Sample Weighted Multiple Kernel K-means via Min-Max Optimization

Yi Zhang[*]
zhangy@nudt.edu.cn
National University of Defense Technology
Changsha, China

Weixuan Liang[*]
weixuanliang@nudt.edu.cn
National University of Defense Technology
Changsha, China

Xinwang Liu[†]
xinwangliu@nudt.edu.cn
National University of Defense Technology
Changsha, China

Sisi Dai
daisisi@nudt.edu.cn
National University of Defense Technology
Changsha, China

Siwei Wang
wangsiwei13@nudt.edu.cn
National University of Defense Technology
Changsha, China

Liyang Xu
xuliyang08@nudt.edu.cn
National University of Defense Technology
Changsha, China

En Zhu
enzhu@nudt.edu.cn
National University of Defense Technology
Changsha, China

ABSTRACT

A representative multiple kernel clustering (MKC) algorithm, termed simple multiple kernel k-means (SMKKM), is recently proposed to optimally mine useful information from a set of pre-specified kernels to improve clustering performance. Different from existing min-min learning framework, it puts a novel min-max optimization manner, which attracts considerable attention in related community. Despite achieving encouraged success, we observe that SMKKM only focuses on combination coefficients among kernels and ignores the relationship among the importance of different samples. As a result, it does not sufficiently consider different contributions of each sample to clustering, and thus cannot effectively obtain the "ideal" similarity structure, leading to unsatisfying performance. To address this issue, this paper proposes a novel sample weighted multiple kernel k-means via min-max optimization (SWMKKM), which sufficiently considers the sum of relationship between one sample and the others to represent the sample weights. Such a weighting criterion helps clustering algorithm pay more attention to samples with more positive effects on clustering and avoids unreliable overestimation for samples with poor quality. Based on SMKKM, we adopt a reduced gradient algorithm with proved convergence to solve the resultant optimization problem. Comprehensive experiments on multiple benchmark datasets demonstrate that our proposed SWMKKM dramatically improves the state-of-the-art MKC algorithms, verifying the effectiveness of our proposed sample weighting criterion.

CCS CONCEPTS

• **Theory of computation → Unsupervised learning and clustering**; • **Computing methodologies → Cluster analysis**.

KEYWORDS

multi-view clustering, multiple kernel clustering, min-max optimization,sample weight

ACM Reference Format:
Yi Zhang, Weixuan Liang, Xinwang Liu, Sisi Dai, Siwei Wang, Liyang Xu, and En Zhu. 2022. Sample Weighted Multiple Kernel K-means via Min-Max Optimization. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisbon, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3547917

1 INTRODUCTION

Multiple Kernel Clustering (MKC) is a popular method for addressing multi-view clustering problem, which usually looks for an appropriate strategy to mine useful information from a group kernels to improve clustering performance. These kernel matrices are often pre-specified and can be constructed from miscellaneous views data. By mapping the original data into a reproducing kernel Hilbert space (RKHS) through kernel tricks to extract non-linear information of features, MKC groups closer samples into the same cluster [4, 5, 8, 9, 11, 14, 15, 17, 19, 34, 39, 46].

MKC has been intensively studied and widely applied to various applications [6, 12, 16, 32, 36, 38, 43, 47]. For example, the works in [28, 49] develop multiple kernel subspace clustering (MKSC), which

[*] First authors with equal contribution
[†] Corresponding author

MM '22, October 10–14, 2022, Lisbon, Portugal
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547917

aims to obtain better representability to express high-dimension non-linear features by combining the advantage of MKC with the characteristic of subspace. The work in [44] assumes the low-rank consensus kernel and block-diagonal self-representation to maintain the "ideal" structure of subspace representation for improved robustness. By constructing graph from kernel matrices, the work in [27] links graph learning with MKC to enhance the clustering performance and tries adopting $l_{2,1}$ norm to make it more robust. The work in [31] proposes to hold local graph structure to extract the information among specific samples and simultaneously preserves global graph structure for the whole clustering, both of which are very important to obtain potential available features. Some methods focus on the localized kernel alignment criterion, which pay more attention to keeping closer sample pairs together and avoiding the unreliable estimation of farther ones [6, 23]. By adopting the paradigm of high-order neighborhood to further mine the localized relationship among samples, the work in [20] holds further localized structure for better clustering. Clustering algorithms with late fusion manner maximally align the consensus partition matrix with a group base partition matrices, computing in partition layer instead of kernel data layer, thus it can substantially reduce computation complexity. Due to the success of late fusion manner, it has been further applied and extended, such as [37, 45].

