Recovery of Corrupted Multiple Kernels for Clustering

Peng Zhou[1,2] , **Liang Du**[1,3*] , **Lei Shi**[1,2] , **Hanmo Wang**[1,2] and **Yi-Dong Shen**[1*]
[1]State Key Laboratory of Computer Science,
Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
[3]School of Computer and Information Technology, Shanxi University, Taiyuan 030006, PR China
{zhoup,duliang,shilei,wanghm,ydshen}@ios.ac.cn

Abstract

Kernel-based methods, such as kernel k-means and kernel PCA, have been widely used in machine learning tasks. The performance of these methods critically depends on the selection of kernel functions; however, the challenge is that we usually do not know what kind of kernels is suitable for the given data and task in advance; this leads to research on multiple kernel learning, i.e. we learn a consensus kernel from multiple candidate kernels. Existing multiple kernel learning methods have difficulty in dealing with noises. In this paper, we propose a novel method for learning a robust yet low-rank kernel for clustering tasks. We observe that the noises of each kernel have specific structures, so we can make full use of them to clean multiple input kernels and then aggregate them into a robust, low-rank consensus kernel. The underlying optimization problem is hard to solve and we will show that it can be solved via alternating minimization, whose convergence is theoretically guaranteed. Experimental results on several benchmark data sets further demonstrate the effectiveness of our method.

1 Introduction

Clustering is a fundamental unsupervised machine learning problem. Kernel-based clustering methods, such as kernel k-means, are widely used due to their effectiveness of separating non-linearly separable clusters [Dhillon *et al.*, 2004]. In real world applications, we can construct different candidate kernels; for example, different kernel functions and parameters can lead to different kernels, and various views or data sources can also generate various kernels. The performance of kernel-based clustering highly depends on the choice of kernels [Yu *et al.*, 2012]. Unfortunately, it is still a challenge to determine a suitable one among an extensive range of possible kernels for the given data and task in advance. Recent years there have been considerable interests in multiple kernel learning [Xu *et al.*, 2008; Tzortzis and Likas, 2012; Huang *et al.*, 2012b; Gönen and Margolin, 2014], which aims at learning a suitable kernel from multiple kernel candidates for classification and clustering.

Unsupervised multiple kernel learning methods usually learn a consensus kernel by linearly combining a set of candidate kernels [Zhao *et al.*, 2009; Kulis *et al.*, 2009; Huang *et al.*, 2012b; 2012a]. Since the original data may be corrupted with noises and outliers, the induced kernel matrix may also be contaminated; moreover, given a dateset each kernel can be viewed as one with a certain degree of noises w.r.t. this dataset. However, to the best of our knowledge, no existing multiple kernel learning method has a mechanism to effectively handle kernel noises [Gönen and Margolin, 2014].

In this paper, we propose a novel Robust Multiple Kernel Clustering (RMKC) method; it contains a special mechanism capturing the structure of noises in multiple kernels. We observe that once an instance is corrupted with noise, both the corresponding row and column of a kernel will be simultaneously contaminated. Thus we introduce an error matrix for each kernel which expresses such row-wise and column-wise noises. We integrate the error matrices into a multiple kernel consensus framework. A valid consensus kernel should be symmetric and positive semi-definite (p.s.d.); in addition, to obtain a clear cluster structure we further impose a low rank constraint [Kulis *et al.*, 2006]. This leads to a hard optimization problem. To solve it we derive an alternating minimization procedure which can be theoretically guaranteed to converge. Note that we solve this problem by directly minimizing the rank of the consensus kernel without using any convex or non-convex approximated surrogates; this differs from existing approximated approaches [Luo *et al.*, 2012; Liu *et al.*, 2013; Xia *et al.*, 2014b; Lu *et al.*, 2014]. We have done empirical studies, which demonstrates the effectiveness of our method and show statistically significants improvements on compared algorithms.

2 Related Work

Multiple kernel learning has been actively studied [Xu *et al.*, 2008; Tzortzis and Likas, 2012; Huang *et al.*, 2012b; Gönen and Margolin, 2014]. Based on the availability of class labels, multiple kernel learning can be categorized into two classes: supervised algorithms and unsupervised methods. Although supervised multiple kernel learning has been extensively studied [Xu *et al.*, 2008; 2010; Hoi *et al.*, 2013;

[*] Corresponding author

Xia *et al.*, 2014a], unsupervised algorithm is more challenging due to the absence of class labels.

Several unsupervised algorithms have been developed in the framework of kernel k-means. [Tzortzis and Likas, 2012] proposed weighted kernel-based multi-view clustering including multi-view kernel k-means and multi-view spectral clustering respectively. [Yu *et al.*, 2012] developed a multiple kernel k-means where the kernel combinational coefficients are optimized automatically. [Huang *et al.*, 2012b] also presented a multiple kernel k-means and extended it to fuzzy k-means problem. [Zhuang *et al.*, 2011] explored local information for multiple kernel learning.

Due to the connection between kernel k-means and spectral clustering, the latter has also been extended to deal with multiple kernels [Dhillon *et al.*, 2004]. [Huang *et al.*, 2012a] aggregated kernels with different weights into a unified one for spectral clustering. [Kumar *et al.*, 2011] linearly combined spectral embeddings to get the final clustering. By resorting to the spectral method, [Gönen and Margolin, 2014] also proposed to solve a multiple kernel k-means associated with two-layer weights.

All the above methods try to learn the consensus clustering via the linear combination of multiple input kernels. As discussed before, such integration schema has no mechanism for handling noises and outliers in kernels.

