DISCRETE MULTI-KERNEL K-MEANS WITH DIVERSE AND OPTIMAL KERNEL LEARNING

Yihang Lu[1,2] *Jitao Lu*[1,2] *Rong Wang*[2*] *Feiping Nie*[1,2]

[1] School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.
[2] School of Artificial Intelligence, Optics and Electronics (iOPEN), and the Key Laboratory of
Intelligent Interaction and Applications (Ministry of Industry and Information Technology),
Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.

ABSTRACT

Multiple Kernel k-means and its variants integrate a group of kernels to improve clustering performance, but it still has some drawbacks: 1) linearly combining base kernels to get the optimal one limits the kernel representability and cuts off the negotiation of kernel learning and clustering; 2) ignoring the correlation among kernels leads to kernel redundancy; 3) solving NP-hard cluster assignment problem by a two-stage strategy leads to information loss. In this paper, we propose the Discrete Multi-kernel k-means with Diverse and Optimal Kernel Learning (DMK-DOK) model, which adaptively seeks for a better kernel by residing in the base kernel neighborhood and negotiates the kernel learning and clustering. Moreover, it implicitly penalizes the highly correlated kernels to enhance the kernel fusion with less redundancy and more diversity. What's more, it jointly learns discrete and relaxed labels in the same optimization objective, which can avoid information loss. Lastly, extensive experiments conducted on real-world datasets illustrated the superiority of our model.

Index Terms— Kernel method, Multiple Kernel Clustering, Multiple Kernel k-means.

1. INTRODUCTION

K-means [1] enjoys huge popularity in clustering, but it cannot work well if the clusters are not linearly separable [2–4]. To address the issue, kernel k-means(KKM) [5, 6] maps data sample $\boldsymbol{x_i}$ to a reproducing kernel hilbert space (RKHS) [7] via a kernel mapping $\phi(\cdot)$, which can be expressed as

$$\min_{\mathbf{Y}\in\text{Ind}} \sum_{i=1}^{n}\sum_{j=1}^{c} y_{ij}\|\phi(\boldsymbol{x}_i)-\boldsymbol{\mu}_j\|_2^2, \tag{1}$$

where n, c are the total number of samples and clusters respectively, and $\boldsymbol{\mu}_j$ is the j-th cluster centroid. Denote $\mathbf{Y}\in$ Ind as the cluster indicator matrix, and its $\langle i,j\rangle$-th element is y_{ij}. If $\boldsymbol{x_i}$ belongs to the j-th cluster, $y_{ij}=1$ otherwise $y_{ij}=0$. The $\langle i,j\rangle$-th element of kernel matrix $\mathbf{K}$ is $\kappa(i,j)=\phi^T(\boldsymbol{x}_i)\phi(\boldsymbol{x}_j)$, so the matrix-vector form of problem (1) is

$$\min_{\mathbf{Y}\in\text{Ind}} \text{Tr}(\mathbf{K}(\mathbf{I}_n-\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T)), \tag{2}$$

which is NP-hard due to the discrete constraints of $\mathbf{Y}$. Let $\mathbf{F}=\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}$ and relax $\mathbf{F}$ to take arbitrary real values as

$$\min_{\mathbf{F}^T\mathbf{F}=\mathbf{I}} \text{Tr}(\mathbf{K}(\mathbf{I}_n-\mathbf{F}\mathbf{F}^T)), \tag{3}$$

which is improved by multiple kernel k-means (MKKM) [8] via integrating a group of kernels $\{\phi_p(\cdot)\}_{p=1}^{v}$. Denote $\mathbf{K}_{\boldsymbol{\alpha}}$ as the new kernel matrix with the $\langle i,j\rangle$-th element $\kappa_{\boldsymbol{\alpha}}(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi_{\boldsymbol{\alpha}}^T(\boldsymbol{x}_i)\phi_{\boldsymbol{\alpha}}(\boldsymbol{x}_j)=\sum_{p=1}^{v}\alpha_p^2\kappa_p(\boldsymbol{x}_i,\boldsymbol{x}_j)$, where $\boldsymbol{\alpha}=[\alpha_1,\cdots,\alpha_v]^T$ is the kernel coefficient. MKKM can be formulated as

$$\begin{gathered}\min_{\mathbf{F},\boldsymbol{\alpha}} \text{Tr}(\mathbf{K}_{\boldsymbol{\alpha}}(\mathbf{I}_n-\mathbf{F}\mathbf{F}^T)),\\ \text{s.t. } \mathbf{F}^T\mathbf{F}=\mathbf{I}, \boldsymbol{\alpha}^T\mathbf{1}=1, \alpha_p\geqslant 0, \forall p,\end{gathered} \tag{4}$$

which can be optimized alternatively: 1) $\mathbf{F}$ with $\boldsymbol{\alpha}$ fixed is defined by c eigenvectors related to the largest c eigenvalues of $\mathbf{K}_{\boldsymbol{\alpha}}$; 2) update $\boldsymbol{\alpha}$ with $\mathbf{F}$ via Quadratic Programming. Upon converging, k-means turns the relaxed $\mathbf{F}$ into discrete labels.

