.AC.UK
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, Surrey, United Kingdom, GU2 7XH

Editor: Sören Sonnenburg, Francis Bach, Cheng Soon Ong

Abstract

Sparsity-inducing multiple kernel Fisher discriminant analysis (MK-FDA) has been studied in the literature. Building on recent advances in non-sparse multiple kernel learning (MKL), we propose a non-sparse version of MK-FDA, which imposes a general ℓ_p norm regularisation on the kernel weights. We formulate the associated optimisation problem as a semi-infinite program (SIP), and adapt an iterative wrapper algorithm to solve it. We then discuss, in light of latest advances in MKL optimisation techniques, several reformulations and optimisation strategies that can potentially lead to significant improvements in the efficiency and scalability of MK-FDA. We carry out extensive experiments on six datasets from various application areas, and compare closely the performance of ℓ_p MK-FDA, fixed norm MK-FDA, and several variants of SVM-based MKL (MK-SVM). Our results demonstrate that ℓ_p MK-FDA improves upon sparse MK-FDA in many practical situations. The results also show that on image categorisation problems, ℓ_p MK-FDA tends to outperform its SVM counterpart. Finally, we also discuss the connection between (MK-)FDA and (MK-)SVM, under the unified framework of regularised kernel machines.

Keywords: multiple kernel learning, kernel fisher discriminant analysis, regularised least squares, support vector machines

1. Introduction

Since their introduction in the mid-1990s, kernel methods (Schölkopf and Smola, 2002; Shawe-Taylor and Cristianini, 2004) have proven successful for many machine learning problems, for example, classification, regression, dimensionality reduction, clustering. Representative methods such as support vector machine (SVM) (Vapnik, 1999; Shawe-Taylor and Cristianini, 2004), kernel Fisher discriminant analysis (kernel FDA) (Mika et al., 1999; Baudat and Anouar, 2000), kernel principal component analysis (kernel PCA) (Schölkopf et al., 1999) have been reported to produce state-of-the-art performance in numerous applications. Kernel methods work by embedding data items in an input space (vector, graph, string, etc.) into a feature space, and applying linear methods in the feature space. This embedding is defined implicitly by specifying an inner product for the feature space via a symmetric positive semidefinite (PSD) kernel function.

It is well recognised that in kernel methods, the choice of kernel function is critically important, since it completely determines the embedding of the data in the feature space. Ideally, this embedding should be learnt from training data. In practice, a relaxed version of this very challenging

problem is often considered: given multiple kernels capturing different "views" of the problem, how to learn an "optimal" combination of them. Among several others (Cristianini et al., 2002; Chapelle et al., 2002; Bousquet and Herrmann, 2003; Ong et al., 2003), Lanckriet et al. (2002, 2004) are one of the pioneering works for this multiple kernel learning (MKL) problem.

Lanckriet et al. (2002, 2004) study a binary classification problem, and their key idea is to learn a linear combination of a given set of base kernels by maximising the margin between the two classes or by maximising kernel alignment. More specifically, suppose one is given n $m \times m$ symmetric PSD kernel matrices $K_j, j = 1, \cdots, n$, and m class labels $y_i \in \{1, -1\}, i = 1, \cdots, m$. A linear combination of the n kernels under an ℓ_1 norm constraint is considered:

$$K = \sum_{j=1}^{n} \beta_j K_j, \ \boldsymbol{\beta} \geq \mathbf{0}, \ \|\boldsymbol{\beta}\|_1 = 1,$$

where $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_n)^T \in \mathbb{R}^n$, and $\mathbf{0}$ is the m dimensional vector of zeros. Geometrically, taking the sum of kernels can be interpreted as taking the Cartesian product of the associated feature spaces. Different scalings of the feature spaces lead to different embeddings of the data in the composite feature space. The goal of MKL is then to learn the optimal scaling of the feature spaces, such that the "separability" of the two classes in the composite feature space is maximised.

Lanckriet et al. (2002, 2004) propose to use the soft margin of SVM as a measure of separability, that is, to learn $\boldsymbol{\beta}$ by maximising the soft margin between the two classes. One of the most commonly used formulations of the resulting MKL problem is the following saddle point problem:

$$\max_{\boldsymbol{\beta}} \min_{\boldsymbol{\alpha}} -\mathbf{y}^T \boldsymbol{\alpha} + \frac{1}{2} \sum_{j=1}^{n} \boldsymbol{\alpha}^T \beta_j K_j \boldsymbol{\alpha} \tag{1}$$

$$\text{s.t.} \quad \mathbf{1}^T \boldsymbol{\alpha} = 0, \ \ \mathbf{0} \leq \mathbf{y}^T \boldsymbol{\alpha} \leq C\mathbf{1}, \ \ \boldsymbol{\beta} \geq \mathbf{0}, \ \ \|\boldsymbol{\beta}\|_1 \leq 1,$$

where $\boldsymbol{\alpha} \in \mathbb{R}^m$, $\mathbf{1}$ is the m dimensional vector of ones, $\mathbf{y}$ is the m dimensional vector of class labels, C is a parameter controlling the trade-off between regularisation and empirical error, and $K_j(\mathbf{x}_i, \mathbf{x}_{i'})$ is the dot product of the i^{th} and the i'^{th} training examples in the j^{th} feature space. Note that in Equation (1), we have replaced the constraint $\|\boldsymbol{\beta}\|_1 = 1$ by $\|\boldsymbol{\beta}\|_1 \leq 1$, which can be shown to have no effect on the solution of the problem, but allows for an easier generalisation.

Several alternative MKL formulations have been proposed (Lanckriet et al., 2004; Bach and Lanckriet, 2004; Sonnenburg et al., 2006; Zien and Ong, 2007; Rakotomamonjy et al., 2008). These formulations essentially solve the same problem as Equation (1), and differ only in the optimisation techniques used. The original semi-definite programming (SDP) formulation (Lanckriet et al., 2004) becomes intractable when m is in the order of thousands, while the semi-infinite linear programming (SILP) formulation (Sonnenburg et al., 2006) and the reduced gradient descent algorithm (Rakotomamonjy et al., 2008) can deal with much larger problems.

Of particular interest to this article is the SILP formulation in Sonnenburg et al. (2006). The authors propose to use a technique called column generation to solve the SILP, which involves dividing a SILP into an inner subproblem and an outer subproblem, and alternating between solving the two subproblems until convergence. A straightforward implementation of column generation leads to a conceptually very simple wrapper algorithm, where finding the optimal $\boldsymbol{\alpha}$ in the inner subproblem corresponds to solving a standard binary SVM. This means the wrapper algorithm can take advantage of existing efficient SVM solvers, and can be reasonably fast for medium-sized

problems already. However, as pointed out by Sonnenburg et al. (2006), solving the whole SVM problem to a high precision is unnecessary and therefore wasteful when the variable $\boldsymbol{\beta}$ in the outer subproblem is still far from the global optimum.

To remedy this, Sonnenburg et al. (2006) propose to optimise $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in an interleaved manner, by incorporating chunking (Joachims, 1988) into the inner subproblem. The key idea of chunking, and more generally decomposition techniques for SVM, is to freeze all but a small subset of $\boldsymbol{\alpha}$, and solve only a small-sized subproblems of the SVM dual in each iteration. The resulting interleaved algorithm in Sonnenburg et al. (2006) avoids the wasteful computation of the whole SVM dual, and as a result has an improved efficiency over the wrapper algorithm. Moreover, with the interleaved algorithm, only columns of the kernel matrices that correspond to the "active" dual variables need to be loaded into memory, extending MKL's applicability to large scale problems.

The learning problem in Equation (1) imposes an ℓ_1 regularisation on the kernel weights. It has been known that ℓ_1 norm regularisation tends to produce sparse solutions (Rätsch, 2001), which means during the learning most kernels are assigned zero weights. Conventionally, sparsity is favoured mainly for two reasons: it offers a better interpretability, and the test process is more efficient with sparse kernel weights. However, sparsity is not always desirable, since the information carried in the zero-weighted kernels is lost. In Kloft et al. (2008) and Cortes et al. (2009), non-sparse versions of MKL are proposed, where an ℓ_2 norm regularisation is imposed instead of ℓ_1 norm. Kloft et al. (2009, 2011) later extended their work to use a general ℓ_p ($p \geq 1$) norm regularisation. To solve the associated optimisation problem, Kloft et al. (2011) propose extensions of the wrapper and the interleaved algorithms in Sonnenburg et al. (2006) respectively. Experiments in Kloft et al. (2008, 2009, 2011) show that the regularisation norm contributes significantly to the performance of MKL, and confirm that in general a smaller regularisation norm produces more sparse kernel weights.

Although many of the above references discuss general loss functions (Lanckriet et al., 2004; Sonnenburg et al., 2006; Kloft et al., 2011), they have mainly been focusing on the binary hinge loss. In this sense, the corresponding MKL algorithms are essentially binary multiple kernel support vector machines (MK-SVMs). In contrast to SVM, which maximises the soft margin, Fisher discriminant analysis (FDA) (Fisher, 1936) maximises the ratio of projected between and within class scatters. Since its introduction in the 1930s, FDA has stood the test of time. Equipped recently with kernelisation (Mika et al., 1999; Baudat and Anouar, 2000) and efficient implementation (Cai et al., 2007), FDA has established itself as a strong competitor of SVM. In many comparative studies, FDA is reported to offer comparable or even better performance than SVM (Mika, 2002; Cai et al., 2007; Ye et al., 2008).

In Kim et al. (2006) and Ye et al. (2008), a multiple kernel FDA (MK-FDA) is introduced, where an ℓ_1 norm is used to regularise the kernel weights. As in the case of ℓ_1 MK-SVM, ℓ_1 MK-FDA tends to produce sparse selection results, which may lead to a loss of information. In this paper, we extend the work of Kim et al. (2006) and Ye et al. (2008) to a general ℓ_p norm regularisation by bringing latest advances in non-sparse MKL to MK-FDA. Our contribution can be summarised as follows:

- We provide a SIP formulation of ℓ_p MK-FDA for both binary and multiclass problems, and adapt the wrapper algorithm in Sonnenburg et al. (2006) to solve it. By considering recent advances in large scale MKL techniques, we also discuss several strategies that could significantly improve the efficiency and scalability of the wrapper-based ℓ_p MK-FDA. (Section 2)

- We carry out extensive experiments on six datasets, including one synthetic dataset, four object and image categorisation benchmarks, and one computational biology dataset. We confirm that as in the case of ℓ_p MK-SVM, in ℓ_p MK-FDA, a smaller regularisation norm in general leads to more sparse kernel weights. We also show that by selecting the regularisation norm p on an independent validation set, the "intrinsic sparsity" of the given set of base kernels can be learnt. As a result, using the learnt optimal norm p in ℓ_p MK-FDA offers better performance than fixed norm MK-FDAs. (Section 3)

- We compare closely the performance of ℓ_p MK-FDA and that of several variants of ℓ_p MK-SVM, and show that on object and image categorisation datasets, ℓ_p MK-FDA has a small but consistent edge. In terms of efficiency, our wrapper-based ℓ_p MK-FDA is comparable to the interleaved ℓ_p MK-SVM on small/medium sized binary problems, but can be significantly faster on multiclass problems. When compared against recently proposed MKL techniques that define the state-of-the-art, such as SMO-MKL (Vishwanathan et al., 2010) and OBSCURE (Orabona et al., 2010), our MK-FDA also compares favourably or similarly. (Section 3)

- Finally, we discuss the connection between (MK-)FDA and (MK-)SVM, from the perspectives of both loss function and version space, under the unified framework of regularised kernel machines. (Section 4)

Essentially, our work builds on Sonnenburg et al. (2006), Ye et al. (2008) and Kloft et al. (2011). However, we believe the empirical findings of this paper, especially the one that (MK-)FDA tends to outperform (MK-)SVM on image categorisation datasets, is important, given that SVM and SVM based MKL are widely accepted as the state-of-the-art classifier in most image categorisation systems. Finally, note that preliminary work to this article has been published previously as conference papers (Yan et al., 2009b,a, 2010). The aim of this article is to consolidate the results into an integrated and comprehensive account and to provide more experimental results in support of the proposed methodology.

2. ℓ_p Norm Multiple Kernel FDA

In this section we first present our ℓ_p regularised MK-FDA for binary problems and then for multiclass problems. In both cases, we first give problem formulation, then solve the associated optimisation problem using a wrapper algorithm. Towards the end of this section, we also discuss several possible improvements over the wrapper algorithm in terms of time and memory complexity, in light of recent advances in MKL optimisation techniques.

2.1 Binary Classification

Given a binary classification problem with m training examples, our goal is to learn the optimal kernel weights $\boldsymbol{\beta} \in \mathbb{R}^n$ for a linear combination of n base kernels under the ℓ_p ($p \geq 1$) constraint:

$$K = \sum_{j=1}^{n} \beta_j K_j, \;\; \beta_j \geq 0, \; \|\boldsymbol{\beta}\|_p^p \leq 1,$$

where the $p \geq 1$ requirement is to ensure that the triangle inequality is satisfied and $\|\cdot\|_p$ is a norm. We define optimality in terms of the class separation criterion of FDA, that is, the learnt kernel

weights $\boldsymbol{\beta}$ are optimal, if the ratio of the projected between and within class scatters is maximised. In this paper we assume each kernel is centred in its feature space. Centring can be performed implicitly (Schölkopf et al., 1999) by $K_j = P\tilde{K}_j P$, where P is the $m \times m$ centring matrix defined as $P = I - \frac{1}{m}\mathbf{1}\cdot\mathbf{1}^T$, $\tilde{K}_j$ is the uncentred kernel matrix, and I is the $m \times m$ identity matrix.

Let m^+ be the number of positive training examples, and $m^- = m - m^+$ be that of negative training examples. For a given kernel K, let $\phi(\mathbf{x}_i^+)$ be the i^{th} positive training point in the implicit feature space associated with K, $\phi(\mathbf{x}_i^-)$ be the i^{th} negative training point in the feature space. Here $\mathbf{x}_i^+$ and $\mathbf{x}_i^-$ can be thought of as training examples in some input space, and ϕ is the mapping to the feature space. Also let $\boldsymbol{\mu}^+$ and $\boldsymbol{\mu}^-$ be the centroids of the positive examples and negative examples in the feature space, respectively:

$$\boldsymbol{\mu}^+ = \frac{1}{m^+}\sum_{i=1}^{m^+}\phi(\mathbf{x}_i^+), \qquad \boldsymbol{\mu}^- = \frac{1}{m^-}\sum_{i=1}^{m^-}\phi(\mathbf{x}_i^-).$$

The within class covariance matrices of the two classes are:

$$C^+ = \frac{1}{m^+}\sum_{i=1}^{m^+}\left(\phi(\mathbf{x}_i^+) - \boldsymbol{\mu}^+\right)\left(\phi(\mathbf{x}_i^+) - \boldsymbol{\mu}^+\right)^T,$$
$$C^- = \frac{1}{m^-}\sum_{i=1}^{m^-}\left(\phi(\mathbf{x}_i^-) - \boldsymbol{\mu}^-\right)\left(\phi(\mathbf{x}_i^-) - \boldsymbol{\mu}^-\right)^T.$$

The between class scatter S_B and within class scatter S_w are then defined as:

$$\begin{aligned} S_B &= \frac{m^+m^-}{m}(\boldsymbol{\mu}^+ - \boldsymbol{\mu}^-)(\boldsymbol{\mu}^+ - \boldsymbol{\mu}^-)^T, \\ S_W &= m^+C^+ + m^-C^-. \end{aligned} \tag{2}$$

The objective of single kernel FDA is to find the projection direction $\mathbf{w}$ in the feature space that maximises $\frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}}$, or equivalently, $\frac{\mathbf{w}^T \frac{m}{m^+m^-} S_B \mathbf{w}}{\mathbf{w}^T S_T \mathbf{w}}$, where $S_T = S_B + S_W$ is the total scatter matrix. In practice a regularised objective function

$$J_{FDA}(\mathbf{w}) = \frac{\mathbf{w}^T \frac{m}{m^+m^-} S_B \mathbf{w}}{\mathbf{w}^T (S_T + \lambda I)\mathbf{w}} \tag{3}$$

is maximised to improve generalisation and numerical stability (Mika, 2002), where λ is a small positive number.