As a representative of MKC, the recently proposed simple multiple kernel k-means (SMKKM) [25] gives a novel min-max optimization manner, which minimizes the alignment for kernel weights and maximizes the alignment for partition matrices instead of the simultaneous minimization for both. After that, a reduced gradient algorithm is introduced to solve the resultant intractable optimization problem. The novelties of objective manner and improvement of clustering performance attract considerable concerns and researches in community.

Despite that recently proposed SMKKM has achieved encouraging success as mentioned above, we observe an obvious and important defect in the existing algorithms. That is, they only focus on the combination coefficients among kernels, and do not sufficiently consider the relationship among the importance of different samples. They indiscriminately consider the contribution of each sample for clustering. However, different samples usually have dissimilar importance in clustering due to their variable effect and quality in practice. More importantly, the low-quality or redundant samples would hurt the "ideal" similarity structure, leading to poor clustering performance. To address this issue, we propose to learn the kernel alignment and partition in a sample weighted manner. Specifically, we consider each sample has different contributions for clustering tasks and qualitatively represent sample weights with the sum of relationship among one sample and the others. Such a weighted criterion guides clustering algorithm to pay more attention to samples with more positive effects on clustering and avoids unreliable overestimation of samples with poor quality. By this way, our proposed SWMKKM could sufficiently consider different contribution of samples according to the kernel information, and thereby enhance the clustering performance. Afterwards, we demonstrate the theoretical connection and difference between SMKKM and our proposed SWMKKM, and point out that the former is a special case of the latter with equal weights for all samples. Based on this observation, we develop the objective function of our proposed SWMKKM and carefully design a optimization strategy with proved convergence to solve the resultant optimization problem. In addition, comprehensive experiments on multiple benchmark datasets are carried out to evaluate the effectiveness of our proposed algorithm. The results have demonstrated that our proposed SWMKKM significantly outperforms the state-of-the-art MKC competitors, verifying the effectiveness of our proposed algorithm. The main contributions of this paper are summarized as follows,

- This paper, for the first time, points out that the recently proposed algorithms can not effectively deal with different contribution of samples, which may become a bottleneck that performance cannot break through. Correspondingly, we, for the first time, develop a sample weighted criterion to address this issue, which is likely to form a new learning framework to further improve and explain the performance of algorithms.
- This paper rigorously proves the convexity and differentiability of our proposed algorithm and give sufficient theoretical support basis for the optimization. Based on this, we introduces the reduced gradient descent method with guaranteed convergence to optimize the resultant problem.
- Comprehensive experiments on multiple benchmark datasets have demonstrated that SWMKKM dramatically outperforms the state-of-the-art MKC algorithms, verifying the effectiveness of the proposed sample weighted criterion.

2 RELATED WORK

In this section, we provide a brief review of MKKM and SMKKM, and then introduce the motivation of our work.

2.1 Multiple Kernel K-means

Given a data matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, k-means clustering aims to assign $\mathbf{X}$ into k clusters by minimizing the sum of squares error, where n and d is the number of samples and feature dimensions. Its objective function can be formulated as follows,

$$\min_{\mathbf{S},\,\mathbf{c}} \frac{1}{n} \sum\nolimits_{i=1}^{n} \sum\nolimits_{q=1}^{k} S_{iq} \|\mathbf{x}_i - \mathbf{c}_q\|^2 \ s.t.\ \mathbf{S1} = \mathbf{1}, \tag{1}$$

