On $p$-norm Path Following in Multiple Kernel Learning for Non-linear Feature Selection

Pratik Jawanpuria
Department of Computer Science & Engineering, Indian Institute of Technology Bombay, India 400 076

Manik Varma
Microsoft Research India, Bangalore, India 560 001

J. Saketha Nath
Department of Computer Science & Engineering, Indian Institute of Technology Bombay, India 400 076

Abstract

Our objective is to develop formulations and algorithms for efficiently computing the feature selection path – i.e. the variation in classification accuracy as the fraction of selected features is varied from null to unity. Multiple Kernel Learning subject to $l_{p \geq 1}$ regularization ($l_p$-MKL) has been demonstrated to be one of the most effective techniques for non-linear feature selection. However, state-of-the-art $l_p$-MKL algorithms are too computationally expensive to be invoked thousands of times to determine the entire path.

We propose a novel conjecture which states that, for certain $l_p$-MKL formulations, the number of features selected in the optimal solution monotonically decreases as $p$ is decreased from an initial value to unity. We prove the conjecture, for a generic family of kernel target alignment based formulations, and show that the feature weights themselves decay (grow) monotonically once they are below (above) a certain threshold at optimality. This allows us to develop a path following algorithm that systematically generates optimal feature sets of decreasing size. The proposed algorithm sets certain feature weights directly to zero for potentially large intervals of $p$ thereby reducing optimization costs while simultaneously providing approximation guarantees.

We empirically demonstrate that our formulation can lead to classification accuracies which are as much as 10% higher on benchmark data sets not only as compared to other $l_p$-MKL formulations and uniform kernel baselines but also leading feature selection methods. We further demonstrate that our algorithm reduces training time significantly over other path following algorithms and state-of-the-art $l_p$-MKL optimizers such as SMO-MKL. In particular, we generate the entire feature selection path for data sets with a hundred thousand features in approximately half an hour on standard hardware. Entire path generation for such data set is well beyond the scaling capabilities of other methods.

1. Introduction

Feature selection is an important problem in machine learning motivated by considerations of elimination of noisy, expensive and redundant features, model compression for learning and predicting on a budget, model interpretability, etc. In many real world applications, one needs to determine the entire feature selection path, i.e. the variation in prediction accuracy as the fraction of selected features is varied from null to unity, so as to determine the most feasible operating point on the path in the context of the given application.

There has been much recent progress in non-linear feature selection (Li et al., 2006; Bach, 2008; Hwang et al., 2011; Song et al., 2012) where the predictor is a non-linear function of the input features. Most algorithms have a parameter influencing the number of selected features and one would need to try thousands of parameter settings to generate the entire feature selection path. This is a challenge since state-of-the-art non-linear feature selection algorithms remain computationally expensive and training them thousands of times can be prohibitive (even with warm restarts).

In particular, Multiple Kernel Learning (MKL) techniques have been shown to be amongst the most effective for non-linear feature selection (Ji et al., 2008; Chen et al., 2008;
Varma & Babu, 2009; Vedaldi et al., 2009; Levinboim & Sha, 2012; Hwang et al., 2012). They have been demonstrated to be superior to a number of linear and non-linear filter and wrapper methods (Varma & Babu, 2009). MKL feature selection techniques were also used to reduce prediction time in the joint winning entry of the competitive PASCAL VOC 2009 object detection in images challenge (Vedaldi et al., 2009). However, even though many specialized MKL optimization techniques have been developed (Kloft et al., 2011; Vishwanathan et al., 2010; Orabona & Jie, 2011; Orabona et al., 2012; Jain et al., 2012), training them thousands of times with different parameter settings is often infeasible.

Our objective, in this paper, is to develop MKL formulations and algorithms for efficiently determining the non-linear feature selection path. Our starting point is a novel conjecture that, for certain MKL formulations subject to $l_p$-regularization, the number of features (kernels) selected in the optimal solution monotonically decreases as $p$ is decreased from an initial value to unity. We first prove that the conjecture is true for a generic family of Kernel Target Alignment (KTA) based $l_p$-MKL formulations. This implies that regulating $p$, in the formulations of this family, provides a principled way of generating the feature selection path (see Fig 1(a), 1(b)). In fact, for this family, we further strengthen the conjecture and prove that the feature weights themselves decay (grow) monotonically once they are below (above) a certain threshold at optimality (see Fig 1(c)-1(e)). This implies that there exist juncture points along the path and an algorithm can exploit these by eliminating certain features from the optimization for potentially large intervals of $p$. It should be noted that these conjectures are non-trivial and we show that they do not hold for the popular square loss based $l_p$-MKL formulation.

Based on these results, we propose a Generalized $l_p$-KTA formulation which extends the KTA formulations of (Cristianini et al., 2001; Lanckriet et al., 2004; Cortes et al., 2012). The proposed formulation is strongly convex and leads to robust feature selection (Bousquet & Elisseeff, 2002; Zou & Hastie, 2005; Kivinen et al., 2006). Furthermore, the feature monotonicity results allow us to develop a predictor-corrector based path following algorithm for Generalized $l_p$-KTA that exploits the presence of juncture points for increased efficiency.