From Theorem 2.1 of Ye et al. (2008), for a given kernel K, the maximal value of Equation (3) is:

$$J^*_{FDA} = \mathbf{a}^T\mathbf{a} - \mathbf{a}^T\left(I + \frac{1}{\lambda}K\right)^{-1}\mathbf{a}, \tag{4}$$

where

$$\mathbf{a} = \left(\frac{1}{m^+}, \cdots, \frac{1}{m^+}, \frac{-1}{m^-}, \cdots, \frac{-1}{m^-}\right)^T \in \mathbb{R}^m$$

contains the centred labels. On the other hand, Lemma 2.1 of Ye et al. (2008) states that the $\mathbf{w}$ that maximises Equation (3) also minimises the following regularised least squares (RLS):

$$J_{RLS}(\mathbf{w}) = \|\phi^T(X)\mathbf{w} - \mathbf{a}\|^2 + \lambda\|\mathbf{w}\|^2, \tag{5}$$

and the minimum of Equation (5) is given by:

$$J^*_{RLS} = \mathbf{a}^T \left(I + \frac{1}{\lambda} K \right)^{-1} \mathbf{a}. \tag{6}$$

In Equation (5), $\phi(X) = (\phi(\mathbf{x}_1^+), \cdots, \phi(\mathbf{x}_{m^+}^+), \phi(\mathbf{x}_1^-), \cdots, \phi(\mathbf{x}_{m^-}^-))$ are the (centred) training data in the feature space such that $\phi(X)^T \phi(X) = K$.

Due to strong duality, the minimal value of Equation (5) is equal to the maximal value of its Lagrangian dual problem, that is,

$$J^*_{RLS} = \max_{\boldsymbol{\alpha}} \mathbf{a}^T \boldsymbol{\alpha} - \frac{1}{4} \boldsymbol{\alpha}^T \boldsymbol{\alpha} + \frac{1}{4\lambda} \boldsymbol{\alpha}^T K \boldsymbol{\alpha},$$

or equivalently

$$J^*_{RLS} = -\min_{\boldsymbol{\alpha}} \left(-\mathbf{a}^T \boldsymbol{\alpha} + \frac{1}{4} \boldsymbol{\alpha}^T \boldsymbol{\alpha} + \frac{1}{4\lambda} \boldsymbol{\alpha}^T K \boldsymbol{\alpha} \right), \tag{7}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^m$. By combining Equation (4), Equation (6) and Equation (7), it follows that the maximal value of the FDA objective in Equation (3) is given by:

$$J^*_{FDA} = \mathbf{a}^T \mathbf{a} + \min_{\boldsymbol{\alpha}} \left(-\mathbf{a}^T \boldsymbol{\alpha} + \frac{1}{4} \boldsymbol{\alpha}^T \boldsymbol{\alpha} + \frac{1}{4\lambda} \boldsymbol{\alpha}^T K \boldsymbol{\alpha} \right). \tag{8}$$

Now instead of a fixed single kernel, consider the case where the kernel K can be chosen from linear combinations of a set of base kernels. The kernel weights must be regularised somehow to make sure Equation (8) remains meaningful and does not become arbitrarily large. In this paper, we propose to impose an ℓ_p regularisation on the kernel weights for any $p \geq 1$, following Kloft et al. (2009, 2011):

$$\tilde{\mathcal{K}} = \left\{ K = \sum_{j=1}^{n} \beta_j K_j : \boldsymbol{\beta} \geq \mathbf{0}, \|\boldsymbol{\beta}\|_p^p \leq 1 \right\}. \tag{9}$$

Combining Equation (9) and Equation (8), and dropping the unimportant constant $\mathbf{a}^T \mathbf{a}$, it can be shown that the optimal $K \in \tilde{\mathcal{K}}$ maximising Equation (4) is found by solving:

$$\max_{\boldsymbol{\beta}} \min_{\boldsymbol{\alpha}} -\mathbf{a}^T \boldsymbol{\alpha} + \frac{1}{4} \boldsymbol{\alpha}^T \boldsymbol{\alpha} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}^T \beta_j K_j \boldsymbol{\alpha} \tag{10}$$

$$\text{s.t. } \boldsymbol{\beta} \geq \mathbf{0}, \ \ \|\boldsymbol{\beta}\|_p^p \leq 1.$$

Note that putting an ℓ_p constraint on $\boldsymbol{\beta}$ or penalizing $\mathbf{w}$ by an $\ell_{2,r}$ block norm are equivalent with $p = r/(2-r)$ (Szafranski et al., 2008). When $p = 1$, we have the ℓ_1 MK-FDA discussed in Ye et al. (2008); while $p = \infty$ leads to $r = 2$, and MK-FDA reduces to standard single kernel FDA with unweighted concatenation of base feature spaces. In this paper, however, we are interested in the general case of any $p \geq 1$.

Equation (10) is an optimisation problem with a quadratic objective and a general p^{th} order constraint. We borrow the idea from ℓ_p MK-SVM (Kloft et al., 2009, 2011) and use second order Taylor expansion to approximate the norm constraint:

$$\|\boldsymbol{\beta}\|_p^p \ \approx \ \frac{p(p-1)}{2} \sum_{j=1}^{n} \tilde{\beta}_j^{p-2} \beta_j^2 - \sum_{j=1}^{n} p(p-2) \tilde{\beta}_j^{p-1} \beta_j + \frac{p(p-3)}{2} + 1 \ := \ \nu(\boldsymbol{\beta}), \tag{11}$$

where $\tilde{\beta}_j$ is the current estimate of β_j in an iterative process, which will be explained in more detail shortly. Substituting Equation (11) into Equation (10), we arrive at the binary ℓ_p MK-FDA saddle point problem:

$$\max_{\boldsymbol{\beta}} \min_{\boldsymbol{\alpha}} -\mathbf{a}^T\boldsymbol{\alpha} + \frac{1}{4}\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^T\beta_j K_j\boldsymbol{\alpha} \qquad (12)$$
$$\text{s.t. } \boldsymbol{\beta} \geq \mathbf{0},\ \ \nu(\boldsymbol{\beta}) \leq 1.$$

In Sonnenburg et al. (2006), the authors propose to transform a saddle point problem similar to Equation (12) to a semi-infinite program (SIP). A SIP is an optimisation problem with a finite number of variables $\mathbf{x} \in \mathbb{R}^{\mathbf{d}}$ on a feasible set described by infinitely many constraints (Hettich and Kortanek, 1993):

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{s.t. } g(\mathbf{x},u) \geq 0 \ \ \forall u \in \mathcal{U},$$

where $\mathcal{U}$ is an infinite index set. Following the similar arguments as in Sonnenburg et al. (2006) and Ye et al. (2008), we show in Theorem 1 that the saddle point problem in Equation (12) can also be transformed into a SIP.

Theorem 1 *Given a set of n kernel matrices $K_1,\cdots,K_n$, the kernel weights $\boldsymbol{\beta}$ that optimise Equation* (12) *are given by solving the following SIP problem:*

$$\max_{\theta,\boldsymbol{\beta}} \ \theta \qquad (13)$$
$$s.t. \quad -\mathbf{a}^T\boldsymbol{\alpha} + \frac{1}{4}\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^T\beta_j K_j\boldsymbol{\alpha} \geq \theta \ \ \forall\boldsymbol{\alpha} \in \mathbb{R}^m, \quad \boldsymbol{\beta} \geq \mathbf{0}, \quad \nu(\boldsymbol{\beta}) \leq 1.$$

Proof Let $\boldsymbol{\alpha}^*$ be the optimal solution to the saddle point problem in Equation (12). By defining

$$\theta^* := -\mathbf{a}^T\boldsymbol{\alpha}^* + \frac{1}{4}\boldsymbol{\alpha}^{*T}\boldsymbol{\alpha}^* + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^{*T}\beta_j K_j\boldsymbol{\alpha}^*$$

as the minimum objective value achieved by $\boldsymbol{\alpha}^*$, we have

$$-\mathbf{a}^T\boldsymbol{\alpha} + \frac{1}{4}\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^T\beta_j K_j\boldsymbol{\alpha} \geq \theta^*$$

$\forall\boldsymbol{\alpha} \in \mathbb{R}^m$. Now define

$$\theta = \min_{\boldsymbol{\alpha}} -\mathbf{a}^T\boldsymbol{\alpha} + \frac{1}{4}\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^T\beta_j K_j\boldsymbol{\alpha}$$

and substitute it into Equation (12), the theorem is proved. ■

We adapt the wrapper algorithm in Sonnenburg et al. (2006) to solve the SIP in Equation (13). This algorithm is based on the column generation technique, where the basic idea is to divide a SIP into an inner subproblem and an outer subproblem. The algorithm alternates between solving the

Algorithm 1 A wrapper algorithm for solving the binary ℓ_p MK-FDA SIP in Equation (13)

Input: $K_1, \cdots, K_n$, $\mathbf{a}$, $\theta^{(1)} = -\infty$, $\beta_j^{(1)} = n^{-1/p} \forall j$, ε.

Output: Learnt kernel weights $\boldsymbol{\beta} = (\beta_1^{(t)}, \cdots, \beta_n^{(t)})^T$.

1: **for** $t = 1, \cdots$ **do**
2: Compute $\boldsymbol{\alpha}^{(t)}$ in Equation (15);
3: Compute $S^{(t)} = -\mathbf{a}^T \boldsymbol{\alpha}^{(t)} + \frac{1}{4} \boldsymbol{\alpha}^{(t)T} \boldsymbol{\alpha}^{(t)} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}^{(t)T} \beta_j^{(t)} K_j \boldsymbol{\alpha}^{(t)}$;
4: **if** $|1 - \frac{S^{(t)}}{\theta^{(t)}}| \leq \varepsilon$ **then**
5: break;
6: **end if**
7: Compute $\{\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}\}$ in Equation (16), where $\nu(\boldsymbol{\beta})$ is defined as in Equation (11) with $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(t)}$;
8: **end for**

two subproblems until convergence. At step t, the inner subproblem ($\boldsymbol{\alpha}$ step) identifies the constraint that maximises the constraint violation for $\{\theta^{(t)}, \boldsymbol{\beta}^{(t)}\}$:

$$\boldsymbol{\alpha}^{(t)} := \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} -\mathbf{a}^T \boldsymbol{\alpha} + \frac{1}{4} \boldsymbol{\alpha}^T \boldsymbol{\alpha} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}^T \beta_j^{(t)} K_j \boldsymbol{\alpha}. \tag{14}$$

Note that the program in Equation (14) is nothing but the single kernel FDA/RLS dual problem using the current estimate $\boldsymbol{\beta}^{(t)}$ as kernel weights. Observing that Equation (14) is an unconstrained quadratic program, $\boldsymbol{\alpha}^{(t)}$ is obtained by solving the following linear system (Ye et al., 2008):

$$\left(\frac{1}{2} I + \frac{1}{2\lambda} \sum_{j=1}^{n} \beta_j^{(t)} K_j \right) \boldsymbol{\alpha}^{(t)} = \mathbf{a}. \tag{15}$$

If $\boldsymbol{\alpha}^{(t)}$ satisfies constraint $-\mathbf{a}^T \boldsymbol{\alpha}^{(t)} + \frac{1}{4} \boldsymbol{\alpha}^{(t)T} \boldsymbol{\alpha}^{(t)} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}^{(t)T} \beta_j^{(t)} K_j \boldsymbol{\alpha}^{(t)} \geq \theta^{(t)}$ then $\{\theta^{(t)}, \boldsymbol{\beta}^{(t)}\}$ is optimal. Otherwise, the constraint is added to the set of constraints and the algorithm proceeds to the outer subproblem of step t.

The outer subproblem ($\boldsymbol{\beta}$ step) is also called the restricted master problem. At step t, it computes the optimal $\{\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}\}$ in Equation (13) for a restricted subset of constraints:

$$\{\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}\} = \underset{\theta, \boldsymbol{\beta}}{\operatorname{argmax}}\, \theta \tag{16}$$

$$\text{s.t.} \quad -\mathbf{a}^T \boldsymbol{\alpha}^{(r)} + \frac{1}{4} \boldsymbol{\alpha}^{(r)T} \boldsymbol{\alpha}^{(r)} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}^{(r)T} \beta_j K_j \boldsymbol{\alpha}^{(r)} \geq \theta \;\; \forall r = 1, \cdots, t, \;\; \boldsymbol{\beta} \geq \mathbf{0}, \;\; \nu(\boldsymbol{\beta}) \leq 1.$$

When $p = 1$, $\nu(\boldsymbol{\beta}) \leq 1$ reduces to a linear constraint. As a result, Equation (16) becomes a linear program (LP) and the ℓ_p MK-FDA reduces to the ℓ_1 MK-FDA in Ye et al. (2008). When $p > 1$, Equation (16) is a quadratically constrained linear program (QCLP) with one quadratic constraint $\nu(\boldsymbol{\beta}) \leq 1$ and $t + n$ linear constraints. This can be solved by off-the-shelf optimisation tools such as Mosek.[1] Note that at time t, $\nu(\boldsymbol{\beta})$ is defined as in Equation (11) with $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(t)}$, that is, the current estimate of $\boldsymbol{\beta}$.

1. Mosek optimisation toolbox can be found at `http://www.mosek.com`.

Normalised maximal constraint violation is used as a convergence criterion. The algorithm stops when $|1-\frac{S^{(t)}}{\theta^{(t)}}| \leq \varepsilon$, where $S^{(t)} := -\mathbf{a}^T\boldsymbol{\alpha}^{(t)} + \frac{1}{4}\boldsymbol{\alpha}^{(t)T}\boldsymbol{\alpha}^{(t)} + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}^{(t)T}\beta_j^{(t)}K_j\boldsymbol{\alpha}^{(t)}$ and ε is a pre-defined accuracy parameter. This iterative wrapper algorithm for solving the binary ℓ_p MK-FDA SIP is summarised in Algorithm 1. It is a special case of a set of semi-infinite programming algorithms known as exchange methods, which are guaranteed to converge (Hettich and Kortanek, 1993). Finally, note that in line 4 of Algorithm 1, $\boldsymbol{\beta}^{(t+1)}$ can also be solved using the analytical update in Kloft et al. (2011) that is adapted to FDA. However, in practice we notice that for MK-FDA, such an analytical update tends to be numerically unstable when p is close to 1.