where $\mathbf{S} \in \{0,1\}^{n\times k}$ is a clustering assignment matrix and $S_{iq} = 1$ if $\mathbf{x}_i$ belongs to the q-th cluster. To handle with non-linear features, the samples are usually mapped into a reproducing kernel Hilbert space (RKHS) [33] by kernel methods. The kernel matrices can be written as $K_{i,j} = \phi_i^\top \phi_j$ with a mapping function $\varphi(\cdot)$. Then, by defining $\mathbf{H} = \mathbf{SL}^{\frac{1}{2}}$ where $\mathbf{L} = diag([s_1^{-1}, \ldots, s_k^{-1}])$ with $s_j = \sum_{i=1}^{n} \mathbf{S}_{ij}$, we can equivalently rewrite its formulation as follows,

$$\min_{\mathbf{H}} \ \mathrm{Tr}\left(\mathbf{K}\left(\mathbf{I} - \mathbf{H}\mathbf{H}^\top\right)\right) \ s.t.\ \ \mathbf{H}^\top\mathbf{H} = \mathbf{I}. \tag{2}$$

Following the multiple kernel learning framework, the optimal consensus kernel $\mathbf{K}_{\boldsymbol{\gamma}}$ is usually assumed as a linear combination of a group base kernel matrices. Therefore, the objective can be extended as follows,

$$\min_{\mathbf{H},\,\boldsymbol{\gamma}} \ \mathrm{Tr}(\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{I} - \mathbf{H}\mathbf{H}^\top)) \ \ s.t.\ \ \mathbf{H}^\top\mathbf{H} = \mathbf{I}, \boldsymbol{\gamma} \in \Delta, \tag{3}$$

where $\Delta = \{\boldsymbol{\gamma} \in \mathbb{R}^m \mid \sum_{p=1}^{m} \gamma_p = 1,\ \gamma_p \geq 0,\ \forall p\}$, $\mathbf{K}_{\boldsymbol{\gamma}} = \sum_{p=1}^{m} \gamma_p^2 \mathbf{K}_p$ and m denotes the number of data kernels. $\mathbf{H}$ and $\boldsymbol{\gamma}$ can be jointly solved by an alternate optimization method in literature [11]. After that, a standard k-means algorithm is performed on the learned partition matrix $\mathbf{H}$ to obtain the final cluster assignments.

2.2 Simple Multiple Kernel K-means

Recently, the work in [26] points out that often-used $\min_{\boldsymbol{\gamma}}$ $\min_{\mathbf{H}}$ paradigm probably can not achieve satisfying clustering performance in real-world applications, sometimes even worse than the baseline kernel k-means. Thus, more new clustering models are encouraged to be designed and studied. Different from the $\min_{\boldsymbol{\gamma}}$ $\min_{\mathbf{H}}$ paradigm, simple multiple kernel k-means (SMKKM) [26] proposes a novel $\min_{\boldsymbol{\gamma}}$ $\max_{\mathbf{H}}$ optimization manner as follows,

$$\min_{\boldsymbol{\gamma}\in\Delta} \max_{\mathbf{H}\in\mathbb{R}^{n\times k}} \operatorname{Tr}(\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{H}\mathbf{H}^{\top}) \ \ s.t. \ \ \mathbf{H}^{\top}\mathbf{H} = \mathbf{I}_k. \tag{4}$$

This new minimization-maximization manner makes Eq. (4) cannot be easily solved by often-used alternate optimization method. To solve the resultant optimization, SMKKM transforms the $\min_{\boldsymbol{\gamma}}$ $\max_{\mathbf{H}}$ into a minimization w.r.t $\boldsymbol{\gamma}$, and designs a reduced gradient descent strategy after proving the differentiability of the resultant minimization formulation.

Although encouraging clustering performance improvement has been brought by the recently proposed SMKKM, we observe that the lack of consideration of the the relationship among the importance of different samples in existing algorithms leads to the unsatisfied similarity structure. To handle with this issue, we develop a sample weighted multiple kernel k-means clustering algorithm and design an optimization method to solve it.

3 SAMPLE WEIGHTED MKKM

In this section, we first develop the formulation of our proposed SWMKKM and provide sufficient theoretical analysis of it. Then we design a reduced gradient descent algorithm to optimize the resultant problem. Finally, the convergence and computational complexity of our proposed algorithm are discussed.