In contrast to our multiple kernel learning, [Xia *et al.*, 2014b] proposed a method which learns a consensus matrix from a set of probabilistic transition matrices. Although this method also has a mechanism for handling noises, our method is designed for multiple kernel clustering while they are developed for transition matrices integration. Consequently, our noise structure and objective function constraints are totally different. The noises in transition matrix are sparse, while kernel noises are more sophisticated, i.e., they are symmetric with row-wise and column-wise sparsity. Besides, the consensus kernel is attached with additional complex constraints, i.e., symmetric, low-rank and p.s.d., which make it hard to solve.

3 Robust Multiple Kernel Learning

In this section, we present a framework for robust multiple kernel learning, and then introduce a method to solve the corresponding optimization problem.

3.1 Formulation

Given n instances, we calculate m kernels $\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, ..., \mathbf{K}^{(m)}$ which are $n \times n$ matrices. The task of multiple kernel learning is to learn a consensus kernel matrix $\mathbf{K}$ from $\mathbf{K}^{(1)}, ..., \mathbf{K}^{(m)}$. As discussed before, instances may contain noises, and inappropriate kernels can be viewed as matrices with certain degree of noises. In particular, we have the following key observation. When the j-th instance is corrupted with noise, both the j-th row and the j-th column of the kernel are simultaneously contaminated. To alleviate the adverse effect of these noises, we introduce the row-wise sparse matrix $\mathbf{E}^{(i)} \in \mathcal{R}^{n \times n}$ to capture noises on the rows of the i-th kernel. Naturally, $\mathbf{E}^{(i)T}$ is a column-wise sparse matrix characterizing the column noises. As a result, the noises of the i-th kernel is obtained by the aggregation of $\mathbf{E}^{(i)}$ and $\mathbf{E}^{(i)T}$, leading to the final symmetric error matrix $\mathbf{E}^{(i)} + \mathbf{E}^{(i)T}$. Then the *cleaned* matrix of the i-th kernel is $\mathbf{K}^{(i)} - (\mathbf{E}^{(i)} + \mathbf{E}^{(i)T})$. To learn the final consensus kernel $\mathbf{K}$ from $\mathbf{K}^{(i)}$'s, we minimize the average disagreements between $\mathbf{K}$ and the cleaned matrices $\mathbf{K}^{(i)} - (\mathbf{E}^{(i)} + \mathbf{E}^{(i)T})$. Thus, we have the following optimization problem for robust multiple kernel clustering.

$$\begin{aligned}\min_{\mathbf{K},\mathbf{E}^{(i)},\alpha_i} \quad & \sum_{i=1}^{m} \alpha_i \Big(\|\mathbf{K} - (\mathbf{K}^{(i)} - \mathbf{E}^{(i)} - \mathbf{E}^{(i)T})\|_F^2 \\ & + \gamma_1 \|\mathbf{E}^{(i)}\|_{2,1} \Big), \\ s.t. \quad & \mathbf{K} = \mathbf{K}^T, \quad \mathbf{K} \succeq 0, \\ & \sum_{i=1}^{m} \alpha_i^{\rho} = 1, \quad \forall i, \ \alpha_i \geq 0.\end{aligned} \tag{1}$$

where γ_1 is a balancing parameter and ρ is a parameter in $(0, 1)$. The $\ell_{2,1}$-norm is used to ensure the row sparsity of $\mathbf{E}^{(i)}$. The constraints on $\mathbf{K}$ ensure that $\mathbf{K}$ satisfies the properties of kernel matrix: symmetric and positive semi-definite. $\alpha_i \geq 0$ is used to control the weight of kernel, and $\sum_{i=1}^{m} \alpha_i^{\rho} = 1$ is to avoid trivial solution.

For the purpose of clustering, to obtain a more cleared cluster structure [Kulis *et al.*, 2006], we also impose the low-rank constraint on the consensus kernel $\mathbf{K}$; this is achieved by directly minimizing the rank of $\mathbf{K}$. To sum up, we obtain the following formulation:

$$\begin{aligned}\min_{\mathbf{K},\mathbf{E}^{(i)},\alpha_i} \quad & \sum_{i=1}^{m} \alpha_i \Big(\|\mathbf{K} - (\mathbf{K}^{(i)} - \mathbf{E}^{(i)} - \mathbf{E}^{(i)T})\|_F^2 \\ & + \gamma_1 \|\mathbf{E}^{(i)}\|_{2,1} \Big) + \gamma_2 \mathrm{rank}(\mathbf{K}), \\ s.t. \quad & \mathbf{K} = \mathbf{K}^T, \quad \mathbf{K} \succeq 0, \\ & \sum_{i=1}^{m} \alpha_i^{\rho} = 1, \quad \forall i, \quad \alpha_i \geq 0.\end{aligned} \tag{2}$$

where γ_2 is a balancing parameter. Note that the key of our formulation of multiple kernel clustering is the explicit characterization of error matrices $\mathbf{E}^{(i)} + \mathbf{E}^{(i)T}$, which makes it robust. Moreover, regarding the low-rank term, we do not use any approximation (neither convex nor non-convex approximation); this is different from previous ones such as those in [Luo *et al.*, 2012; Liu *et al.*, 2013; Xia *et al.*, 2014b; Lu *et al.*, 2014].

3.2 Optimization

The optimization problem in Eq.(2) involves three groups variables, i.e., $\mathbf{K}$, $\mathbf{E}^{(i)}$, and α_i. We develop an alternating minimization algorithm to solve this problem.