In recent years, lots of studies try to improve MKKM. Gönen *et al.* applied localized data fusion (LMKKM) to adaptively change the kernel coefficients [9]. Liu *et al.* used the self-adaptive kernel learning (ONKC) to improve the kernel's reproducing capability [10]. Then, Yao *et al.* employed subset selection to increase the diversity of kernels (MKKM-RK) [11]. In this paper, we propose a novel Discrete Multi-kernel k-means with Diverse and Optimal Kernel Learning (DMK-DOK) model, which adaptively seeks for the optimal kernel by residing in base kernel neighborhood and enhances the connection between kernel learning and clustering. Besides, it can improve kernel fusion with higher

[*] Corresponding author. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1403501, in part by the Natural Science Basic Research Program of Shaanxi (Program No. 2021JM-071), in part by the National Natural Science Foundation of China under Grant 62176212, Grant 61936014 and Grant 61772427, and in part by the Fundamental Research Funds for the Central Universities under Grant G2019KY0501.

diversity and less redundancy, by assigning small coefficients to highly correlated kernels. What's more, it jointly learns the discrete labels and the relaxed ones in the same objective via improved spectral rotation (ISR) [12] to guarantee a discrete solution, which can avoid information loss and overreliance on k-means. Extensive experiments on real-world datasets demonstrated the effectiveness and superiority of our model.

Notations. In matrix $\mathbf{M}$: $\boldsymbol{m}_i$ is the i-th column and m_{ij} is the $\langle i, j\rangle$-th element. u_i is the i-th element of vector $\boldsymbol{u}$. $\operatorname{Tr}(\mathbf{M}) = \sum_i m_{ii}$, $\|\mathbf{M}\|_F = \sqrt{\operatorname{Tr}(\mathbf{M}^T\mathbf{M})}$ are the trace and Frobenius norm of $\mathbf{M}$. $\mathbf{1}$ is the all one vector. $\operatorname{Ind} \triangleq \{\mathbf{Y} \in \{0,1\}^{n\times c} | \mathbf{Y}\mathbf{1} = \mathbf{1}\}$ denotes the set of indicator matrices.

2. METHODOLOGY

Most MKKM methods consider the optimal kernel as the linear combination of base kernels (*i.e.*, $\mathbf{K}_{\boldsymbol{\alpha}} = \sum_{p=1}^{v} \alpha_p \mathbf{K}_p$), so the optimal kernel is roughly placed on a hyperplane parameterized by $\boldsymbol{\alpha}$, which over reduces the feasible sets of the optimal kernel and limits its representability [13–15]. On the other hand, $\mathbf{K}_{\boldsymbol{\alpha}}$ does not sufficiently consider the effect of clustering matrix $\mathbf{F}$ on kernel learning, cutting off the negotiation between them [16–18]. Besides, the correlation among kernels is always ignored [19], which inevitably leads to kernel redundancy and weakens the complementary information. What's worse, faced with NP-hard label assigning task, most of the existing works adopt a two-stage strategy: clustering label relaxing and label discretization by k-means, which may lead to severe information loss [20, 21]. To address the issue, we propose DMK-DOK which can be expressed as

$$\begin{aligned} \min_{\mathbf{F},\mathbf{G},\mathbf{R},\mathbf{Y},\boldsymbol{\alpha}} \; & \operatorname{Tr}(\mathbf{G}(\mathbf{I}_n - \mathbf{F}\mathbf{F}^T)) + \gamma\|\mathbf{G} - \mathbf{K}_{\boldsymbol{\alpha}}\|_F^2 \\ & + \lambda\|\mathbf{F}\mathbf{R} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}\|_F^2, \\ \text{s.t. } & \mathbf{F}^T\mathbf{F} = \mathbf{I}, \mathbf{G} \succeq 0, \mathbf{R}^T\mathbf{R} = \mathbf{I}, \mathbf{Y} \in \operatorname{Ind}, \\ & \boldsymbol{\alpha}^T\mathbf{1} = 1, \alpha_p \geqslant 0, \forall p, \end{aligned} \tag{5}$$

where $\mathbf{G}$ is the optimal kernel being Positive Semi-Definite (PSD) to approximate the linear consensus kernel $\mathbf{K}_{\boldsymbol{\alpha}}$. $\mathbf{R}$ is a rotation matrix, and γ, λ are regularization parameters. Our model adaptively seeks the optimal kernel $\mathbf{G}$ in the neighborhood of $\mathbf{K}_{\boldsymbol{\alpha}}$, which enlarges the region of a more suitable kernel. In optimization, kernel learning and clustering are seamlessly coupled and negotiate with each other via Eq. (14).

