Multiple Indefinite Kernel Learning for Feature Selection

Hui Xue[1,2,*], **Yu Song**[1,2] and **Hai-Ming Xu**[1,2]
, [1]School of Computer Science and Engineering, Southeast University, Nanjing, 210096, China
[2]MOE Key Laboratory of Computer Network and Information Integration (Southeast University), China
{hxue, song_yu, heimingx}@seu.edu.cn

Abstract

Multiple kernel learning for feature selection (MKL-FS) utilizes kernels to explore complex properties of features and performs better in embedded methods. However, the kernels in MKL-FS are generally limited to be positive definite. In fact, indefinite kernels often emerge in actual applications and can achieve better empirical performance. But due to the non-convexity of indefinite kernels, existing MKL-FS methods are usually inapplicable and the corresponding research is also relatively little. In this paper, we propose a novel multiple indefinite kernel feature selection method (MIK-FS) based on the primal framework of indefinite kernel support vector machine (IKSVM), which applies an indefinite base kernel for each feature and then exerts an l_1-norm constraint on kernel combination coefficients to select features automatically. A two-stage algorithm is further presented to optimize the coefficients of IKSVM and kernel combination alternately. In the algorithm, we reformulate the non-convex optimization problem of primal IKSVM as a difference of convex functions (DC) programming and transform the non-convex problem into a convex one with the affine minorization approximation. Experiments on real-world datasets demonstrate that MIK-FS is superior to some related state-of-the-art methods in both feature selection and classification performance.

1 Introduction

Feature selection is an important problem in machine learning and has been studied extensively, which aims to select only a subset of relevant features in order to speed up learning process, eliminate some noises and provide better model interpretability [Chandrashekar and Sahin, 2014]. In general, feature selection methods can be divided into three categories: "*filter*" which ranks the features according to some discrimination measures independent of learning algorithms, "*wrapper*" which evaluates the features by learning algorithms, and "*embedded*" which embeds feature selection into learning process.

*Corresponding author.

In the past few years, multiple kernel learning for feature selection (MKL-FS) has attracted more attention in embedded methods. By means of MKL, MKL-FS can not only uncover complicated properties of the features effectively, but also convert the selection of the features into the learning on a sparse combination of multiple kernels [Xu *et al.*, 2009; Varma and Babu, 2009]. Chen et al. treated the feature selection problem of gene expression data as an ordinary multiple parameter learning problem based on multiple kernel support vector machine [Chen *et al.*, 2007]. Dileep and Sekhar further applied this method in image categorization tasks and showed its superiority over principal component analysis (PCA) [Dileep and Sekhar, 2009]. Varma and Babu proposed a more generalized MKL scheme for feature selection where the combination of base kernels can be generalized to be nonlinear [Varma and Babu, 2009]. Xu et al. presented a non-monotonic feature selection method to select a specific number of features and the corresponding combinatorial optimization problem was approximated by an MKL problem [Xu *et al.*, 2009]. Tan et al. focused on sparse support vector machines (SVM) with l_0-norm whose convex relaxation can be further formulated as an MKL problem, where each kernel corresponds to a sparse feature subset [Tan *et al.*, 2010]. Yamada et al. proposed an alternative feature-wise kernelized Lasso to capture nonlinear input-output dependency and select informative features according to the kernel coefficients [Yamada *et al.*, 2014].

However, MKL-FS methods usually require that base kernels should be positive definite (PD) and satisfy the Mercer's condition in order to obtain a bi-convex optimization problem. In fact, standard PD kernels are inapplicable in many practical situations. For example, kernels obtained from similarity measures often violate Mercer's conditions and are not positive definite [Saigo *et al.*, 2004; Chen *et al.*, 2009]. On the contrary, indefinite kernels have played an increasingly important role in machine learning and shown much better performance in some scenarios than PD kernels. Kowalski et al. focused on using mixed norm regularization to reach better sparsity [Kowalski *et al.*, 2009]. Liwicki et al. applied an indefinite robust gradient-based kernel in an incremental kernel principal component analysis (KPCA) algorithm for visual tracking and achieved more efficient results [Liwicki *et al.*, 2012]. Xue et al. integrated additional problem-specific prior knowledge in the construction of indef-

inite kernels and presented its superiority in supervised and semi-supervised classification [Xue and Chen, 2014]. Xu et al. further solved single indefinite kernel SVM with difference of convex functions (DC) programming. [Xu *et al.*, 2017].

Recently, indefinite kernels have been widely studied in dimensionality reduction. Liu utilized an indefinite fractional power polynomial kernel in KPCA in face recognition, which can achieve higher recognition accuracies than the KPCA using PD polynomial kernels [Liu, 2004]. Haasdonk and Pekalska proposed indefinite kernel discriminant analysis (IKDA) to extend traditional KDA in indefinite kernel scenarios [Haasdonk and Pekalska, 2010]. Huang et al. addressed PCA with indefinite kernels (IKPCA) in the framework of least squares support vector machine and then gave IKPCA a feature space interpretation [Huang *et al.*, 2016]. However, the problem of indefinite kernels applied in feature selection has got relatively little research. Furthermore, due to the non-convexity of indefinite kernels, almost all existing MKL-FS methods can not be used.