Optimize $\mathbf{E}^{(i)}$ by Fixing K and α_i

While fixing $\mathbf{K}$ and α_i, we get

$$\min_{\mathbf{E}^{(i)}} \quad \|\mathbf{K} - (\mathbf{K}^{(i)} - \mathbf{E}^{(i)} - \mathbf{E}^{(i)T})\|_F^2 + \gamma_1 \|\mathbf{E}^{(i)}\|_{2,1} \tag{3}$$

We introduce a diagonal matrix $\mathbf{D}^{(i)}$, where the j-th diagonal element is $\frac{1}{2\|E_j^{(i)}\|_2}$, and $\|E_j^{(i)}\|_2$ is the ℓ_2-norm of the j-th row of $\mathbf{E}^{(i)}$.

By setting the derivative of Eq.(3) w.r.t. $\mathbf{E}^{(i)}$ to zero, we get

$$(2\mathbf{I}+\gamma_1\mathbf{D}^{(i)})\mathbf{E}^{(i)}+2\mathbf{E}^{(i)T}=\mathbf{K}^{(i)}+\mathbf{K}^{(i)T}-\mathbf{K}-\mathbf{K}^T \quad (4)$$

where $\mathbf{I}$ is an identity matrix. Denote $\mathbf{A}^{(i)}=\mathbf{K}^{(i)}+\mathbf{K}^{(i)T}-\mathbf{K}-\mathbf{K}^T$, $d_{jj}^{(i)}$ as the (j,j)-th element of $\mathbf{D}^{(i)}$ and $e_{jk}^{(i)}$ as the (j,k)-th element in $\mathbf{E}^{(i)}$. Considering the (j,k)-th element and (k,j)-th element on both sides of Eq.(4), we obtain

$$\begin{cases} (2+\gamma_1 d_{jj}^{(i)})e_{jk}^{(i)}+2e_{kj}^{(i)}=A_{jk}^{(i)} \\ (2+\gamma_1 d_{kk}^{(i)})e_{kj}^{(i)}+2e_{jk}^{(i)}=A_{kj}^{(i)} \end{cases} \quad (5)$$

Solving Eq.(5), we get

$$\begin{cases} e_{kj}^{(i)}=\frac{(2+\gamma_1 d_{jj}^{(i)})A_{kj}^{(i)}-2A_{jk}^{(i)}}{\gamma_1^2 d_{kk}^{(i)}d_{jj}^{(i)}+2\gamma_1 d_{kk}^{(i)}+2\gamma_1 d_{jj}^{(i)}} \\ e_{jk}^{(i)}=\frac{(2+\gamma_1 d_{kk}^{(i)})A_{jk}^{(i)}-2A_{kj}^{(i)}}{\gamma_1^2 d_{jj}^{(i)}d_{kk}^{(i)}+2\gamma_1 d_{jj}^{(i)}+2\gamma_1 d_{kk}^{(i)}} \end{cases}$$

Note that $\mathbf{K}$ and $\mathbf{K}^{(i)}$ are symmetric, thus $\mathbf{A}^{(i)}$ is also symmetric. We can simplify $e_{kj}^{(i)}$ as

$$e_{kj}^{(i)}=\frac{d_{jj}^{(i)}A_{kj}^{(i)}}{\gamma_1 d_{kk}^{(i)}d_{jj}^{(i)}+2d_{kk}^{(i)}+2d_{jj}^{(i)}}. \quad (6)$$

Theorem 1. *Applying Eq.(6) monotonically decreases the objective function Eq.(3).*

Proof. The detailed proof is similar as that in [Yang *et al.*, 2011]. □

Optimize K by Fixing $\mathbf{E}^{(i)}$ and α_i

While fixing $\mathbf{E}^{(i)}$ and α_i, the Eq.(2) becomes

$$\begin{aligned} \min_{\mathbf{K}} \quad & \|\mathbf{K}-\frac{\sum_{i=1}^m \alpha_i(\mathbf{K}^{(i)}-\mathbf{E}^{(i)}-\mathbf{E}^{(i)T})}{\sum_{i=1}^m \alpha_i}\|_F^2 \\ & +\frac{\gamma_2}{\sum_{i=1}^m \alpha_i}\text{rank}(\mathbf{K}), \\ s.t. \quad & \mathbf{K}=\mathbf{K}^T, \quad \mathbf{K}\succeq 0. \end{aligned} \quad (7)$$

Let $\mathbf{B}=\frac{\sum_{i=1}^m \alpha_i(\mathbf{K}^{(i)}-\mathbf{E}^{(i)}-\mathbf{E}^{(i)T})}{\sum_{i=1}^m \alpha_i}$, $\tau=\frac{\gamma_2}{\sum_{i=1}^m \alpha_i}$, and simplify Eq.(7) as

$$\begin{aligned} \min_{\mathbf{K}} \quad & \|\mathbf{K}-\mathbf{B}\|_F^2+\tau\text{rank}(\mathbf{K}), \\ s.t. \quad & \mathbf{K}=\mathbf{K}^T, \quad \mathbf{K}\succeq 0. \end{aligned} \quad (8)$$

