Localized Incomplete Multiple Kernel k-means

Xinwang Liu[1], Xinzhong Zhu[2,7], Miaomiao Li[1], En Zhu[1], Li Liu[3,4], Zhiping Cai[1], Jianping Yin[5] and Wen Gao[6]
[1] School of Computer Science, National University of Defense Technology, Changsha, China, 410073
[2] College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua, China, 321004
[3] College of System Engineering, National University of Defense Technology, Changsha, China, 410073
[4] University of Oulu, Finland, [5] Dongguan University of Technology, Guangdong, China
[6] School of Electronics Engineering and Computer Science, Peking University, Beijing, China, 100871
[7] School of Electronic Engineering, XIDIAN University, Xi'an, Shanxi, China, 710126

Abstract

The recently proposed multiple kernel k-means with incomplete kernels (MKKM-IK) optimally integrates a group of pre-specified incomplete kernel matrices to improve clustering performance. Though it demonstrates promising performance in various applications, we observe that it does not *sufficiently consider the local structure among data and indiscriminately forces all pairwise sample similarity to equally align with their ideal similarity values*. This could make the incomplete kernels less effectively imputed, and in turn adversely affect the clustering performance. In this paper, we propose a novel localized incomplete multiple kernel k-means (LI-MKKM) algorithm to address this issue. Different from existing MKKM-IK, LI-MKKM only requires the similarity of a sample to its k-nearest neighbors to align with their ideal similarity values. This helps the clustering algorithm to focus on closer sample pairs that shall stay together and avoids involving unreliable similarity evaluation for farther sample pairs. We carefully design a three-step iterative algorithm to solve the resultant optimization problem and theoretically prove its convergence. Comprehensive experiments on eight benchmark datasets demonstrate that our algorithm significantly outperforms the state-of-the-art comparable algorithms proposed in the recent literature, verifying the advantage of considering local structure.

1 Introduction

Multiple kernel clustering (MKC) aims to optimally combine a group of pre-specified base kernels to perform clustering, which has been intensively studied during the past several years [Yu *et al.*, 2012; Li *et al.*, 2014; Gönen and Margolin, 2014; Liu *et al.*, 2016; Li *et al.*, 2016; Liu *et al.*, 2017b; Zhang *et al.*, 2015; Cao *et al.*, 2015; Gao *et al.*, 2015; Nie *et al.*, 2014; Cai *et al.*, 2013; Xu *et al.*, 2015a]. A common assumption adopted by these MKC algorithms is that all the pre-specified base kernel matrices are complete. However, it is not uncommon to see that some views of a sample are absent in practical applications [Xiang *et al.*, 2013; Kumar *et al.*, 2013]. Consequently, this will cause the corresponding rows and columns of related base kernel matrices unfilled.

The presence of incomplete base kernel matrices makes it more challenging to utilize the information of all views for clustering. Many efforts have been devoted to address this issue [Ghahramani and Jordan, 1993; Trivedi *et al.*, 2010; Yin *et al.*, 2015; Xu *et al.*, 2015b; Shao *et al.*, 2015; Bhadra *et al.*, 2016; Liu *et al.*, 2017b]. They can roughly be grouped into two categories. The first one firstly fills the incomplete kernels with an imputation algorithm and then applies a standard MKC algorithm with these imputed kernels. The widely used imputation algorithms include zero-filling, mean value filling, k-nearest-neighbor filling and expectation-maximization (EM) filling [Ghahramani and Jordan, 1993]. In contrast, the other category proposes to integrate the imputation and clustering into a single optimization procedure, leading to a clustering-oriented imputation [Liu *et al.*, 2017a]. By this way, these two procedures are seamlessly connected to achieve better clustering performance.

Although the aforementioned algorithms demonstrate promising clustering performance in various applications, we observe that they, no matter separately or jointly optimizing imputation and clustering, *do not sufficiently consider the local structure among data*, which is crucial for unsupervised learning tasks like clustering [Li *et al.*, 2016]. As a consequence, this could make the incomplete kernels less effectively imputed, and in turn adversely affect the clustering performance. To address this issue, we propose to integrate the local structure among data into incomplete multiple kernel clustering tasks, with the aim to further improve the clustering performance. As an instantiation, we design a localized incomplete multiple kernel k-means (LI-MKKM) algorithm and build it upon the latest incomplete multiple kernel clustering framework developed in the literature [Liu *et al.*, 2017a]. Our algorithm inherits the advantage of [Liu *et al.*, 2017a] that unifies the imputation and clustering into a single optimization procedure. The clustering result at the last iteration guides the imputation of absent kernel elements, and the latter is used in turn to conduct the subsequent clustering. These two learning processes negotiate with each other to achieve the optimal clustering. More importantly, different from [Liu *et al.*, 2017a] which rigidly forces closer and farther sample pairs to be equally aligned to the same ideal similarity, our algorithm only requires that the similarity of a sample to its k-nearest neighbours be aligned with the ideal similarity matrix. Such an alignment helps the clustering algorithm to better focus on closer sample pairs that shall stay together and avoids involving unreliable similarity evaluation for farther sample pairs. The optimization objective of our algorithm is carefully designed and an efficient algorithm with proved convergence is developed to solve the resultant optimization problem. Extensive experimental study is carried out on eight multiple kernel learning (MKL) benchmark data sets to evaluate the clustering performance of the proposed algorithm. As demonstrated, our algorithm significantly outperforms existing two-stage imputation methods and the recently proposed algorithm [Liu *et al.*, 2017a], validating the advantage of incorporating the local structure of data.

The main contributions of this paper are briefly summarized as follows. i) We, for the first time, identify the global kernel alignment issue in incomplete multiple kernel clustering and propose an effective solution; ii) We develop a general parametrization model to impute incomplete kernel matrices with theoretically proved feasibility; and iii) We conduct extensive experiments to validate our identification of this issue and the effectiveness of our solution.