In this paper, we propose a novel multiple indefinite kernel feature selection method (MIK-FS) based on the primal framework of indefinite kernel support vector machine (IKSVM) in embedded method scenarios. MIK-FS uses an indefinite base kernel to represent each feature respectively and an l_1-norm constraint is then enforced on kernel combination coefficients in order to select features naturally. A two-stage algorithm is further presented to optimize the coefficients of IKSVM and kernel combination alternately. Concretely, when the kernel combination coefficients are fixed, the non-convex primal problem of IKSVM is reformulated as a difference of convex functions (DC) programming and then converted into a convex one by the affine minorization approximation. Once the coefficients of IKSVM have been solved, kernel combination coefficients can be optimized by projected gradient descent. We further prove that the value of the objective function in MIK-FS is strictly monotonic decreasing with each iteration in the algorithm. Experimental results on real-world datasets have shown that MIK-FS outperforms some related methods in terms of feature selection and classification.

2 Related Work

Given a training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, where $\boldsymbol{x}_i \in \mathbb{R}^M$ is a training sample and $y_i \in \{-1, +1\}$ is the corresponding class label. MKL-FS firstly applies a base kernel k_m on each feature of the samples and then combines these kernels into a kernel combination including two ways: additive and multiplicative combinations.

Most MKL-FS methods integrated the kernels into an additive combination [Dileep and Sekhar, 2009]:

$$k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \sum_{m=1}^{M} d_m k_m\left(x_{i,m}, x_{j,m}\right), d_m \geq 0 \tag{1}$$

where $x_{i,m}$ denotes the mth feature of $\boldsymbol{x}_i$ and d_m represents the coefficient of the kernel k_m.

Varma and Babu further extended the linear combination Eq. (1) into a nonlinear product form [Varma and Babu, 2009]:

$$k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \prod_{m=1}^{M} d_m k_m\left(x_{i,m}, x_{j,m}\right), d_m \geq 0 \tag{2}$$

MKL-FS aims to learn a sparse combination of the kernels so that the feature can be selected naturally. Consequently, SVM-based MKL-FS methods further embeds the kernel combination into the dual problem of SVM, which can learn the coefficients of SVM and the combination simultaneously:

$$\begin{aligned} \min_{\boldsymbol{\alpha}, \boldsymbol{d}} \quad & \tfrac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) + \lambda \|\boldsymbol{d}\|_1 - \sum_{i=1}^{n} \alpha_i \\ s.t. \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C, i = 1, \cdots, n \\ & d_m \geq 0, m = 1, \cdots, M \end{aligned} \tag{3}$$

where $\boldsymbol{d}$ is the kernel combination coefficients. An l_1-norm regularizer is used on $\boldsymbol{d}$ to obtain a spares solution.

When the kernels are PD, MKL-FS can select the informative features while training a good classifier SVM. However, when the kernels become indefinite, it more likely leads to bad classification performance if we directly embed the combination into the dual problem of indefinite kernel SVM (IKSVM). In the case of indefinite kernels, the primal and dual problems of IKSVM are both non-convex. Consequently, there is a dual gap between the two problems and their solutions are not equivalent [Xu *et al.*, 2017]. As a result, the selected features and the classifier's coefficients learned from the dual IKSVM are more likely not beneficial for the subsequent classification. In other words, it is more reasonable that the indefinite kernel combination should be embedded into the primal IKSVM. However, existing SVM-based MKL-FS methods mostly focus on the dual problem and will fail in solving the non-convex indefinite kernel problems.

3 MIK-FS

In this section, we will present our multiple indefinite kernel feature selection method (MIK-FS). In MIK-FS method, each feature of the samples is characterized by an indefinite kernel and the global optimization problem would come to be non-convex. In order to avoid suffering from the dual gap, we attempt to focus on the primal problem of IKSVM and its objective function can be formulated as an unconstrained optimization problem:

$$\min_{\boldsymbol{f} \in \mathcal{K}, b} \quad \lambda \left\langle \boldsymbol{f}, \boldsymbol{f} \right\rangle_{\mathcal{K}} + \sum_{i=1}^{n} L(y_i, \boldsymbol{f}(\boldsymbol{x}_i) + b) \tag{4}$$

where $\mathcal{K}$ represents a Reproducing Kernel Kreĭn Space (RKKS) and L is a loss function. According to [Ong *et al.*, 2004], the Representer Theorem still holds in RKKS and the solution to Eq. (4) can be expressed as

$$\boldsymbol{f}^* = \sum_{i=1}^{n} \beta_i k(\boldsymbol{x}_i, \cdot) \tag{5}$$

where $\beta_i \in \mathbb{R}$, $k(\cdot, \cdot)$ is a kernel function in RKKS and the corresponding kernel matrix can be indefinite. Combining Eqs. (4) and (5), the kernelized primal form of IKSVM can be formulated as:

$$\min_{\boldsymbol{\beta}, b} \quad \lambda \sum_{i,j=1}^{n} \beta_i \beta_j k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) + \sum_{i=1}^{n} L(y_i, \sum_{j=1}^{n} \beta_j k\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) + b) \tag{6}$$

We further embed the indefinite kernel combination of Eq. (1) into Eq. (6) to obtain our MIK-FS. An l_1-norm regularizer

is added to get sparse kernel combination coefficients. The corresponding model of MIK-FS is:

$$\begin{aligned} \min_{\boldsymbol{\beta},b,\boldsymbol{d}} \quad & \lambda_1 \boldsymbol{\beta}^T \boldsymbol{K} \boldsymbol{\beta} + \lambda_2 \|\boldsymbol{d}\|_1 + \sum_{i=1}^{n} L(y_i, \boldsymbol{K}^i \boldsymbol{\beta} + b) \\ s.t \quad & d_m \geq 0, m = 1, \cdots, M \end{aligned} \tag{7}$$

where $\boldsymbol{K} = \sum_{m=1}^{M} d_m \boldsymbol{K}_m$ is an indefinite kernel matrix derived from associated kernel function $\boldsymbol{K}_{ij} = k(\boldsymbol{x_i}, \boldsymbol{x_j})$ described in Eq. (1) and $\boldsymbol{K}_m$ is derived from kernel function k_m. $\boldsymbol{K}^i$ represents the ith row of $\boldsymbol{K}$.