Let $\mathbf{U_K\Sigma_K V_K}^T=\mathbf{K}$ be the Singular Value Decomposition (SVD) of $\mathbf{K}$. Since $\mathbf{K}$ should be symmetric and positive semi-definite, $\mathbf{U_K}=[\mathbf{u_1},...,\mathbf{u_n}]$, $\mathbf{V_K}=[\mathbf{u_1},...,\mathbf{u_n}]$, and

$$\mathbf{\Sigma_K}=\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$$

where $\lambda_1,...,\lambda_n$ are n eigenvalues of $\mathbf{K}$, $\mathbf{u_1},...,\mathbf{u_n}$ are corresponding eigenvectors of $\mathbf{K}$. Similarly, let $\mathbf{U_B\Sigma_B V_B}^T=\mathbf{B}$ be the SVD decomposition of $\mathbf{B}$. Because $\mathbf{B}$ is symmetric, $\mathbf{U_B}=[\mathbf{v_1},\mathbf{v_2},...,\mathbf{v_n}]$, $\mathbf{V_B}=[sign(\sigma_1)*\mathbf{v_1},...,sign(\sigma_n)*\mathbf{v_n}]$, and

$$\mathbf{\Sigma_B}=\begin{pmatrix} |\sigma_1| & & \\ & \ddots & \\ & & |\sigma_n| \end{pmatrix}$$

where $\sigma_1,...,\sigma_n$ are n eigenvalues of $\mathbf{B}$, $\mathbf{v_1},...,\mathbf{v_n}$ are corresponding eigenvectors, and $sign(\cdot)$ is a sign function, i.e. $sign(x)=-1$ if x is negative and $sign(x)=1$ otherwise.

Theorem 2. *Let $\mathbf{U_K\Sigma_K V_K}^T=\mathbf{K}$ be the SVD decomposition of $\mathbf{K}$, $\mathbf{U_B\Sigma_B V_B}^T=\mathbf{B}$ be the SVD decomposition of $\mathbf{B}$, and λ_i, σ_i be as denoted before, then the solution of Eq.(8) is*

$$\mathbf{U_K}=\mathbf{U_B}, \quad \mathbf{V_K}=\mathbf{V_B},$$
$$\lambda_i=\begin{cases} \sigma_i, & \sigma_i\geq\sqrt{\tau} \\ 0, & \sigma_i<\sqrt{\tau} \end{cases}$$

Proof.

$$\begin{aligned} & \|\mathbf{K}-\mathbf{B}\|_F^2 \quad (9) \\ =& tr(\mathbf{KK}^T)-2tr(\mathbf{KB}^T)+tr(\mathbf{BB}^T) \\ =& \sum_{i=1}^n \lambda_i^2-2tr(\mathbf{KB}^T)+tr(\mathbf{BB}^T) \end{aligned}$$

where $tr(\cdot)$ is the trace function. According to Von Neumanns trace inequality, we have $tr(\mathbf{KB}^T)\leq tr(\mathbf{\Sigma_K\Sigma_B})$, then

$$\begin{aligned} & tr(\mathbf{U_K\Sigma_K V_K}^T\mathbf{B})=tr(\mathbf{KB}^T)\leq tr(\mathbf{\Sigma_K\Sigma_B}) \\ & =tr(\mathbf{U_B\Sigma_K V_B^T V_B\Sigma_B U_B^T})=tr(\mathbf{U_B\Sigma_K V_B}^T\mathbf{B}) \end{aligned} \quad (10)$$

which leads to

$$\|\mathbf{U_K\Sigma_K V_K}^T-\mathbf{B}\|_F^2\geq\|\mathbf{U_B\Sigma_K V_B}^T-\mathbf{B}\|_F^2 \quad (11)$$

Thus, to minimize Eq.(8), we should set $\mathbf{U_K}=\mathbf{U_B}$ and $\mathbf{V_K}=\mathbf{V_B}$, i.e., the eigenvectors of $\mathbf{K}$ should be the same as eigenvectors of $\mathbf{B}$. Note that $\mathbf{K}\succeq 0$, which means $\lambda_i\geq 0$ leads to that if $sign(\sigma_i)=-1$, i.e. $\sigma_i<0$, then λ_i must be zero.

To handle the rank function, we define function f:

$$f(x)=\begin{cases} 1, & x\neq 0 \\ 0, & x=0 \end{cases}$$

Since the rank of $\mathbf{K}$ is the number of non-zero singular values of $\mathbf{K}$, Eq.(8) can be rewritten as:

$$\begin{aligned} \min_{\lambda} \quad & \sum_{i=1}^n(\lambda_i-|\sigma_i|)^2+\tau\sum_{i=1}^n f(\lambda_i), \\ s.t. \quad & \lambda_i\geq 0. \end{aligned} \quad (12)$$

Note that if $\sigma_i<0$, then $\lambda_i=0$, so we drop all negative σ_i in Eq.(12):

$$\min_{\lambda} \quad \sum_{\sigma_i\geq 0}\left((\lambda_i-\sigma_i)^2+\tau f(\lambda_i)\right), \quad s.t. \quad \lambda_i\geq 0. \quad (13)$$

Eq.(13) can be solved easily:

$$\lambda_i = \begin{cases} \sigma_i, & \sigma_i \geq \sqrt{\tau} \\ 0, & 0 \leq \sigma_i < \sqrt{\tau} \end{cases} \tag{14}$$

Next considering the case $\sigma_i < 0$ in addition, we get the final solution of λ_i:

$$\lambda_i = \begin{cases} \sigma_i, & \sigma_i \geq \sqrt{\tau} \\ 0, & \sigma_i < \sqrt{\tau} \end{cases} \tag{15}$$