2 Related Work

Multiple advanced algorithms have recently been proposed to address incomplete multiple kernel clustering [Trivedi *et al.*, 2010; Xu *et al.*, 2015b; Shao *et al.*, 2015; Yin *et al.*, 2015; Bhadra *et al.*, 2016]. With the help of one complete view, the work in [Trivedi *et al.*, 2010] constructs a complete kernel matrix for the other incomplete view. The work in [Xu *et al.*, 2015b] proposes an algorithm to accomplish multi-view learning with incomplete views by exploiting the connections of multiple views, where different views are assumed to be generated from a shared subspace. A multi-incomplete-view clustering algorithm is proposed in [Shao *et al.*, 2015]. It learns latent feature matrices for all the views and generates a consensus matrix by minimizing the difference between each view and the consensus. In addition, the approach in [Bhadra *et al.*, 2016] proposes to predict missing rows and columns of a base kernel by modelling both within-view and between-view relationships among kernel values.

One drawback shared by the above-mentioned "two-stage" algorithms is that the processes of imputation and clustering are disconnected, and this prevents the two learning processes from negotiating with each other to achieve the optimal clustering. To overcome this drawback, some pioneering work are proposed to integrate imputation and clustering into a single optimization procedure [Liu *et al.*, 2017a]. These two procedures are interacted to achieve better clustering performance. In the following, we give an introduction to the newly proposed MKKM with incomplete kernels [Liu *et al.*, 2017a] upon which we develop a novel algorithm.

Let $\mathbf{s}_p$ denote the available sample indices from the p-th $(1 \leq p \leq m)$ view and $\mathbf{K}_p^{(cc)} \in \mathbb{R}^{n_p \times n_p}$ be a kernel sub-matrix computed with these observed samples, where n_p is the length of $\mathbf{s}_p$. The recently proposed MKKM-IK [Liu *et al.*, 2017a] optimally integrates these incomplete kernel matrices $\{\mathbf{K}_p^{(cc)}\}_{p=1}^m$ to improve clustering performance. It unifies the imputation and clustering procedure into a single optimization objective and alternately optimizes each of them, which is mathematically fulfilled as follows,

$$\begin{aligned} \min\nolimits_{\mathbf{H},\, \boldsymbol{\beta},\, \{\mathbf{K}_p\}_{p=1}^m} \quad & \mathrm{Tr}(\mathbf{K}_{\boldsymbol{\beta}}(\mathbf{I}_n - \mathbf{H}\mathbf{H}^\top)) \\ s.t. \quad & \mathbf{H} \in \mathbb{R}^{n\times k}, \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k, \;\; \boldsymbol{\beta}^\top \mathbf{1}_m = 1, \beta_p \geq 0, \\ & \mathbf{K}_p(\mathbf{s}_p, \mathbf{s}_p) = \mathbf{K}_p^{(cc)}, \; \mathbf{K}_p \succeq 0, \; \forall p, \end{aligned} \tag{1}$$

where $\mathbf{K}_{\boldsymbol{\beta}} = \sum_{p=1}^m \beta_p^2 \mathbf{K}_p$, $\mathbf{H}$ is the clustering matrix, and n and k are the number of samples to be clustered and clusters. The constraint $\mathbf{K}_p(\mathbf{s}_p, \mathbf{s}_p) = \mathbf{K}_p^{(cc)}$ is imposed to ensure that $\mathbf{K}_p$ maintains the known entries during the course. MKKM-IK develops a three-step alternate optimization algorithm to solve Eq.(1), which simultaneously imputes the missing entries of base kernels and clustering. Interested readers are referred to [Liu *et al.*, 2017a].

Although the unification of clustering and imputation into a single procedure is elegant, it is implemented via *globally maximizing the alignment between the combined kernel matrix* $\mathbf{K}_{\boldsymbol{\beta}}$ *and the ideal kernel matrix* $\mathbf{H}\mathbf{H}^\top$, as shown in Eq.(1). This criterion inappropriately exploits the local distribution of data and indiscriminately forces closer and farther sample pairs to be equally aligned to the same ideal similarity. As a result, this could make these base kernels less effectively utilized, and in turn adversely affect the clustering performance. In the following, we design a novel algorithm, termed localized incomplete multiple kernel k-means (LI-MKKM) to address these issues.

3 The Proposed LI-MKKM

Although it is well recognized that preserving the local structure of data is crucial in unsupervised learning tasks such as clustering analysis [Liu *et al.*, 2014], it is not sufficiently considered in MKKM with incomplete kernel matrices. In light of this, we propose to incorporate the local structure of data by locally aligning the similarity of each sample to its k-nearest neighbours with corresponding ideal kernel matrix, which is flexible and able to well handle the intra-cluster variations.

Let $\mathcal{N}^{(i)} \in \{0,1\}^{n\times\tau} (1 \leq i \leq n)$ denote the neighborhood indication matrices of the i-th sample. For example, $\mathcal{N}_{jv}^{(i)} = 1$ denotes $\mathbf{x}_j$ is the v-th nearest neighbor of $\mathbf{x}_i$, where $1 \leq v \leq \tau$ and τ is the number of nearest neighbors. The local kernel alignment for the i-th sample is calculated as $\langle (\mathcal{N}^{(i)})^\top \mathbf{K}_{\boldsymbol{\beta}} \mathcal{N}^{(i)}, (\mathcal{N}^{(i)})^\top (\mathbf{I} - \mathbf{H}\mathbf{H}^\top) \mathcal{N}^{(i)} \rangle$. As seen, this local kernel alignment takes a sub-matrix corresponding to its neighbors from the whole $\mathbf{K}_{\boldsymbol{\beta}}$, and let it align with the ideal sub-matrix. By taking over the local kernel alignment for each sample and defining $\mathbf{B}^{(i)} = \mathcal{N}^{(i)} \mathcal{N}^{(i)^\top}$, we obtain the objective of localized incomplete MKKM (LI-MKKM) as follows,

$$\begin{aligned}\min_{\boldsymbol{\beta},\,\{\mathbf{K}_p\}_{p=1}^m,\mathbf{H}}\; & \sum\nolimits_{i=1}^n \mathrm{Tr}(\mathbf{K}_{\boldsymbol{\beta}}(\mathbf{B}^{(i)}-\mathbf{B}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{B}^{(i)}))\\ s.t.\;\; & \mathbf{H}\in\mathbb{R}^{n\times k},\;\mathbf{H}^\top\mathbf{H}=\mathbf{I}_k,\;\boldsymbol{\beta}^\top\mathbf{1}_m=1,\;\beta_p\geq 0,\\ & \mathbf{K}_p(\mathbf{s}_p,\mathbf{s}_p)=\mathbf{K}_p^{(cc)},\;\mathbf{K}_p\succeq 0,\;\forall p.\end{aligned} \tag{2}$$

In the following, we carefully design a three-step algorithm to solve the optimization problem in Eq.(2).