Furthermore, in order to ensure the MIK-FS model is continuously differentiable, we select the smooth quadratic hinge loss function as $L(\cdot)$ and the optimization problem becomes

$$\begin{aligned} \min_{\boldsymbol{\beta},b,\boldsymbol{d}} \quad & \lambda_1 \boldsymbol{\beta}^T \boldsymbol{K} \boldsymbol{\beta} + \lambda_2 \|\boldsymbol{d}\|_1 + \sum_{i=1}^{n} \max\left(0, 1 - y_i(\boldsymbol{K}^i \boldsymbol{\beta} + b)\right)^2 \\ s.t \quad & d_m \geq 0, m = 1, \cdots, M \end{aligned} \tag{8}$$

In Eq. (8), the first term is the regularizer related to hypothesis $\boldsymbol{f}$. Specially, it is worth noting that $\boldsymbol{\beta}$ is unconstrained which is different to the Lagrange multiplier $\boldsymbol{\alpha}$ in Eq. (3). The second term is the l_1-norm regularizer related to $\boldsymbol{d}$. If the coefficient d_m equals to zero, it means that the corresponding feature has no effect on the classification and can be discarded.

Compared to traditional MKL-FS methods, the proposed MIK-FS has two advantages. Firstly, the kernels used in MIK-FS can actually involve both PD and indefinite kernels. Consequently, MIK-FS is a broader method to exploit more generalized kernels. Secondly, we construct MIK-FS on the the primal form of IKSVM and avoid the dual gap in the non-convex optimization problem effectively.

4 Optimization Algorithm

We adopt a two-stage algorithm to optimize the coefficients $(\boldsymbol{\beta}, b)$ of IKSVM and $\boldsymbol{d}$ of kernel combination alternately. The complete algorithm is described in Algorithm 1. We will introduce the two stages in detail in the following subsections.

4.1 Solving $(\boldsymbol{\beta}, b)$

Firstly, we denote the objective function of MIK-FS as

$$\begin{aligned} F(\boldsymbol{\beta}, b, \boldsymbol{d}) = & \lambda_1 \boldsymbol{\beta}^T \boldsymbol{K} \boldsymbol{\beta} + \lambda_2 \|\boldsymbol{d}\|_1 \\ & + \sum_{i=1}^{n} \max\left(0, 1 - y_i(\boldsymbol{K}^i \boldsymbol{\beta} + b)\right)^2 \end{aligned} \tag{9}$$

When the coefficients $\boldsymbol{d}$ are fixed, Eq. (9) degenerates into a non-convex problem of IKSVM with a single indefinite kernel. That is

$$f(\boldsymbol{\beta}, b) = \lambda_1 \boldsymbol{\beta}^T \boldsymbol{K} \boldsymbol{\beta} + L + \text{Constant}_{\boldsymbol{d}} \tag{10}$$

where $L = \sum_{i=1}^{n} \max\left(0, 1 - y_i(\boldsymbol{K}^i \boldsymbol{\beta} + b)\right)^2$ and $\text{Constant}_{\boldsymbol{d}} = \lambda_2 \|\boldsymbol{d}\|_1$ is a constant.

According to [Xu *et al.*, 2017], the non-convex problem of Eq (10) can be reformulated as a DC programming [Tao and An, 1997; Dinh and Le Thi, 2014] equivalently due to the favorable property of the spectra for indefinite kernel matrices. Concretely, the objective function can be decomposed as

$$\begin{aligned} & f(\boldsymbol{\beta}, b) = g(\boldsymbol{\beta}, b) - h(\boldsymbol{\beta}, b) \\ \text{with} \quad & g(\boldsymbol{\beta}, b) = \lambda_1 \boldsymbol{\beta}^T (\rho \boldsymbol{I}) \boldsymbol{\beta} + L + \text{Constant}_{\boldsymbol{d}} \\ & h(\boldsymbol{\beta}, b) = \lambda_1 \boldsymbol{\beta}^T (\rho \boldsymbol{I} - \boldsymbol{K}) \boldsymbol{\beta} \end{aligned} \tag{11}$$

where the positive number ρ satisfies the condition: $\rho \geq \eta$ and the number η is the maximum eigenvalue of the kernel matrix $\boldsymbol{K}$. As a result, the functions $g(\boldsymbol{\beta}, b)$ and $h(\boldsymbol{\beta}, b)$ can both be guaranteed to be convex.