In Eq. (5), $\|\mathbf{G} - \mathbf{K}_{\boldsymbol{\alpha}}\|_F^2$ implicitly includes $\operatorname{Tr}(\mathbf{K}_{\boldsymbol{\alpha}}\mathbf{K}_{\boldsymbol{\alpha}}) = \sum_{p=1}^{v}\sum_{q=1}^{v} \alpha_p\alpha_q \operatorname{Tr}(\mathbf{K}_p\mathbf{K}_q)$ measuring the correlation of $\mathbf{K}_p$ and $\mathbf{K}_q$: the larger $\operatorname{Tr}(\mathbf{K}_p\mathbf{K}_q)$ means $\mathbf{K}_p$ and $\mathbf{K}_q$ are more correlated, and vice versa. Thus, if $\mathbf{K}_p$ and $\mathbf{K}_q$ are highly correlated, minimizing $\alpha_p\alpha_q \operatorname{Tr}(\mathbf{K}_p\mathbf{K}_q)$ avoids simultaneously assigning α_p and α_q with large weights, which enhances kernel diversity and reduces kernel redundancy.

Lastly, a discrete transformation strategy (*i.e.*, ISR) is used to adjust the relaxed continuous-valued label $\mathbf{F}$ into the discrete label $\mathbf{Y}$, which can directly obtain a discrete solution and avoid the information loss with overreliance on k-means.

Algorithm 1 Generalized power iteration for problem (6)

```
Input: $\mathbf{G} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times c}, \lambda \in \mathbb{R}$
Output: Optimal $\mathbf{F} \in \mathbb{R}^{n\times c}$
Randomly initialize $\mathbf{F} \in \mathbb{R}^{n\times c}$ satisfying $\mathbf{F}^T\mathbf{F} = \mathbf{I}_c$.
while not converge do
    Calculate $\mathbf{M} = 2\mathbf{G}\mathbf{F} + 2\lambda\mathbf{B}$
    Singular value decomposition: $\mathbf{M} = \mathbf{U}_{\mathbf{M}}\boldsymbol{\Sigma}_{\mathbf{M}}\mathbf{V}_{\mathbf{M}}^T$
    Update $\mathbf{F} = \mathbf{U}_{\mathbf{M}}\mathbf{V}_{\mathbf{M}}^T$
end while
return F
```

3. OPTIMIZATION

Problem (5) can be optimized in an alternative fashion, one set of variables at a time with the others fixed as follows.

Step 1: Update $\mathbf{F}$ with $\mathbf{G}, \mathbf{R}, \mathbf{Y}, \boldsymbol{\alpha}$ fixed, Problem (5) turns

$$\max_{\mathbf{F}^T\mathbf{F}=\mathbf{I}} \operatorname{Tr}(\mathbf{F}^T\mathbf{G}\mathbf{F}) + 2\lambda \operatorname{Tr}(\mathbf{F}^T\mathbf{B}), \tag{6}$$

where $\mathbf{B} \triangleq \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{\frac{1}{2}}\mathbf{R}^T$ and the Lagrangian function is

$$\mathcal{L}(\mathbf{F}, \boldsymbol{\Gamma}) = \operatorname{Tr}(\mathbf{F}^T\mathbf{G}\mathbf{F}) + 2\lambda \operatorname{Tr}(\mathbf{F}^T\mathbf{B}) - \operatorname{Tr}(\boldsymbol{\Gamma}(\mathbf{F}^T\mathbf{F} - \mathbf{I}_c)), \tag{7}$$

where $\boldsymbol{\Gamma} \in \mathbb{R}^{c\times c}$ whose elements γ_{ij} are Lagrangian multipliers. The KKT condition can be written as

$$\frac{\partial\mathcal{L}}{\partial\mathbf{F}} = 2\mathbf{G}\mathbf{F} + 2\lambda\mathbf{B} - 2\mathbf{F}\boldsymbol{\Gamma} = 0, \tag{8}$$

which is difficult to solve directly. Thus, we use generalized power iteration [22] to solve problem (6) in Algorithm 1.