3.1 Proposed Formulation

As seen from Eq. (4), with the novel min-max optimization paradigm, SMKKM has explored the ideas of clustering optimization and achieved encouraging clustering performance. However, we observe that it only focuses on combination coefficients among kernels and ignores the relationship among the importance of different samples which shall have various contribution. As a result, the lack of above considerations leads to the limited performance improvement. To address this issue, we propose a sample weighted multiple kernel k-means clustering algorithm, which learns the kernel alignment and partition in a sample weighted manner.

To be specific, we regard kernel matrices as similarity, and the differences in importance can be extracted from the overall similarity relationship. We also inherit the assumption of MKKM, that is, the optimal weights can be expressed by a linear combination of a group of weights from different views.For example, if a sample is similar to most other samples, this strategy considers it very important and gives it a larger weight. And this sample is likely to be a key node. Large weight values can make the learning of cluster centroids more reliable and stable.Otherwise, the sample may be an "unsatisfactory" sample or an outlier. A small weight can effectively reduce the excessive impact of such samples on cluster centroids.

To do so, we firstly initialize the weight matrix of single view as follow,

$$\mathbf{W}_p = \operatorname{diag}(\mathbf{K}_p\mathbf{1}), \tag{5}$$

where $\mathbf{K}_p$ denotes the kernel matrix of the p-th view. Next, we combine the sample weights from all views with a power parameter and obtain

$$\mathbf{W}_{\boldsymbol{\gamma}} = \left(\sum\nolimits_{p=1}^{m} \boldsymbol{\gamma}_p^2 \mathbf{W}_p\right)^{\frac{\lambda}{2}}, \tag{6}$$

where $\boldsymbol{\gamma} \in \Delta$ and $\Delta = \{\boldsymbol{\gamma} \in \mathbb{R}^m \mid \sum_{p=1}^{m} \gamma_p = 1, \ \gamma_p \geq 0, \ \forall p\}$. Finally, by merging Eq. (6) into Eq. (4), we obtain the formulation of our proposed SWMKKM as follow,

$$\begin{gathered}\min_{\boldsymbol{\gamma}\in\Delta} \max_{\mathbf{H}\in\mathbb{R}^{n\times k}} \operatorname{Tr}(\mathbf{H}^{\top}\mathbf{W}_{\boldsymbol{\gamma}}^{\top}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{W}_{\boldsymbol{\gamma}}\mathbf{H}) \\ s.t. \ \ \mathbf{H}^{\top}\mathbf{H} = \mathbf{I}_k, \boldsymbol{\gamma} \in \Delta,\end{gathered} \tag{7}$$

where $\mathbf{K}_{\boldsymbol{\gamma}} = \sum_{p=1}^{m} \boldsymbol{\gamma}_p^2 \mathbf{K}_p$. As seen from Eq. (6), $\mathbf{W}_{\boldsymbol{\gamma}}$ is also the diagonal matrix where each term indicates the weight of the corresponding sample. Furthermore, our proposed formulation is close to the objective function of SMKKM in form, therefore we can solve the resultant problem by the similar method. Based on this, the formulation in Eq. (7) can be equivalently expressed as,

$$\min_{\boldsymbol{\gamma}\in\Delta} \mathcal{F}(\boldsymbol{\gamma}), \tag{8}$$

where

$$\mathcal{F}(\boldsymbol{\gamma}) = \max_{\mathbf{H}} \operatorname{Tr}(\mathbf{H}^{\top}\mathbf{W}_{\boldsymbol{\gamma}}^{\top}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{W}_{\boldsymbol{\gamma}}\mathbf{H}), \ s.t. \ \mathbf{H}^{\top}\mathbf{H} = \mathbf{I}_k. \tag{9}$$

Through this transform technique, the $\min_{\boldsymbol{\gamma}}$ & $\max_{\mathbf{H}}$ optimization problem is converted to a minimization one. As meanwhile, we let $\hat{\mathbf{K}}_{\boldsymbol{\gamma}} = \mathbf{W}_{\boldsymbol{\gamma}}^{\top}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{W}_{\boldsymbol{\gamma}}$ and the objective function $\mathcal{F}(\boldsymbol{\gamma})$ can be treated as a kernel k-means optimal solution problem.