Optimizing $\mathbf{H}$ with fixed $\boldsymbol{\beta}$ and $\{\mathbf{K}_p\}_{p=1}^m$

Given $\boldsymbol{\beta}$ and $\{\mathbf{K}_p\}_{p=1}^m$, the optimization w.r.t $\mathbf{H}$ in Eq.(2) reduces to

$$\max_{\mathbf{H}}\;\mathrm{Tr}(\mathbf{H}^\top\sum\nolimits_{i=1}^n(\mathbf{B}^{(i)}\mathbf{K}_{\boldsymbol{\beta}}\mathbf{B}^{(i)})\mathbf{H})\quad s.t.\quad \mathbf{H}^\top\mathbf{H}=\mathbf{I}_k, \tag{3}$$

which is a conventional kernel k-means optimization problem and readily solved by existing off-the-shelf packages.

Optimizing $\{\mathbf{K}_p\}_{p=1}^m$ with fixed $\boldsymbol{\beta}$ and $\mathbf{H}$

Given $\boldsymbol{\beta}$ and $\mathbf{H}$, the optimization w.r.t $\{\mathbf{K}_p\}_{p=1}^m$ in Eq.(2) is

$$\begin{aligned}\min_{\{\mathbf{K}_p\}_{p=1}^m}\; & \sum\nolimits_{p=1}^m \beta_p^2\mathrm{Tr}(\mathbf{K}_p\sum\nolimits_{i=1}^n \mathrm{Tr}(\mathbf{B}^{(i)}-\mathbf{B}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{B}^{(i)}))\\ s.t.\;\; & \mathbf{K}_p(\mathbf{s}_p,\mathbf{s}_p)=\mathbf{K}_p^{(cc)},\;\mathbf{K}_p\succeq 0,\;\forall p,\end{aligned} \tag{4}$$

Directly solving the optimization problem in Eq.(4) appears to be computationally intractable because it involves multiple kernel matrices. Looking into this optimization problem, we can find that the constraints are separately defined on each $\mathbf{K}_p$ and that the objective function is a sum over each $\mathbf{K}_p$. Therefore, the problem in Eq.(4) can be equivalently rewritten as m independent sub-problems, as stated in Eq.(5),

$$\min_{\mathbf{K}_p}\;\mathrm{Tr}(\mathbf{K}_p\mathbf{T})\quad s.t.\quad \mathbf{K}_p(\mathbf{s}_p,\mathbf{s}_p)=\mathbf{K}_p^{(cc)},\,\mathbf{K}_p\succeq 0. \tag{5}$$

where $\mathbf{T}=\sum_{i=1}^n(\mathbf{B}^{(i)}-\mathbf{B}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{B}^{(i)})$.

At the first glance, it seems that the equality and PSD constraints imposed on $\mathbf{K}_p$ make Eq.(5) be difficult to solve. To efficiently solve this problem, we propose to parameterize each $\mathbf{K}_p$ as

$$\mathbf{K}_p=\begin{bmatrix}\mathbf{K}_p^{(cc)} & \mathbf{K}_p^{(cc)}\mathbf{W}_p\\ \mathbf{W}_p^\top\mathbf{K}_p^{(cc)} & \mathbf{W}_p^\top\mathbf{K}_p^{(cc)}\mathbf{W}_p\end{bmatrix}, \tag{6}$$

where $\mathbf{W}_p\in\mathbb{R}^{c\times m}$. The missing kernel entries are assumed to be represented by the observed ones. The following Theorem 3.1 shows that $\mathbf{K}_p$ in Eq.(6) satisfies both constraints by this parametrization.

Theorem 3.1 *$\mathbf{K}_p$ in Eq.(6) is a feasible set of the optimization problem Eq.(5).*

Proof 1 *Firstly, it is not difficult to check that the equality constraint is satisfied. For any vector $\mathbf{x}\in\mathbb{R}^n$, we have $\mathbf{x}=[\mathbf{x}_c^\top,\mathbf{x}_m^\top]^\top$. We have*

$$\begin{aligned}\mathbf{x}^\top\mathbf{K}_p\mathbf{x}=&[\mathbf{x}_c^\top,\mathbf{x}_m^\top]\begin{bmatrix}\mathbf{K}_p^{(cc)} & \mathbf{K}_p^{(cc)}\mathbf{W}_p\\ \mathbf{W}_p^\top\mathbf{K}_p^{(cc)} & \mathbf{W}_p^\top\mathbf{K}_p^{(cc)}\mathbf{W}_p\end{bmatrix}[\mathbf{x}_c^\top,\mathbf{x}_m^\top]^\top\\ =&(\mathbf{x}_c+\mathbf{W}_p\mathbf{x}_m)^\top\mathbf{K}_p^{(cc)}(\mathbf{x}_c+\mathbf{W}_p\mathbf{x}_m)\geq 0.\end{aligned} \tag{7}$$

This verifies the parametrization of $\mathbf{K}_p$ is PSD. It completes the proof.