Step 2: Update $\mathbf{R}$ with $\mathbf{F}, \mathbf{G}, \mathbf{Y}, \boldsymbol{\alpha}$ fixed. Problem (5) turns

$$\max_{\mathbf{R}^T\mathbf{R}=\mathbf{I}} \operatorname{Tr}(\mathbf{R}^T\mathbf{N}), \tag{9}$$

where $\mathbf{N} \triangleq \mathbf{F}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}} \in \mathbb{R}^{c\times c}$. Perform singular value decomposition (SVD) of $\mathbf{N}$: $\mathbf{N} = \mathbf{U}_{\mathbf{N}}\boldsymbol{\Sigma}_{\mathbf{N}}\mathbf{V}_{\mathbf{N}}^T$, where $\mathbf{U}_{\mathbf{N}}, \boldsymbol{\Sigma}_{\mathbf{N}}, \mathbf{V}_{\mathbf{N}} \in \mathbb{R}^{c\times c}$. Based on the theorem in [23], the optimal $\mathbf{R}$ can be easily obtained by $\mathbf{R} = \mathbf{U}_{\mathbf{N}}\mathbf{V}_{\mathbf{N}}^T$.

Step 3: Update $\mathbf{Y}$ with $\mathbf{F}, \mathbf{G}, \mathbf{R}, \boldsymbol{\alpha}$ fixed. Problem (5) turns

$$\begin{aligned} & \min_{\mathbf{Y}\in\operatorname{Ind}} \|\mathbf{F}\mathbf{R} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}\|_F^2 \\ \Leftrightarrow & \max_{\mathbf{Y}\in\operatorname{Ind}} \operatorname{Tr}(\mathbf{D}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}) \\ \Leftrightarrow & \max_{\mathbf{Y}\in\operatorname{Ind}} g(\mathbf{Y}) = \sum_{j=1}^{c} \frac{\boldsymbol{d}_j^T\boldsymbol{y}_j}{\sqrt{\boldsymbol{y}_j^T\boldsymbol{y}_j}}, \end{aligned} \tag{10}$$

where $\mathbf{D} = \mathbf{F}\mathbf{R}$, $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_c]$, $\mathbf{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_c]$ and $\boldsymbol{y}_j$,$\boldsymbol{d}_j$ are the j-th columns of $\mathbf{Y}$, $\mathbf{D}$. Now, we are going to update the discrete cluster label matrix $\mathbf{Y}$ row by row via coordinate descent technique [24], during which all variables are fixed except the i-th row being updated. Futhermore, there are c kinds of situations of the i-th row including $\{[1, 0, \cdots, 0], \cdots, [0, 0, \cdots, 1]\}$ with varying positions of 1, which can be denoted as $\{\mathbf{Y}^{(1)}, \cdots, \mathbf{Y}^{(c)}\}$, where we choose the one that maximizes $g(\mathbf{Y})$ as the updated optimal $\mathbf{Y}$.

Step 4: Update $\boldsymbol{\alpha}$ with $\mathbf{F}, \mathbf{G}, \mathbf{Y}, \mathbf{R}$ fixed. Problem (5) turns

$$\begin{aligned} &\min_{\boldsymbol{\alpha}^T\mathbf{1}=1,\alpha_p\geqslant 0,\forall p} \|\mathbf{G}-\mathbf{K}_{\boldsymbol{\alpha}}\|_F^2 \\ \Leftrightarrow &\min_{\boldsymbol{\alpha}^T\mathbf{1}=1,\alpha_p\geqslant 0,\forall p} \operatorname{Tr}(\mathbf{K}_{\boldsymbol{\alpha}}\mathbf{K}_{\boldsymbol{\alpha}}) - 2\operatorname{Tr}(\mathbf{G}^T\mathbf{K}_{\boldsymbol{\alpha}}) \qquad (11) \\ \Leftrightarrow &\min_{\boldsymbol{\alpha}^T\mathbf{1}=1,\alpha_p\geqslant 0,\forall p} \boldsymbol{\alpha}^T\mathbf{M}\boldsymbol{\alpha} + \boldsymbol{f}^T\boldsymbol{\alpha}, \end{aligned}$$

where we define $\boldsymbol{f}$ and $\mathbf{M}$ as follows

$$\boldsymbol{f} \triangleq \begin{bmatrix} -2\operatorname{Tr}(\mathbf{G}^T\mathbf{K}_1) \\ -2\operatorname{Tr}(\mathbf{G}^T\mathbf{K}_2) \\ \vdots \\ -2\operatorname{Tr}(\mathbf{G}^T\mathbf{K}_v) \end{bmatrix}, \qquad (12)$$

$$\mathbf{M} \triangleq \begin{bmatrix} \operatorname{Tr}(\mathbf{K}_1\mathbf{K}_1) & \cdots & \operatorname{Tr}(\mathbf{K}_1\mathbf{K}_v) \\ \vdots & \ddots & \vdots \\ \operatorname{Tr}(\mathbf{K}_v\mathbf{K}_1) & \cdots & \operatorname{Tr}(\mathbf{K}_v\mathbf{K}_v) \end{bmatrix}. \qquad (13)$$