3.2 Theoretical Analysis

In this subsection, we make a theoretical analysis of our proposed algorithm, including the positive semidefiniteness, convexity and differentiability.

Theorem 1. *Each $\mathbf{K}_{\boldsymbol{\gamma}}$ is positive semidefinite, then each $\hat{\mathbf{K}}_{\boldsymbol{\gamma}}(1 \leq \gamma \leq m)$ is positive semidefinite.*

Proof. As $\forall \boldsymbol{\gamma}$, $\mathbf{K}_{\boldsymbol{\gamma}}$ is positive semidefinite, we can obtain $\forall \mathbf{x} \neq 0, \mathbf{x}^{\top}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{x} \geq 0$. Then $\mathbf{x}^{\top}\hat{\mathbf{K}}_{\boldsymbol{\gamma}}\mathbf{x} = (\mathbf{W}_{\boldsymbol{\gamma}}\mathbf{x})^{\top}\mathbf{K}_{\boldsymbol{\gamma}}(\mathbf{W}_{\boldsymbol{\gamma}}\mathbf{x}) \geq 0$, therefore $\hat{\mathbf{K}}_{\boldsymbol{\gamma}} = \mathbf{W}_{\boldsymbol{\gamma}}^{\top}\mathbf{K}_{\boldsymbol{\gamma}}\mathbf{W}_{\boldsymbol{\gamma}}$ is positive semidefinite. □

Theorem 1 indicates that each $\hat{\mathbf{K}}_{\boldsymbol{\gamma}}$ still remains positive semidefinite (PSD) after aforementioned sample weighting processing and so that the formulation is lower bounded by zero.

Theorem 2. *When $\lambda \geq 1$, $\mathcal{F}(\boldsymbol{\gamma})$ in Eq. (8) is convex.*

The proof of Theorem 2 is given in appendix due to space limit. By Theorem 2, the convexity of $\mathcal{F}(\boldsymbol{\gamma})$ is proved, and this is a necessary condition of the following theorem, which is about the differentiability of the proposed objective.

THEOREM 3. *$\mathcal{F}(\boldsymbol{\gamma})$ in Eq. (8) is differentiable. The parital derivative of $\mathcal{F}(\boldsymbol{\gamma})$ at γ_p is*

$$\begin{aligned}\frac{\partial\mathcal{F}(\boldsymbol{\gamma})}{\partial\gamma_p} =& 2\gamma_p \mathrm{Tr}(\hat{\mathbf{H}}^\top \mathbf{W}_{\boldsymbol{\gamma}} \mathbf{K}_p \mathbf{W}_{\boldsymbol{\gamma}} \hat{\mathbf{H}}) \\ &+ 2\lambda\gamma_p \mathrm{Tr}(\hat{\mathbf{H}}^\top \mathbf{W}_p (\sum_{p=1}^{m} \gamma_p^2 \mathbf{W}_p))^{\frac{\lambda}{2}-1} \mathbf{K}_{\boldsymbol{\gamma}} \mathbf{W}_{\boldsymbol{\gamma}} \hat{\mathbf{H}}),\end{aligned} \tag{10}$$

where $\hat{\mathbf{H}} = \arg\max_{\mathbf{H}} \mathcal{F}(\boldsymbol{\gamma})$, s.t. $\mathbf{H}^\top\mathbf{H} = \mathbf{I}_k$.

The form of Eq. (8) seems similar to the formula of Danskin's Theorem [3]. It can be known that only if $\hat{\mathbf{H}}$ is the unique solution, $\mathcal{F}(\boldsymbol{\gamma})$ can be shown differentiable by Danskin's Theorem. Unfortunately, there is more than one maximizer $\hat{\mathbf{H}}$ with some fixed $\boldsymbol{\gamma}$. To address this issue, we elegantly transform it into an equivalent optimization problem by the following lemma.