Then, the conjugate dual problem of Eq. (11) can be formulated as $f^*(\overline{\boldsymbol{\beta}}, \overline{b}) = h^*(\overline{\boldsymbol{\beta}}, \overline{b}) - g^*(\overline{\boldsymbol{\beta}}, \overline{b})$, where $g^*(\overline{\boldsymbol{\beta}}, \overline{b}) = \sup\{\langle(\overline{\boldsymbol{\beta}}, \overline{b}), (\boldsymbol{\beta}, b)\rangle - g(\boldsymbol{\beta}, b), (\boldsymbol{\beta}, b) \in \mathbb{R}^n \times \mathbb{R}\}$ is the conjugate function of g and h^* is the conjugate function of h . And the relationship between variables $\{(\boldsymbol{\beta}, b)\}$ and $\{(\overline{\boldsymbol{\beta}}, \overline{b})\}$ is

$$(\overline{\boldsymbol{\beta}}, \overline{b}) \in \partial h(\boldsymbol{\beta}, b), \ (\boldsymbol{\beta}, b) \in \partial g^*(\overline{\boldsymbol{\beta}}, \overline{b}) \tag{12}$$

Using Eq. (12), we can approximate the functions h and g^* with their affine minorization at points $(\boldsymbol{\beta}^k, b^k)$ and $(\overline{\boldsymbol{\beta}}^k, \overline{b}^k)$ respectively

$$\begin{aligned} h(\boldsymbol{\beta}, b) &= h(\boldsymbol{\beta}^k, b^k) + \left\langle \boldsymbol{\beta} - \boldsymbol{\beta}^k, \overline{\boldsymbol{\beta}}^k \right\rangle \\ g^*(\overline{\boldsymbol{\beta}}, \overline{b}) &= g^*(\overline{\boldsymbol{\beta}}^k, \overline{b}^k) + \left\langle \overline{\boldsymbol{\beta}} - \overline{\boldsymbol{\beta}}^k, \boldsymbol{\beta}^{k+1} \right\rangle \end{aligned} \tag{13}$$

where $(\overline{\boldsymbol{\beta}}^k, \overline{b}^k) \in \partial h(\boldsymbol{\beta}^k, b^k)$ and $(\boldsymbol{\beta}^{k+1}, b^{k+1}) \in \partial g^*(\overline{\boldsymbol{\beta}}^k, \overline{b}^k)$. Embedding Eq. (13) into the primal and conjugate dual problems of IKSVM, we have

$$\begin{aligned} & \arg\min_{(\boldsymbol{\beta},b)} \{g(\boldsymbol{\beta}, b) - h(\boldsymbol{\beta}, b)\} \\ & = \arg\min\{(\boldsymbol{\beta}^{k+1}, b^{k+1}) : g(\boldsymbol{\beta}, b) - \left\langle \boldsymbol{\beta}, \overline{\boldsymbol{\beta}}^k \right\rangle\} \\ & \arg\min_{(\overline{\boldsymbol{\beta}},\overline{b})} \{h^*(\overline{\boldsymbol{\beta}}, \overline{b}) - g^*(\overline{\boldsymbol{\beta}}, \overline{b})\} \\ & = \arg\min\{(\overline{\boldsymbol{\beta}}^{k+1}, \overline{b}^{k+1}) : h^*(\overline{\boldsymbol{\beta}}, \overline{b}) - \left\langle \overline{\boldsymbol{\beta}}, \boldsymbol{\beta}^{k+1} \right\rangle\} \end{aligned} \tag{14}$$

Algorithm 1 MIK-FS

Inputs:
T: the maximum number of iterations
λ_1, λ_2: the regularization parameters
ϵ: the tolerance value for convergence

Process:
set $t = 0$, initialize $\boldsymbol{d}_t, \text{obj}_t$;
while $(t < T)$ **do**
  set $t = t + 1$;
  fix $\boldsymbol{d}_{t-1}$ and solve Eq. (8) to obtain $(\boldsymbol{\beta}_t, b_t)$;
  fix $(\boldsymbol{\beta}_t, b_t)$ and solve Eq. (8) to obtain a solution $\boldsymbol{d}_t$;
  calculate the value of objective function obj_t;
  if $|\text{obj}_t - \text{obj}_{t-1}| \leq \epsilon$ **then**
    MIK-FS converges and break;
  end if
end while
return $\boldsymbol{\beta}_t, b_t, \boldsymbol{d}_t$;

So these two problems of IKSVM become convex after the transformation in Eq. (14). We can construct two sequences $\{(\boldsymbol{\beta}, b)\}$ and $\{(\overline{\boldsymbol{\beta}}, \overline{b})\}$ by solving Eq. (14).

Furthermore, in order to simplify the solving process [Dinh and Le Thi, 2014], we directly compute $\{(\overline{\boldsymbol{\beta}}, \overline{b})\}$ in the way that $\{(\overline{\boldsymbol{\beta}}^k, \overline{b}^k)\} \in \partial h(\boldsymbol{\beta}^k, b^k)$. Aa a result, the two sequences $\{(\boldsymbol{\beta}, b)\}$ and $\{(\overline{\boldsymbol{\beta}}, \overline{b})\}$ can be constructed as follows:

$$\begin{aligned} &(\overline{\boldsymbol{\beta}}^k, \overline{b}^k) \in \partial h(\boldsymbol{\beta}^k, b^k) \\ &(\boldsymbol{\beta}^{k+1}, b^{k+1}) = \arg\min\{g(\boldsymbol{\beta}, b) - \langle \boldsymbol{\beta}, \overline{\boldsymbol{\beta}}^k \rangle\} \end{aligned} \tag{15}$$

Since $g(\boldsymbol{\beta}, b)$ is a convex function and the related convex optimization problem in Eq. (15) can be easily solved. The detailed steps for solving $(\boldsymbol{\beta}, b)$ in Eq. (10) are described in Algorithm 2.