2.2 Multiclass Classification

In this section we consider the multiclass case. Let c be the number of classes, and m_k be the number of training examples in the k^{th} class. In multiclass FDA, the following objective is commonly maximised (Ye et al., 2008):

$$J_{MC-FDA}(W) = \operatorname{trace}\left(\left(W^T(S_T+\lambda I)W\right)^{-1}W^T S_B W\right), \tag{17}$$

where W is the projection matrix, the within class scatter S_W is defined in a similar way as in Equation (2) but with c classes, and the between class scatter is $S_B = \phi(X)HH^T\phi(X)^T$, where $\phi(X) = (\phi(\mathbf{x}_1),\phi(\mathbf{x}_2),\cdots,\phi(\mathbf{x}_m))$ is the set of m training examples in the feature space, and $H = (\mathbf{h}_1,\mathbf{h}_2,\cdots,\mathbf{h}_c)$ is an $m\times c$ matrix with $\mathbf{h}_k = (h_{1k},\cdots,h_{mk})^T$ and

$$h_{ik} = \begin{cases} \sqrt{\frac{m}{m_k}} - \sqrt{\frac{m_k}{m}} & \text{if } y_i = k \\ -\sqrt{\frac{m_k}{m}} & \text{if } y_i \neq k. \end{cases} \tag{18}$$

Similar to the binary case, using duality theory and the connection between FDA and RLS, Ye et al. (2008) show that the maximal value of Equation (17) is given by (up to an additive constant determined by the labels):

$$J^*_{MC-FDA} \sim \min_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(-\mathbf{h}_k^T\boldsymbol{\alpha}_k + \frac{1}{4}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k + \frac{1}{4\lambda}\boldsymbol{\alpha}_k^T K\boldsymbol{\alpha}_k\right),$$

where $\boldsymbol{\alpha}_k \in \mathbb{R}^m$ for $k = 1,\cdots,c$. When choosing from linear combinations of a set of base kernels with kernel weights regularised with an ℓ_p norm, the optimal kernel weights are given by:

$$\max_{\boldsymbol{\beta}}\min_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(-\mathbf{h}_k^T\boldsymbol{\alpha}_k + \frac{1}{4}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k\right) \tag{19}$$
$$\text{s.t.}\quad \boldsymbol{\beta}\geq\mathbf{0},\ \ \|\boldsymbol{\beta}\|_p^p \leq 1.$$

We use again second order Taylor expansion to approximate the norm constraint and arrive at the multiclass ℓ_p MK-FDA saddle point problem:

$$\max_{\boldsymbol{\beta}}\min_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(-\mathbf{h}_k^T\boldsymbol{\alpha}_k + \frac{1}{4}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k + \frac{1}{4\lambda}\sum_{j=1}^{n}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k\right)$$
$$\text{s.t.}\quad \boldsymbol{\beta}\geq\mathbf{0},\ \ \nu(\boldsymbol{\beta}) \leq 1,$$

Algorithm 2 A wrapper algorithm for solving the multiclass ℓ_p MK-FDA SIP in Equation (20)

Input: $K_1, \cdots, K_n$, $\mathbf{a}$, $\theta^{(1)} = -\infty$, $\beta_j^{(1)} = n^{-1/p} \forall j$, ε.

Output: Learnt kernel weights $\boldsymbol{\beta} = (\beta_1^{(t)}, \cdots, \beta_n^{(t)})^T$.

1: **for** $t = 1, \cdots$ **do**
2: Compute $\boldsymbol{\alpha}_k^{(t)}$ in Equation (21);
3: Compute $S^{(t)} = \sum_{k=1}^{c} \big(-\mathbf{h}_k^T \boldsymbol{\alpha}_k^{(t)} + \frac{1}{4} \boldsymbol{\alpha}_k^{(t)T} \boldsymbol{\alpha}_k^{(t)} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}_k^{(t)T} \beta_j^{(t)} K_j \boldsymbol{\alpha}_k^{(t)} \big)$;
4: **if** $|1 - \frac{S^{(t)}}{\theta^{(t)}}| \leq \varepsilon$ **then**
5: break;
6: **end if**
7: Compute $\{\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}\}$ in Equation (22), where $\nu(\boldsymbol{\beta})$ is defined as in Equation (11) with $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(t)}$;
8: **end for**

where $\nu(\boldsymbol{\beta})$ is defined as in Equation (11).

Again similar to the binary case, Equation (19) can be reformulated as a SIP:

$$\max_{\theta, \boldsymbol{\beta}} \quad \theta \tag{20}$$
$$\text{s.t.} \quad \sum_{k=1}^{c} \left(-\mathbf{h}_k^T \boldsymbol{\alpha}_k + \frac{1}{4} \boldsymbol{\alpha}_k^T \boldsymbol{\alpha}_k + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}_k^T \beta_j K_j \boldsymbol{\alpha}_k \right) \geq \theta \ \ \forall \boldsymbol{\alpha}_k \in \mathbb{R}^m, \ \ \boldsymbol{\beta} \geq \mathbf{0}, \ \ \nu(\boldsymbol{\beta}) \leq 1,$$

and the SIP can be solved using a column generation algorithm that is similar to Algorithm 1. In the inner subproblem, the only difference is that instead of one linear system, here c linear systems need to be solved, one for each $\mathbf{h}_k$:

$$\left(\frac{1}{2} I + \frac{1}{2\lambda} \sum_{j=1}^{n} \beta_j^{(t)} K_j \right) \boldsymbol{\alpha}_k^{(t)} = \mathbf{h}_k. \tag{21}$$

Accordingly, the outer subproblem for computing the optimal $\{\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}\}$ is adapted to work with multiple classes:

$$(\theta^{(t+1)}, \boldsymbol{\beta}^{(t+1)}) = \operatorname*{argmax}_{\theta, \boldsymbol{\beta}} \theta \tag{22}$$
$$\text{s.t.} \quad \sum_{k=1}^{c} \left(-\mathbf{h}_k^T \boldsymbol{\alpha}_k^{(r)} + \frac{1}{4} \boldsymbol{\alpha}_k^{(r)T} \boldsymbol{\alpha}_k^{(r)} + \frac{1}{4\lambda} \sum_{j=1}^{n} \boldsymbol{\alpha}_k^{(r)T} \beta_j K_j \boldsymbol{\alpha}_k^{(r)} \right) \geq \theta \quad \forall r = 1, \cdots, t$$
$$\boldsymbol{\beta} \geq \mathbf{0}, \quad \nu(\boldsymbol{\beta}) \leq 1.$$

When $p = 1$, Equation (22) reduces to an LP and our formulation reduces to that in Ye et al. (2008). For $p > 1$, Equation (22) is an QCLP with one quadratic constraint and $t + n$ linear constraints, as in the binary case. The iterative wrapper algorithm for solving the multiclass ℓ_p MK-FDA SIP is summarised in Algorithm 2.

2.3 Addressing Efficiency Issues

In this section we discuss several possible improvements over the wrapper-based ℓ_p MK-FDA method proposed in the previous sections. In particular, we address time and memory complexity issues, in light of recent advances in MKL optimisation techniques. We show that by exploiting

the equivalence between kernel FDA and least squares SVM (LSSVM) (Suykens and Vandewalle, 1999), the interleaved method in Sonnenburg et al. (2006) and Kloft et al. (2011) can be applied to MK-FDA. Furthermore, we demonstrate that the formulation in Vishwanathan et al. (2010) that tackles directly the MKL dual problem can also be adapted to work with MK-FDA. Both new formulations discussed in this section are equivalent to previous ones in terms of learnt kernel weights, but can potentially lead to significant efficiency improvement. However, note that we describe these new formulations only briefly, and do not show their efficiency in the experiments section and their implementation details, since these are not in the main scope of this paper. Note also that in the following we focus only on multiclass formulations, as the corresponding binary ones can be derived in a very similar fashion, or as special cases.

2.3.1 INTERLEAVED OPTIMISATION OF THE SADDLE POINT PROBLEM

We consider the multiclass MKL problem for a general convex loss function $V(\xi_{ik}, h_{ik})$:

$$\begin{aligned} &\min_{\mathbf{w}_{jk},\xi_{ik},\boldsymbol{\beta}} \sum_{k=1}^{c} \left(\frac{1}{2} \sum_{j=1}^{n} \frac{||\mathbf{w}_{jk}||^2}{\beta_j} + C \sum_{i=1}^{m} V(\xi_{ik}, h_{ik}) \right) \\ &\text{s.t.} \quad \sum_{j=1}^{n} \mathbf{w}_{jk}^T \phi_j(\mathbf{x}_i) = \xi_{ik}, \ \forall i, \ \forall k; \ \ \boldsymbol{\beta} \geq \mathbf{0}; \ \ ||\boldsymbol{\beta}||_p^2 \leq 1, \end{aligned} \tag{23}$$

where h_{ik} is as defined in Equation (18), and we have replaced the constraint $||\boldsymbol{\beta}||_p^p \leq 1$ equivalently by $||\boldsymbol{\beta}||_p^2 \leq 1$. When $V(\xi_{ik}, h_{ik})$ is the square loss $V(\xi_{ik}, h_{ik}) = \frac{1}{2}(\xi_{ik} - h_{ik})^2$, Equation (23) is essentially multiclass multiple kernel regularised least squares (MK-RLS). It can be shown (see Appendix A for details) that this multiclass MK-RLS can be reformulated as the following saddle point problem:

$$\begin{aligned} &\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\alpha}_k} \sum_{k=1}^{c} \left(\mathbf{h}_k^T \boldsymbol{\alpha}_k - \frac{1}{2C} \boldsymbol{\alpha}_k^T \boldsymbol{\alpha}_k - \frac{1}{2} \sum_{j=1}^{n} \boldsymbol{\alpha}_k^T \beta_j K_j \boldsymbol{\alpha}_k \right) \\ &\text{s.t.} \quad \boldsymbol{\beta} \geq \mathbf{0}; \ \ ||\boldsymbol{\beta}||_p^2 \leq 1. \end{aligned} \tag{24}$$

Making substitutions $\boldsymbol{\alpha}_k \to \frac{C}{2}\boldsymbol{\alpha}_k$ and then $C \to \frac{1}{\lambda}$, it directly follows that the MK-RLS in Equation (24) is equivalent to the MK-FDA in Equation(19). In the previous sections, we proposed to use a conceptually very simple wrapper algorithm to solve it. However, as pointed out in Sonnenburg et al. (2006) and Kloft et al. (2011), such an algorithm has two disadvantages: solving the whole single kernel problem in the $\boldsymbol{\alpha}$ step is unnecessary therefore wasteful, and all kernels need to be loaded into memory. These problems, especially the second one, significantly limit the scalability of wrapper-based MKL algorithms. For example, 50 kernel matrices of size 20000×20000 would usually not fit into memory since they require approximately 149GB of memory (Kloft et al., 2011).

Exploiting the fact that LSSVM, RLS and kernel FDA are equivalent (Rifkin, 2002; Gestel et al., 2002; Keerthi and Shevade, 2003), sequential minimal optimisation (SMO) techniques (Joachims, 1988) developed for LSSVM (Keerthi and Shevade, 2003; Lopez and Suykens, 2011) can be employed to remedy these problems. This effectively leads to an interleaved algorithm that is similar to Algorithm 2 in Kloft et al. (2011), but applies to square loss instead of to hinge loss. Such an interleaved optimisation strategy allows for a very cheap update of a minimal subset of the dual

variables $\boldsymbol{\alpha}_k$ in each $\boldsymbol{\alpha}$ step, without having to have access to the whole kernel matrices, and as a result extends the applicability of MK-FDA to large scale problems. We omit details of the resulting interleaved MK-FDA algorithm, the interested reader is referred to Keerthi and Shevade (2003) and Lopez and Suykens (2011).

2.3.2 WORKING DIRECTLY WITH THE DUAL

The MK-FDA algorithms considered so far, including the wrapper method and the interleaved method, are all based on the intermediate saddle point formulation Equation (24), or equivalently, Equation(19). Recently, a "direct" formulation of MKL was proposed in Vishwanathan et al. (2010), where the idea is to eliminate $\boldsymbol{\beta}$ from the saddle point problem, and deal directly with the dual. Consider again MKL with a general convex loss, but following Vishwanathan et al. (2010) this time we impose the norm constraint in the form of Tikhonov regularisation instead of Ivanov regularisation:

$$\min_{\mathbf{w}_{jk},\xi_{ik},\boldsymbol{\beta}} \sum_{k=1}^{c}\left(\frac{1}{2}\sum_{j=1}^{n}\frac{\|\mathbf{w}_{jk}\|^2}{\beta_j}+C\sum_{i=1}^{m}V(\xi_{ik},h_{ik})\right)+\frac{\mu}{2}\|\boldsymbol{\beta}\|_p^2 \qquad (25)$$
$$\text{s.t.} \quad \sum_{j=1}^{n}\mathbf{w}_{jk}^T\phi_j(\mathbf{x}_i)=\xi_{ik},\ \forall i,\ \forall k;\ \ \boldsymbol{\beta}\geq\mathbf{0}.$$

Note that the two formulations in Equation (25) and Equation (23) are equivalent, in the sense that for any given C there exists a μ (and vice versa) such that the optimal solutions to both problems are identical (Kloft et al., 2011).

It can be shown (see Appendix B for details) that for the special case of square loss, which corresponds to MK-FDA/MK-RLS, the dual of Equation(25) is:

$$\max_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(\mathbf{h}_k^T\boldsymbol{\alpha}_k-\frac{1}{2C}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k\right)-\frac{1}{8\mu}\left\|\left(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^TK_j\boldsymbol{\alpha}_k\right)_{j=1}^{n}\right\|_q^2, \qquad (26)$$

where $q=\frac{p}{p-1}$ is the dual norm of p, and once the optimal $\boldsymbol{\alpha}_k$ are found by solving Equation (26), the kernel weights are given by:

$$\beta_j=\frac{1}{2\mu}\left(\sum_{j=1}^{n}(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^TK_j\boldsymbol{\alpha}_k)^q\right)^{\frac{1}{q}-\frac{1}{p}}(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^TK_j\boldsymbol{\alpha}_k)^{\frac{q}{p}}.$$

Equation (26) can be viewed as an extension of Equation (9) in Vishwanathan et al. (2010) to multiclass problems. Another difference is that Equation (9) in Vishwanathan et al. (2010) considers a hinge loss, while Equation (26) is for square loss. Similarly as in Vishwanathan et al. (2010), for any $p>1$, Equation (26) can be solved using an SMO type of algorithm, with the update rule for the minimal subset of dual variables adapted to work with square loss (Keerthi and Shevade, 2003; Lopez and Suykens, 2011). On the other hand, observing that Equation (26) is an unconstrained optimisation problem and the objective function is differentiable everywhere for $p>1$, an alternative approach is the quasi-Newton descent methods, for example, the limited memory variant (Liu and Nocedal, 1989). In fact, Equation (26) can also be thought of as an extension of the smooth variant of group Lasso considered in Kloft et al. (2011) to multiclass case. Note however that Equation (26) has a term of ℓ_q norm squared, while the smooth group Lasso formulation in Kloft et al. (2011) has a term of ℓ_q norm. This is a direct result of the fact that the two formulations use Tikhonov regularisation and Ivanov regularisation over $\boldsymbol{\beta}$, respectively.

3. Experiments

In this section we validate the usefulness of the proposed ℓ_p MK-FDA with experimental evidence on six datasets. The experiments can be divided into four groups:

- We first demonstrate in Section 3.1 and 3.2 the different behaviour of the sparse ℓ_1 MK-FDA and a non-sparse version of MK-FDA (ℓ_2 norm) on synthetic data and the Pascal VOC2008 object recognition dataset (Everingham et al., 2008). The goal of these two experiments is to confirm that ℓ_1 and ℓ_2 regularisations indeed lead to sparse and non-sparse kernel weights respectively.

- Next in Section 3.3, 3.4 and 3.5 we carry out experiments on another three object and image categorisation benchmarks, namely, Pascal VOC2007 (Everingham et al., 2007), Caltech101 (Fei-Fei et al., 2006), and Oxford Flower17 (Nilsback and Zisserman, 2008). We show that by selecting the regularisation norm p on an independent validation set, the intrinsic sparsity of the given set of base kernels can be learnt. As a result, using the learnt optimal norm p in the proposed ℓ_p MK-FDA offers better performance than ℓ_1 or ℓ_∞ MK-FDAs. Moreover, we compare the performance of ℓ_p MK-FDA and that of several variants of ℓ_p MK-SVM, and show that on image categorisation problems ℓ_p MK-FDA tends to have a small but consistent edge over its SVM counterpart.

- In Section 3.6 we further compare ℓ_p MK-FDA and ℓ_p MK-SVM on the protein subcellular localisation problem studied in Zien and Ong (2007) and Ong and Zien (2008). On this dataset ℓ_p MK-SVM outperforms ℓ_p MK-FDA by a small margin, and the results suggest that given the same set of base kernels, the two MKL algorithms may favour slightly different norms.

- Finally, in Section 3.7, the training speed of our wrapper-based ℓ_p MK-FDA and several ℓ_p MK-SVM implementations is analysed empirically on a few small/medium sized problems, where MK-FDA compares favourably or similarly against state-of-the-art MKL techniques.