Thus, problem (11) becomes a Quadratic Programming (QP) problem with v decision variables $\{\alpha_1, \ldots, \alpha_v\}$ and one equality constraint. It can be solve by any standard QP solver.

Step 5: Update $\mathbf{G}$ with $\mathbf{F}, \mathbf{R}, \mathbf{Y}, \boldsymbol{\alpha}$ fixed. Problem (5) turns

$$\begin{aligned} &\min_{\mathbf{G}\succeq 0} \operatorname{Tr}(\mathbf{G}(\mathbf{I}_n - \mathbf{F}\mathbf{F}^T)) + \gamma\|\mathbf{G}-\mathbf{K}_{\boldsymbol{\alpha}}\|_F^2 \\ \Leftrightarrow &\min_{\mathbf{G}\succeq 0} \operatorname{Tr}\left(\mathbf{G}^T\mathbf{G} - 2\mathbf{G}^T\mathbf{K}_{\boldsymbol{\alpha}} + \frac{1}{\gamma}\mathbf{G}(\mathbf{I}_n - \mathbf{F}\mathbf{F}^T)\right) \qquad (14) \\ \Leftrightarrow &\min_{\mathbf{G}\succeq 0} \|\mathbf{G}-\mathbf{B}\|_F^2 \end{aligned}$$

Since $\mathbf{B} \triangleq \mathbf{K}_{\boldsymbol{\alpha}} - \frac{1}{2\gamma}(\mathbf{I}_n - \mathbf{F}\mathbf{F}^T)$, the cluster label and the optimal kernel are optimized in the same objective, which can negotiate the clustering task and the kernel learning. Perform SVD of $\mathbf{B}$: $\mathbf{B} = \mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{B}}\mathbf{V}_{\mathbf{B}}^T$, where $\mathbf{U}_{\mathbf{B}}, \boldsymbol{\Sigma}_{\mathbf{B}}, \mathbf{V}_{\mathbf{B}} \in \mathbb{R}^{n\times n}$. Based on the theorem and conclusions in [10], we can easily get the conclusion that $\|\mathbf{G}-\mathbf{B}\|_F^2 \geqslant \operatorname{Tr}(\boldsymbol{\Sigma}_{\mathbf{G}}\boldsymbol{\Sigma}_{\mathbf{G}}) + \operatorname{Tr}(\mathbf{B}^T\mathbf{B}) - 2\operatorname{Tr}(\mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{G}}\mathbf{V}_{\mathbf{B}}^T\mathbf{B}) = \|\mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{G}}\mathbf{V}_{\mathbf{B}}^T - \mathbf{B}\|_F^2$. Thus, the optimal solution of problem (14) is $\mathbf{G}^* = \mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{B}}^+\mathbf{V}_{\mathbf{B}}^T$, where $\boldsymbol{\Sigma}_{\mathbf{B}}^+$ is a diagonal matrix formed by the nonnegative eigenvalues of $\mathbf{B}$ and setting the negative ones to zeros.

The overall optimization procedure is summarized in Algorithm 2, which will be terminated when the decreasing rate of the objective is lower than threshold 10^{-6}. In Algorithm 2, the time complexity of DMK-DOK is composed of 5 main parts. For convenience, we ignore the inner iterations of each step and focus on the most expensive operations in the subproblems of optimizing $\mathbf{F}$, $\mathbf{R}$, $\mathbf{Y}$, $\boldsymbol{\alpha}$ and $\mathbf{G}$, which are $\mathcal{O}(n^2)$, $\mathcal{O}(nc^2)$, $\mathcal{O}(n^2)$, $\mathcal{O}(n^2v)$ and $\mathcal{O}(n^3)$, respectively. Therefore, the overall time complexity of the proposed DMK-DOK is $\mathcal{O}(n^3)$, which is comparable with previous works.