To sum up, we get the global optima of this sub problem $\mathbf{K} = \mathbf{U_B}\mathbf{\Sigma_K}\mathbf{V_B}^T$, where $\mathbf{\Sigma_K}$ is obtained by Eq.(15). Since the input $\mathbf{K}^{(i)}$ is kernel matrix which is symmetric, $\mathbf{B}$ is also symmetric and it leads to that $\mathbf{K}$ is also symmetric. □

Optimize α_i by Fixing $\mathbf{E}^{(i)}$ and $\mathbf{K}$

When $\mathbf{K}$ and $\mathbf{E}^{(i)}$ are fixed, the optimization problem w.r.t. α_i is:

$$\min_{\alpha} \quad \sum_{i=1}^{m} \alpha_i r_i, \quad s.t. \quad \sum_{i=1}^{m} \alpha_i^{\rho} = 1, \quad \forall i,\ \alpha_i \geq 0. \tag{16}$$

where $r_i = \|\mathbf{K} - (\mathbf{K}^{(i)} - \mathbf{E}^{(i)} - \mathbf{E}^{(i)T})\|_F^2 + \gamma_1 \|\mathbf{E}^{(i)}\|_{2,1}$.

Introducing Lagrange multiplier l, we get

$$\mathcal{L} = \sum_{i=1}^{m} \alpha_i r_i + l(\sum_{i=1}^{m} \alpha_i^{\rho} - 1).$$

Setting the derivative w.r.t. α_i to zero and considering $\sum_i \alpha_i^{\rho} = 1$, we solve the l as $l = -\frac{1}{\rho}(\sum_i r_i^{\frac{\rho}{\rho-1}})^{\frac{\rho-1}{\rho}}$ and then obtain the solution of α_i:

$$\alpha_i = \frac{r_i^{\frac{1}{\rho-1}}}{(\sum_i r_i^{\frac{\rho}{\rho-1}})^{\frac{1}{\rho}}} \tag{17}$$

To sum up, we alternatively optimize $\mathbf{E}^{(i)}$, $\mathbf{K}$, and α_i until it converges. Algorithm 1 summarizes the whole process.

3.3 Convergence Analysis

Theorem 3. *The iterative approach in Algorithm 1 converges.*

Proof. Theorem 1 shows that updating $\mathbf{E}^{(i)}$ as Eq.(6) can monotonically decreases the objective function Eq.(2). When updating $\mathbf{K}$ and α_i, we find the global optima of the subproblem which also monotonically decreases the objective function. In addition, the objective function is always greater than 0. Thus Algorithm 1 converges. □

3.4 Complexity Analysis

In each iteration, when updating $\mathbf{E}^{(i)}$, there are only elementwise optimizations, thus the complexity is $O(n^2 m)$, where n is the number of instances and m the number of kernels. When updating $\mathbf{K}$, the complexity is $O(n^3)$ due to the SVD decomposition. Computing α_i just costs $O(m)$. To sum up, in one iteration, the time complexity is $O(n^2 m + n^3)$, thus the total complexity is $O((n^2 m + n^3)q)$, where q is the number of iterations.

Algorithm 1 Robust Multiple Kernel Clustering

```
Input: multiple kernels $\mathbf{K}^{(1)}, ..., \mathbf{K}^{(m)}$, parameters $\gamma_1$, $\gamma_2$, $\rho$.
Output: the consensus kernel matrix $\mathbf{K}$, error matrix of each kernel $\mathbf{E}^{(i)}$, kernel weights $\alpha$.
Initialize $\mathbf{K} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{K}^{(i)}$ , $\alpha_i = (\frac{1}{m})^{\frac{1}{\rho}}$ and $\mathbf{E}^{(i)} = \frac{1}{2}(\mathbf{K}^{(i)} - \mathbf{K})$.
while not converge do
    Compute $\mathbf{E}^{(i)}$ as Eq.(6).
    Compute $\mathbf{U_B}, \mathbf{\Sigma_B}, \mathbf{V_B}$ as SVD decomposition of $\mathbf{B}$.
    Compute $\mathbf{\Sigma_K}$ as Eq.(15) and obtain $\mathbf{K}$ as $\mathbf{K} = \mathbf{U_B}\mathbf{\Sigma_K}\mathbf{V_B}^T$.
    Compute $\alpha$ as Eq.(17).
end while
```

Table 1: Description of the datasets.

	#instances	#features	#classes
YALE	165	1024	15
WINE	178	13	3
JAFFE	213	676	10
CSTR	476	1000	4
BBCNews	737	1000	5
TR31	927	10128	7
UCI Digits	2000	76	10
Hitech	2301	22498	6
News4a	3840	4989	4

4 Experiment

To demonstrate the effectiveness of out method, we apply RMKC for clustering tasks and compare it with several state-of-the-art multiple kernel clustering methods on benchmark datasets.

4.1 Datasets

We collect 9 datasets, including 5 text corpora, i.e. CSTR [Du *et al.*, 2012], BBCNews, Tr31, Hitech [Greene and Cunningham, 2005], News4a [Wu and Schölkopf, 2006]; 3 image datasets, i.e., YALE [Belhumeur *et al.*, 1997], JAFFE [Lyons *et al.*, 1999], UCI Digits[1]and 1 chemical analysis dataset of wine, i.e., WINE [2], which are frequently used in clustering task. Datasets from different areas serve as a good test bed for a comprehensive evaluation. The important statistics of these datasets are summarized in Table 1.