We perform extensive experiments to demonstrate that the proposed Generalized $l_p$-KTA formulation can lead to classification accuracies which are as much as 10% higher on benchmark data sets not only as compared to other $l_p$-MKL formulations and uniform kernel baselines but also leading feature selection methods. We further demonstrate that our algorithm reduces training time significantly over other path following algorithms and state-of-the-art $l_p$-MKL optimizers such as SMO-MKL. In particular, we generate the entire feature selection path for data sets with a hundred thousand features in approximately half an hour on standard hardware. Entire path generation for such data set is well beyond the scaling capabilities of other methods.

Our contributions are as follows: (a) we propose, and prove, a novel conjecture regarding the monotonicity of selected features, and feature weights themselves, for certain $l_p$-regularized MKL formulations; (b) we propose a Generalized $l_p$-KTA formulation for robust non-linear feature selection that is capable of achieving significantly higher classification accuracies as compared to state-of-the-art; and (c) our algorithm is many times faster than other leading techniques.

### 2. Related Work

Considerable progress has been made in the area of non-linear feature selection. A popular approach is to map the input features to a kernel-induced feature space while simultaneously performing feature selection in the original input space. For example, (Li et al., 2006) propose to perform LASSO regression in the induced space and thereby perform non-linear feature selection while (Weston et al., 2000; Grandvalet & Canu, 2002) pose this problem as that of tuning the hyper-parameters of the kernel. Another promising direction is to find un-correlated or independent features in the kernel induced feature space. While (Wu et al., 2005; Cao et al., 2007) aim at selecting orthogonal features in the feature space, (Chen et al., 2008; Song et al., 2012) employ Hilbert-Schmidt Independence Criterion based measures. Most of these approaches are either computationally expensive or resort to approximately minimizing their non-convex objectives.

Multiple Kernel Learning based techniques for non-linear feature selection have been explored in settings such as multi-label classification (Ji et al., 2008), bio-informatics (Chen et al., 2008; Levinboim & Sha, 2012) and object categorization (Vedaldi et al., 2009; Hwang et al., 2012). In (Varma & Babu, 2009), MKL techniques for non-linear feature selection were shown to be better than boosting (Baluja & Rrowley, 2007), lasso (Andrew & Gao, 2007), sparse SVM (Chan et al., 2007), LP-SVM (Fung & Mangasarian, 2002) and BAHSIC (Song et al., 2012). Various MKL formulations have been developed including $l_{p \geq 1}$-norm regularization over the kernel weights (Lanckriet et al., 2004; Rakotomamonjy et al., 2008; Cortes et al., 2009a; Kloft et al., 2011; Orabona & Jie, 2011; Orabona et al., 2012), mixed-norm regularizers (Aflalo et al., 2011), non-linear combinations of base kernels (Cortes et al., 2009b; Varma & Babu, 2009), Bregman divergence based regularizers (Vishwanathan et al., 2010) and for regularized kernel discriminant analysis (Ye...
et al., 2008). State-of-the-art $l_{p\geq1}$-MKL optimization techniques such as SMO-MKL (Vishwanathan et al., 2010) and SPG-GMKL (Jain et al., 2012) have been shown to scale to a million kernels. Nevertheless, these techniques take more than a day to train on a standard desktop and so computing them thousands of times for determining the entire feature selection path is infeasible.

Path following over the regularization parameter has been studied in the context of both non-sparse regularization (Hastie et al., 2004) as well as sparse linear classifiers (Zhu et al., 2003). Some of the other settings where path following has been studied are: $l_1$-MKL (Bach et al., 2004), general non-linear regularization paths (Rosset, 2004), boosting (Zhao & Yu, 2007) and $l_1$ regularized feature selection (Li & Smichiesescu, 2010). Tracing the solution path for other hyper-parameters has also been explored. While (Gunter & Zhu, 2005; Wang et al., 2006) perform path following over the tube width parameter in support vector regression, (Wang et al., 2007) follow the path for the kernel hyper-parameter in SVMs.

To the best of our knowledge, $p$-norm path following has not been studied in the literature so far. The closest competing technique to ours is the path following $l_1$-MKL approach of (Bach et al., 2004) and we present comparative results in Section 6.

3. Generalized $l_p$-KTA

To perform non-linear feature selection, we associate a non-linear base kernel with each individual feature, such as RBF kernels defined per feature, and then learn a sparse combination of base kernels. We introduce our Generalized $l_p$-KTA formulation for sparse kernel learning using the following notation.

Let $k_1, \ldots, k_r$ denote the given base kernel functions (one per feature) and let $K_1, \ldots, K_r$ denote the corresponding centered gram matrices obtained from the training data so that the features induced in the RKHS have zero mean (Cortes et al., 2012). Let $y$ denote the vector with entries as the labels of the training data. We are interested in learning a kernel, $k$, that is a conic combination of the given base kernels, i.e. $k = \sum_{i=1}^{r} \eta_i k_i$, $\eta_i \geq 0 \forall i$. We focus on the following family of Bregman divergence based $l_p$-KTA formulations for learning the kernel weights $\eta_i$.