Algorithm 2 Primal IKSVM

Inputs:
T_1: the maximum number of iterations
ϵ: the tolerance value for convergence
$\boldsymbol{d}$: the kernel coefficients
$\boldsymbol{K}_1 \dots \boldsymbol{K}_M$: matrix obtained from features
δ: the offset for dc decomposition

Process:
1: calculate $\boldsymbol{K} = \sum_{m=1}^{M} d_m \boldsymbol{K}_m$;
2: calculate η, the maximum eigenvalue of $\boldsymbol{K}$;
3: set $\rho = |\eta| + \delta$;
4: set $k = 0$, initialize $\boldsymbol{\beta}^k$;
5: **while** $(k < T_1)$ **do**
6: calculate $\overline{\boldsymbol{\beta}}^k = \lambda_1(\rho \boldsymbol{I} - \boldsymbol{K})\boldsymbol{\beta}^k$;
7: calculate $(\boldsymbol{\beta}^{k+1}, b^{k+1}) = \text{argmin}\{g(\boldsymbol{\beta}, b) - \langle \boldsymbol{\beta}, \overline{\boldsymbol{\beta}}^k \rangle\}$;
8: set $\boldsymbol{\alpha}^{k+1} = (\boldsymbol{\beta}^{k+1}, b^{k+1})$ and $\boldsymbol{\alpha}^k = (\boldsymbol{\beta}^k, b^k)$;
9: **if** $\|\boldsymbol{\alpha}^{k+1} - \boldsymbol{\alpha}^k\|_2^2 \leq \epsilon$ **then**
10: the algorithm converges and break the loop;
11: **end if**
12: set $k = k + 1$;
13: **end while**
14: **return** $\boldsymbol{\beta}^k, b^k$;

Algorithm 2 firstly integrates the kernel matrixes $\boldsymbol{K}_m$ into an additive combination with coefficients $\boldsymbol{d}$ and performs eigenvalue decomposition on $\boldsymbol{K}$ to find the maximum eigenvalue for DC decomposition (Step 1-3). Within the loop, algorithm 2 calculates the solution of conjugate dual problem (Step 6) and then solves primal problem by approximating $h(\boldsymbol{\beta}, b)$ with its affine minorization (Step 7). Difference between two points obtained in adjacent iterations is calculated for measuring convergence (Step 8-11).

4.2 Solving $\boldsymbol{d}$

When $(\boldsymbol{\beta}, b)$ is fixed, Eq. (8) can be reformulated as:

$$\begin{aligned} \min_{\boldsymbol{d}} \quad & \boldsymbol{d}^T \boldsymbol{\gamma} + \sum_{i=1}^{n} \max\left(0, 1 - y_i\left(\boldsymbol{\theta}^i \boldsymbol{d} + b\right)\right)^2 \\ s.t. \quad & d_m \geq 0, m = 1, ..., M \end{aligned} \tag{16}$$

where $\boldsymbol{\gamma} = [\lambda_1 \boldsymbol{\beta}^T K_1 \boldsymbol{\beta} + \lambda_2, \dots, \lambda_1 \boldsymbol{\beta}^T K_M \boldsymbol{\beta} + \lambda_2]^T$, $\boldsymbol{\theta} = [K_1\boldsymbol{\beta}, \cdots, K_M\boldsymbol{\beta}]$ and $\boldsymbol{\theta}^i$ represents the ith row of $\boldsymbol{\theta}$. Eq. (16) can be solved by projected gradient descent (PGD). The gradient of $\boldsymbol{d}$ at d_m is:

$$\nabla_{d_m} = \gamma_m + \sum_{i \in SV} \left(1 - y_i\left(\boldsymbol{K}^i \boldsymbol{\beta} + b\right)\right)\left(-y_i K_m^i \boldsymbol{\beta}\right) \tag{17}$$

where $SV = \left\{\boldsymbol{x_i} \in X | y_i\left(\boldsymbol{K}^i \boldsymbol{\beta} + b\right) \leq 1\right\}$ is the set of support vectors in current iteration. PGD method for solving $\boldsymbol{d}$ is summarized in Algorithm 3.

Algorithm 3 PGD

Inputs:
T_2: the maximum number of iterations
ϵ: the tolerance value for convergence

Process:
1: set $k = 0$, initialize $\boldsymbol{d}^k$;
2: **while** $(k < T_2$ and $\|\nabla_{\boldsymbol{d}^k}\|_2^2 \geq \epsilon)$ **do**
3: calculate $\nabla_{\boldsymbol{d}^k}$ according to Eq. (17);
4: calculate step size α by Armijo rule ;
5: set $\boldsymbol{d}^{k+1} = \boldsymbol{d}^k - \alpha * \nabla_{\boldsymbol{d}^k}$;
6: set $\boldsymbol{d}^{k+1} = \max(\boldsymbol{d}^{k+1}, 0)$;
7: set $k = k + 1$;
8: **end while**
9: **return** $\boldsymbol{d}^k$;

Algorithm 3 calculates the gradient of Eq. (16) and the step size α based on the Armijo rule (Step 3-4). After that, $\boldsymbol{d}$ is updated and projected to feasible sets (Step 5-6).

4.3 Convergence Analysis

We further present a theoretical analysis for the convergence of MIK-FS. As mentioned above, MIK-FS is a two-stage algorithm and thus we will analyze the convergence of these two stages respectively. First of all, when the kernel combination coefficients $\boldsymbol{d}$ are fixed, MIK-FS can be formulated as a primal IKSVM problem. We solve it by DC programming which can guarantee that the objective function of IKSVM is monotonically decreasing [Xu *et al.*, 2017].