Among the six datasets used in the experiments, three of them (synthetic, VOC08, VOC07) are binary problems and the rest (Caltech101, Flower17, Protein) are multiclass ones. In our experiments the wrapper-based ℓ_p MK-FDA is implemented in Matlab with the outer-subproblem solved using the Mosek optimisation toolbox. The code of our ℓ_p MK-FDA implementation is available online.[2] Once the kernel weights have been learnt, we use a spectral regression based efficient kernel FDA implementation (Cai et al., 2007; Tahir et al., 2009) to compute the optimal projection directions, the code of which is also available online.[3] On binary problems, we compare ℓ_p MK-FDA with two implementations of binary ℓ_p MK-SVM, namely, MK-SVM Shogun (Sonnenburg et al., 2006, 2010),[4] and SMO-MKL (Vishwanathan et al., 2010);[5] while on multiclass problems, we compare ℓ_p MK-FDA with two variants of multiclass ℓ_p MK-SVM: MK-SVM Shogun and MK-SVM OBSCURE (Orabona et al., 2010; Orabona and Jie, 2011).[6] In both ℓ_p MK-FDA and ℓ_p MK-SVM

2. The code of our ℓ_p MK-FDA is available at `http://www.featurespace.org`
3. The code of spectral regression FDA can be found at `http://www.zjucadcg.cn/dengcai/SR/index.html`.
4. Version 0.10.0 of the Shogun toolbox, the latest version as of the writing of this paper, can be found at `http://www.shogun-toolbox.org`.
5. The code of SMO-MKL is available at `http://research.microsoft.com/en-us/um/people/manik/code/SMO-MKL/download.html`.
6. The code of OBSCURE can be found at `http://dogma.sourceforge.net`.

Shogun, the stopping threshold ε is set to 10^{-4} unless stated otherwise. Parameters in MK-SVM OBSCURE and SMO-MKL are set to default values unless stated otherwise.

All kernels used in the experiments have been normalised. For the first five datasets, due to the kernel functions used, the kernel matrices are by definition spherically normalised: all data points lie on the unit hypersphere in the feature space. For the protein localisation dataset, the kernels are multiplicatively normalised following Ong and Zien (2008) and Kloft et al. (2011) to allow comparison with Kloft et al. (2011). After normalisation, the kernels are then centred in the feature spaces, as required by ℓ_p MK-FDA. Note that ℓ_p MK-SVM is not affected by centring. Kernels used in the experiments (except for those in the simulation and in training speed experiments) are also available online.[7]

3.1 Simulation

We first perform simulation to illustrate the different behaviour of ℓ_1 MK-FDA and a special case of ℓ_p MK-FDA, namely, the case of $p = 2$. We simulate two classes by sampling 100 points from two 2-dimensional Gaussian distributions, 50 points from each. The means of the two distributions in both dimensions are drawn from a uniform distribution between 1 and 2, and the covariances of the two distributions are also randomly generated. A radial basis function (RBF) kernel is then constructed using these 2-dimensional points. Similarly, 100 test points are sampled from the same distributions, 50 from each, and an RBF kernel is built for the test points. Kernel FDA is then applied to find the best projection direction in the feature space and compute the error rate on the test set. Figure 1 (a) gives 3 examples of the simulated points. It shows that due to the parameters used in the two Gaussian distributions, the two classes are heavily, but not completely, overlapping. As a result, the error rate given by single kernel FDA is around 0.43: slightly better than a random guess.

The above process of mean/covariance generation, sampling, and kernel building is repeated n times, resulting in n training kernels (and n corresponding test kernels). These n training kernels, although generated independently, can be thought of as kernels that capture different "views" of a single binary classification problem. With this interpretation in mind, we apply ℓ_1 and ℓ_2 MK-FDAs to learn optimal kernel weights for this classification problem. We vary the number n from 5 to 50 at a step size of 5. For each value of n, ℓ_1 and ℓ_2 MK-FDAs are applied and the resulting error rates are recorded. This process is repeated 100 time for each value of n to compute the mean and standard deviation of error rates. The results for various n values are plotted in Figure 1 (c).

It is clear in Figure 1 (c) that as the number of kernels increases, the error rates of both methods drop. This is expected, since more kernels bring more discriminative information. Another observation is that ℓ_1 MK-FDA slightly outperforms ℓ_2 MK-FDA when the number of kernels is 5, and vice versa when the number of kernels is 10 or 15. When there are 20 kernels, the advantage of ℓ_2 MK-FDA becomes clear. As the number of kernels keeps increasing, its advantage becomes more and more evident.

The different behaviour of ℓ_1 and ℓ_2 MK-FDAs can be explained by the different weights learnt from them. Two typical examples of such weights, learnt using $n = 5$ kernels and $n = 30$ kernels respectively, are plotted in Figure 1 (b). It has been known that ℓ_1 norm regularisation tends to produce sparse solutions (Rätsch, 2001; Kloft et al., 2008). When kernels carry complementary information, this will lead to a loss of information and hence degraded performance. When the

7. The kernels can be downloaded at `http://www.featurespace.org`.

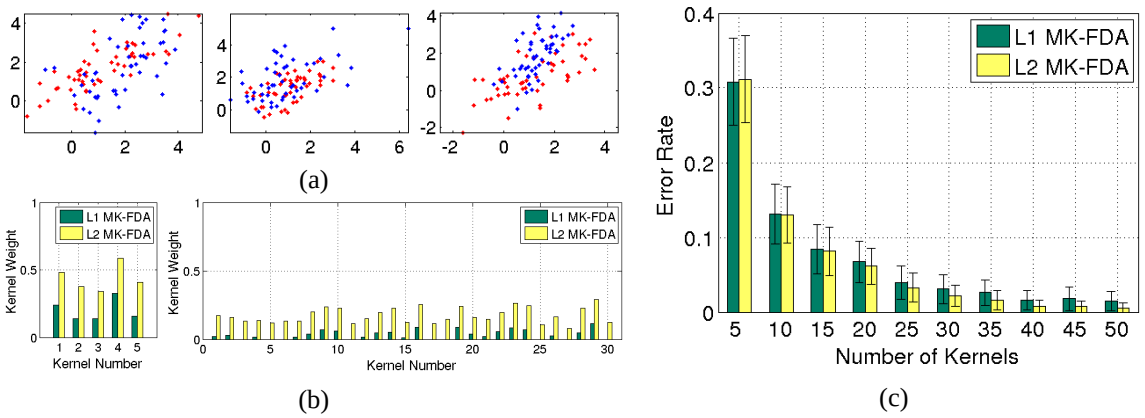


Figure 1: Simulation: (a) Three examples of the two Gaussian distributions. (b) Comparing the kernel weights learnt from ℓ_1 MK-FDA and ℓ_2 MK-FDA. Left: using 5 kernels. Right: using 30 kernels. (c) Mean and standard deviation of error rates of ℓ_1 MK-FDA and ℓ_2 MK-FDA using various number of kernels.

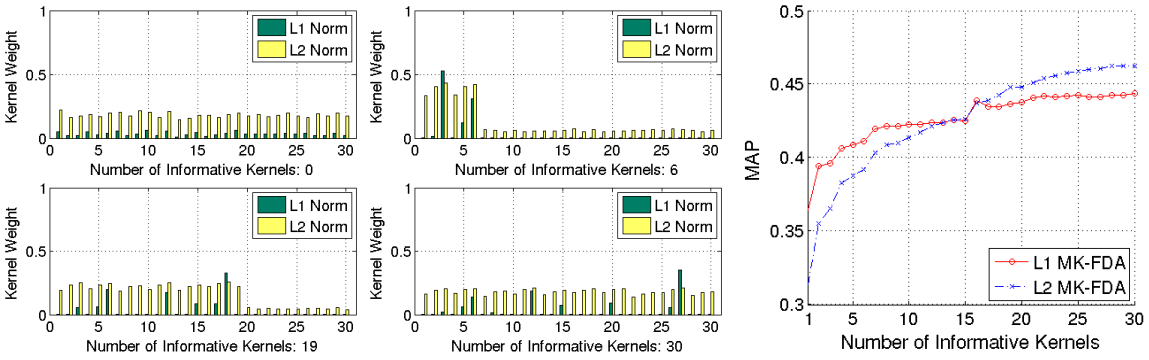
number of kernels is sufficiently small, however, this effect does not occur: as can be seen in the left plot of Figure 1 (b), when there are only 5 kernels, all of them get non-zero weights in both ℓ_1 and ℓ_2 MK-FDAs.

As the number of kernels increases, eventually there are enough of them for the over-selectiveness of ℓ_1 regularisation to exhibit itself. As the right plot of Figure 1 (b) shows, when 30 kernels are used, many of them are assigned zero weights by ℓ_1 MK-FDA. This leads to a loss of information. By contrast, the weights learnt in ℓ_2 MK-FDA are non-sparse, hence the better performance. Finally, it is worth noting that the sparsity of learnt kernel weights, which captures the sparsity of information in the kernel set, is not to be confused with the numerical sparsity of the kernel matrices. For example, when the RBF kernel function is used, the kernel matrices will not contain any zero, regardless of the sparsity of kernel weights.

3.2 Pascal VOC2008

In this section, we demonstrate again the different behaviour of ℓ_1 and ℓ_2 MK-FDAs, but this time on a real world dataset: the Pascal visual object classes (VOC) challenge 2008 development dataset. The VOC challenge provides a yearly benchmark for comparison of object classification methods, with one of the most challenging datasets in the object recognition / image classification community. The VOC2008 development dataset consists of 4332 images of 20 object classes such as aeroplane, cat, person, etc. The dataset is divided into a pre-defined training set with 2111 images and a validation set with 2221 images. In our experiments, the training set is used for training and the validation set for testing. VOC2008 test set is not used as the class labels are not publicly available.

Pascal VOC2008 is a multilabel dataset in the sense that each image can contain multiple classes of objects. To tackle this multilabel problem, the classification of the 20 object classes is treated as 20 independent binary problems. In our experiments, average precision (AP) (Snoek et al., 2006) is used to measure the performance of each binary classifier. Average precision is particularly suitable for evaluating the performance of a retrieval system, since it emphasises higher ranked relevant



(a) (b)

Figure 2: VOC2008: (a) Learnt kernel weights in ℓ_1 MK-FDA and ℓ_2 MK-FDA. "motorbike" class. (b) MAPs of ℓ_1 MK-FDA and ℓ_2 MK-FDA with various composition of kernel set.

instances. The mean of the APs of the 20 classes in the dataset, MAP, is used as a measure of the overall performance.

The SIFT descriptor (Lowe, 2004; Mikolajczyk and Schmid, 2005) and spatial pyramid match kernel (SPMK) (Grauman and Darrell, 2007; Lazebnik et al., 2006) based on bag-of-words model (Zhang et al., 2007; Gemert et al., 2008) are used to build base kernels. The combination of two sampling strategies (dense sampling and Harris-Laplace interest point sampling), 5 colour variants of SIFT descriptors (Sande et al., 2008), and 3 ways of dividing an image into spatial location grids results in $2 \times 5 \times 3 = 30$ "informative" kernels. We also generate 30 sets of random vectors, and build 30 RBF kernels from them. These random kernels are then mixed with the informative ones, to study how the properties of kernels affect the performance of ℓ_1 and ℓ_2 MK-FDAs.

The number of kernels used in each run is fixed to 30. In the first run, only the 30 random kernels are used. In the following runs the number of informative kernels is increased and that of random kernels decreased, until the 31^{st} run, where all 30 kernels are informative. In each run, we apply both ℓ_1 and ℓ_2 MK-FDAs to the 20 binary problems, compute the MAP for each algorithm, and record the learnt kernel weights.

Figure 2 (a) plots the kernel weights learnt from ℓ_1 MK-FDA and ℓ_2 MK-FDA. In each subplot, the weights of the informative kernels are plotted towards the left end and those of random ones towards the right. We clearly observe again the "over-selective" behaviour of ℓ_1 norm: it sets the weights of most kernels, including informative kernels, to zero. By contrast, the proposed ℓ_2 MK-FDA always assigns non-zero weights to the informative kernels. However, ℓ_2 MK-FDA is "under-selective", in the sense that it assigns non-zero weights to the random kernels. It is also worth noting that the kernels that do get selected by ℓ_1 MK-FDA are usually the ones that get highest weights in ℓ_2 MK-FDA.

The MAPs of both ℓ_1 and ℓ_2 MK-FDAs are shown in Figure 2 (b). In order to improve the clarity of the interest region, in Figure 2 (b), the MAP of the first run, that is, when all kernels are random, is not plotted. In such a situation, both versions of MK-FDAs reduce to a chance classifier, which has an MAP of around 0.007. It can be seen from Figure 2 (b) that, as expected, ℓ_1 MK-FDA outperforms ℓ_2 MK-FDA when the noise level is high and vice versa when the noise level is low. Another interpretation of this observation is that when the "intrinsic" sparsity of the base kernels is

high then ℓ_1 norm regularisation is appropriate, and vice versa. This suggests that if we can learn this intrinsic sparsity of base kernels on a validation set, we will be able to find the most appropriate regularisation norm p, and get improved performance over a fix norm MK-FDA. We validate this idea in the next section.

3.3 Pascal VOC2007

Similar to Pascal VOC2008, Pascal VOC2007 is a multilabel object recognition dataset consisting of the same 20 object categories. The dataset is divided into training, validation and test sets, with 2501, 2510 and 4952 images respectively. As in the case of VOC2008, the classification of the 20 object classes is treated as 20 independent binary problems, and MAP is used as a measure of overall performance.

We generate 14 base kernels by combining 7 colour variants of local descriptors (Sande et al., 2008) and two kernel functions, namely, SPMK (Lazebnik et al., 2006; Grauman and Darrell, 2007) and RBF kernel with χ^2 distance (Zhang et al., 2007). We first perform supervised dimensionality reduction on the descriptors to improve their discriminability, following Cai et al. (2011). The descriptors with reduced dimensionality are clustered with k-means to learn codewords (Csurka et al., 2004). The soft assignment scheme in Gemert et al. (2008) is then employed to generate a histogram for each image as its representation. Finally, the two kernel functions are applied to the histograms to build kernel matrices.

We investigate the idea of learning the intrinsic sparsity of the base kernels by tuning the regularisation norm p on a validation set, using both ℓ_p MK-SVM and ℓ_p MK-FDA. For both methods, we learn the parameter p on the validation set from 12 values: $\{1, 1+2^{-6}, 1+2^{-5}, 1+2^{-4}, 1+2^{-3}, 1+2^{-2}, 1+2^{-1}, 2, 3, 4, 8, 10^6\}$. For ℓ_p MK-SVM, the regularisation parameter C is learnt jointly with p from 10 values that are logarithmically spaced over 2^{-2} to 2^7. Similarly, for ℓ_p MK-FDA, the regularisation parameter λ is learnt jointly with p from a set of 10 values that are logarithmically spaced over 4^{-5} to 4^4. The sets of values of C and λ are chosen to cover the areas in the parameter spaces that give the best performance for MK-SVM and MK-FDA, respectively.

Plotted in Figure 3 are the weights learnt on the training set in ℓ_p MK-FDA and ℓ_p MK-SVM with various p values for the "aeroplane" class. For ℓ_p MK-FDA, for each p value, the weights learnt with the optimal λ value are plotted; while for ℓ_p MK-SVM, for each p value, we show the weights learnt with the optimal C value. It is clear that as p increases, in both MKL algorithms, the sparsity of the learnt weights decreases. As expected, when $p = 10^6$ (practically infinity), the kernel weights become ones, that is, ℓ_∞ MK-FDA/MK-SVM produces uniform kernel weights. Note that for the same norm p, the weights learnt in ℓ_p MK-FDA and ℓ_p MK-SVM can be different. This is especially evident when p is small. Note also that results reported in this section are obtained using the Shogun implementation of MK-SVM, which is based on the saddle point formulation of the problem. The recently proposed SMO-MKL works directly with the dual and can be more efficient, especially on large scale problems. However, as discussed in Section 2.3, these two formulations are equivalent and produce identical kernel weights. Considering this, we only present the results of SMO-MKL in terms of training speed in Section 3.7.