Let $F$ be a strictly convex and differentiable function and let $\nabla F$ denote its gradient. Then, the Bregman divergence generated by the function $F$ is given by $B_F(x) = F(x) - F(x_0) - (x - x_0)^\top \nabla F(x_0)$, where $x_0$ is some fixed and given point in the domain of $F$. As an example, $F(x) = \langle x, x \rangle$ leads to $B_F(x) = \|x - x_0\|^2$, the squared Euclidean distance. The proposed Generalized $l_p$-KTA formulations have the following form:

$$
\min_{\eta \geq 0} \lambda_1 \tilde{B}_F(\eta) + \lambda_2 \sum_{i=1}^{r} \eta_i^p - \sum_{i=1}^{r} \eta_i y^\top K_i y, \quad (1)
$$

where the first term representing the Bregman divergence based regularizer is decomposable as $\tilde{B}_F(\eta) = \sum_i B_F(\eta_i)$, the second term is the sparsity inducing $l_p$ regularizer and the third term captures the alignment of the learnt kernel to the ideal kernel $yy^\top$. Note that $p \geq 1$, $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$ are regularization parameters.

A large number of popular Bregman divergences such as the squared Euclidean distance, generalized KL-divergence etc. are decomposable (Banerjee et al., 2005) and hence admissible under the Generalized $l_p$-KTA formulation. Such Bregman divergences are known to promote robust feature and kernel selection (Bousquet & Elisseeff, 2002; Zou & Hastie, 2005; Kivinen et al., 2006) while the $l_p$-norm regularizer achieves variable sparsity (Kloft et al., 2011). Note that the classical KTA formulations studied in (Lanckriet et al., 2004; Cristianini et al., 2001) can be obtained as special cases of our formulation by taking $F$ to be the squared.

Figure 1: (a) The feature selection path for the proposed Generalized $l_p$-KTA formulation. Note that generating the entire path for the largest data set (Dorothea) is well beyond the scaling capabilities of all existing algorithms while our proposed algorithm generated the path in only half an hour on standard hardware; (b) plot validating our weak Generalized $l_p$-KTA monotonicity conjecture that the number of selected features decreases monotonically with $p$; (c) & (d) plots validating our strong Generalized $l_p$-KTA monotonicity conjecture that feature (kernel) weights themselves vary monotonically beyond juncture points but vary non-monotonically in between (e). Figure best viewed in color under magnification.
Euclidean distance and having \( \lambda_2 = 0 \). On the other hand, substituting \( \lambda_1 = 0 \) results in the KTA formulation studied in Proposition (7) in (Cortes et al., 2012). The normalized KTA formulation in Proposition (9) in (Cortes et al., 2012) employs a non-decomposable Bregman divergence and hence is not a special case of (1). However we empirically demonstrate in Section 6 that the proposed Generalized \( l_p \)-KTA formulation outperforms this normalized KTA formulation both in terms of generalization accuracy as well as computational cost. In particular, the normalized KTA formulation needs to operate on an \( r \times r \) dense matrix which becomes infeasible for large \( r (r = 10^6) \) for the standard Dorothea feature selection data set.

The final classification results are obtained by training an SVM using the learnt kernel \( k = \sum_{i=1}^{r} \eta_i k_i \). This corresponds to non-linear feature selection since features for which \( \eta_i \) is zero do not contribute to the kernel and can be dropped from the data set.

We conclude this section with the following observation. Though (1) optimizes the kernel weights independently for a given \( \lambda_1 \) and \( \lambda_2 \), the proposed algorithm employs cross-validation for optimizing \( \lambda_2 \). Hence, the final optimal kernel weights need not be independent.

4. Monotonic Feature & Kernel Selection

Path Following

In this section, we prove the monotonicity conjecture for the proposed Generalized \( l_p \)-KTA formulation. Let \( \eta^*_i (p') \) denote the optimal weight of the kernel \( k_i \) obtained with (1) at \( p = p' \) and let \( K(p') \) denote the set of kernels active at \( p = p' \). A kernel \( k_i \) is said to be active at \( p = p' \), i.e. \( k_i \in K(p') \), if and only if \( \eta^*_i (p') > \epsilon \), where \( \epsilon > 0 \) is a user-defined tolerance. The proposed conjecture can now be formally stated as

\[
k_j \notin K(p') \Rightarrow k_j \notin K(p) \text{ } \forall \text{ } 1 < p < p'
\]

We begin our analysis by noting the following theorem:

**Theorem 1.** The path of the optimal solutions of (1) with respect to \( p \), i.e. \( \eta^*(p) \), is unique, smooth (provided \( F \) is twice-differentiable) and, in general, non-linear.