Based on Theorem 3.1, the optimization problem in Eq.(5) is equivalent to the following unconstrained one,

$$\min\nolimits_{\mathbf{W}_p}\;\mathrm{Tr}\left(\begin{bmatrix}\mathbf{K}_p^{(cc)} & \mathbf{K}_p^{(cc)}\mathbf{W}_p\\ \mathbf{W}_p^\top\mathbf{K}_p^{(cc)} & \mathbf{W}_p^\top\mathbf{K}_p^{(cc)}\mathbf{W}_p\end{bmatrix}\begin{bmatrix}\mathbf{T}^{(cc)} & \mathbf{T}^{(cm)}\\ \mathbf{T}^{(cm)\top} & \mathbf{T}^{(mm)}\end{bmatrix}\right), \tag{8}$$

where the matrix $\mathbf{T}$ is expressed in a blocked form as $\begin{bmatrix}\mathbf{T}^{(cc)} & \mathbf{T}^{(cm)}\\ \mathbf{T}^{(cm)\top} & \mathbf{T}^{(mm)}\end{bmatrix}$.

By taking the derivative of Eq.(8) with respect to $\mathbf{W}_p$ and letting it vanish, we can obtain

$$\mathbf{W}_p=-\mathbf{T}^{(cm)}(\mathbf{T}^{(mm)})^{-1}. \tag{9}$$

By substituting $\mathbf{W}_p$ in Eq.(9) into Eq.(6), we have a closed-form expression for the optimal $\mathbf{K}_p$. It is worth pointing out that Theorem 3.1 provides a general parametrization model to impute incomplete kernel matrices. In fact, the imputation in MKKM-IK [Liu *et al.*, 2017a] can be treated as a special case of Eq.(6). Some regularization such as low-rank constraint on $\mathbf{W}_p$ in Eq.(6) will be incorporated to further improve the clustering performance in the future work.

As observed, Eq.(5) exploits the local structure of data via $\mathbf{T}$ to guide the imputation of each base kernel matrix. It locally aligns the similarity of each sample to its τ-nearest neighbors with corresponding ideal kernel matrix, which is flexible and able to well handle the intra-cluster variations. By this way, the incomplete kernels could be more effectively imputed, leading to improved clustering performance.

Optimizing $\boldsymbol{\beta}$ with fixed $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$

Given $\{\mathbf{K}_p\}_{p=1}^m$ and $\mathbf{H}$, it is not difficult to show that Eq.(2) w.r.t. $\boldsymbol{\beta}$ is as follows,

$$\min_{\boldsymbol{\beta}}\;\sum\nolimits_{p=1}^m z_p\beta_p^2\quad s.t.\;\;\boldsymbol{\beta}^\top\mathbf{1}_m=1,\,\beta_p\geq 0, \tag{10}$$

where $z_p=\mathrm{Tr}(\mathbf{K}_p\mathbf{V})$ and $\mathbf{V}=\sum_{i=1}^n(\mathbf{A}^{(i)}-\mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})$.

The optimization in Eq.(10) has a closed-form solution if $z_p\geq 0\,(1\leq p\leq m)$. The following Theorem 3.2 shows that this optimization problem can be analytical solved.

Theorem 3.2 *The optimization in Eq.(10) has a closed-form solution as follows,*

$$\beta_p=w_p/\sum\nolimits_{p=1}^m w_p,\;\forall p, \tag{11}$$

where $w_p=1/z_p$.

Proof 2 *By denoting $\mathbf{H}=[\mathbf{h}_1,\cdots,\mathbf{h}_k]$, we can see that $\mathbf{H}\mathbf{H}^\top\mathbf{h}_c=\mathbf{h}_c(1\leq c\leq k)$ since $\mathbf{H}^\top\mathbf{H}=\mathbf{I}_k$. This means $\mathbf{H}\mathbf{H}^\top$ has k eigenvalue with 1. Meanwhile, its rank is no more than k, which implies its has $n-k$ eigenvalue with 0. Correspondingly, $\mathbf{I}_n-\mathbf{H}\mathbf{H}^\top$ has $n-k$ and k eigenvalue with 1 and 0. As a result, $\mathbf{A}^{(i)}(\mathbf{I}_n-\mathbf{H}\mathbf{H}^\top)\mathbf{A}^{(i)}$ is PSD, and this guarantees that $\mathbf{V}=\sum_{i=1}^n(\mathbf{A}^{(i)}-\mathbf{A}^{(i)}\mathbf{H}\mathbf{H}^\top\mathbf{A}^{(i)})$ is PSD. Therefore, we have $z_p=\mathrm{Tr}(\mathbf{K}_p\mathbf{V})\geq 0,\;\forall p$ since both $\mathbf{K}_p$ and $\mathbf{V}$ are PSD. The proof is completed by taking the derivative of the Lagrangian function of Eq.(10) on β_p and letting it vanish.*

In sum, our algorithm for solving Eq.(2) is outlined in Algorithm 1, where the absent elements of $\{\mathbf{K}_p^{(0)}\}_{p=1}^m$ are initially imputed with zeros and $\mathrm{obj}^{(t)}$ denotes the objective value at the t-th iteration. It is worth pointing out that the neighborhood of each sample is kept unchanged during the optimization. The τ-nearest neighbors of each sample are measured by $\mathbf{K}_{\boldsymbol{\beta}^{(0)}}$. By doing so, the objective of Algorithm 1 is guaranteed to be monotonically decreased when optimizing one variable with the others fixed at each iteration. At the same time, the objective is lower-bounded by zero. As a result, our algorithm is guaranteed to converge to a local minimum. Also, as shown in the experimental study, it usually converges in less than 10 iterations.