4.2 Compared Methods

We compare the following algorithms

- **Single kernel methods.** Since we have multiple kernels, we run kernel k-means and spectral clustering on each kernel separately. Both the best and the average results over all kernels are reported, which are referred to as KKM-b, KKM-a, SC-b, SC-a, respectively. Due to space limitation, the worst results of single kernel are not

[1] https://archive.ics.uci.edu/ml/datasets/Multiple+Features

[2] https://archive.ics.uci.edu/ml/datasets/Wine

Table 2: Clustering results with K-means based methods

Data	Metrics	KKM-b	KKM-a	KKM-ew	MKKM	RMKC-KM
YALE	ACC	0.4558	0.3211	0.3588	0.4491	**0.5006**
	NMI	0.4745	0.3386	0.3794	0.4900	**0.5236**
WINE	ACC	0.9618	0.7369	0.9590	0.9607	**0.9663**
	NMI	0.9618	0.7468	0.9590	0.8613	**0.9663**
JAFFE	ACC	0.7512	0.6183	0.6507	0.6981	**0.7718**
	NMI	0.7746	0.6451	0.6789	**0.7756**	**0.7901**
CSTR	ACC	0.6832	0.4454	0.4804	0.5164	**0.6840**
	NMI	0.7354	0.5105	0.5400	0.6069	**0.7434**
BBCNews	ACC	0.6016	0.3827	0.4140	0.4752	**0.6114**
	NMI	0.6357	0.4597	0.4620	0.5328	**0.6427**
TR31	ACC	0.4920	0.3702	0.1868	0.1780	**0.5175**
	NMI	0.5249	0.4211	0.3817	0.3808	**0.5249**
UCI Digits	ACC	0.6495	0.5577	0.6346	0.6523	**0.6651**
	NMI	0.6660	0.5785	0.6522	0.6442	**0.6825**
Hitech	ACC	0.4415	0.3045	0.3472	0.3371	**0.4209**
	NMI	0.5211	0.3596	0.4345	0.4200	**0.5511**
News4a	ACC	0.5423	0.3164	0.2702	0.2637	**0.5758**
	NMI	0.5514	0.3194	0.2721	0.2650	**0.5861**

reported. It should be pointed out that the worst results are often far below the average.

- **Equal weighted methods.** The multiple input kernels are combined into a single kernel with equal weights. The results are referred to as KKM-ew and SC-ew.
- **MKKM.**[3] The MKKM (Multiple Kernel K-Means) is proposed in [Huang *et al.*, 2012b] which extends kernel k-means in multiple kernel setting.
- **Co-regSC.**[4] The Co-regSC is a co-regularized multi-view spectral clustering proposed by [Kumar *et al.*, 2011].
- **LMKKM.**[5] The LMKKM is proposed in [Gönen and Margolin, 2014]. It transfers the kernel k-means to spectral methods and uses the results from eigenvalue decomposition to clustering. Thus it is a spectral method in fact.
- **RMSC.**[6] The RMSC (Robust Multiview Spectral Clustering) is proposed by [Xia *et al.*, 2014b]. We first transform the kernels into probabilistic transition matrices following [Xia *et al.*, 2014b], and then apply RMSC to get the final clustering results.
- **RMKC**, which is our proposed method. We use RMKC to learn the consensus kernel, and then apply kernel k-means and spectral clustering on the learned kernel, which are referred to as RMKC-KM and RMKC-SC, respectively.

4.3 Experiment Setup

Following the similar experimental protocol of other multiple kernel learning methods, we apply 8 different kernel functions as basis for multiple kernel clustering. These kernels are 5 RBF kernels $\mathcal{K}(\mathbf{x_i}, \mathbf{x_j}) = exp(-\|\mathbf{x_i} - \mathbf{x_j}\|^2/(2t^2))$ with $t = t_0 * d_{max}$, where d_{max} is the maximal distance between samples and t_0 varies in the range of $\{0.01, 0.1, 1, 10, 100\}$, 2 polynomial kernels $\mathcal{K}(\mathbf{x_i}, \mathbf{x_j}) = (\mathbf{x_i}^T \mathbf{x_j})^a$ with $a = 2, 4$ and a linear kernel. Finally, all kernels are normalized to normalized-cut weighted form as [Xu and Gong, 2004] did and then rescaled to $[0, 1]$.

The number of clusters is set to the true number of classes for all the datasets and clustering algorithms. The results of most of these compared algorithms depend on the initialization. We independently repeat the experiments for 10 times with random initializations and report the average results and t-test results. In our method, we tune ρ in $\{0.3, 0.5, 0.7\}$, and tune γ_1 from $[10^{-3}, 10^3]$, γ_2 from $[10^{-2}, 10^2]$ by grid search. For other compared methods, we tune the parameters as suggested in their papers.

Two clustering evaluation metrics are adopted to measure the clustering performance, including clustering Accuracy (ACC) and Normalized Mutual Information (NMI).

4.4 Experimental Results

We conduct two groups of experiment: we compare our methods RMKC-KM and RMKC-SC with k-means based methods and spectral clustering methods respectively. Table 2 and Table 3 show the results. Bold font indicates that the difference is statically significant (the p-value of t-test is smaller than 0.05). Note that since we aim to compare with other multiple kernel methods, we do not calculate the p-value of KKM-b, KKM-a, SC-b and SC-a.