LEMMA 4. *With some fixed PSD matrix $\mathbf{K}$, the following statement holds:*

$$\max_{\mathbf{H}} \mathrm{Tr}(\mathbf{K}\mathbf{H}\mathbf{H}^\top) = \max_{\mathbf{P}\in\mathcal{P}_k} \mathrm{Tr}(\mathbf{K}\mathbf{P})$$

where $\mathbf{H} \in \mathbb{R}^{n\times k}$, $\mathbf{H}^\top\mathbf{H} = \mathbf{I}_k$ and $\mathcal{P}_k \subset \mathbb{R}^{n\times n}$ is the space of rank-k orthogonal projection.

According to Theorem 1 and Lemma 4, the objective can be written as $\mathcal{F}(\gamma) = \max_{\mathbf{P}\in\mathcal{P}_k} \mathrm{Tr}(\mathbf{W}_{\boldsymbol{\gamma}}^\top \mathbf{K}_{\boldsymbol{\gamma}} \mathbf{W}_{\boldsymbol{\gamma}} \mathbf{P})$. The proof of Lemma 4 is given in appendix due the limited space. Further, to verify that the objective meets the conditions of Danskin's Theorem, we need to prove the following two lemmas.

LEMMA 5. *With fixed $\boldsymbol{\gamma}$, the solution of $\max_{\mathbf{P}\in\mathcal{P}_k} F(\boldsymbol{\gamma})$ is unique, where $F(\boldsymbol{\gamma}) = \mathrm{Tr}(\mathbf{W}_{\boldsymbol{\gamma}}^\top \mathbf{K}_{\boldsymbol{\gamma}} \mathbf{W}_{\boldsymbol{\gamma}} \mathbf{P})$.*

LEMMA 6. *$\mathcal{P}$ is compact.*

The proofs of Lemma 5 and Lemma 6 are also given in appendix. Now we complete the proof of Theorem 3.

PROOF. Denote that $f(\boldsymbol{\gamma}, \mathbf{P}) = \mathrm{Tr}(\mathbf{W}_{\boldsymbol{\gamma}}^\top \mathbf{K}_{\boldsymbol{\gamma}} \mathbf{W}_{\boldsymbol{\gamma}} \mathbf{P})$. It is easy to check that $\forall\boldsymbol{\gamma} \in \Delta$ and $\mathbf{P} \in \mathcal{P}$, $f(\boldsymbol{\gamma}, \mathbf{P})$ is continuous. According to Lemma 5 and Lemma 6, we know that $\mathcal{P}$ is compact and the maximizer of $f(\boldsymbol{\gamma}, \mathbf{P})$ is unique with fixed $\boldsymbol{\gamma}$. The conditions of Danskin's Theorem [3] hold, thus $\max_{\mathbf{P}} f(\boldsymbol{\gamma}, \mathbf{P})$ is differentiable. With Lemma 4, we can obtain that $F(\boldsymbol{\gamma}) = \max_{\mathbf{P}} f(\boldsymbol{\gamma}, \mathbf{P})$ is also differentiable. The proof is complete. □

Remark. *By Theorem 3, we can optimize $\mathcal{F}(\boldsymbol{\gamma})$ by gradient descent algorithm due to its differentiability. As stated in Theorem 2, the objective function is convex and it will convergence to the global minimum by the following optimization strategy.*

3.3 Min-Max Optimization

First, since $\mathcal{F}(\boldsymbol{\gamma})$ can be treated as a kernel k-means optimal value function, for any fixed $\boldsymbol{\gamma}$, we can easily obtain the optimal solution of maximization in Eq. (9) by eigenvalue decomposition as

$$\mathbf{H}^* = \left\{\arg\max_{\mathbf{H}} \ \mathrm{Tr}\left(\mathbf{H}^\top \hat{\mathbf{K}}_{\boldsymbol{\gamma}} \mathbf{H}\right) \ s.t.\ \mathbf{H}^\top\mathbf{H} = \mathbf{I}_k\right\}. \tag{11}$$