Algorithm 1 Proposed LI-MKKM

```
**Input**: $\{\mathbf{K}_p^{(cc)}, \mathbf{s}_p\}_{p=1}^m$, $k$, $\tau$ and $\epsilon_0$.
**Output**: $\mathbf{H}$, $\boldsymbol{\beta}$ and $\{\mathbf{K}_p\}_{p=1}^m$.
Initialize $\boldsymbol{\beta}^{(0)} = \mathbf{1}_m/m$, $\{\mathbf{K}_p^{(0)}\}_{p=1}^m$ and $t = 1$.
Generating $\mathbf{S}^{(i)}$ for $i$-th samples $(1 \leq i \leq n)$ by $\mathbf{K}_{\boldsymbol{\beta}^{(0)}}$.
**repeat**
    $\mathbf{K}_{\boldsymbol{\beta}^{(t)}} = \sum_{p=1}^m \left(\beta_p^{(t-1)}\right)^2 \mathbf{K}_p^{(t-1)}$.
    Update $\mathbf{H}^{(t)}$ by solving Eq.(3) with $\mathbf{K}_{\boldsymbol{\beta}^{(t)}}$.
    Update $\{\mathbf{K}_p^{(t)}\}_{p=1}^m$ with $\mathbf{H}^{(t)}$ and $\boldsymbol{\beta}^{(t)}$ by Eq.(5).
    Update $\boldsymbol{\beta}^{(t)}$ by solving Eq.(10) with $\mathbf{H}^{(t)}$ and $\{\mathbf{K}_p^{(t)}\}_{p=1}^m$.
    $t = t + 1$.
**until** $\left(\mathrm{obj}^{(t-1)} - \mathrm{obj}^{(t)}\right)/\mathrm{obj}^{(t)} \leq \epsilon_0$
```

We end up this section by discussion the computational complexity of the proposed algorithm. Specifically, the complexity of our algorithm is $\mathcal{O}(n^3 + \sum_{p=1}^m n_p^3)$ per iteration, where n_p $(n_p \leq n)$ is the number of observed samples of $\mathbf{K}_p$. It is comparable to the case of existing MKKM-IK [Liu *et al.*, 2017a]. Furthermore, it is worth pointing out that $\mathbf{K}_p$ can be trivially calculated in a parallel way, because each of them is independent. In this way, our algorithm can scale well with respect to the number of base kernels. Meanwhile, although the kernel alignment is optimized in a localized way, the resultant computational complexity is not altered significantly and brings not much extra computation when compared with existing MKKM-IK [Liu *et al.*, 2017a], as validated by the running time comparison in Table 3.

4 Experiments

4.1 Experimental settings

The proposed algorithm is experimentally evaluated on eight widely used MKL benchmark data sets shown in Table 2. They are Oxford Flower17 and Flower102[1] and Caltech102[2]. For these datasets, all kernel matrices are pre-computed and can be publicly downloaded from the above websites. Meanwhile, Caltech102-5 means the number of samples belonging to each cluster is 5, and so on. We compare the proposed algorithm with several commonly used imputation methods, including zero filling (ZF), mean filling (MF), k-nearest-neighbor filling (KNN) and the alignment-maximization filling (AF) proposed in [Trivedi *et al.*, 2010]. The widely used MKKM [Gönen and Margolin, 2014] is then applied with these imputed base kernels. These two-stage methods are termed MKKM+ZF, MKKM+MF, MKKM+KNN and MKKM+AF, respectively. In addition, we compare with the newly proposed MKKM-IK [Liu *et al.*, 2017a], which jointly optimizes the imputation and clustering. We donot incorporate the algorithms in [Xu *et al.*, 2015b; Shao *et al.*, 2015; Zhao *et al.*, 2016] into our experimental comparison since they only consider the absence of input features while not the rows/columns of base kernels. For all data sets, it is assumed that the true number of clusters is known and it is set as the true number of classes. We follow the approach in [Liu *et al.*, 2017a] to generate the missing vectors $\{\mathbf{s}_p\}_{p=1}^m$. The parameter ε, termed missing ratio in this experiment, controls the percentage of samples that have absent views, and it affects the performance of the algorithms in comparison. In order to show this point in depth, we compare these algorithms with respect to ε. Specifically, ε on all the four data sets is set as $[0.1 : 0.1 : 0.9]$.

Table 2: Datasets used in our experiments.

Dataset	#Samples	#Kernels	#Classes
Flower17	1360	7	17
Flower102	8189	4	102
Caltech102-5	510	48	102
Caltech102-10	1020	48	102
Caltech102-15	1530	48	102
Caltech102-20	2040	48	102
Caltech102-25	2550	48	102
Caltech102-30	3060	48	102

The widely used clustering accuracy (ACC) and normalized mutual information (NMI) are applied to evaluate the clustering performance. For all algorithms, we repeat each experiment for 50 times with random initialization to reduce the effect of randomness caused by k-means, and report the best result. Meanwhile, we randomly generate the "incomplete" patterns for 10 times in the above-mentioned way and report the statistical results. The aggregated ACC and NMI are used to evaluate the goodness of the algorithms in comparison. Taking the aggregated ACC for example, it is obtained by averaging the averaged ACC achieved by an algorithm over different ε.

4.2 Experimental results

Figure 1 presents the ACC and NMI comparison of the above algorithms with different missing ratios on Flower17, Flower102, Caltech102-25 and Caltech102-30 datasets. We have the following observations: i) The recently proposed MKKM-IK [Liu *et al.*, 2017a] (in blue) achieves compara-