The results reveal some interesting points:

- The performances of kernel k-means and spectral clustering are largely determined by the choice of kernel

[3] http://imp.iis.sinica.edu.tw/IVCLab/research/Sean/mkfc/code.rar

[4] http://www.umiacs.umd.edu/~abhishek/code_cospectral.zip

[5] https://github.com/mehmetgonen/lmkkmeans

[6] http://ss.sysu.edu.cn/~py/RMSC.zip

Table 3: Clustering results with Spectral Clustering based methods

Data	Metrics	SC-b	SC-a	SC-ew	Coreg-SC	LMKKM	RMSC	RMKC-SC
YALE	ACC	0.5721	0.4716	0.5497	**0.5739**	0.5509	**0.5879**	**0.5809**
	NMI	0.5827	0.5001	0.5737	0.5887	0.5762	**0.6013**	**0.5961**
WINE	ACC	0.9663	0.8065	0.9551	0.9607	0.9494	0.9551	**0.9663**
	NMI	0.8730	0.5849	0.9590	0.8613	0.8324	0.8447	**0.8748**
JAFFE	ACC	0.8953	0.7800	0.8357	**0.8864**	**0.8873**	**0.8685**	**0.8920**
	NMI	0.9180	0.8178	0.8719	**0.9154**	**0.9140**	**0.9093**	**0.9099**
CSTR	ACC	0.8137	0.5749	0.6158	0.7006	0.5899	0.6105	**0.8198**
	NMI	0.6274	0.3425	0.4817	0.5948	0.4432	0.5562	**0.6254**
BBCNews	ACC	0.5434	0.4348	0.4551	0.5280	0.4072	0.4763	**0.5563**
	NMI	0.3450	0.1929	0.2516	0.3082	0.2094	0.2872	**0.3302**
TR31	ACC	0.5714	0.4369	0.4005	0.6091	0.4142	0.4641	**0.6745**
	NMI	0.4190	0.1524	0.1177	0.3771	0.4328	0.3509	**0.4667**
UCI Digits	ACC	0.6886	0.5992	0.6764	0.6764	0.6325	0.6698	**0.6864**
	NMI	0.6367	0.5454	0.6253	0.6231	0.5965	0.6221	**0.6417**
Hitech	ACC	0.4534	0.3409	0.4202	0.4435	0.4058	0.4086	**0.5044**
	NMI	0.3165	0.1371	0.2818	0.3128	0.3038	0.2690	**0.3351**
News4a	ACC	0.6194	0.3555	0.3863	0.5727	0.2714	0.5530	**0.6578**
	NMI	0.4310	0.1222	0.1480	0.3290	0.2755	0.3676	**0.4406**

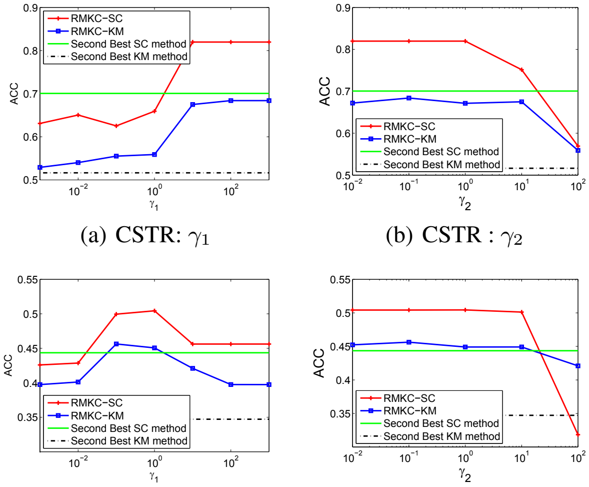
functions. With a proper kernel function, these methods usually present good results. However, the performances are significantly deteriorated on inappropriate kernels. Such observations also motivate the development of multiple kernel clustering.

- With proper kernel weight learning schema, the multiple kernel clustering approaches usually improve the results over simple equally weighted combination.
- The proposed RMKC outperforms other methods significantly in most of the datasets. The performance of RMKC is usually close to or even better than the result of the best single kernel. Note that, RMKC does not need perform exhaustive search on a predefined pool of kernels. Such results well demonstrate the superiority of the proposed method.

4.5 Parameter Study

We explore the effect of parameters (γ_1 and γ_2) on clustering performance by tuning γ_1 from $[10^{-3}, 10^3]$ and γ_2 from $[10^{-2}, 10^2]$. We show the ACC on CSTR and Hitech datasets. The results on other datasets are similar.

Figure 1 shows the ACC results, from which we can see that our algorithm is better than the closest baseline method for most of the γ_1 and γ_2 values. It indicates that the performance of our method is stable across a wide range of the parameters.



(a) CSTR: γ_1

(b) CSTR : γ_2

(c) Hitech : γ_1

(d) Hitech : γ_2

Figure 1: ACC w.r.t γ_1 and γ_2 on CSTR and Hitech.

5 Conclusion

In this paper, we proposed a novel robust multiple kernel clustering method. We observed that the noises of kernels have specific structures, i.e., they are symmetric, row-wise and column-wise. Based on this observation, we introduced both row-sparse and column-sparse matrices to our multiple kernel formulation, such that robust and low-rank consensus kernel can be learned by minimizing the disagreement over the cleaned kernels. We provided an iterative algorithm to solve the hard optimization problem. Experimental results on real world datasets show that our method outperforms other compared methods.

In the future, we will study scalability issue with multiple kernel learning. Besides, we will further explore some common structure of noises in multiple kernels.

Acknowledgments

This work is supported in part by China National 973 program 2014CB340301 and NSFC grant 61379043, 61273291.