Algorithm 1 Sample Weighted Multiple Kernel K-means via min-max optimization

```
Input: $\{\mathbf{K}_p\}_{p=1}^m$, $k$, $\lambda$.
Initialize $\boldsymbol{\gamma}^{(1)} = \mathbf{1}/m$, $\{\mathbf{W}_p\}_{p=1}^m = diag(\mathbf{K}_p\mathbf{1})$ and t = 1.
repeat
    Compute $\mathbf{W}_{\boldsymbol{\gamma}^{(t)}}$ with Eq. (6).
    Compute $\mathbf{H}^*$ by solving a kernel k-means with $\hat{\mathbf{K}}_{\boldsymbol{\gamma}^{(t)}} = \mathbf{W}_{\boldsymbol{\gamma}^{(t)}}^\top \mathbf{K}_{\boldsymbol{\gamma}^{(t)}} \mathbf{W}_{\boldsymbol{\gamma}^{(t)}}$.
    Compute $\frac{\partial\mathcal{F}(\boldsymbol{\gamma}^{(t)})}{\partial\gamma_p}$ $(p = 1, \cdots, m)$.
    Compute the gradient $\nabla\mathcal{F}(\boldsymbol{\gamma}^{(t)})$ by Eq. (12, 13).
    Compute the descent direction $\boldsymbol{d}^{(t)}$ by Eq. (14).
    Update $\boldsymbol{\gamma}$ via the scheme $\boldsymbol{\gamma}^{(t+1)} \longleftarrow \boldsymbol{\gamma}^{(t)} + \alpha\boldsymbol{d}^{(t)}$.
    $t \longleftarrow t + 1$.
until $\max|\boldsymbol{\gamma}^{(t+1)} - \boldsymbol{\gamma}^{(t)}| \leq 1e - 4$
```

Next, we adopt a reduced gradient descent algorithm to optimize $\mathcal{F}(\boldsymbol{\gamma})$ by following [25, 30]. We can calculate the gradient of $\mathcal{F}(\boldsymbol{\gamma})$ by Theorem 3 and Eq. (11), and then update $\boldsymbol{\gamma}$ through the descent gradient with keeping the equality and non-negativity constraints. We let γ_u be the largest component of $\boldsymbol{\gamma}$ and $\nabla\mathcal{F}(\boldsymbol{\gamma})$ denotes the reduced gradient of $\mathcal{F}(\boldsymbol{\gamma})$. The p-th $(1 \leq p \leq m)$ element of $\nabla\mathcal{F}(\boldsymbol{\gamma})$ is

$$\left[\nabla\mathcal{F}(\boldsymbol{\gamma})\right]_p = \frac{\partial\mathcal{F}(\boldsymbol{\gamma})}{\partial\gamma_p} - \frac{\partial\mathcal{F}(\boldsymbol{\gamma})}{\partial\gamma_u} \ \ \forall\, p \neq u, \tag{12}$$

and

$$\left[\nabla\mathcal{F}(\boldsymbol{\gamma})\right]_u = \sum_{p=1, p\neq u}^{m} \left(\frac{\partial\mathcal{F}(\boldsymbol{\gamma})}{\partial\gamma_u} - \frac{\partial\mathcal{F}(\boldsymbol{\gamma})}{\partial\gamma_p}\right). \tag{13}$$

Thus the equality constraint $\sum_{p=1}^m \gamma_p = 1$ guaranteed to be satisfied by updating $\boldsymbol{\gamma}$ with the gradient in Eq. (12) and Eq. (13). After that, by taking the positive constraints on $\boldsymbol{\gamma}$ into consideration in the descent direction, we set it as

$$d_p = \begin{cases} 0, & \text{if } \gamma_p = 0 \text{ and } \left[\nabla\mathcal{F}(\boldsymbol{\gamma})\right]_p > 0, \\ -\left[\nabla\mathcal{F}(\boldsymbol{\gamma})\right]_p, & \text{if } \gamma_p > 0 \text{ and } p \neq u, \\ -\left[\nabla\mathcal{F}(\boldsymbol{\gamma})\right]_u, & \text{if } p = u. \end{cases} \tag{14}$$

Finally, $\boldsymbol{\gamma}$ can be updated via the scheme $\boldsymbol{\gamma} \longleftarrow \boldsymbol{\gamma} + \alpha\boldsymbol{d}$ where $\boldsymbol{d} = [d_1, \cdots, d_m]^\top$ and α denotes the descent direction and a learning step size. The optimal α can be selected by Armijo's rule. The complete algorithm procedure of our proposed SWMKKM is outlined in Algorithm 1.