**Proof** The optimal solution path of (1), i.e. \( \eta^*(p) \), is unique since the objective of (1) is strictly convex. In order to prove the smoothness of the optimal solution path, we begin by deriving the necessary and sufficient conditions of optimality for (1). To this end, we first define

\[
g(\eta_i) = \lambda_1 B_F(\eta_i) + \lambda_2 \eta_i^p - \eta_i y^\top K_i y
\]

Next, we consider two cases:

**Case 1:** \( \eta^*_i = 0 \) for a given \( p \): The necessary and sufficient conditions of optimality for a convex function is that the gradient of the function at optimality, \( g'(\eta^*_i) \), should lie in the normal cone of the feasibility set (Ben-Tal & Nemirovski, 2001). This results in the following inequality:

\[
\lambda_1 (F'(0) - F'(\eta^*_i)) - y^\top K_i y \geq 0
\]

Since \( F \) is a convex function, we have \( F(\eta_i) \geq F(\eta^*_i) + F'(\eta^*_i)(\eta_i - \eta^*_i) \) as well as \( F(\eta_i) \geq F(\eta^*_i) + F'(\eta^*_i)(\eta_i - \eta^*_i) \). Summing these two inequalities, we get \( \eta_i \geq (F'(\eta_i) - F'(\eta^*_i))(\eta_i - \eta^*_i) \). Now substituting \( \eta_i = 0 \), we get \( \eta_i \geq 0 \) (\( F'(0) - F'(\eta^*_i) \)). Since \( \eta^*_i \geq 0 \) (feasible set of \( \eta_i \) is \( \geq 0 \)), it follows \( 0 \geq F'(0) - F'(\eta^*_i) \). Hence, in the LHS of (3), \( (F'(0) - F'(\eta^*_i)) \leq 0 \) and \( -y^\top K_i y \leq 0 \). It follows that the optimality conditions for \( \eta^*_i = 0 \) are

\[
F'(0) - F'(\eta^*_i) = 0 \text{ and } y^\top K_i y = 0
\]

Note that both the above equalities are independent of \( p \). It follows that if \( \exists p' > 1 \) s.t. \( \eta^*_i (p') = 0 \) then \( \eta^*_i (p) = 0 \text{ } \forall \text{ } p > 1 \). Thus, the optimal solution path, \( \eta^*_i (p) \), is smooth, in fact constant, when \( \exists p' > 1 \) s.t. \( \eta^*_i (p') = 0 \).

**Case 2:** \( \eta^*_i > 0 \) for a given \( p \): In this case, the necessary and sufficient conditions of optimality (Ben-Tal & Nemirovski, 2001) simplifies to \( g'(\eta^*_i) = 0 \). Thus we get the following optimality condition

\[
G_i(\eta_i, p) = \lambda_1 (F'(\eta_i) - F'(\eta^*_i)) + \lambda_2 p \eta_i^{p-1}
\]

\[
- y^\top K_i y = 0
\]

In the following, we prove that the path of the optimal solution, \( \eta^*_i (p) \), of (5) is smooth for the non trivial case: \( \eta^*_i (p) > 0 \text{ } \forall \text{ } p > 1 \).

Since the pair \( (\eta^*_i (p), p) \) always satisfy the equality in (5), we must have that \( dG_i(\eta^*_i (p), p) = \sum_j \frac{dG_j}{\eta_j} d\eta_j + \frac{dG_j}{dp} dp = 0 \text{ } \forall \text{ } p > 1 \). This leads to

\[
\frac{d\eta_i^p (p)}{dp} = -\frac{\lambda_2 \eta_i^p (p)^{p-1} (1 + p \ln \eta_i^p (p))}{\lambda_1 F''(\eta_i^p (p)) + \lambda_2 p (p - 1) \eta_i^p (p)^{p-2}}
\]

The terms \( \ln \eta_i^p (p) \) and \( \eta_i^p (p)^{p-2} \) are always finite as \( \eta_i^p (p) > 0 \) and the denominator of (6) is always non-zero, in fact positive, because for any convex, twice-differentiable \( F \) we have: \( F''(\eta_i^p (p)) \geq 0 \). Hence, the derivative along the optimal solution path (6) is well defined and itself a continuous function; proving that the optimal solution path is smooth (but generally non-linear) in this case too.

The next theorem states the key result that proves the monotonicity conjecture for the proposed Generalized \( l_p \)-KTA formulation.

**Theorem 2.** Given \( \eta^*_i (p') \), the following holds as \( p \) decreases from \( p' \) to unity: (1) \( \eta^*_i (p) \) decreases monotonically whenever \( \eta^*_i (p') < e^{-1} \); (2) \( \eta^*_i (p) \) increases monotonically whenever \( \eta^*_i (p') > e^{-1} \).
The key idea is to exploit the monotonicity in the active proximate the solution path at a lower computational cost.