[1] `http://www.robots.ox.ac.uk/~vgg/data/flowers/`

[2] `http://files.is.tue.mpg.de/pgehler/projects/iccv09/`

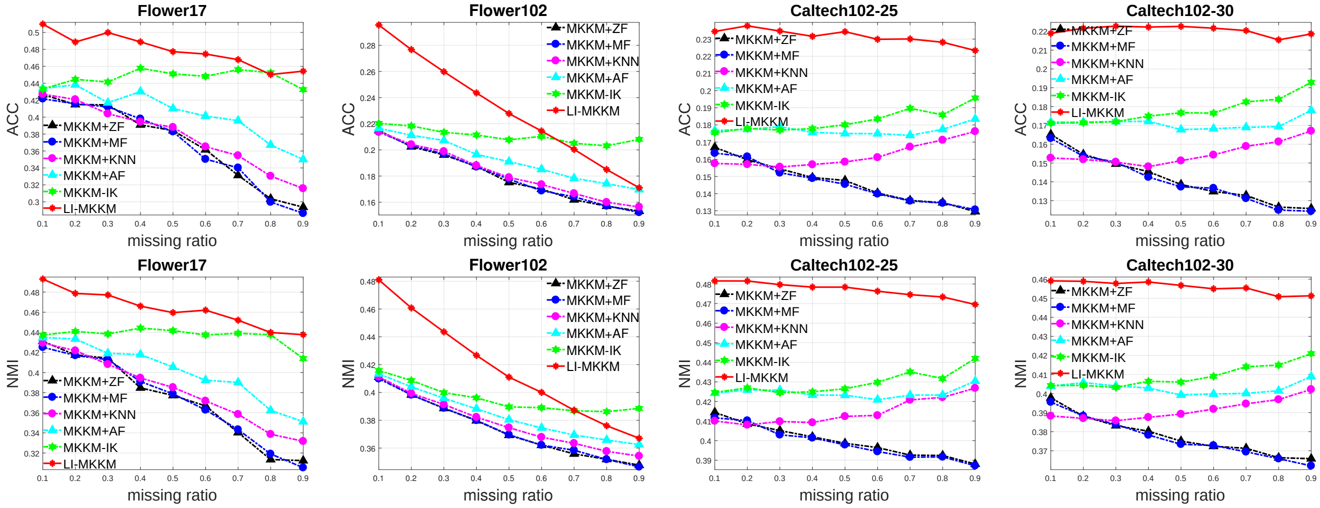


Figure 1: ACC and NMI comparison with the variation of missing ratios on Flower17, Flower102, Caltech102-25 and Caltech102-30 datasets. The curves on other datasets are similar and omitted due to space limit.

Table 1: Aggregated ACC and NMI comparison (mean±std) of different clustering algorithms on eight benchmark datasets.

Datasets	MKKM+ZF	MKKM+MF	MKKM+KNN	MKKM+AF	MKKM-IK	LI-MKKM
				[Trivedi *et al.*, 2010]	[Liu *et al.*, 2017a]	Proposed
			ACC			
Flower17	36.9 ± 0.8	36.8 ± 0.6	37.8 ± 0.6	40.5 ± 0.7	44.6 ± 0.6	$\mathbf{47.9 \pm 0.3}$
Flower102	18.0 ± 0.2	18.0 ± 0.2	18.2 ± 0.1	19.2 ± 0.1	21.1 ± 0.2	$\mathbf{23.1 \pm 0.2}$
Calt102-5	26.1 ± 0.3	25.7 ± 0.3	27.3 ± 0.3	29.0 ± 0.3	28.9 ± 0.3	$\mathbf{31.3 \pm 0.3}$
Calt102-10	19.7 ± 0.2	19.7 ± 0.2	21.5 ± 0.2	22.6 ± 0.2	22.7 ± 0.2	$\mathbf{27.1 \pm 0.3}$
Calt102-15	17.1 ± 0.2	17.1 ± 0.2	18.9 ± 0.1	20.3 ± 0.2	20.8 ± 0.2	$\mathbf{25.0 \pm 0.2}$
Calt102-20	15.7 ± 0.1	15.7 ± 0.2	17.3 ± 0.2	18.9 ± 0.2	19.5 ± 0.2	$\mathbf{24.0 \pm 0.2}$
Calt102-25	14.7 ± 0.2	14.6 ± 0.1	16.2 ± 0.1	17.7 ± 0.2	18.3 ± 0.2	$\mathbf{23.2 \pm 0.2}$
Calt102-30	14.2 ± 0.1	14.1 ± 0.1	15.5 ± 0.2	17.1 ± 0.2	17.8 ± 0.2	$\mathbf{22.1 \pm 0.2}$
			NMI			
Flower17	37.3 ± 0.4	37.3 ± 0.5	38.2 ± 0.5	40.1 ± 0.4	43.7 ± 0.3	$\mathbf{46.3 \pm 0.2}$
Flower102	37.4 ± 0.1	37.4 ± 0.1	37.8 ± 0.1	38.4 ± 0.1	39.6 ± 0.1	$\mathbf{41.7 \pm 0.1}$
Calt102-5	64.3 ± 0.2	63.9 ± 0.1	65.9 ± 0.2	66.6 ± 0.1	66.5 ± 0.2	$\mathbf{67.0 \pm 0.1}$
Calt102-10	53.6 ± 0.1	53.7 ± 0.1	55.2 ± 0.1	55.7 ± 0.2	55.8 ± 0.1	$\mathbf{58.6 \pm 0.1}$
Calt102-15	47.4 ± 0.1	47.4 ± 0.1	48.8 ± 0.1	49.7 ± 0.1	50.1 ± 0.1	$\mathbf{53.5 \pm 0.1}$
Calt102-20	43.1 ± 0.1	43.1 ± 0.2	44.5 ± 0.1	45.6 ± 0.2	46.0 ± 0.1	$\mathbf{50.3 \pm 0.1}$
Calt102-25	40.0 ± 0.1	39.9 ± 0.1	41.5 ± 0.1	42.5 ± 0.2	43.0 ± 0.2	$\mathbf{47.7 \pm 0.1}$
Calt102-30	37.8 ± 0.1	37.7 ± 0.1	39.2 ± 0.1	40.3 ± 0.1	40.9 ± 0.1	$\mathbf{45.6 \pm 0.1}$

Table 3: Running time comparison of the aforementioned algorithms on all datasets (in seconds).