Next, we plot in Figure 4 top-left the APs on the validation and test sets for the "bird" class with various p values, using ℓ_p MK-FDA, where again for each p value, the APs with the λ value that gives the best AP on the validation set are plotted. It is clear that the two curves match well, which implies that learning p in addition to λ should help. Shown in the middle and right columns

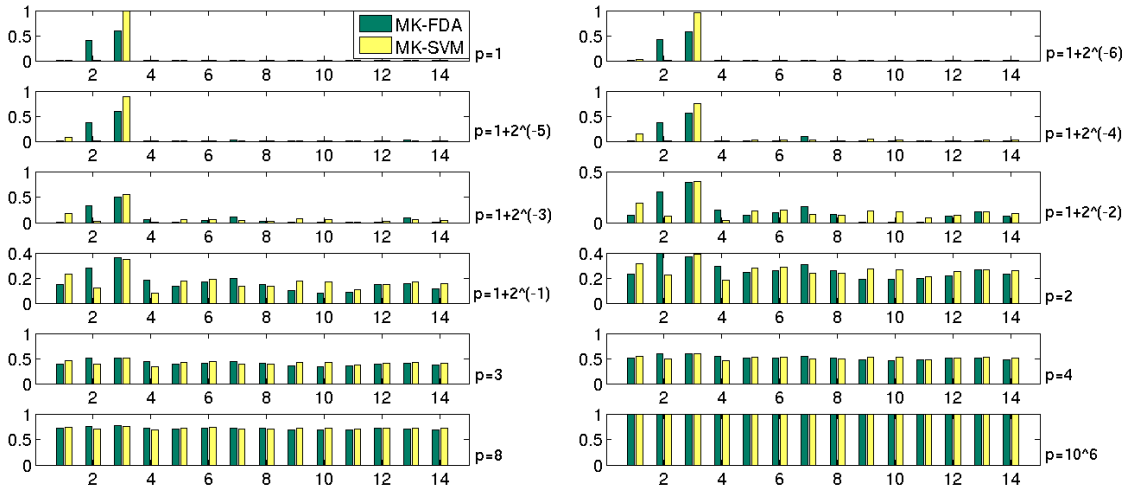


Figure 3: VOC2007: Kernel weights learnt on the training set in ℓ_p MK-FDA and ℓ_p MK-SVM with various p values. "aeroplane" class.

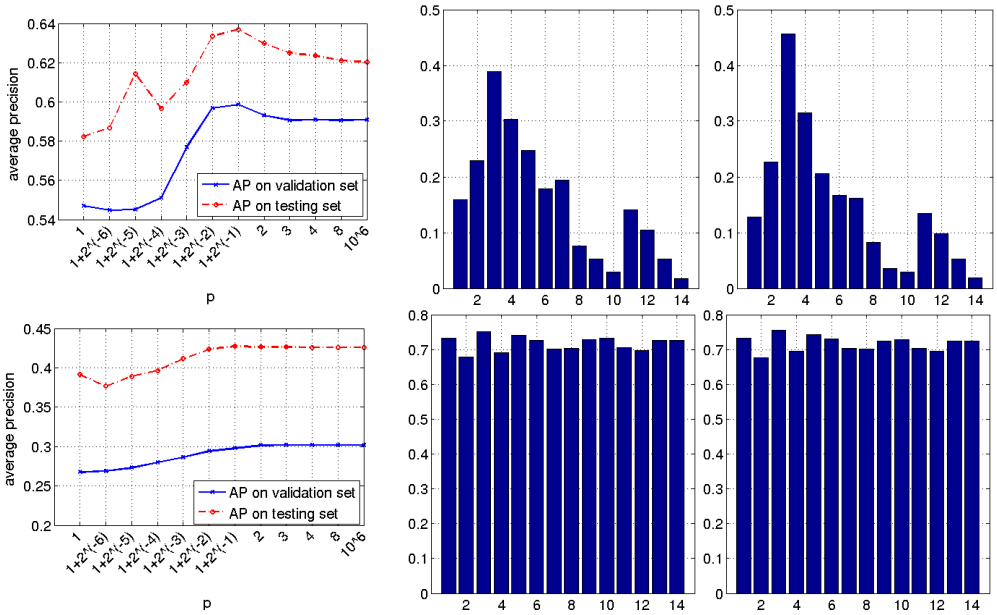


Figure 4: VOC2007: Learning the norm p for MK-FDA on the validation set. Top row: "bird" class. Bottom row: "pottedplant" class; left column: APs on the validation set and test set with various p values; middle column: kernel weights learnt on the training set with the optimal $\{p,\lambda\}$ combination; right column: kernel weights learnt on the training+validation set with the same $\{p,\lambda\}$ combination.

	MK-SVM			MK-FDA		
	ℓ_1	ℓ_∞	ℓ_p	ℓ_1	ℓ_∞	ℓ_p
aeroplane	78.8	79.6	79.6	79.9	79.5	**80.9**
bicycle	63.4	65.0	64.7	64.7	67.6	**67.8**
bird	57.3	61.0	61.0	57.1	62.0	**63.7**
boat	**71.1**	70.1	**71.1**	70.9	70.1	70.8
bottle	29.1	**29.9**	29.7	27.5	29.7	29.4
bus	62.9	64.9	**65.5**	63.4	66.1	66.1
car	77.9	78.8	78.8	79.1	79.5	**80.9**
cat	56.7	56.4	57.1	57.1	56.9	**58.3**
chair	52.3	**53.0**	**53.0**	51.9	52.5	52.9
cow	38.7	41.4	41.4	42.3	41.5	**43.4**
	table continued in the right column.					

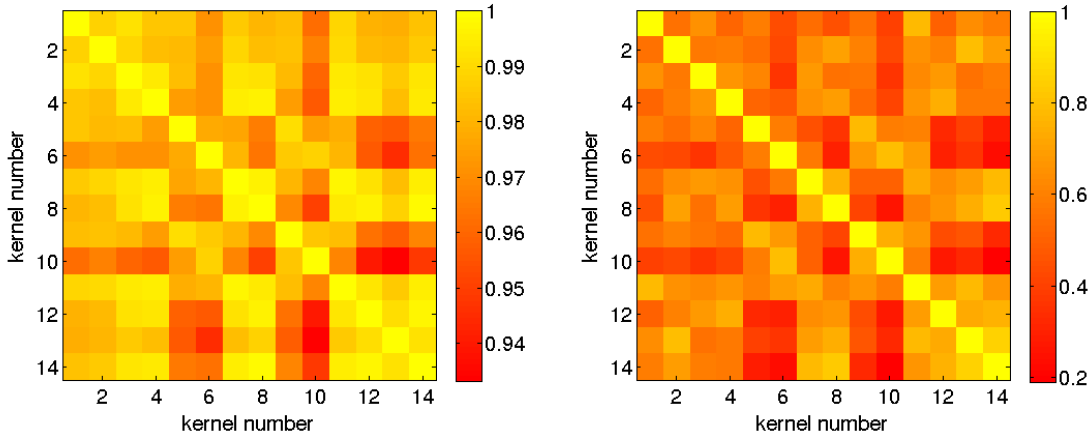
	MK-SVM			MK-FDA		
	ℓ_1	ℓ_∞	ℓ_p	ℓ_1	ℓ_∞	ℓ_p
din. table	52.4	57.3	56.6	57.2	59.2	**61.4**
dog	42.8	45.8	44.6	44.2	**46.1**	45.1
horse	78.9	80.6	80.6	80.0	**81.1**	81.0
moterbike	66.3	66.8	66.8	67.8	67.8	**68.8**
person	86.7	88.0	88.0	86.8	88.1	**88.8**
pot. plant	31.8	41.0	40.5	32.5	**42.6**	42.5
sheep	40.2	**46.0**	**46.0**	39.0	44.4	43.9
sofa	44.0	43.8	44.0	43.5	43.7	**45.9**
train	81.3	82.4	82.4	83.2	84.2	**85.1**
tvmonitor	53.3	53.7	53.7	52.5	54.1	**56.9**
MAP	58.3	60.3	60.3	59.0	60.8	**61.7**

Table 1: VOC2007: Average precisions of six MKL methods

of the top row of Figure 4 are the learnt kernel weights with the optimal $\{p, \lambda\}$ combination on the training set and on the training + validation set, respectively. Since for the "bird" class the optimal p found on the validation set is $1 + 2^{-1}$, both sets of weights are non-sparse. For this particular binary problem, the intrinsic sparsity of the set of base kernels is medium. Similarly, the bottom row of Figure 4 shows the results for the "pottedplant" class. We again observe that the AP on the validation set and that on the test set show similar patterns. However, for the "pottedplant" class, the optimal p on the validation set is found to be 8, which implies that the intrinsic sparsity of the kernels is low.

When keeping the norm p fixed at 1, 10^6 and learning only the C/λ parameter, the ℓ_p MK-SVM/MK-FDA reduces to ℓ_1 and ℓ_∞ MK-SVM/MK-FDA, respectively. The APs and MAPs of the six MKL methods are shown in Table 1. The results in Table 1 demonstrate that learning the regularisation norm p indeed improves the performance of MK-FDA. However, it is worth noting that this improvement is achieved at a computational price of cross validating for an additional parameter, the regularisation norm p. In the case of MK-SVM, the learnt optimal p yields the same MAP as ℓ_∞ MK-SVM. However, this does not mean learning p is not bringing anything, because a priori we would not know that ℓ_∞ is the most appropriate norm. Instead, the conclusion we can draw from the MK-SVM results is that the sparsity of the base kernels, according to MK-SVM, is very low. Another observation from Table 1 is that in all three cases: ℓ_1, ℓ_∞ and ℓ_p tuned, MK-FDA outperforms MK-SVM on the majority of classes.

The pairwise alignment of the 14 kernel matrices w.r.t. the Frobenius dot product (Golub and van Loan, 1996), $\mathcal{A}(i,j) = \frac{<K_i, K_j>_F}{\|K_i\|_F \|K_j\|_F}$, is plotted in Figure 5, where subplot (a) shows the alignment of uncentred kernels and subplot (b) shows that of centred kernels. Kernel alignment has been used to analyse the property of a given kernel set (Nakajima et al., 2009; Kloft et al., 2011). We argue, however, that kernel alignment by itself cannot reveal completely the sparsity of a kernel set. First of all, as shown in Figure 5 (a) and (b), centring the kernel matrices changes significantly the alignment of the kernels. On the other hand, it is well known that centring does not change the effective weights learnt in MKL, since the shape of the data in the feature space is translation invariant. Second, kernel alignment does not take into account label information. For a multilabel dataset such as VOC07, all object classes share the same set of images (hence the same kernels),



(a) (b)

Figure 5: VOC07: Alignment of the 14 kernels. (a) Spherically normalised kernels. (b) Spherically normalised and centred kernels. Note the scale difference between the two plots as indicated by the colorbars.

and the labels are different depending on which object class (i.e., which binary problem) is being considered. It is clear from Table 1 that for both ℓ_p MK-FDA and ℓ_p MK-SVM, the sparsity of the kernel set is class dependent. This means kernel alignment, which is class independent, by itself cannot be expected to identify the kernel set sparsity for all classes. Instead, we hypothesise that correlation analysis using projected labels (Braun et al., 2008) is probably more appropriate.

Finally, note that due to different parameter sets and different normalisation methods used (spherical normalisation in this paper while unit trace normalisation in Yan et al., 2010), the results on VOC07, Caltech101 and Flower17 reported in this paper are slightly different from those in Yan et al. (2010). However, the trends in the results remain the same, and all conclusions drawn from the results remain unchanged.

3.4 Caltech101

In the following three sections, we compare the proposed ℓ_p MK-FDA with several variants of ℓ_p MK-SVM on multiclass problems. We start in this section with the Caltech101 object recognition dataset. Caltech101 is a multiclass object recognition benchmark with 101 object categories. We follow the popular practice of using 15 randomly selected images per class for training, up to 50 randomly selected images per class for testing, and compute the average accuracy over all classes. This process is repeated 3 times, and we report the mean of the average accuracies on the test set that is achieved with the optimal parameter (C for MK-SVM and λ for MK-FDA). Validation is omitted, as the training of multiclass MK-SVM Shogun on this dataset can be very time consuming.

We generate 10 kernels in a similar way as in the VOC2007 experiments. In addition to these "informative" kernels, we also construct 10 RBF kernels from 10 sets of random vectors. To test the robustness of the MKL methods, we repeat the experiment 6 times. We start with only the informative kernels, and add two more random kernels in each subsequent run.

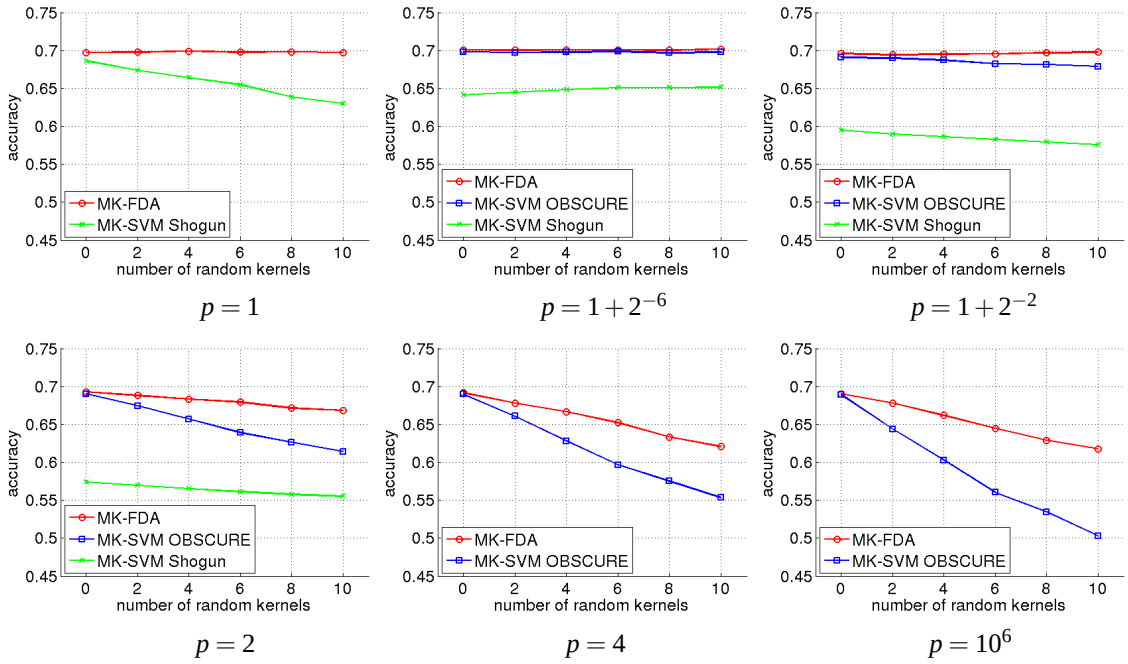


Figure 6: Caltech101: Accuracy comparison of three multiclass MKL methods.

Two multiclass MK-SVM implementations are compared against multiclass MK-FDA, namely, MK-SVM Shogun, and the recently proposed online MK-SVM algorithm OBSCURE (Orabona et al., 2010). For OBSCURE, the parameters are set to default values, except for the MKL norm p and the regularisation parameter C. In our experiments, C and λ are chosen from the same set of values that are logarithmically spaced over 4^{-5} to 4^4. We use the same set of 12 p values as in the VOC07 experiments. Note however that in OBSCURE, the MKL norm p is specified equivalently through the block norm r, where $r = 2p/(p+1)$. Moreover, OBSCURE requires that $r > 1$, so $p = r = 1$ in the set of p values is not used for OBSCURE.

The performance of the three MKL methods with various numbers of random kernels is illustrated in Figure 6, where we show results for six p values, covering the spectrum from highly sparsity-inducing norm, to uniform weighting. When p is large, MK-SVM Shogun does not converge within 24 hours, so its performance is not plotted for $p = 4$ and $p = 10^6$. We can see from Figure 6 that, when p is small, both MK-SVM OBSCURE and MK-FDA are robust to the added noise, and MK-FDA has a marginal advantage over OBSCURE (e.g., $\sim$0.003 when $p = 1+2^{-6}$). When p is large, as expected, the performance of all three methods in general degrades as the number of random kernels increases. However, MK-FDA does so more gracefully than OBSCURE. On the other hand, both MK-FDA and MK-SVM OBSCURE outperform MK-SVM Shogun by a large margin on this multiclass problem.