We modify the standard PC algorithm so as to closely approximate the solution path at a lower computational cost. The key idea is to exploit the monotonicity in the active set size and directly set certain kernel weights to zero for all the subsequent $p$ values. Algorithm 1 summarizes the proposed algorithm. The algorithm maintains an active-set that is initialized to those kernels whose optimal weight at $p_0$ is above $\epsilon$. At every iteration, the first order PC step is used to determine the kernel weights for the next $p$ value. Kernels whose weight falls below $\epsilon$ are eliminated from the active set for the remainder of the optimization due to the monotonicity property. The error incurred by the proposed method can be bounded in terms of $\epsilon$. The following Lemma holds when the squared Euclidean distance is used as the Bregman divergence:

**Lemma 1.** For any $p$, the deviation in the objective value of (1) obtained using the approximate path following algorithm from the true optimal objective is upper bounded by $r(\lambda_1\epsilon^2 + \lambda_2(p-1)\epsilon^p)$.

This result follows from the optimality conditions (5) and Theorem 2. We also state an analogous lemma for the generalized KL-divergence in the supplementary material.

**Algorithm 1** Generic Algorithm for Computing Solution Path of the Kernel Weights

**Input:** $y^T K_1 y, \ldots, y^T K_r y$, step size $\Delta p (> 0)$, tolerance $\epsilon (> 0)$, start and end $p$ values: $p_0$ and $p_e$ respectively with $p_0 > p_e$.

**Output:** $\eta^*(p)$ for all $p$ at $\Delta$ intervals between $[p_e, p_0]$.

**Initialization:** $p = p_0$, Index set of active kernels $K = \{1, \ldots, r\}$

Solve (1) at $p = p_0$ to obtain the optimal solution $\eta^*(p_0)$

if $(\eta^*_i(p_0)) < \epsilon$ for any $i \in K$

Set $\eta^*_i(p_0) = 0$

Update $K = K \setminus \{i\}$

end if

repeat

Update $p = p - \Delta p$

Set $\eta^*_i(p) = 0$ for all $i \notin K$

for $i \in K$

**Predictor:** Initialize $\eta^*_i(p)$ using warm start or first or second order approximation

**Corrector:** Run Newton’s method to obtain $\eta^*_i(p)$

end for

if $(\eta^*_i(p)) < \epsilon$ for any $i \in K$

Set $\eta^*_i(p) = 0$

Update $K = K \setminus \{i\}$

end if

until $p \leq p_e$

---

<table>
<thead>
<tr>
<th>Data set</th>
<th>Num</th>
<th>Dim</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arcene</td>
<td>100</td>
<td>10000</td>
</tr>
<tr>
<td>Relathe</td>
<td>1427</td>
<td>4322</td>
</tr>
<tr>
<td>Basehock</td>
<td>1993</td>
<td>4862</td>
</tr>
<tr>
<td>Madeleon</td>
<td>2000</td>
<td>500</td>
</tr>
<tr>
<td>Pemac</td>
<td>1943</td>
<td>3289</td>
</tr>
<tr>
<td>Dorothea</td>
<td>800</td>
<td>10000</td>
</tr>
</tbody>
</table>
Table 2: The maximum classification accuracy achieved (with the corresponding number of selected features) along the feature selection path. Generalized \( l_p \)-KTA achieves significantly higher accuracies as compared to state-of-the-art KTA and \( l_p \geq 1 \)-MKL formulations as well as leading feature selection techniques such as BAHSIC. The table reports mean results averaged over 5-fold cross validation (see the supplementary material for standard deviations). '-' denote results where the data set was too large for the feature selection algorithm to generate results. 

<table>
<thead>
<tr>
<th>Data set &amp; kernel</th>
<th>Gen ( l_p )-KTA (RBF)</th>
<th>Gen ( l_p )-KTA (Linear)</th>
<th>Inf. Gain</th>
<th>Chi-Square</th>
<th>Fisher Score</th>
<th>mRMR</th>
<th>ReliefF</th>
<th>Spectrum</th>
<th>Gini Index</th>
<th>K.-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arcene</td>
<td>73.40 (16)</td>
<td>92.00 (9999)</td>
<td>72.00 (110)</td>
<td>71.20 (120)</td>
<td>66.20 (65)</td>
<td>68.20 (60)</td>
<td>68.40 (170)</td>
<td>64.60 (185)</td>
<td>64.20 (95)</td>
<td></td>
</tr>
<tr>
<td>Madelon</td>
<td>62.04 (12)</td>
<td>85.16 (9999)</td>
<td>61.63 (5)</td>
<td>61.69 (10)</td>
<td>61.47 (10)</td>
<td>61.87 (5)</td>
<td>62.06 (15)</td>
<td>59.43 (25)</td>
<td>55.04 (65)</td>
<td></td>
</tr>
<tr>
<td>Relathe</td>
<td>88.39 (190)</td>
<td>85.16 (9999)</td>
<td>84.39 (190)</td>
<td>85.35 (180)</td>
<td>83.35 (180)</td>
<td>75.01 (60)</td>
<td>77.01 (200)</td>
<td>69.90 (175)</td>
<td>70.97 (200)</td>
<td></td>
</tr>
<tr>
<td>Pcmac</td>
<td>89.88 (180)</td>
<td>85.16 (9999)</td>
<td>88.88 (180)</td>
<td>88.24 (180)</td>
<td>88.24 (180)</td>
<td>83.34 (145)</td>
<td>80.76 (200)</td>
<td>66.67 (185)</td>
<td>65.20 (200)</td>
<td></td>
</tr>
<tr>
<td>Basehock</td>
<td>94.76 (189)</td>
<td>85.16 (9999)</td>
<td>94.76 (189)</td>
<td>94.76 (189)</td>
<td>94.61 (200)</td>
<td>88.88 (70)</td>
<td>86.05 (200)</td>
<td>69.79 (200)</td>
<td>70.37 (200)</td>
<td></td>
</tr>
<tr>
<td>Dorothea</td>
<td>93.60 (12)</td>
<td>85.16 (9999)</td>
<td>93.60 (12)</td>
<td>95.28 (160)</td>
<td>86.30 (20)</td>
<td>93.33 (35)</td>
<td>93.33 (20)</td>
<td>90.28 (140)</td>
<td>90.08 (65)</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: The maximum classification accuracy achieved (with the corresponding number of selected features) along the feature selection path. In keeping with the ASU experimental protocol, all algorithms are restricted to selecting at most 200 features and are allowed to train on only half the data. Generalized \( l_p \)-KTA (RBF) outperforms all the linear techniques and this demonstrates the advantages of non-linear feature selection. Amongst the linear methods, our proposed method with linear features is the best in general.