Dataset	MKKM+ZF	MKKM+MF	MKKM+KNN	MKKM+AF	MKKM-IK	LI-MKKM
				[Trivedi *et al.*, 2010]	[Liu *et al.*, 2017a]	Proposed
Flower17	2.49 ± 0.04	2.42 ± 0.05	3.36 ± 0.05	2.95 ± 0.08	7.25 ± 0.49	6.31 ± 0.08
Flower102	193.50 ± 4.50	210.22 ± 12.41	329.49 ± 27.59	239.97 ± 9.43	415.98 ± 7.71	418.08 ± 10.91
Caltech102-5	3.81 ± 0.13	3.75 ± 0.11	6.36 ± 0.14	4.33 ± 0.23	31.89 ± 3.25	8.08 ± 0.12
Caltech102-10	14.68 ± 0.31	14.80 ± 0.41	26.70 ± 0.53	16.41 ± 0.14	71.22 ± 16.27	36.79 ± 0.20
Caltech102-15	58.63 ± 0.31	59.12 ± 0.16	86.52 ± 2.19	60.99 ± 2.68	202.33 ± 28.91	143.70 ± 0.73
Caltech102-20	119.48 ± 7.28	118.31 ± 5.79	204.80 ± 18.23	130.82 ± 16.29	335.08 ± 42.03	290.62 ± 5.41
Caltech102-25	235.52 ± 11.31	220.39 ± 7.87	395.45 ± 25.46	215.56 ± 6.32	599.44 ± 87.33	537.31 ± 1.59
Caltech102-30	370.85 ± 17.21	367.84 ± 28.16	648.88 ± 36.59	360.89 ± 16.90	874.63 ± 28.56	837.72 ± 15.01

ble or better clustering performance when compared with existing two-stage imputation methods. These results verify the effectiveness of the joint optimization on imputation and clustering. ii) The proposed LI-MKKM (in red) consistently and significantly further improves the clustering performance of MKKM-IK, as shown in all subfigures from Fig.1(a) to 1(h). This clearly demonstrates the advantage of well utilizing the local structure of data. iii) The improvement of our algorithm is more significant with the decrease of missing ratios. For example, it improves the second best algorithm (MKKM-IK) by 5 percentage points on Caltech102-30 in terms of clustering accuracy when the missing ratio is 0.1 (see Figure 1(d)).

We attribute the superiority of our algorithm as two aspects: i) *Well utilizing the local structure of data*. Our local kernel alignment criterion is flexible and allows the pre-specified kernels to be aligned for better clustering, making the pre-specified kernels be well utilized; and ii) *The joint optimization on imputation and clustering*. On one hand, the imputation is guided by the clustering results, which makes the imputation more directly targeted at the ultimate goal. On the other hand, this meaningful imputation is beneficial to refine the clustering results. These two learning processes negotiate with each other, leading to improved clustering performance. In contrast, MKKM+ZF, MKKM+MF, MKKM+KNN and MKKM+AF algorithms do not fully take advantage of the connection between the imputation and clustering procedures. This could produce imputation that does not well serve the subsequent clustering as originally expected, affecting the clustering performance. Both factors bring the significant improvements on clustering performance.

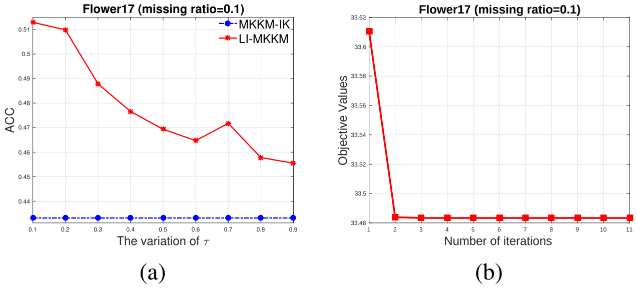
We also report the aggregated ACC and NMI, and the standard deviation in Table 1, where the best and second ones are marked in red and blue color, respectively. Again, we observe that the proposed algorithm significantly outperforms MKKM+ZF, MKKM+MF, MKKM+KNN, MKKM+AF and MKKM-IK. For example, our algorithm exceeds the second best one (MKKM-IK) by nearly 2.3%, 4.4%, 4.2%, 4.5%, 4.9%, 4.3% in terms of clustering accuracy on Caltech102, respectively. These results are consistent with our observations in Figure 1. Meanwhile, we observe that LF-MKKM is inferior to MKKM-IK in the presence of intensive missing ratios. We conjecture that the neighbors of each sample calculated by $\mathbf{K}_{\boldsymbol{\beta}^{(0)}}$ may no longer be reliable when the missing ratio is relatively intensive, and this adversely affects the clustering performance. We plan to further explore this point in the future work.

From the above experiments, we conclude that the proposed algorithm well exploits the local structure of data, bringing forth significant improvements on clustering performance.

As aforementioned, LI-MKKM is with comparable computational complexity with existing MKKM-IK [Liu *et al.*, 2017a]. In Table 3, we report their running time (in seconds) on eight datasets. As observed, LI-MKKM slightly improves the running time of MKKM-IK. This is because LI-MKKM requires less iterations to achieve the same convergence criterion compared with MKKM-IK. For example, LI-MKKM needs five iterations to satisfy the criterion, while this number is 14 for MKKM-IK. This motivates us to study the impact of incorporating the local structure of data on the algorithm convergence in the future.

4.3 Parameter Sensitivity and Convergence

As can be seen in Eq.(2), LI-MKKM introduces the number of neighbors τ as an extra parameter. In all the above experiments, we empirically set $\tau = 0.1$ for all datasets, and observe that our algorithm demonstrate very promising performance. In the following, we conduct experiments to show the effect of this parameter on the performance of LI-MKKM on Flower17 dataset.



(a) (b)

Figure 2: The sensitivity of LI-MKKM with the variation of τ (missing ratio=0.1), and its convergence on Flower17.

Figure 2(a) plots the ACC of LI-MKKM by varying τ in a large range $[0.1, 0.2, \cdots, 0.9] * n$ on Flower17. The NMI of MKKM-IK is also provided as a baseline. From the figure, we observe that: i) the NMI monotonically decreases with the increase of τ, clearly demonstrating the effectiveness of preserving the local structure of data; and ii) LI-MKKM significantly outperforms the recently proposed MKKM-IK and shows stable performance across a wide range of τ. These results demonstrate that LI-MKKM is stable across a wide range of τ. The curves on other datasets are similar and omitted due to space limit.