3.5 Oxford Flower17

Oxford Flower17 is a multiclass dataset consisting of 17 categories of flowers with 80 images per category. This dataset comes with 3 predefined splits into training (17×40 images), validation (17×20 images) and test (17×20 images) sets. Moreover, Nilsback and Zisserman (2008) precomputed

method	accuracy and std. dev.	parameters tuned on val. set
product	85.5 ± 1.2	C
averaging	84.9 ± 1.9	C
MKL (SILP)	85.2 ± 1.5	C
MKL (Simple)	85.2 ± 1.5	C
CG-Boost	84.8 ± 2.2	C
LP-β	85.5 ± 3.0	$C_j, j = 1, \cdots, n$ and $\delta \in (0,1)$
LP-B	85.4 ± 2.4	$C_j, j = 1, \cdots, n$ and $\delta \in (0,1)$
ℓ_p MK-SVM Shogun	86.0 ± 2.4	p and C jointly
ℓ_p MK-SVM OBSCURE	85.6 ± 0.0	p and C jointly
ℓ_p MK-FDA	$\mathbf{87.2 \pm 1.6}$	p and λ jointly

Table 2: Flower17: Comparison of ten kernel fusion methods.

7 distance matrices using various features, and the matrices are available online.[8] We use these distance matrices and follow the same procedure as in Gehler and Nowozin (2009) to compute 7 kernels: $K_j(\mathbf{x}_i, \mathbf{x}_{i'}) = \exp(-D_j(\mathbf{x}_i, \mathbf{x}_{i'})/\eta_j)$, where η_j is the mean of the pairwise distances for the j^{th} feature.

Table 2 compares ℓ_p MK-SVM Shogun, ℓ_p MK-SVM OBSCURE, ℓ_p MK-FDA, and 7 kernel combination techniques discussed in Gehler and Nowozin (2009). Note that these methods are directly comparable since they share the same kernel matrices and the same splits. For ℓ_p MK-SVM Shogun, ℓ_p MK-SVM OBSCURE and ℓ_p MK-FDA, the parameters p, C and λ are tuned on the validation set from the same sets of values as in the Caltech101 experiments. For the other seven methods, the corresponding entries in the table are taken directly from Gehler and Nowozin (2009), where: "product" and "sum" refer to the two simplest kernel combination methods, namely, taking the element-wise geometric mean and arithmetic mean of the kernels, respectively; "MKL (SILP)" and "MKL (Simple)" are essentially ℓ_1 MK-SVM; while "CG-Boost", "LP-β" and "LP-B" are three boosting based kernel combination methods.

We can see from Table 2 that the boosting based methods, although performing well on other datasets in Gehler and Nowozin (2009), fail to outperform the baseline methods "product" and "averaging". On the other hand, ℓ_p MK-FDA not only shows a considerable improvement over all the methods discussed in Gehler and Nowozin (2009), but also outperforms both ℓ_p MK-SVM Shogun and ℓ_p MK-SVM OBSCURE. Note that the optimal test accuracy over all combinations of parameters achieved by OBSCURE is comparable to that by MK-FDA. However, the performance on the validation set and that on the test set do not match as well for OBSCURE as for MK-FDA,[9] resulting in the lower test accuracy of OBSCURE. Parameters that need to be tuned on the validation set in these methods are also compared in Table 2.

3.6 Protein Subsellular Localisation

In the previous three sections, we have shown that on both binary and multiclass object recognition problems, ℓ_p MK-FDA tends to outperform ℓ_p MK-SVM by a small margin. In this section, we further compare ℓ_p MK-FDA and ℓ_p MK-SVM on a computational biology problem, namely, the prediction of subcellular localisation of proteins (Zien and Ong, 2007; Ong and Zien, 2008).

8. The distance matrices can be found at `http://www.robots.ox.ac.uk/~vgg/research/flowers/index.html`.
9. This is indicated by, for example, a lower Spearman or Kendall rank correlation coefficient.

norm p		1	32/31	16/15	8/7	4/3	2	4	8	16	∞
plant	MK-SVM	**8.18** $\pm$ 0.47	8.22 $\pm$ 0.45	8.20 $\pm$ 0.43	8.21 $\pm$ 0.42	8.43 $\pm$ 0.42	9.47 $\pm$ 0.43	11.00 $\pm$ 0.47	11.61 $\pm$ 0.49	11.91 $\pm$ 0.55	11.85 $\pm$ 0.60
	MK-FDA	10.86 $\pm$ 0.42	11.02 $\pm$ 0.43	10.96 $\pm$ 0.46	11.07 $\pm$ 0.43	10.85 $\pm$ 0.43	**10.69** $\pm$ 0.37	11.28 $\pm$ 0.45	11.28 $\pm$ 0.45	11.04 $\pm$ 0.43	11.35 $\pm$ 0.46
nonpl	MK-SVM	**8.97** $\pm$ 0.26	9.01 $\pm$ 0.25	9.08 $\pm$ 0.26	9.19 $\pm$ 0.27	9.24 $\pm$ 0.29	9.43 $\pm$ 0.32	9.77 $\pm$ 0.32	10.05 $\pm$ 0.32	10.23 $\pm$ 0.32	10.33 $\pm$ 0.31
	MK-FDA	10.93 $\pm$ 0.31	10.59 $\pm$ 0.33	10.91 $\pm$ 0.31	10.89 $\pm$ 0.32	**10.84** $\pm$ 0.31	11.00 $\pm$ 0.33	12.12 $\pm$ 0.41	12.12 $\pm$ 0.41	11.81 $\pm$ 0.38	12.15 $\pm$ 0.41
psortNeg	MK-SVM	9.99 $\pm$ 0.35	9.91 $\pm$ 0.34	**9.87** $\pm$ 0.34	10.01 $\pm$ 0.34	10.13 $\pm$ 0.33	11.01 $\pm$ 0.32	12.20 $\pm$ 0.32	12.73 $\pm$ 0.34	13.04 $\pm$ 0.33	13.33 $\pm$ 0.35
	MK-FDA	9.89 $\pm$ 0.34	10.07 $\pm$ 0.36	9.95 $\pm$ 0.35	9.87 $\pm$ 0.37	9.75 $\pm$ 0.39	**9.74** $\pm$ 0.37	11.39 $\pm$ 0.35	11.25 $\pm$ 0.34	11.27 $\pm$ 0.35	11.50 $\pm$ 0.35
psortPos	MK-SVM	13.07 $\pm$ 0.66	**13.01** $\pm$ 0.63	13.41 $\pm$ 0.67	13.17 $\pm$ 0.62	13.25 $\pm$ 0.61	14.68 $\pm$ 0.67	15.55 $\pm$ 0.72	16.43 $\pm$ 0.81	17.36 $\pm$ 0.83	17.63 $\pm$ 0.80
	MK-FDA	**12.59** $\pm$ 0.75	13.16 $\pm$ 0.80	13.07 $\pm$ 0.80	13.34 $\pm$ 0.80	13.45 $\pm$ 0.74	13.63 $\pm$ 0.70	16.86 $\pm$ 0.85	16.37 $\pm$ 0.89	16.56 $\pm$ 0.87	16.94 $\pm$ 0.84

Table 3: Protein Subcellular Localisation: comparing ℓ_p MK-FDA and ℓ_p MK-SVM w.r.t. prediction error and its standard error. Prediction error is measured as $1-$ average MCC in percentage.

The protein subcellular localisation problem contains 4 datasets, corresponding to 4 different sets of organisms: plant (*plant*), non-plant eukaryotes (*nonpl*), Gram-positive (*psortPos*) and Gram-negative bacteria (*psortNeg*). Each of the 4 datasets can be considered as a multiclass classification problem, with the number of classes ranging between 3 and 5. For each dataset, 69 kernels that capture diverse aspects of protein sequences are available online.[10] We download the kernel matrices and follow the experimental setup in Kloft et al. (2011) to enable a direct comparison. More specifically, for each dataset, we first multiplicatively normalise the kernel matrices. Then for each of the 30 predefined splits, we use the first 20% of examples for testing and the rest for training.

In Kloft et al. (2011), the multiclass problem associated with each dataset is decomposed into binary problems using the one-vs-rest strategy. This is not necessary in the case of FDA, since FDA by its natures handles both binary and multiclass problems in a principled fashion. For each dataset, we consider the same set of values for the norm p as in Kloft et al. (2011): $\{1, 32/31, 16/15, 8/7, 4/3, 2, 4, 8, \infty\}$. In Kloft et al. (2011), the regularisation constant C for MK-SVM is taken from a set of 9 values: $\{1/32, 1/8, 1/2, 1, 2, 4, 8, 32, 128\}$. In our experiments, the regularisation constant λ for MK-FDA is also taken from a set of 9 values, and the values are logarithmically spaced over 10^{-8} to 10^{0}.

Again following Kloft et al. (2011), for each p/λ combination, we evaluate the performance of ℓ_p MK-FDA w.r.t. average (over the classes) Matthews correlation coefficient (MCC), and report in Table 3 the average of $1-$ M*CC* over 30 splits and its standard error. For ease of comparison, we also show in Table 3 the performance of ℓ_p MK-SVM as reported in Kloft et al. (2011).

10. The kernels can be downloaded at `http://www.fml.tuebingen.mpg.de/raetsch/suppl/protsubloc`.

Table 3 demonstrates that overall the performance of ℓ_p MK-FDA lags behind that of ℓ_p MK-SVM, except on *psortNeg* and on *psortPos*, where it has a small edge. Another observation is that the optimal norm p identified by MK-SVM does not necessarily agree with that by MK-FDA. On *psortPos* they are in close agreement and both methods favour sparsity-inducing norms. On *plant*, *nonpl* and *psortNeg*, on the other hand, the norms picked by MK-FDA are larger than those picked by MK-SVM. Note, however, that this observation can be slightly misleading, because on the latter three datasets, the performance curve of ℓ_p MK-FDA is quite "flat" in the area of optimal performance. As a result, the optimal norm estimated may not be stable.

3.7 Training Speed

In this section we provide an empirical analysis of the efficiency of the wrapper-based ℓ_p MK-FDA and various implementations of ℓ_p MK-SVM. We use $p = 1$ (or in some cases $1 + 2^{-5}$, $1 + 2^{-6}$) and $p = 2$ as examples of sparse/non-sparse MKL respectively,[11] and study the scalability of MK-FDA and MK-SVM w.r.t. the number of examples and the number of kernels, on both binary and multiclass problems.

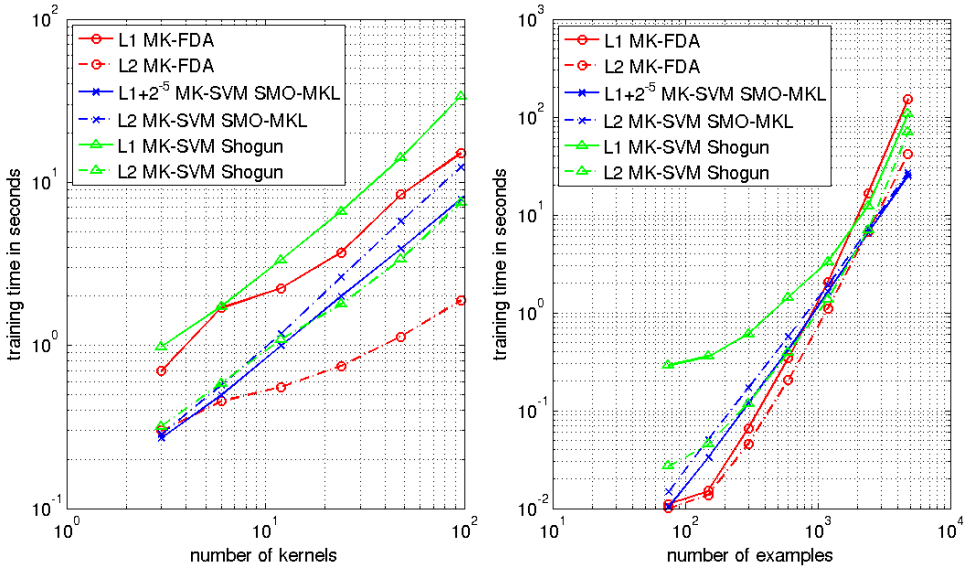
3.7.1 BINARY CASE: VOC2007

We first compare on the VOC2007 dataset the training speed of three binary MKL methods: the wrapper based binary ℓ_p MK-FDA in Section 2.1, the binary ℓ_p MK-SVM Shogun implementation (Sonnenburg et al., 2006, 2010), and the SMO-MKL in Vishwanathan et al. (2010). In the experiments, interleaved optimisation and analytical update of β are used for MK-SVM Shogun.

We first fix the number of training examples to 1000, and vary the number of kernels from 3 to 96. We record the time taken to learn the kernel weights, and average over the 20 binary problems. Figure 7 (a) shows the training time of the six MKL algorithms as functions of the number of kernels. Next, we fix the number of kernels to 14, and vary the number of examples from 75 to 4800. Similarly, in Figure 7 (b) we plot the average training time as functions of the number of examples.

Figure 7 (a) demonstrates that for small/medium sized problems, when a sparsity-inducing norm is used, SMO-MKL is the most efficient; while when $p = 2$, MK-FDA can be significantly faster than the competing methods. On the other hand, when training efficiency is measured as a function of the number of examples, there is no clear winner, as indicated in Figure 7 (b). However, the trends in Figure 7 (b) suggest that on large scale problems, SMO-MKL is likely to be more efficient than MK-FDA and MK-SVM Shogun. In both cases, MK-FDA has a comparable or better efficiency than MK-SVM Shogun, despite the fact that MK-SVM Shogun uses the interleaved algorithm while MK-FDA employs the somewhat wasteful wrapper-based implementation. Again, this trend is likely to flip over on large scale problems. For such problems, one can adopt either the square loss counterpart of the interleaved algorithm, or the square loss counterpart of the SMO-MKL algorithm, or the limited memory quasi-Newton method, to improve the efficiency of ℓ_p MK-FDA, as discussed in Section 2.3.

11. Both SMO-MKL and OBSCURE require that $p > 1$. Moreover, SMO-MKL is numerically unstable when $p = 1 + 2^{-6}$. As a result, we use $p = 1 + 2^{-5}$ and $p = 1 + 2^{-6}$ as sparsity-inducing norms for SMO-MKL and OBSCURE, respectively.



(a) (b)

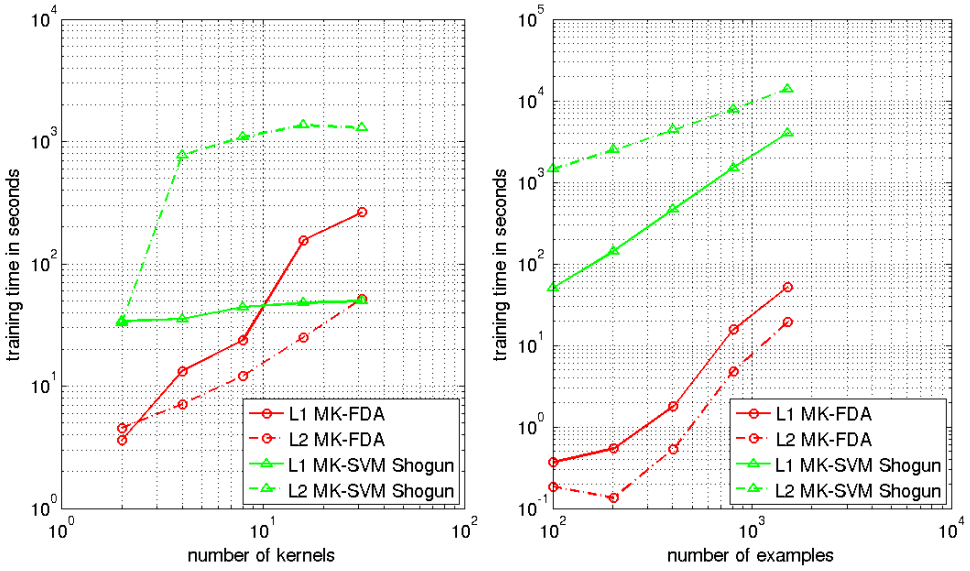
Figure 7: Training speed on a binary problem: VOC2007. (a) Training time vs. number of kernels, where number of examples is fixed at 1000. (b) Training time vs. number of examples, where number of kernels is fixed at 14. $\lambda = 1$ for MK-FDA, and $C = 1$ for MK-SVM Shogun and MK-SVM OBSCURE.