<table>
<thead>
<tr>
<th>Data set &amp; kernel</th>
<th>Gen ( l_p )-KTA (RBF)</th>
<th>Gen ( l_p )-KTA (Linear)</th>
<th>Inf. Gain</th>
<th>Chi-Square</th>
<th>Fisher Score</th>
<th>mRMR</th>
<th>ReliefF</th>
<th>Spectrum</th>
<th>Gini Index</th>
<th>K.-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arcene</td>
<td>73.40 (16)</td>
<td>92.00 (9999)</td>
<td>72.00 (110)</td>
<td>71.20 (120)</td>
<td>66.20 (65)</td>
<td>68.20 (60)</td>
<td>68.40 (170)</td>
<td>64.60 (185)</td>
<td>64.20 (95)</td>
<td></td>
</tr>
<tr>
<td>Madelon</td>
<td>62.04 (12)</td>
<td>85.16 (9999)</td>
<td>61.63 (5)</td>
<td>61.69 (10)</td>
<td>61.47 (10)</td>
<td>61.87 (5)</td>
<td>62.06 (15)</td>
<td>59.43 (25)</td>
<td>55.04 (65)</td>
<td></td>
</tr>
<tr>
<td>Relathe</td>
<td>88.39 (190)</td>
<td>85.16 (9999)</td>
<td>84.39 (190)</td>
<td>85.35 (180)</td>
<td>83.35 (180)</td>
<td>75.01 (60)</td>
<td>77.01 (200)</td>
<td>69.90 (175)</td>
<td>70.97 (200)</td>
<td></td>
</tr>
<tr>
<td>Pcmac</td>
<td>89.88 (180)</td>
<td>85.16 (9999)</td>
<td>88.88 (180)</td>
<td>88.24 (180)</td>
<td>88.24 (180)</td>
<td>83.34 (145)</td>
<td>80.76 (200)</td>
<td>66.67 (185)</td>
<td>65.20 (200)</td>
<td></td>
</tr>
<tr>
<td>Basehock</td>
<td>94.76 (189)</td>
<td>85.16 (9999)</td>
<td>94.76 (189)</td>
<td>94.76 (189)</td>
<td>94.61 (200)</td>
<td>88.88 (70)</td>
<td>86.05 (200)</td>
<td>69.79 (200)</td>
<td>70.37 (200)</td>
<td></td>
</tr>
<tr>
<td>Dorothea</td>
<td>93.60 (12)</td>
<td>85.16 (9999)</td>
<td>93.60 (12)</td>
<td>95.28 (160)</td>
<td>86.30 (20)</td>
<td>93.33 (35)</td>
<td>93.33 (20)</td>
<td>90.28 (140)</td>
<td>90.08 (65)</td>
<td></td>
</tr>
</tbody>
</table>

6. Experiments

We carry out experiments to determine both the classification accuracy as well as the training time of the proposed Generalized \( l_p \)-KTA over the feature selection path.

Data sets & kernels: We present results on a variety of feature selection data sets taken from the NIPS 2003 Feature Selection Challenge (Guyon et al., 2006) and the ASU Feature Selection Repository (Zhao et al., 2010). Table 2 lists the number of instances and features in each data set. Unless otherwise stated, results are averaged via 5-fold cross-validation (standard deviations are reported in the supplementary material for lack of space). We define an RBF kernel per feature as our base kernels and center and trace normalize them as recommended in (Cortes et al., 2012).