Proposition 1. *For the sequence* $\{\boldsymbol{\alpha}^k = (\boldsymbol{\beta}^k, b^k)\}$*, we have*

$$(g - h)(\boldsymbol{\beta}^k, b^k) - (g - h)(\boldsymbol{\beta}^{k+1}, b^{k+1}) \geq \tau \|d(\boldsymbol{\alpha})\|^2,$$

the equality holds if and only if $\tau\|d(\boldsymbol{\alpha})\|^2 = 0$*, where* τ *is a positive parameter to make functions* g *and* h *strongly convex.*

Furthermore, the local minimum would be obtained when $\|d(\boldsymbol{\alpha})\|^2 = 0$ is satisfied.

Then, when the coefficients $(\boldsymbol{\beta}, b)$ of IKSVM are fixed, MIK-FS is transformed into a convex problem to solve the kernel combination coefficients. Thus, we can obtain the following proposition for the whole algorithm MIK-FS.

Proposition 2. *For the sequence* $\{(\boldsymbol{\beta}^k, b^k, \boldsymbol{d}^k)\}$ *, we have*

$$F(\boldsymbol{\beta}^{k+1}, b^{k+1}, \boldsymbol{d}^{k+1}) \leq F(\boldsymbol{\beta}^k, b^k, \boldsymbol{d}^k),$$

that is, the objective function $F(\boldsymbol{\beta}^k, b^k, \boldsymbol{d}^k)$ *is strictly monotonic decreasing along the solution sequence.*

When $F(\boldsymbol{\beta}^k, b^k, \boldsymbol{d}^k) = F(\boldsymbol{\beta}^{k+1}, b^{k+1}, \boldsymbol{d}^{k+1})$ comes true, the algorithm MIK-FS can converge to a stationary point.

Table 1: Datasets description.

Datasets	#Num	#Feature
ALLAML	72	7129
Colon	62	2000
Gli_85	85	22283
Prostate_GE	102	5966
Central_Nervous_System	60	7129
Lung_Cancer	181	12533
Dbworld	64	4702
Isolet	120	617
Glioma	21	4434
Carcinom	34	9182

5 Experiments

We conduct a series of experiments on several real-world datasets to compare our MIK-FS to some related state-of-the-art algorithms.

5.1 Experimental Setup

We select ten datasets from three different repositories for experiments: (a) seven datasets from a feature selection repository[1], namely ALLAML, Colon, Gli_85, Prostate_GE, Isolet, Glioma, Carcinom; (b) two binary datasets from an online repository[2] of high-dimensional biomedical datasets, namely Central_Nervous_System, Lung_Cancer; (c) one dataset Dbworld from UCI Machine Learning Repository. Table 1 lists a brief description of these ten datasets, including the number of samples and the number of features in each sample. As we can see from the table, the number of features in the ten datasets are all very large. So there are more likely high redundancies among these features.

We randomly divide the samples into two non-overlapping training and testing sets which contain almost half of the samples in each class. The processes are repeated ten times to generate ten independent runs for each dataset and then the average results are reported. Since the three datasets Isolet, Glioma and Carcinom are designed for multi-class classification , we choose the first two classes.

In our experiments, we choose the indefinite sigmoid kernel $k = \tanh(a \cdot \boldsymbol{x_i}^T \boldsymbol{x_j} - r)$ as the base feature kernel in MIK-FS. Gaussian kernel is used for two MKL-FS methods. For all the algorithms, the regularization parameters and the kernel parameters are chosen from the set $\{10^{-2}, 10^{-1}, 1, 10^1, 10^2\}$. A feature is discarded if the corresponding kernel combination coefficient d_i is less than a small threshold, e.g., 10^{-5}.

We compare the proposed MIK-FS with the following algorithms:

- l_1-SVM [Bradley and Mangasarian, 1998]: SVM with l_1-norm regularizer .
- ElasticNet-SVM [Wang *et al.*, 2006]: SVM with mixed l_1-norm and l_2-norm regularizer.
- RFMKL [Dileep and Sekhar, 2009]: An SVM-based MKL-FS algorithm with additive kernel combination.
- GMKL [Varma and Babu, 2009]: An SVM-based MKL-FS algorithm with multiplicative kernel combination. .
- IKPCA [Huang *et al.*, 2016]: Kernel principal component analysis with indefinite kernels.

5.2 Experimental Results

Table 2 lists the average classification accuracies and the corresponding number of selected features in each compared algorithm on the ten datasets. The best results are highlighted in bold.

As shown in Table 2, l_1-SVM is simple and fast, but it performs poorly compared to other algorithms. Elasticnet-SVM outperforms l_1-SVM in terms of classification accuracies but tends to select too many features. GMKL achieves similar results with RFMKL on most datasets except that it performs worse than RFMKL on Gli_85, Central_Nervous_System and Lung_Cancer. MIK-FS excels GMKL and RFMKL on all datasets. The reduced dimensions in IKPCA are limited by the number of samples and it performs even worse than l_1-SVM on most datasets. Overall, our proposed MIK-FS achieves higher accuracies than other algorithms on almost all the datasets while selecting a smaller number of features.

Figure 1 shows the classification accuracies corresponding to the specified number of features of five embedded feature selection algorithms on six datasets including ALLAML, Colon, Central_Nervous_System, Isolet, Glioma and Carcinom. The maximum number of selected feature is set to 60 which is enough for most algorithms to reach the highest accuracies on the datasets.

As shown in Figure 1, our algorithm obviously outperforms the other algorithms in terms of the highest classification accuracies, whose accuracies can even exceed the others' beyond 7% on the datasets Carcinom and Glioma. With the increasing of the number of selected features, the classification accuracies of MIK-FS rise more quickly than the other algorithms on most datasets. MIK-FS can achieve the highest accuracies on all the six datasets when the number of selected feature is larger than 30.