Algorithm 2 Discrete Multi-kernel k-means with Diverse and Optimal Kernel Learning (DMK-DOK) in problem (5)

```
Input: Precomputed kernel matrices $\{\mathbf{K}_p \in \mathbb{R}^{n\times n}\}_{p=1}^v$
Output: Final cluster indicator $\mathbf{Y} \in \text{Ind}$
Let $\alpha_p = 1/v, \forall p \in \{1, \ldots, v\}$; random initialize $\mathbf{F}$, $\mathbf{R}$
  and $\mathbf{Y}$ satisfying $\mathbf{F}^T\mathbf{F} = \mathbf{I}_c$, $\mathbf{R}^T\mathbf{R} = \mathbf{I}_c$ and $\mathbf{Y} \in \text{Ind}$;
  initialize $\mathbf{G}$ by $\mathbf{G}^* = \mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{B}}^+\mathbf{V}_{\mathbf{B}}^T$
while not converge do
    Update $\mathbf{F}$ by Algorithm 1
    Update $\mathbf{R}$ by $\mathbf{R} = \mathbf{U}_{\mathbf{N}}\mathbf{V}_{\mathbf{N}}^T$
    Update $\mathbf{Y}$ by Eq. (10) via coordinate descent method
    Update $\boldsymbol{\alpha}$ by solving problem (11)
    Update $\mathbf{G}$ by $\mathbf{G}^* = \mathbf{U}_{\mathbf{B}}\boldsymbol{\Sigma}_{\mathbf{B}}^+\mathbf{V}_{\mathbf{B}}^T$
end while
return $\mathbf{Y} \in \text{Ind}$
```

4. EXPERIMENTS

4.1. Evaluation Protocol

Dataset Descriptions. We evaluate the clustering performance of the proposed model on five real-world benchmark datasets, covering different scenarios and applications. *TR11*, *TR41* and *TR45* are text datasets derived from TREC collections and preprocessed by the TF-IDF feature extractor[1]. Dataset *Wine* is derived from the UCI Machine Learning Repository [25]. Dataset *Palm* is the face dataset[2]. 12 different kernel functions are utilized to build the kernel matrices from the raw data following Du *et al.* [26]. These datasets are sufficient enough to evaluate the clustering performance, with different number of samples, features, and clusters.

Comparison Methods and Clustering Metrics. We compare the proposed method with classic and state-of-the-art multiple kernel k-means clustering methods, including *MKKM* [8], *LMKKM* [9], *ONKC* [10], *MKKM-RK* [11]. We employ grid search among all the comparison models to determine their hyperparameters following guidance in relevant papers and select the optimal results. The source code can be downloaded from the author's page. Three widely adopted metrics are employed to evaluate the clustering performance, including clustering accuracy (ACC), normalized mutual information (NMI) and adjusted Rand index (ARI).

Experimental Settings. We implement the experiments in MATLAB R2020b environment. For all comparison methods with hyperparameter(s), we follow the guidance of their authors to search for the values and report the best results. There are two hyperparameters λ and γ in our proposed method, and

[1] https://git.io/Jqfuh

[2] https://sites.google.com/site/feipingnie/file

we choose them both ranging with $[2^{-7}, 2^{-6}, \dots, 2^{7}]$, according to the selected datasets and specific clustering tasks.

Dataset	Metric	MKKM	LMK -KM	ONKC	MKKM -RK	DMK -DOK
Palm	ACC	0.8685	0.8645	0.8625	0.8805	**0.9110**
	NMI	0.9585	0.9609	0.9596	0.9650	**0.9673**
	ARI	0.8512	0.8484	0.8507	0.8720	**0.8834**
TR11	ACC	0.5459	0.5580	0.6643	0.6232	**0.7560**
	NMI	0.5665	0.5720	0.6145	0.5809	**0.6881**
	ARI	0.4503	0.4601	0.5709	0.5527	**0.6626**
TR41	ACC	0.5774	0.6128	0.6720	0.6298	**0.7118**
	NMI	0.6288	0.6214	0.6655	0.6391	**0.6765**
	ARI	0.5011	0.5075	0.6070	0.5397	**0.6194**
TR45	ACC	0.7478	0.7174	0.7870	0.7203	**0.8130**
	NMI	0.7181	0.6784	0.7086	0.6689	**0.7346**
	ARI	0.6648	0.6099	0.6877	0.6065	**0.6933**
Wine	ACC	0.9719	0.9663	0.9775	0.9663	**0.9831**
	NMI	0.8829	0.8635	0.9065	0.8748	**0.9261**
	ARI	0.9122	0.8962	0.9309	0.8992	**0.9471**

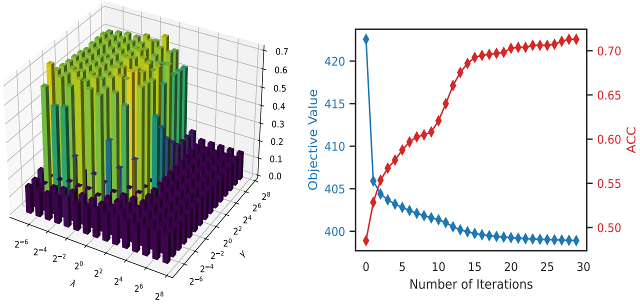
Table 1: Clustering performance. Best results are in bold.