Baseline techniques: We compare the proposed approach to a number of baseline techniques including state-of-the-art Centered KTA formulation (Cortes et al., 2012), highly optimized \( l_p \)-MKL techniques such as SMO-MKL (Vishwanathan et al., 2010), leading feature selection methods such as BAHSIC (Song et al., 2012) as well as path following approaches for \( l_1 \)-MKL (PF-\( l_1 \)-MKL) (Bach et al., 2004). While evaluating classification accuracy we also compare to the uniform kernel combination baseline (\( \eta_i = 1/r \forall i \)) referred to as Uniform as well as path following linear feature selection approaches (PF-\( l_1 \)-SVM) (Zhu et al., 2003). We also compare our classification accuracy to eight other state-of-the-art feature selection techniques whose details are given in Zhao et al. (2010). The final classification results for all the feature selection algorithms are obtained using an SVM with the kernel computed as

\[ k = \sum_{i=1}^{p} \eta_i k_i, \]

where \( \eta_i \) are the feature weights provided by the algorithm. While evaluating computational cost, we also compare the cost of our proposed approximate first order predictor-corrector algorithm to the exact path following algorithm.

Parameter settings: For the Generalized \( l_p \)-KTA formulation we set \( \lambda_1, \lambda_2 \) and the SVM misclassification penalty \( C \) by 5-fold cross-validation while varying \( p \) from 2 to 1 in decrements of 0.01. We use the squared Euclidean distance as the Bregman divergence with \( x_0 = 0 \). BAHSIC, PF-\( l_1 \)-MKL, and PF-\( l_1 \)-SVM are parameter free when computing the feature selection path. The parameters of the other techniques were also set via extensive cross-validation except for the computationally intensive SMO-MKL where this was feasible only for the smaller data sets. On the
larger data sets, we follow the SMO-MKL’s authors’ experimental protocol and fix $\lambda = 1$ and validate over $C$ with $p \in \{1.01, 1.33, 1.66, 2\}$. As in the case of previous $l_p$-MKL algorithms (Vishwanathan et al., 2010; Orabona & Jie, 2011; Orabona et al., 2012; Jain et al., 2012), we employ the common strategy of thresholding to obtain sparse solutions for the proposed Generalized $l_p$-KTA.

**Results:** Table 2 lists the maximum classification accuracy achieved along the entire feature selection path and the corresponding number of selected features (i.e. corresponding to the best oracle results on the test set). Our proposed Generalized $l_p$-KTA formulation gets significantly higher classification accuracies than all competing methods. For instance, on Arcene data set, Generalized $l_p$-KTA achieves a classification accuracy which is 10% higher than the closest competing method. Table 3 presents results obtained following the ASU experimental protocol where only 50% of the data is used for training and the number of selected features is restricted to be less than 200. This tests the capabilities of feature selection algorithms under the demanding conditions of both limited training data and limited prediction budget. As can be seen, Gen $l_p$-KTA (RBF) clearly outperforms all the linear methods thereby demonstrating the power of non-linear feature selection which is the focus of our paper. Amongst all the linear feature selection methods, our proposed Gen $l_p$-KTA (Linear) is the best in general. It is the best method on 3 data sets – on Relathe it is better than the second best method by 4%, on Arcene by 1.4% and on Dorothea by 0.27%. On 2 data sets it is the second best method – lagging behind the best method on Madelon by 0.02% and on Pcmac by 0.11%. These results demonstrate that the Generalized $l_p$-KTA formulation can lead to better results not only as compared to other KTA and $l_p$-MKL formulations but also as compared to leading feature selection techniques.

Table 4 assesses the cost of computing the feature selection path. Note that algorithms such as BAHISTIC compute the path a feature at a time while other algorithms, such PF-$l_1$-MKL, have a much sparser sampling of the path (in the limit Centered KTA produces only a single point on the path). Therefore, for a fair comparison, we report the total time taken by each algorithm divided by the number of points generated by it along the path. This measures the average time taken by each algorithm to generate a point along the path. Table 4 shows that the proposed approximate first order predictor-correction algorithm can be up to 878 times faster than competing techniques. Furthermore, on the largest data set, Dorothea with $10^5$ features, Centered-KTA, SMO-MKL and PF-$l_1$-MKL were not able to generate even a single point on the path whereas our algorithm was able to generate the entire path in 21 minutes using pre-computed kernels and 36 minutes using kernels computed on the fly. BAHISTIC generates the path by adding a single feature at each iteration and was able to generate the initial path segment up to 500 features but at a high computational cost of more than 3 hours. All experiments were carried out on a standard 2.40 GHz Intel Xeon desktop with 32 GB of RAM. Finally, we note that our approximate path following algorithm was also found to be faster than the exact path following algorithm implemented with first order predictor or with warm restarts without exploiting juncture points and eliminating features from the optimization. The speedups against both the baselines varied with more than 3 times on Madelon and almost 4 times on Dorothea while the maximum relative deviation from the exact path was only $7.3 \times 10^{-6}$ and $3.6 \times 10^{-4}$ respectively. Figure 1(a) shows the feature selection path as computed by our approximate algorithm on all the data sets.