The experiments about the convergence of MIK-FS are conducted on six datasets the same as previous experiments. We plot the value of $\{F(\boldsymbol{\beta}^k, b^k, \boldsymbol{d}^k) - F(\boldsymbol{\beta}^{k+1}, b^{k+1}, \boldsymbol{d}^{k+1})\}$ during the iterations in MIK-FS. Figure 2 shows difference value of objective function in MIK-FS changing with iterations on six datasets. Obviously, MIK-FS method converges rapidly within 10 iterations on all the six datasets.

6 Conclusion

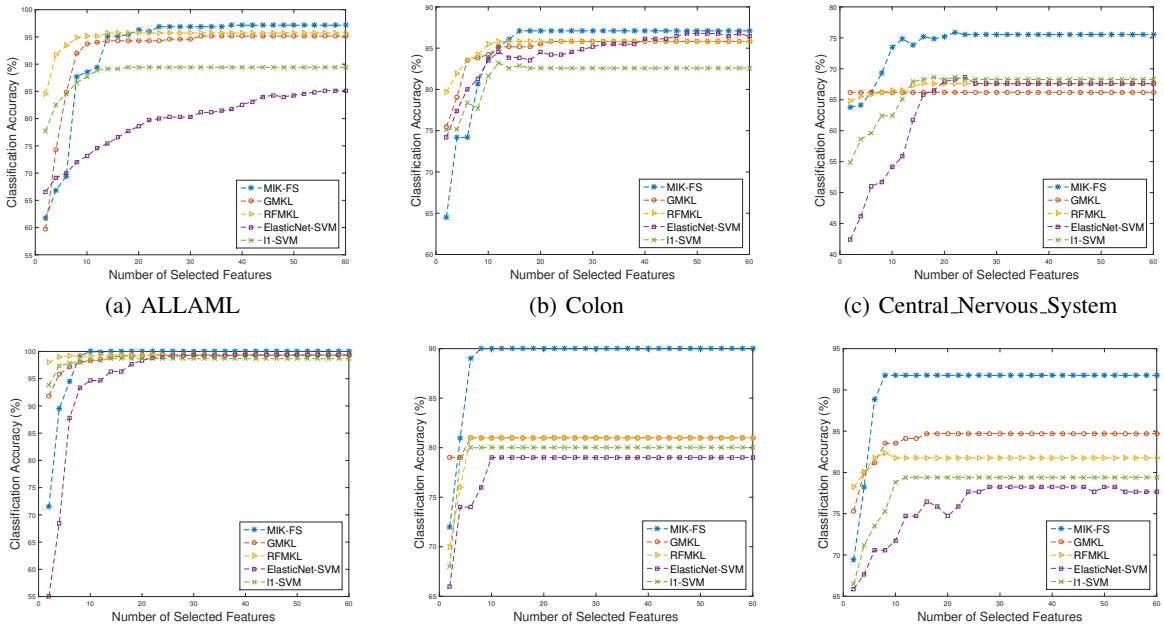
In this paper, we propose an embedded feature selection method MIK-FS based on the primal framework of IKSVM, which applies an indefinite base kernel for each feature and generates sparse kernel combination coefficients by l_1-norm to select features automatically. A two-stage algorithm is accordingly designed to optimize the coefficients of IKSVM and kernel combination alternately. Experiments on real-world datasets validate the effectiveness of MIK-FS in feature selection and classification.

[1] http://featureselection.asu.edu/datasets.php

[2] http://datam.i2r.a-star.edu.sg/datasets/krbd/

Table 2: Classification accuracies and the number of selected features (mean (#dimension)) of each compared algorithm on ten real-world datasets.

	l_1-SVM	ElasticNet-SVM	RFMKL	GMKL	IKPCA	MIK-FS
ALLAML	89.43 (18)	95.14 (558)	95.71 (14)	95.14 (34)	87.71 (13)	**97.14 (18)**
Colon	82.58 (13)	**88.07 (87)**	85.81 (11)	85.80 (20)	77.74 (16)	87.09 (17)
Gli_85	76.67 (5)	75.24 (8)	75.00 (28)	73.10 (4)	69.52 (38)	**78.09 (14)**
Prostate_GE	95.49 (15)	95.10 (71)	95.88 (14)	95.88 (29)	50.98 (45)	**95.88 (8)**
Central_Nervous_System	71.04 (9)	68.62 (19)	67.93 (10)	66.20 (3)	65.52 (27)	**75.52(17)**
Lung_Cancer	98.34 (12)	98.89 (15)	97.78 (33)	91.44 (428)	83.33 (9)	**99.89 (46)**
Dbworld	85.16 (14)	90.00 (232)	88.71 (18)	87.74 (154)	89.68 (6)	**90.00 (11)**
Isolet	98.67 (11)	99.34 (55)	99.34 (12)	99.34 (15)	99.83 (19)	**100.00 (9)**
Glioma	80.00 (5)	81.00 (104)	81.00 (5)	81.00 (6)	70.00 (9)	**90.00 (3)**
Carcinom	79.41 (4)	80.59 (488)	81.76 (7)	84.70 (16)	76.47 (15)	**91.70 (11)**



(a) ALLAML (b) Colon (c) Central_Nervous_System

(d) Isolet (e) Glioma (f) Carcinom

Figure 1: Classification accuracies versus the number of selected features of five embedded feature selection algorithms.