4.2. Result Analyses

Clustering Performance. Table 1 explicitly displays the results of 3 widely-adopted clustering performance metrics, which we can draw the following conclusions: *a*) Among 5 datasets, our proposed model obtained the best clustering performance in all the metrics, which indicates its superiority. *b*) On text dataset *TR11*, our method surpasses the second best method (ONKC) by $+\mathbf{0.0917}$ ACC, $+\mathbf{0.0736}$ NMI, $+\mathbf{0.0917}$ ARI, which is a great improvement. Same improvement achieved in *TR41*, *TR45*. Thus, our method may be especially suitable for text data clustering. *c*) Compared with the methods using the two-stage optimization strategy (*i.e.*, MKKM, LMKKM, ONKC, MKKM-RK), our model has great advantages, which shows that jointly learning discrete and relaxed labels in the same objective can effectively avoid information loss and improve clustering performance.

Parameter Sensitivity. We present the clustering performances in terms of ACC among all models under different parameter settings. *a*) λ negotiates the relaxed label $\mathbf{F}$ and the discrete one $\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-\frac{1}{2}}$. As shown in Fig. 1, the results go well when λ is in the range of $[2^{-6}, 2^{-5}, \dots, 2^{-1}]$. When λ is too large, the effectiveness of the algorithm will drop drastically, because spectral rotation would dominate the model discarding the clustering objective which leads to a trivial solution. Generally, DMK-DOK is well-controlled when λ is set to appropriate values and contributes to negotiating kernel learning and clustering. *b*) γ helps to choose the optimal kernel $\mathbf{G}$ approaching the linear consensus kernel $\mathbf{K}_{\boldsymbol{\alpha}}$, so γ is able to measure how much effort we need to spend seeking for a better kernel $\mathbf{G}$ in the neighborhood of $\mathbf{K}_{\boldsymbol{\alpha}}$. In specific, if γ is too large, $\mathbf{K}_{\boldsymbol{\alpha}}$ and $\mathbf{G}$ are infinitely close, the optimal kernel tends to be the linear combination of base kernels. Besides, a small γ may limit communication between $\mathbf{G}$ and $\mathbf{K}_{\boldsymbol{\alpha}}$. Generally, when $\gamma > 2^{-5}$, our model achieved excellent results. All in all, our model always presents approving and promising performance, with small λ and large γ.

Convergence Analyses. To demonstrate the efficiency of our alternative optimization algorithm, we plot the convergence curves in Fig. 1, where the x-axes denotes the number of iterations and the y-axes denotes the value of ACC. It can be seen that the value of the objective function decreases monotonically with iterations. Thus, our optimization algorithm is of great effectiveness. Furthermore, the clustering performance increases with iterations, which means that optimizing our model can improve the clustering performance. Thus, our optimization algorithm is of high efficiency.



(a) Parameter Sensitivity (b) Convergence Analyses

Fig. 1: (a) On dataset *TR41* the clustering performances in terms of ACC are ploted. (b) How objective values and the values of ACC evolves on dataset *TR41*.

5. CONCLUSION

MKKM uses a group of kernels to improve the clustering performance, but it gains the optimal kernel by linearly combining base kernels, which limits the kernel representability. Second, most existing methods ignore the correlation among kernels, which may lead to kernel redundancy. Third, a two-stage strategy is used to solve the NP-hard cluster assignment problem, which may lead to information loss. In this paper, we proposed the DMK-DOK model to adaptively seek for a better kernel in the base kernel neighborhood. Moreover, our method helps to assign small coefficients to highly correlated kernels, which enhances the kernel diversity. What's more, DMK-DOK jointly learns the discrete and relaxed labels in the same objective via ISR, which can avoid information loss. Lastly, extensive experiments conducted on a number of datasets demonstrated the superiority of our model.

6. REFERENCES

[1] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 100–108, 1979.

[2] Feiping Nie, Dong Xu, Ivor W. Tsang, and Changshui Zhang, "Spectral embedded clustering," in *IJCAI*, 2009, pp. 1181–1186.