3.7.2 MULTICLASS CASE: CALTECH101

Next we compare three multiclass ℓ_p MKL algorithms on the Caltech101 dataset, namely, the wrapper-based multiclass ℓ_p MK-FDA in Section 2.2, multiclass ℓ_p MK-SVM Shogun, and MK-SVM OBSCURE. We compare the first two methods following similar protocols as in the binary case. In Figure 8 (a) we show the average training time over the 3 splits as functions of the number of kernels (from 2 to 31) when the number of examples is fixed to 101 (one example per class); while plotted in Figure 8 (b) is the average training time as functions of the number of examples (from 101 to 1515, that is, from one example per class to 15 examples per class) when the number of kernels is fixed to 10.

Figure 8 shows that on small/medium sized multiclass problems, MK-FDA is in most cases one or two orders of magnitude faster than MK-SVM Shogun. The only exception is that as the number of kernels increases, the efficiency of ℓ_1 MK-SVM Shogun degrades more gracefully than ℓ_1 MK-FDA, and eventually overtakes. Another observation from both Figure 7 and Figure 8 is that, ℓ_2 MK-FDA tends to be more efficient than ℓ_1 MK-FDA, despite the fact that in the outer subproblem, the LP solver employed in ℓ_1 MK-FDA is slightly faster than the QCLP solver in ℓ_2 MK-FDA. This is because ℓ_1 MK-FDA usually takes a few tens of iterations to converge, while the ℓ_2 version typically takes less than 5. This difference in the number of iterations reverses the efficiency advantage of LP over QCLP.

Due to its online nature, the efficiency of OBSCURE has to be measured differently to allow a fair comparison. The OBSCURE algorithm is a two-stage algorithm, and each stage involves an iterative process with a parameter $T1/T2$ controlling the number of iterations. In general the



(a) (b)

Figure 8: Training speed on a multiclass problem: Caltech101. MK-FDA vs. MK-SVM Shogun. (a) Training time vs. number of kernels, where number of examples is fixed at 101. (b) Training time vs. number of examples, where number of kernels is fixed at 10. $\lambda = 1$ for MK-FDA, and $C = 1$ for MK-SVM Shogun.

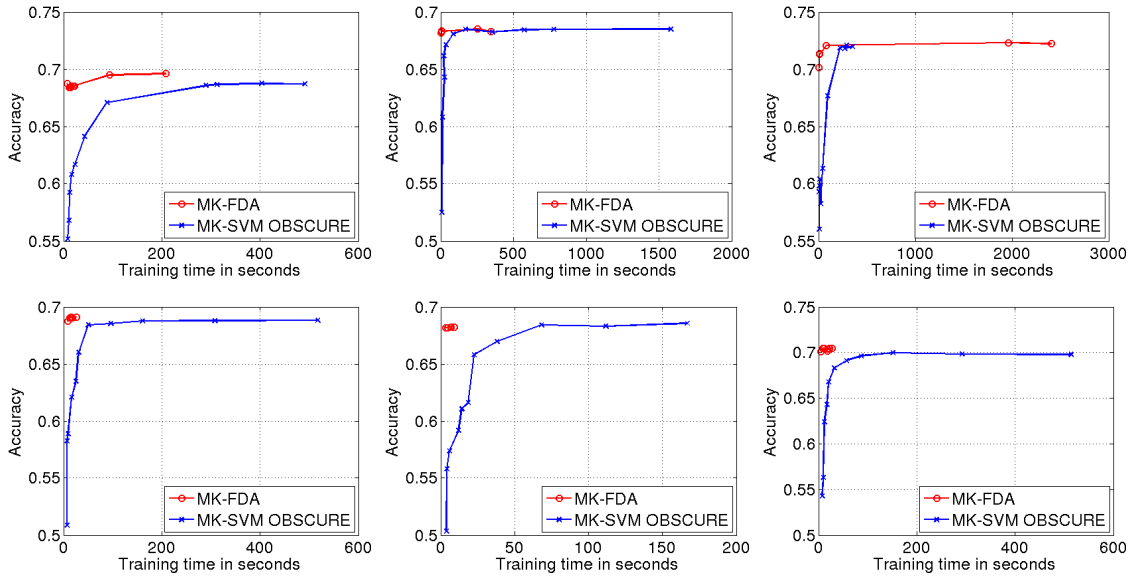


Figure 9: Training speed on a multiclass problem: Caltech101. MK-FDA vs. MK-SVM OBSCURE. Top row: $p = 1 + 2^{-6}$. Bottom row: $p = 2$. The three columns correspond to the three splits. 10 kernels and $101 \times 15 = 1515$ training examples.

larger the values of $T1$ and $T2$, the longer it takes to train, but the more accurate the learnt model. We set $T1 = T2 = T$ and vary T in a set of 11 values from 2^0 to 2^{10}. This allows us to plot a curve of model accuracy against training time. For MK-FDA, the similar curve can be plotted by varying the convergence threshold ε in a set of 11 values: $\{2^0, \cdots, 2^{-6}, 10^{-2}, \cdots, 10^{-5}\}$. Note that the regularisation parameters (λ for MK-FDA and C for OBSCURE) are set to values that yield the highest classification accuracy.

The resulting time-accuracy curves for all 3 splits of the dataset are presented in Figure 9, where the top row corresponds to the case of $p = 1 + 2^{-6}$ and the bottom row to $p = 2$, and each column corresponds to one split. It is evident that MK-FDA typically reaches its optimum faster than OBSCURE, especially in the case of $p = 2$. Moreover, the optimum achieved by MK-FDA is at least as accurate as that by OBSCURE, confirming our findings in Section 3.4. All the training time reported in this section is measured on a single core of an Intel Xeon E5520 2.27GHz processor.

4. Discussion: FDA vs. SVM

The empirical observation that MK-FDA tends to outperform MK-SVM on image categorisation datasets matches well with our experience with single kernel FDA and single kernel SVM on several other object/image/video classification benchmarks, including VOC2008, VOC2009, VOC2010,[12] Trecvid2008, Trecvid2009,[13] and ImageCLEF2010.[14] In this section, we discuss the connection between (MK-)SVM and (MK-)FDA from perspectives of both loss function and version space, and attempt to explain their different performance.

It is well known that many machine learning problems essentially boil down to function learning. In the supervised scenario, it is intuitive to learn the function by minimising the empirical loss for the given set of labelled input/output pairs $\{\mathbf{x}_i, y_i\}_{i=1}^m$, with respect to some loss function. However, such an empirical risk minimisation principle is ill-posed and therefore does not generalise (Tikhonov and Arsenin, 1977; Vapnik, 1999). Regularisation tries to restore well-posedness of the learning problem, by restricting the complexity of the function set over which the empirical loss is minimised. By (implicitly) mapping the data into a high dimensional feature space, this can be conveniently done in the form of Tikhonov regularisation:

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m} V(f(\phi(\mathbf{x}_i)), y_i), \tag{27}$$

where $\phi(\mathbf{x}_i)$ is the mapping to the feature space, $f(\phi(\mathbf{x}_i)) = \mathbf{w}^T\phi(\mathbf{x}_i)$ is the linear function to be learnt, the complexity of the function set is regularised by $\frac{1}{2}||\mathbf{w}||^2$, and $V(\cdot,\cdot)$ measures the empirical loss. Learning machines with the form of Equation (27) are collectively termed regularised kernel machines, a name capturing the two key aspects of them: regularisation, and kernel mapping. Note that in the formulation above, the unregularised bias term b in standard SVM is absent from the linear function. As shown in Keerthi and Shevade (2003); Poggio et al. (2004), the two formulations, with and without b, can be made equivalent by transforming the kernel function.

The setting in Equation (27) is very general, in the sense that many state-of-the-art machine learning techniques can be realised by plugging in different loss functions. For example, the hinge loss $V(f(\phi(\mathbf{x})), y) = (1 - yf(\phi(\mathbf{x})))_+$, where $(\cdot)_+ = \max(\cdot, 0)$, gives rise to the well known SVM,

12. More information on VOC can be found at `http://pascallin.ecs.soton.ac.uk/challenges/VOC`.
13. More information on Trecvid can be found at `http://www-nlpir.nist.gov/projects/trecvid`.
14. More information on ImageCLEF can be found at `http://www.imageclef.org/2010`.

probably the most popular learning machine in the past ten years. On the other hand, along with the success of SVM, regularised kernel machines using the square loss $V(f(\phi(\mathbf{x})),y) = (y - f(\phi(\mathbf{x})))^2$ have emerged several times under various names, including: regularised networks (RN) (Girosi et al., 1995; Evgeniou et al., 2000), regularised least squares (RLS) (Rifkin, 2002), kernel ridge regression (KRR) (Saunders et al., 1998; Hastie et al., 2002), least squares support vector machines (LSSVM) (Suykens and Vandewalle, 1999; Gestel et al., 2002), proximal support vector machines (PSVM) (Fung and Mangasarian, 2001). In particular, shortly after the proposal of kernel FDA (Mika et al., 1999; Baudat and Anouar, 2000), its regularised version was shown to be yet another equivalent formulation (Duda et al., 2000; Rifkin, 2002; Gestel et al., 2002).

There is a long list of literature which compares the performance of FDA and SVM, for example, Mika (2002), Rifkin (2002), Cai et al. (2007) and Ye et al. (2008), with most of them reporting both methods yield virtually identical performance, and the rest claiming there is a small advantage towards one method or the other. It is speculated in Mika et al. (1999) that the superior performance of FDA over SVM in their experiments is due to the fact that FDA uses all training examples in the test stage while SVM uses only the support vectors. However, a more elegant way of explaining the different performance of SVM and FDA is probably from the perspective of version space. Version space is the space of all consistent hypotheses, that is, all $\mathbf{w}$'s that correspond to hyperplanes with zero training error (Rujan, 1997). Note that with a full rank kernel matrix, linear separability in the feature space and therefore the existence of version space is guaranteed. It is shown in Rujan (1997) that the optimal hyperplane in the Bayes sense, which requires the knowledge of the joint distribution on $\mathcal{X} \times \mathcal{Y}$ thus not obtainable in practice, is arbitrarily close (with increasing training sample size) to the centre of mass of the version space.

Algorithms that explicitly approximate the Bayes point were later termed Bayes point machine (BPM) in Herbrich et al. (2001). Herbrich et al. (2001) also prove that the hyperplane found by SVM corresponds to the centre of the largest inscribed ball of the version space. In this light, SVM can be viewed as an approximation to BPM. This approximation is reasonable if the version space is regularly shaped, but can be weak otherwise (Rujan, 1997; Herbrich et al., 2001; Mika, 2002). For example, experiments in Herbrich et al. (2001) show that BPM consistently outperforms SVM. Recently, an ellipsoid SVM was proposed (Momma et al., 2010), where the idea is to improve the approximation to the Bayes point by using the centre of the largest inscribed ellipsoid, instead of that of the ball. We conjecture that for certain kernels (e.g., kernels generated using local descriptors and bag-of-words model, as those used in image categorisation problems), due to the different loss functions used, the hyperplane given by FDA is closer to the Bayes point than that given by SVM, resulting in the superior performance of (MK-)FDA in our experiments. How to decide without a validation process whether (MK-)FDA or (MK-SVM) is more suitable for a given kernel (set), and how to incorporate explicit BPM approximation into MKL, are interesting research directions for the future.

5. Conclusions

In this paper we have incorporated latest advances in both non-sparse MKL formulation and MKL optimisation techniques into MK-FDA. We have presented a non-sparse version of MK-FDA based on an ℓ_p norm regularisation of kernel weights, and have discussed several of its reformulations and associated optimisation strategies, including wrapper and interleaved algorithms for its saddle point formulation, and an SMO-based scheme for its dual formulation.

We carried out extensive evaluation on six datasets from various application areas. Our results indicate that the optimal norm p, and therefore the "intrinsic sparsity" of the base kernels, can be estimated on an independent validation set. This estimation can be exploited in many practical applications where there is no prior knowledge on how informative the channels are. We have also compared closely the performance of ℓ_p MK-FDA and that of several variants of ℓ_p MK-SVM. On object and image categorisation problems, MK-FDA tends to have a small advantage. This observation is consistent with our findings elsewhere regarding the performance of single kernel FDA/SVM. In terms of training time, the wrapper-based MK-FDA implementation has similar or favourable efficiency on small to medium sized problems when compared against state-of-the-art MKL techniques. On large scale problems, alternative optimisation strategies discussed in the paper should be employed to improve the efficiency and scalability of MK-FDA.

Finally, we have provided a discussion on the connection between (MK-)FDA and (MK-)SVM from the perspectives of both loss function and version space, under the unified framework of regularised kernel machines.

Acknowledgments

We would like to thank Sören Sonnenburg, Gunnar Rätsch, Alexander Binder for their help on the Shogun toolbox, Jianhui Chen and Jieping Ye for help on their discriminant MKL package, and Marius Kloft for helpful discussions. We would also like to thank the anonymous reviewers and the editors for their valuable comments and suggestions. The work presented in this paper is supported by the EPSRC/UK grant EP/F069626/1 ACASVA Project.

Appendix A. Multiclass ℓ_p MK-FDA Saddle Point Formulation

In this appendix, we first derive the saddle point formulation of multiclass MKL for a general convex loss. Multiclass MK-FDA saddle point problem is then derived as a special case of it. Using the output encoding scheme in Equation (18), multiclass MKL for a general convex loss function $V(\xi_{ik}, h_{ik})$ can be stated as:

$$\begin{aligned} &\min_{\mathbf{w}_{jk},\xi_{ik},\boldsymbol{\beta}} \sum_{k=1}^{c} \left(\frac{1}{2} \sum_{j=1}^{n} \frac{||\mathbf{w}_{jk}||^2}{\beta_j} + C \sum_{i=1}^{m} V(\xi_{ik}, h_{ik}) \right) \\ &\text{s.t.} \quad \sum_{j=1}^{n} \mathbf{w}_{jk}^T \phi_j(\mathbf{x}_i) = \xi_{ik}, \ \forall i, \ \forall k; \ \ \boldsymbol{\beta} \geq \mathbf{0}; \ \ ||\boldsymbol{\beta}||_p^2 \leq 1. \end{aligned} \tag{28}$$

We build the Lagrangian of Equation (28):

$$\begin{aligned} \mathcal{L} = &\sum_{k=1}^{c} \left(\frac{1}{2} \sum_{j=1}^{n} \frac{||\mathbf{w}_{jk}||^2}{\beta_j} + C \sum_{i=1}^{m} V(\xi_{ik}, h_{ik}) \right) + \zeta(\frac{1}{2}||\boldsymbol{\beta}||_p^2 - \frac{1}{2}) \\ &- \sum_{k=1}^{c} \sum_{i=1}^{m} \alpha_{ik} \left(\sum_{j=1}^{n} \mathbf{w}_{jk}^T \phi_j(x_i) - \xi_{ik} \right), \end{aligned}$$

set to zero the derivatives of the Lagrangian w.r.t. $\mathbf{w}_{jk}$, and substitute back. After some rearrangements we have:

$$\mathcal{L} = \sum_{k=1}^{c}\left(C\sum_{i=1}^{m}V(\xi_{ik},h_{ik}) + \sum_{i=1}^{m}\alpha_{ik}\xi_{ik} - \frac{1}{2}\sum_{j=1}^{n}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k\right) + \zeta(\frac{1}{2}||\boldsymbol{\beta}||_p^2 - \frac{1}{2}),$$

where $\boldsymbol{\alpha}_k = (\alpha_{1k},\cdots,\alpha_{mk})^T$. Following Theorem 1 of Kloft et al. (2011) it can be shown that at the optimum $||\boldsymbol{\beta}||_p^2 = 1$. Using this fact we arrive at the multiclass MKL saddle point problem for a general loss function:

$$\min_{\xi_{ik},\boldsymbol{\beta}}\max_{\alpha_{ik}}\sum_{k=1}^{c}\left(C\sum_{i=1}^{m}V(\xi_{ik},h_{ik}) + \sum_{i=1}^{m}\alpha_{ik}\xi_{ik} - \frac{1}{2}\sum_{j=1}^{n}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k\right) \tag{29}$$
$$\text{s.t.} \quad \boldsymbol{\beta}\geq\mathbf{0};\ \ ||\boldsymbol{\beta}||_p^2 \leq 1.$$

At this point any convex loss function can be plugged into Equation (29). Take the square loss $V(\xi_{ik},h_{ik}) = \frac{1}{2}(\xi_{ik}-h_{ik})^2$ as an example. Setting to zero the derivatives of $\mathcal{L}$ w.r.t. ξ_{ik} we have $\xi_{ik} = h_{ik} - \alpha_{ik}/C$. Plugging this into Equation (29) and rearranging we arrive at the multiclass MKL saddle point problem for square loss, that is, multiclass multiple kernel regularised least squares (MK-RLS):

$$\min_{\boldsymbol{\beta}}\max_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(\mathbf{h}_k^T\boldsymbol{\alpha}_k - \frac{1}{2C}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k - \frac{1}{2}\sum_{j=1}^{n}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k\right) \tag{30}$$
$$\text{s.t.} \quad \boldsymbol{\beta}\geq\mathbf{0};\ \ ||\boldsymbol{\beta}||_p^2 \leq 1,$$

where the c classes are coupled through the common set of kernel weights $\boldsymbol{\beta}$. By making substitutions $\boldsymbol{\alpha}_k \to \frac{C}{2}\boldsymbol{\alpha}_k$ and then $C \to \frac{1}{\lambda}$, it directly follows that the MK-RLS in Equation (30) is equivalent to the MK-FDA in Equation(19).