LI-MKKM is theoretically guaranteed to converge to a local minimum. In the above experiments, we observe that the objective value of our algorithm does monotonically decrease at each iteration and that it usually converges in less than 10 iterations. One example of the evolution of the objective value on Flower17 are demonstrated in Figure 2(b).

5 Conclusion

While the recently proposed MKKM-IK is able to handle multi-kernel clustering with incomplete kernels, it does not sufficiently utilize the local structure of data, which is crucial for clustering analysis. This paper proposes LF-MKKM to calculate the kernel alignment in a local manner to address this issue. LF-MKKM effectively solves the resultant optimization problem, and demonstrates well improved clustering performance via extensive experiments on benchmark datasets. In the future, we plan to explore the parametrization on incomplete kernel matrices in order to further improve its computational complexity and clustering performance.

Acknowledgements

This work was supported by the Natural Science Foundation of China (project no. 61672528).

References

[Bhadra *et al.*, 2016] Sahely Bhadra, Samuel Kaski, and Juho Rousu. Multi-view kernel completion. In *arXiv:1602.02518*, 2016.

[Cai *et al.*, 2013] Xiao Cai, Feiping Nie, Weidong Cai, and Heng Huang. Heterogeneous image features integration via multi-modal semi-supervised learning model. In *ICCV*, pages 1737–1744, 2013.

[Cao *et al.*, 2015] Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. Diversity-induced multi-view subspace clustering. In *CVPR*, pages 586–594, 2015.

[Gao *et al.*, 2015] Hongchang Gao, Feiping Nie, Xuelong Li, and Heng Huang. Multi-view subspace clustering. In *ICCV*, pages 4238–4246, 2015.

[Ghahramani and Jordan, 1993] Zoubin Ghahramani and Michael I. Jordan. Supervised learning from incomplete data via an EM approach. In *NIPS*, pages 120–127, 1993.

[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.

[Kumar *et al.*, 2013] Ritwik Kumar, Ting Chen, Moritz Hardt, David Beymer, Karen Brannon, and Tanveer Fathima Syeda-Mahmood. Multiple kernel completion and it-s application to cardiac disease discrimination. In *ISBI*, pages 764–767, 2013.

[Li *et al.*, 2014] Shao-Yuan Li, Yuan Jiang, and Zhi-Hua Zhou. Partial multi-view clustering. In *AAAI*, pages 1968–1974, 2014.

[Li *et al.*, 2016] Miaomiao Li, Xinwang Liu, Lei Wang, Y-ong Dou, Jianping Yin, and En Zhu. Multiple kernel clustering with local kernel alignment maximization. In *IJCAI*, pages 1704–1710, 2016.

[Liu *et al.*, 2014] Xinwang Liu, Lei Wang, Jian Zhang, Jianping Yin, and Huan Liu. Global and local structure p-reservation for feature selection. *IEEE Trans. Neural Netw. Learning Syst.*, 25(6):1083–1095, 2014.

[Liu *et al.*, 2016] Xinwang Liu, Yong Dou, Jianping Yin, Lei Wang, and En Zhu. Multiple kernel k-means clustering with matrix-induced regularization. In *AAAI*, pages 1888–1894, 2016.

[Liu *et al.*, 2017a] Xinwang Liu, Miaomiao Li, Lei Wang, Yong Dou, Jianping Yin, and En Zhu. Multiple kernel k-means with incomplete kernels. In *AAAI*, pages 2259–2265, 2017.

[Liu *et al.*, 2017b] Xinwang Liu, Sihang Zhou, Yueqing Wang, Miaomiao Li, Yong Dou, En Zhu, and Jianping Yin. Optimal neighborhood kernel clustering with multiple kernels. In *AAAI*, pages 2266–2272, 2017.

[Nie *et al.*, 2014] Feiping Nie, Xiaoqian Wang, and Heng Huang. Clustering and projected clustering with adaptive neighbors. In *KDD*, pages 977–986, 2014.

[Shao *et al.*, 2015] Weixiang Shao, Lifang He, and Philip S. Yu. Multiple incomplete views clustering via weighted nonnegative matrix factorization with $\ell_{2,1}$ regularization. In *ECML PKDD*, pages 318–334, 2015.

[Trivedi *et al.*, 2010] Anusua Trivedi, Piyush Rai, Hal Daumé III, and Scott L. DuVall. Multiview clustering with incomplete views. In *NIPS 2010: Machine Learning for Social Computing Workshop ,Whistler, Canada*, 2010.

[Xiang *et al.*, 2013] Shuo Xiang, Lei Yuan, Wei Fan, Yalin Wang, Paul M. Thompson, and Jieping Ye. Multi-source learning with block-wise missing data for alzheimer's disease prediction. In *ACM SIGKDD*, pages 185–193, 2013.

[Xu *et al.*, 2015a] Chang Xu, Dacheng Tao, and Chao Xu. Multi-view intact space learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(12):2531–2544, 2015.

[Xu *et al.*, 2015b] Chang Xu, Dacheng Tao, and Chao Xu. Multi-view learning with incomplete views. *IEEE Trans. Image Processing*, 24(12):5812–5825, 2015.

[Yin *et al.*, 2015] Qiyue Yin, Shu Wu, and Liang Wang. Incomplete multi-view clustering via subspace learning. In *CIKM*, pages 383–392, 2015.

[Yu *et al.*, 2012] Shi Yu, Léon-Charles Tranchevent, Xinhai Liu, Wolfgang Glänzel, Johan A. K. Suykens, Bart De Moor, and Yves Moreau. Optimized data fusion for k-ernel k-means clustering. *IEEE TPAMI*, 34(5):1031–1039, 2012.

[Zhang *et al.*, 2015] Changqing Zhang, Huazhu Fu, Si Liu, Guangcan Liu, and Xiaochun Cao. Low-rank tensor constrained multiview subspace clustering. In *ICCV*, pages 1582–1590, 2015.

[Zhao *et al.*, 2016] Handong Zhao, Hongfu Liu, and Yun Fu. Incomplete multimodal visual data grouping. In *IJCAI*, pages 2392–2398, 2016.