References

[Belhumeur *et al.*, 1997] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *PAMI, IEEE Transactions on*, 19(7):711–720, 1997.

[Dhillon *et al.*, 2004] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis. Kernel k-means: Spectral clustering and normalized cuts. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 551–556, 2004.

[Du *et al.*, 2012] Liang Du, Xuan Li, and Yi-Dong Shen. Robust nonnegative matrix factorization via half-quadratic minimization. In *ICDM*, pages 201–210, 2012.

[Gönen and Margolin, 2014] Mehmet Gönen and Adam A Margolin. Localized data fusion for kernel k-means clustering with application to cancer biology. In *NIPS*, pages 1305–1313, 2014.

[Greene and Cunningham, 2005] Derek Greene and Padraig Cunningham. Producing accurate interpretable clusters from high-dimensional data. In *PKDD*, pages 486–494, 2005.

[Hoi *et al.*, 2013] Steven C. H. Hoi, Rong Jin, Peilin Zhao, and Tianbao Yang. Online multiple kernel classification. *Machine Learning*, 90(2):289–316, 2013.

[Huang *et al.*, 2012a] Hsin-Chien Huang, Yung-Yu Chuang, and Chu-Song Chen. Affinity aggregation for spectral clustering. In *CVPR, IEEE Conference on*, pages 773–780. IEEE, 2012.

[Huang *et al.*, 2012b] Hsin-Chien Huang, Yung-Yu Chuang, and Chu-Song Chen. Multiple kernel fuzzy clustering. *Fuzzy Systems, IEEE Transactions on*, 20(1):120–134, 2012.

[Kulis *et al.*, 2006] Brian Kulis, Mátyás A. Sustik, and Inderjit S. Dhillon. Learning low-rank kernel matrices. In *ICML*, pages 505–512, 2006.

[Kulis *et al.*, 2009] Brian Kulis, Sugato Basu, Inderjit Dhillon, and Raymond Mooney. Semi-supervised graph clustering: a kernel approach. *Machine learning*, 74(1):1–22, 2009.

[Kumar *et al.*, 2011] Abhishek Kumar, Piyush Rai, and Hal Daume. Co-regularized multi-view spectral clustering. In *NIPS*, pages 1413–1421, 2011.

[Liu *et al.*, 2013] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *PAMI, IEEE Transactions on*, 35(1):171–184, 2013.

[Lu *et al.*, 2014] Canyi Lu, Jinhui Tang, Shuicheng Yan, and Zhouchen Lin. Generalized nonconvex nonsmooth low-rank minimization. In *CVPR, IEEE Conference on*. IEEE, 2014.

[Luo *et al.*, 2012] Dijun Luo, Heng Huang, Feiping Nie, and Chris H Ding. Forging the graphs: A low rank and positive semidefinite graph learning approach. In *NIPS*, pages 2960–2968, 2012.

[Lyons *et al.*, 1999] Michael J. Lyons, Julien Budynek, and Shigeru Akamatsu. Automatic classification of single facial images. *PAMI, IEEE Transactions on*, 21(12):1357–1362, 1999.

[Tzortzis and Likas, 2012] Grigorios Tzortzis and Aristidis Likas. Kernel-based weighted multi-view clustering. In *ICDM*, pages 675–684, 2012.

[Wu and Schölkopf, 2006] Mingrui Wu and Bernhard Schölkopf. A local learning approach for clustering. In *NIPS*, pages 1529–1536, 2006.

[Xia *et al.*, 2014a] Hao Xia, Steven C. H. Hoi, Rong Jin, and Peilin Zhao. Online multiple kernel similarity learning for visual search. *PAMI, IEEE Transactions on*, 36(3):536–549, 2014.

[Xia *et al.*, 2014b] Rongkai Xia, Yan Pan, Lei Du, and Jian Yin. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 2149–2155, 2014.

[Xu and Gong, 2004] Wei Xu and Yihong Gong. Document clustering by concept factorization. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 202–209, New York, NY, USA, 2004. ACM.

[Xu *et al.*, 2008] Zenglin Xu, Rong Jin, Irwin King, and Michael R. Lyu. An extended level method for efficient multiple kernel learning. In *NIPS*, pages 1825–1832, 2008.

[Xu *et al.*, 2010] Zenglin Xu, Rong Jin, Haiqin Yang, Irwin King, and Michael R. Lyu. Simple and efficient multiple kernel learning by group lasso. In *ICML*, pages 1175–1182, 2010.

[Yang *et al.*, 2011] Yi Yang, Heng Tao Shen, Zhigang Ma, Zi Huang, and Xiaofang Zhou. l2, 1-norm regularized discriminative feature selection for unsupervised learning. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1589, 2011.

[Yu *et al.*, 2012] Shi Yu, L-C Tranchevent, Xinhai Liu, Wolfgang Glanzel, Johan AK Suykens, Bart De Moor, and Yves Moreau. Optimized data fusion for kernel k-means clustering. *PAMI, IEEE Transactions on*, 34(5):1031–1039, 2012.

[Zhao *et al.*, 2009] Bin Zhao, James T Kwok, and Changshui Zhang. Multiple kernel clustering. In *SDM*, pages 638–649. SIAM, 2009.

[Zhuang *et al.*, 2011] Jinfeng Zhuang, Jialei Wang, Steven C. H. Hoi, and Xiangyang Lan. Unsupervised multiple kernel learning. In *ACML*, pages 129–144, 2011.