### 7. Conclusions

We developed an efficient $p$-norm path following algorithm for non-linear feature selection in this paper. Our starting point was a novel conjecture that the number of selected features, and the feature weights themselves, vary monotonically with $p$ in $l_p$-MKL formulations. We proposed a Generalized $l_p$-KTA formulation and proved that the conjecture holds for this formulation but not for other popular formulations such as the square loss based $l_p$-MKL. The
monotonicity property of the Generalized \( l_p \)-KTA formulation allows us to eliminate features whose weight goes below a certain threshold from the optimization for large intervals of \( p \) leading to efficient path following. It was theoretically and empirically demonstrated that this resulted in only a minor deviation from the exact path. Experiments also revealed that our proposed formulation could yield significantly higher classification accuracies as compared to state-of-the-art MKL and feature selection methods (by as much as 10% in some cases) and be many times faster than them. All in all, we were able to effortlessly compute the entire path for problems involving a hundred thousand features which are well beyond the scaling capabilities of existing path following techniques.

Acknowledgments

We thank the anonymous reviewers for the valuable comments.

A. Analysis of \( l_p \)-MKL with Square Loss

In this section, we analyze the proposed conjecture in the context of the \( l_p \)-MKL formulation for the ridge regression (Cortes et al., 2009a):

\[
\min_{\eta \geq 0} \max_{\alpha \in \mathbb{R}^m} y^\top \alpha - \frac{1}{2} \alpha^\top Q_\eta \alpha + \lambda_2 \sum_{i=1}^r \eta_i^p
\]

(7)

where \( Q_\eta = \sum_i \eta_i K_i + \frac{I}{\lambda_1} \), \( I \) is the \( m \times m \) identity matrix and \( \lambda_1, \lambda_2 > 0 \) are the regularization parameters. Firstly, we consider a setting employed in (Lanckriet et al., 2004; Cristianini et al., 2001; 2000), which leads to the selection of a low-dimensional subspace: unit rank base kernels \( K_i = u_i u_i^\top \) s.t. trace(\( K_i \)) = 1 and \( \langle K_i, K_j \rangle = 0 \) \( \forall i \neq j \). The following theorem holds in this setting:

**Theorem 3.** Let \( \eta_i^*(p) \) denote the optimal weight corresponding to the \( i \)-th kernel in (7) at \( p \). Given \( \eta_i^*(p') \), the following holds as \( p \) decreases from \( p' \) to unity: (1) \( \eta_i^*(p') \) decreases monotonically whenever \( \eta_i^*(p') < e^{-1} \); (2) \( \eta_i^*(p') \) increases monotonically whenever \( \eta_i^*(p') > e^{-\frac{1}{p'}} \)

The proof involves deriving and analyzing the \( d\eta_i^*(p)/dp \) term and is provided in the supplementary material.

Needless to say, the above theorem implies that the conjecture is true in the case mentioned above. We now present an interesting example where the conjecture does not hold with \( \epsilon = 0 \):

**Theorem 4.** Consider the regression formulation in (7) with two given base kernels of unit rank and unit trace: \( K_1 \) and \( K_2 \). Additionally, let \( y^\top K_2 K_1 y > y^\top K_1 y > 0 \). Then for some \( \lambda_1, \lambda_2 \), \( \exists p' > 1 \) such that \( \eta_i = 0 \) if and only if \( p = p' \).

The proof involves deriving and analyzing the necessary conditions for the optimal value of a kernel weight in (7) to be zero and is detailed in the supplementary material.

The above theorem shows a case in which the kernel weight, after attaining the lowest feasible value (zero), at \( p = p' \), grows as \( p \) further decreases from \( p' \). Figure 2 shows an instance from a real world data set (Parkinson disease data set from the UCI Repository) where the optimal kernel weight in (7) grows after attaining zero. We can observe that at around \( p = 1.5 \), the optimal kernel weight corresponding to Feature 1 is zero, and it again starts growing as \( p \) is further decreased. This observation shows that the conjecture is not universally true for low values of \( \epsilon \). However, in the same setting as in Theorem 4, the conjecture may be true for some other \( \epsilon > 0 \). The proposed algorithm is usable wherever such an \( \epsilon \) is small\(^1\).

![Figure 2: \( \eta_1^p \) decrease as \( p \) is decreased from 2, attains zero at around \( p = 1.5 \) and again grows as \( p \) decreases further. This example proves that the monotonicity conjecture does not universally hold for (7) when \( \epsilon = 0 \).](image)

In summary, the proposed conjecture is itself non-trivial and requires careful analysis for different \( l_p \)-MKL formulations. Interestingly, in case of the proposed Generalized \( l_p \)-KTA, it holds for small enough tolerance, \( \epsilon < e^{-1} \).

**References**


\(^1\)Interestingly, in our initial simulations on benchmark data sets, we found that many existing \( l_p \)-MKL formulations do satisfy the conjecture at a reasonably low \( \epsilon > 0 \). We postpone theoretical analysis of this interesting and open question to future.
On \( p \)-norm Path Following in Multiple Kernel Learning for Non-linear Feature Selection