Appendix B. Multiclass ℓ_p MK-FDA Dual Formulation

In this appendix, we derive the dual formulation of multiclass MK-FDA. We again consider multiclass MKL with a general convex loss, but following Vishwanathan et al. (2010) this time we impose the norm constraint in the form of Tikhonov regularisation instead of Ivanov regularisation:

$$\min_{\mathbf{w}_{jk},\xi_{ik},\boldsymbol{\beta}}\sum_{k=1}^{c}\left(\frac{1}{2}\sum_{j=1}^{n}\frac{\|\mathbf{w}_{jk}\|^2}{\beta_j} + C\sum_{i=1}^{m}V(\xi_{ik},h_{ik})\right) + \frac{\mu}{2}\|\boldsymbol{\beta}\|_p^2 \tag{31}$$
$$\text{s.t.} \quad \sum_{j=1}^{n}\mathbf{w}_{jk}^T\phi_j(\mathbf{x}_i) = \xi_{ik},\ \forall i,\ \forall k;\ \ \boldsymbol{\beta}\geq\mathbf{0}.$$

Note however that the switching from Ivanov to Tikhonov regularisation is not essential for the derivation in the following. The dual program for Ivanov regularisation in Equation (28) can be derived in a similar way.

Building the Lagrangian of Equation (31):

$$\mathcal{L} = \sum_{k=1}^{c}\left(\frac{1}{2}\sum_{j=1}^{n}\frac{\|\mathbf{w}_{jk}\|^2}{\beta_j} + C\sum_{i=1}^{m}V(\xi_{ik},h_{ik})\right) + \frac{\mu}{2}\|\boldsymbol{\beta}\|_p^2 - \sum_{j=1}^{n}\gamma_j\beta_j$$
$$-\sum_{k=1}^{c}\sum_{i=1}^{m}\alpha_{ik}\left(\sum_{j=1}^{n}\mathbf{w}_{jk}^T\phi_j(x_i) - \xi_{ik}\right),$$

and setting to zero the derivatives w.r.t. β_j, we have:

$$\mu(\sum_{j=1}^{n}\beta_j^p)^{\frac{2}{p}-1}\beta_j^{p-1} = \gamma_j + \frac{1}{2}\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j \boldsymbol{\alpha}_k. \tag{32}$$

Multiplying both sides of Equation (32) by β_j and then taking summation over j gives us:

$$\mu\|\boldsymbol{\beta}\|_p^2 = \sum_{j=1}^{n}\beta_j(\gamma_j + \frac{1}{2}\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j \boldsymbol{\alpha}_k),$$

or equivalently:

$$\sum_{j=1}^{n}\gamma_j\beta_j = -\frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T\beta_j K_j\boldsymbol{\alpha}_k + \mu\|\boldsymbol{\beta}\|_p^2. \tag{33}$$

On the other hand, raise both sides of Equation (32) to power $\frac{p}{p-1}$ and then take summation over j, we have:

$$\mu\|\boldsymbol{\beta}\|_p^2 = \frac{1}{\mu}\left\|\left(\gamma_j + \frac{1}{2}\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j\boldsymbol{\alpha}_k\right)_{j=1}^{n}\right\|_q^2, \tag{34}$$

where $q = \frac{p}{p-1}$ is the dual norm of p.

Now let us set the derivatives of $\mathcal{L}$ w.r.t. $\mathbf{w}_{jk}$ also to zero, and substitute the result and Equation (33), Equation (34) back into $\mathcal{L}$. Using the fact that $\gamma_j = 0$ at the optimum (Vishwanathan et al., 2010), and after some rearrangements we arrive at:

$$\mathcal{L} = \sum_{k=1}^{c}\left(C\sum_{i=1}^{m}V(\xi_{ik},h_{ik}) + \sum_{i=1}^{m}\alpha_{ik}\xi_{ik}\right) - \frac{1}{8\mu}\left\|\left(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j\boldsymbol{\alpha}_k\right)_{j=1}^{n}\right\|_q^2. \tag{35}$$

At this point any convex loss function can be plugged into Equation (35) to recover the corresponding multiclass MKL dual. We again take the square loss $V(\xi_{ik},h_{ik}) = \frac{1}{2}(\xi_{ik}-h_{ik})^2$ as an example. Setting to zero the derivatives of $\mathcal{L}$ w.r.t. ξ_{ik} we have $\xi_{ik} = h_{ik} - \alpha_{ik}/C$. Plugging this into Equation (35) and rearranging we arrive at the multiclass MK-RLS dual problem:

$$\max_{\boldsymbol{\alpha}_k}\sum_{k=1}^{c}\left(\mathbf{h}_k^T\boldsymbol{\alpha}_k - \frac{1}{2C}\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k\right) - \frac{1}{8\mu}\left\|\left(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j\boldsymbol{\alpha}_k\right)_{j=1}^{n}\right\|_q^2. \tag{36}$$

Unlike the saddle point formulation in Equation (30), the kernel weights $\boldsymbol{\beta}$ have been eliminated from Equation (36). Despite this, Equation (30) and Equation (36) are equivalent, in the sense that for any given C there exist a μ (and vice versa) such that the optimal solutions to both problems are identical (Kloft et al., 2011).

Finally, substituting Equation (34) and $\gamma_j = 0$ into Equation (32), we show that once the optimal $\boldsymbol{\alpha}_k$ are found by solving Equation (36), the kernel weights $\boldsymbol{\beta}$ are given by:

$$\beta_j = \frac{1}{2\mu}\left(\sum_{j=1}^{n}(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j\boldsymbol{\alpha}_k)^q\right)^{\frac{1}{q}-\frac{1}{p}}(\sum_{k=1}^{c}\boldsymbol{\alpha}_k^T K_j\boldsymbol{\alpha}_k)^{\frac{q}{p}}.$$

References

F. Bach and G. Lanckriet. Multiple kernel learning, conic duality, and the smo algorithm. In *International Conference on Machine Learning*, 2004.

G. Baudat and F. Anouar. Generalized discriminant analysis using a kernel approach. *Neural Computation*, 12:2385–2404, 2000.

O. Bousquet and D. Herrmann. On the complexity of learning the kernel matrix. In *Advances in Neural Information Processing Systems*, 2003.

M. Braun, J. Buhmann, and K. Müller. On relevant dimensions in kernel feature spaces. *Journal of Machine Learning Research*, 9:1875–1908, 2008.

D. Cai, X. He, and J. Han. Efficient kernel discriminant analysis via spectral regression. In *International Conference on Data Mining*, 2007.

H. Cai, K. Mikolajczyk, and J. Matas. Learning linear discriminant projections for dimensionality reduction of image descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2):338–352, 2011.

O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46:131–159, 2002.

C. Cortes, M . Mohri, and A. Rostamizadeh. L2 regularization for learning kernels. In *Uncertainty in Artificial Intelligence*, 2009.

N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. In *Advances in Neural Information Processing Systems*, 2002.

G. Csurka, C. Dance, L. Fan, J. Willamowski, and C Bray. Visual categorization with bags of keypoints. In *ECCV workshop on Statistical Learning in Computer Vision*, 2004.

R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley, 2000.

M. Everingham, L. van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, 2007.

M. Everingham, L. van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html, 2008.

T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13:1–50, 2000.

L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006.

R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

G. Fung and O. L. Mangasarian. Proximal support vector machine classifier. In *International Conference on Knowledge Discovery and Data Mining*, 2001.

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

J. Gemert, J. Geusebroek, C. Veenman, and A. Smeulders. Kernel codebooks for scene categorization. In *European Conference on Computer Vision*, 2008.

T. Gestel, J. Suykens, G. Lanckriet, A. Lambrechts, B. Moor, and J. Vandewalle. Bayesian framework for least-squares support vector machine classifiers, gaussian processes, and kernel fisher discriminant analysis. *Machine Learning*, 14(5):1115–1147, 2002.

F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219–269, 1995.

G. Golub and C. van Loan. *Matrix Computations*. John Hopkins University Press, third edition, 1996.

K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *Journal of Machine Learning Research*, 8:725–760, 2007.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, second edition, 2002.

R. Herbrich, T. Graeple, and C. Campbell. Bayes point machines. *Journal of Machine Learning Research*, 1:245–279, 2001.

R. Hettich and K. Kortanek. Semi-infinite programming: Theory, methods, and applications. *SIAM Review*, 35(3):380–429, 1993.

T. Joachims. *Making Large-Scale Support Vector Machine Learning Practical*. MIT Press, Cambridge, MA, 1988.

S. Keerthi and S. Shevade. Smo algorithm for least squares svm formulations. *Neural Computation*, 15(2):487–507, 2003.

S. Kim, A. Magnani, and S. Boyd. Optimal kernel selection in kernel fisher discriminant analysis. In *International Conference on Machine Learning*, 2006.

M. Kloft, U. Brefeld, P. Laskov, and S. Sonnenburg. Non-sparse multiple kernel learning. In *NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels*, 2008.

M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien. Efficient and accurate lp-norm mkl. In *Advances in Neural Information Processing Systems*, 2009.

M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien. Lp norm multiple kernel learning. *Journal of Machine Learning Research*, 12:953–997, 2011.

G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. Jordan. Learning teh kernel matrix with semi-definite programming. In *International Conference on Machine Learning*, 2002.

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

S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *International Conference on Computer Vision and Pattern Recognition*, 2006.

D. Liu and J. Nocedal. On the limited memory method for large scale optimization. *Mathematical Programming B*, 45(3):503–528, 1989.

J. Lopez and J. Suykens. First and second order smo algorithms for ls-svm classifiers. *Neural Processing Letters*, 33(1):31–44, 2011.

D. Lowe. Distincetive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

S. Mika. Kernel fisher discriminants. PhD Thesis, University of Technology, Berlin, Germany, 2002.

S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K. Müller. Fisher discriminant analysis with kernels. In *IEEE Signal Processing Society Workshop: Neural Networks for Signal Processing*, 1999.

K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

M. Momma, K. Hatano, and H. Nakayama. Ellipsoidal support vector machines. In *Asian Conference on Machine Learning*, 2010.

S. Nakajima, A. Binder, C. Muller, W. Wojcikiewicz, M. Kloft, U. Brefeld, K. Müller, and M. Kawanabe. Multiple kernel learning for object classification. Technical Report on Information-Based Induction Sciences, 2009.

M. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

C. Ong and A. Zien. An automated combination of kernels for predicting protein subcellular localization. In *Workshop on Algorithms in Bioinformatics*, 2008.

C. Ong, A. Smola, and R. C. Williamson. Hyperkernels. In *Advances in Neural Information Processing Systems*, 2003.

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

F. Orabona, L. Jie, and B. Caputo. Online-batch strongly convex multi kerenl learning. In *International Conference on Computer Vision and Pattern Recognition*, 2010.

T. Poggio, S. Mukherjee, R. Rifkin, A. Rakhlin, and A. Verri. B. In *Conference on Uncertainty in Geometric Computations*, 2004.

A. Rakotomamonjy, F. Bach, Y. Grandvalet, and S. Canu. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

G. Rätsch. Robust boosting via convex optimization. PhD Thesis, University of Potsdam, Potsdam, Germany, 2001.

R. Rifkin. Everything old is new again: a fresh look at historical approaches in machine learning. PhD Thesis, Massachusetts Institute of Technology, Boston, USA, 2002.

P. Rujan. Playing billiard in version space. *Neural Computation*, 9:99–122, 1997.

K. Sande, T. Gevers, and C. Snoek. Evaluation of color descriptors for object and scene recognition. In *International Conference on Computer Vision and Pattern Recognition*, 2008.

C. Saunders, A. Gammerman, and V. Vovk. Ridge regression learning algorithm in dual variables. In *International Conference on Machine Learning*, 1998.

B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002.

B. Schölkopf, A. Smola, and K. Müller. Kernel principal component analysis. *Advances in Kernel Methods: Support Vector Learning*, pages 327–352, 1999.

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

C. Snoek, M. Worring, J. Gemert, J. Geusebroek, and A. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *ACM Multimedia Conference*, 2006.

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

S. Sonnenburg, G. Rätsch, S. Henschel, C. Widmer, J. Behr, A. Zien adn F. Bona, A. Binder, C. Gehl, and V. Franc. The shogun machine learning toolbox. *Journal of Machine Learning Research*, 11: 1799–1802, 2010.

J. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9:293–300, 1999.

M. Szafranski, Y. Grandvalet, and A. Rakotomamonjy. Composite kernel learning. In *International Conference on Machine Learning*, 2008.

A. Tahir, J. Kittler, K. Mikolajczyk, F. Yan, K. Sande, and T. Gevers. Visual category recognition using spectral regression and kernel discriminant analysis. In *International Workshop on Subspace Methods*, 2009.

A. Tikhonov and V. Arsenin. *Solutions of Ill-Posed Problems*. Winston, Washington DC, 1977.

V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1999.

S. Vishwanathan, Z. Sun, and N. Theera-Ampornpunt. Multiple kernel learning and the smo algorithm. In *Advances in Neural Information Processing Systems*, 2010.

F. Yan, J. Kittler, K. Mikolajczyk, and A. Tahir. Non-sparse multiple kernel learning for fisher discriminant analysis. In *International Conference on Data Mining*, 2009a.

F. Yan, K. Mikolajczyk, J. Kittler, and A. Tahir. A comparison of l1 norm and l2 norm multiple kernel svms in image and video classification. In *International Workshop on Content-Based Multimedia Indexing*, 2009b.

F. Yan, K. Mikolajczyk, M. Barnard, H. Cai, and J. Kittler. Lp norm multiple kernel fisher discriminant analysis for object and image categorisation. In *International Conference on Computer Vision and Pattern Recognition*, 2010.

J. Ye, S. Ji, and J. Chen. Multi-class discriminant kernel learning via convex programming. *Journal of Machine Learning Research*, 9:719–758, 2008.

J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.

A. Zien and C. Ong. Multiclass multiple kernel learning. In *International Conference on Machine Learning*, 2007.