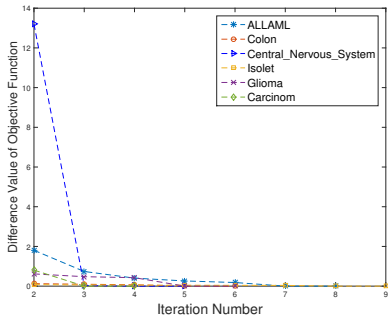


Figure 2: Convergence of MIK-FS on six datasets.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant Nos. 61375057, 61300165 and 61403193), Natural Science Foundation of Jiangsu Province of China (Grant No. BK20131298) and the National Key Research and Development Program of China (Grant No. 2016YFC1306704). Furthermore, the work was also supported by Collaborative Innovation Center of Wireless Communications Technology.

References

[Bradley and Mangasarian, 1998] Paul S Bradley and Olvi L Mangasarian. Feature selection via concave minimization and support vector machines. In *Proceedings of Fifteenth International Conference on Machine Learning*, volume 98, pages 82–90, 1998.

[Chandrashekar and Sahin, 2014] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[Chen *et al.*, 2007] Zhenyu Chen, Jianping Li, and Liwei Wei. A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue. *Artificial Intelligence in Medicine*, 41(2):161–175, 2007.

[Chen *et al.*, 2009] Yihua Chen, Maya R Gupta, and Benjamin Recht. Learning kernels from indefinite similarities. In *Proceedings of the Twenty-sixth Annual International Conference on Machine Learning*, pages 145–152. ACM, 2009.

[Dileep and Sekhar, 2009] Aroor Dinesh Dileep and C Chandra Sekhar. Representation and feature selection using multiple kernel learning. In *Proceedings of the Twenty-second International Joint Conference on Neural Networks*, pages 717–722. IEEE, 2009.

[Dinh and Le Thi, 2014] Tao Pham Dinh and Hoai An Le Thi. Recent advances in dc programming and dca. In *Transactions on Computational Intelligence XIII*, pages 1–37. Springer, 2014.

[Haasdonk and Pekalska, 2010] Bernard Haasdonk and Elżbieta Pekalska. Indefinite kernel discriminant analysis. In *Proceedings of the Sixteenth International Conference on Computational Statistics*, pages 221–230. Springer, 2010.

[Huang *et al.*, 2016] Xiaolin Huang, Andreas Maier, Joachim Hornegger, and Johan AK Suykens. Indefinite kernels in least squares support vector machines and principal component analysis. *Applied and Computational Harmonic Analysis*, 2016.

[Kowalski *et al.*, 2009] Matthieu Kowalski, Marie Szafranski, and Liva Ralaivola. Multiple indefinite kernel learning with mixed norm regularization. In *International Conference on Machine Learning*, pages 545–552, 2009.

[Liu, 2004] Chengjun Liu. Gabor-based kernel pca with fractional power polynomial models for face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 26(5):572–581, 2004.

[Liwicki *et al.*, 2012] Stephan Liwicki, Stefanos Zafeiriou, Georgios Tzimiropoulos, and Maja Pantic. Efficient online subspace learning with an indefinite kernel for visual tracking and recognition. *IEEE transactions on neural networks and learning systems*, 23(10):1624–1636, 2012.

[Ong *et al.*, 2004] Cheng Soon Ong, Xavier Mary, Stéphane Canu, and Alexander J Smola. Learning with non-positive kernels. In *Proceedings of the twenty-first international conference on Machine learning*, page 81. ACM, 2004.

[Saigo *et al.*, 2004] Hiroto Saigo, Jean-Philippe Vert, Nobuhisa Ueda, and Tatsuya Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, 2004.

[Tan *et al.*, 2010] Mingkui Tan, Li Wang, and Ivor W. Tsang. Learning sparse svm for feature selection on very high dimensional datasets. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 1047–1054, 2010.

[Tao and An, 1997] Pham Dinh Tao and Le Thi Hoai An. Convex analysis approach to dc programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997.

[Varma and Babu, 2009] Manik Varma and Bodla Rakesh Babu. More generality in efficient multiple kernel learning. In *Proceedings of the Twenty-sixth Annual International Conference on Machine Learning*, pages 1065–1072. ACM, 2009.

[Wang *et al.*, 2006] Li Wang, Ji Zhu, and Hui Zou. The doubly regularized support vector machine. *Statistica Sinica*, pages 589–615, 2006.

[Xu *et al.*, 2009] Zenglin Xu, Rong Jin, Jieping Ye, Michael R Lyu, and Irwin King. Non-monotonic feature selection. In *Proceedings of the Twenty-sixth Annual International Conference on Machine Learning*, pages 1145–1152. ACM, 2009.

[Xu *et al.*, 2017] Hai-Ming Xu, Hui Xue, Xiao-Hong Chen, and Yun-Yun Wang. Solving indefinite kernel support vector machine with difference of convex functions programming. In *Proceedings of the Thirty-first AAAI Conference on Artificial Intelligence*, 2017.

[Xue and Chen, 2014] Hui Xue and Songcan Chen. Discriminality-driven regularization framework for indefinite kernel machine. *Neurocomputing*, 133:209–221, 2014.

[Yamada *et al.*, 2014] Makoto Yamada, Wittawat Jitkrittum, Leonid Sigal, Eric P Xing, and Masashi Sugiyama. High-dimensional feature selection by feature-wise kernelized lasso. *Neural computation*, 26(1):185–207, 2014.