[3] Feiping Nie, Xiaoqian Wang, and Heng Huang, "Clustering and projected clustering with adaptive neighbors," in *KDD*, 2014, pp. 977–986.

[4] J. Chang, G. Meng, L. Wang, S. Xiang, and C. Pan, "Deep self-evolution clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 809–823, 2020.

[5] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1998.

[6] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *KDD*, 2004, pp. 551–556.

[7] Leena C Vankadara and Debarghya Ghoshdastidar, "On the optimality of kernels for high-dimensional clustering," in *AISTATS*, 2020, pp. 2185–2195.

[8] Shi Yu, Leon Tranchevent, Xinhai Liu, Wolfgang Glanzel, Johan AK Suykens, Bart De Moor, and Yves Moreau, "Optimized data fusion for kernel k-means clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1031–1039, 2011.

[9] Mehmet Gönen and Adam A Margolin, "Localized data fusion for kernel k-means clustering with application to cancer biology," in *NeurIPS*, 2014, pp. 1305–1313.

[10] Xinwang Liu, Sihang Zhou, Yueqing Wang, Miaomiao Li, Yong Dou, En Zhu, and Jianping Yin, "Optimal neighborhood kernel clustering with multiple kernels," in *AAAI*, 2017, pp. 2266–2272.

[11] Yaqiang Yao, Yang Li, Bingbing Jiang, and Huanhuan Chen, "Multiple kernel k-means clustering by selecting representative kernels," *IEEE Trans. Neural Netw. Learn. Syst.*, 2020.

[12] Xiaojun Chen, Feiping Nie, Joshua Zhexue Huang, and Min Yang, "Scalable normalized cut with improved spectral rotation.," in *IJCAI*, 2017, pp. 1518–1524.

[13] Mehmet Gönen and Ethem Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, 2011.

[14] Xinwang Liu, Lei Wang, Jian Zhang, and Jianping Yin, "Sample-adaptive multiple kernel learning," in *AAAI*, 2014, pp. 1975–1981.

[15] Xinwang Liu, Xinzhong Zhu, Miaomiao Li, Lei Wang, En Zhu, Tongliang Liu, Marius Kloft, Dinggang Shen, Jianping Yin, and Wen Gao, "Multiple kernel k-means with incomplete kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1191–1204, 2019.

[16] Bin Zhao, James T Kwok, and Changshui Zhang, "Multiple kernel clustering," in *SDM*, 2009, pp. 638–649.

[17] Xinwang Liu, Lei Wang, Xinzhong Zhu, Miaomiao Li, En Zhu, Tongliang Liu, Li Liu, Yong Dou, and Jianping Yin, "Absent multiple kernel learning algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1303–1316, 2019.

[18] Xinwang Liu, Miaomiao Li, Chang Tang, Jingyuan Xia, Jian Xiong, Li Liu, Marius Kloft, and En Zhu, "Efficient and effective regularized incomplete multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2634–2646, 2021.

[19] Rong Wang, Jitao Lu, Yihang Lu, Feiping Nie, and Xuelong Li, "Discrete multiple kernel k-means," in *IJCAI*, 2021, pp. 3111–3117.

[20] Jin Huang, Feiping Nie, and Heng Huang, "Spectral rotation versus k-means in spectral clustering," in *AAAI*, 2013.

[21] Yang Yang, Fumin Shen, Zi Huang, and Heng Tao Shen, "A unified framework for discrete spectral clustering," in *IJCAI*, 2016, pp. 2273–2279.

[22] Feiping Nie, Rui Zhang, and Xuelong Li, "A generalized power iteration method for solving quadratic problem on the stiefel manifold," *Science China Information Sciences*, vol. 60, no. 11, pp. 112101, 2017.

[23] Rong Wang, Feiping Nie, Zhen Wang, Fang He, and Xuelong Li, "Scalable graph-based clustering with nonnegative relaxation for large hyperspectral image," *IEEE Trans. Geosci. Remote. Sens.*, vol. 57, no. 10, pp. 7352–7364, 2019.

[24] Stephen J. Wright, "Coordinate descent algorithms," *Math. Program.*, vol. 151, no. 1, pp. 3–34, 2015.

[25] Dheeru Dua and Casey Graff, "UCI machine learning repository," 2017.

[26] Liang Du, Peng Zhou, Lei Shi, Hanmo Wang, Mingyu Fan, Wenjian Wang, and Yi-Dong Shen, "Robust multiple kernel k-means using $\ell_{2,1}$-norm," in *IJCAI*, 2015, pp. 3476–3482.