3.4 Discussion

Our proposed SWMKKM adopts the reduced gradient descent algorithm and the objective value is monotonically decreased according to the literature [30]. Furthermore, the convexity of the proposed formulation is proved in Theorem 2, therefore, our proposed algorithm is guaranteed to achieve the convergence of a global optimum.

From Eq. (1), at each iteration, SWMKKM needs to calculate a weight matrix with computational complexity $O(n\log_2 n)$, solve a kernel k-means problem with computational complexity $O(n^3)$, compute the gradient and the descent direction with computational complexity $O(mn^2k)$ and search optimal step size with computational complexity $O(ml)$. Therefore, its computation complexity at each iteration is $O(n^3 + mn^2k + m\ell)$ where ℓ is the is the max number of flops to find the optimal α.

Table 1: Datasets used in our experiments.

Dataset	#Samples	#Kernels	#Clusters
Wpbc	194	10	2
Sonar	207	10	2
Wisconsin	265	2	5
Politicsuk	419	9	5
Cal-5	441	5	7
MFeat	2000	3	10
4Area	4236	2	4
Reuters	18758	5	6

4 EXPERIMENT AND ANALYSIS

In this section, we conduct comprehensive experiments on eight benchmark datasets to evaluate effectiveness of our proposed SWMKKM. The clustering performance, evolution of the objective value and the learned **H**, weight coefficients, parameter sensitivity and computational efficiency are studied carefully.

4.1 Experiment Setting

Eight often-used and representative multi-view datasets are adopted in the experiments to evaluate the clustering performance of our proposed SWMKKM, including *Wpbc*[1], *Sonar*[1], *Wisconsin*[1], *Football*[2], *Politicsuk*[2], *Cal-5*[3], *MFeature*[4], *4Area*[29], *Reuters*[5]. Among them, Wisconsin is a Webkb dataset used in [2, 42]; MFeat is a handwritten digital dataset used in [35, 50]; Politicsuk is a Politics dataset used in [13, 41]; Wpbc is Wisconsin Prognostic Breast Cancer used in [40, 48]. The detail information of datasets is outlined in Table 1. As seen, the numbers of samples, kernels and clusters vary over a considerable range, which ensures the reasonable comparison and evaluation among different clustering algorithms. For all datasets, it is assumed that the real number of clusters k is given and taken as the input of algorithms.

We adopt four often-used criteria to evaluate the clustering performance of all comparison algorithms, i.e. clustering accuracy (ACC), normalized mutual information (NMI), purity (PUR) and rand index (RI). For all algorithms, each experiment is repeated 50 times with performing various initialization to reduce the adverse impact of randomness caused by k-means, and their means and the corresponding standard deviations are reported. Our all experiments are conducted on a PC with Intel Core i9-10900X CPU and 64G RAM in MATLAB R2020b environment.

In our experiments, we compare the proposed algorithm with dozen state-of-the-art multi-view clustering baseline algorithms, including: **Average kernel k-means (Avg-KKM)**. Avg-KKM constructs the consensus kernel by averagely combining all kernels. **Single Best kernel k-means (SB-KKM)**. SB-KKM directly takes each base kernel as as the input of kernel k-means algorithm, and choose the best kernel. **Multiple kernel k-means (MKKM)** [10]. MKKM aims to learn the kernel coefficients while performing KKM. **Localized multiple kernel k-means (LMKKM)** [7]. LMKKM combines the kernels in a localized way in order to capture more information. **Optimal neighborhood kernel clustering (ONKC)** [24]. ONKC chooses the optimal kernel from the neighbor of linear combination of base kernels. **Multiple kernel k-means with matrix-induced regularization (MKKM-MR)** [21]. MKKM-MR introduces a matrix-induced regularization term in order to enhance the diversity and reduce the redundancy of the chosen kernels. **Mulitple kernel clustering with local alignment maximization (LKAM)** [18]. LKAM aligns the ideal similarity with the similarity of samples to k-nearest neighbors rather than all samples. **Multi-view clustering via late fusion alignment maximization (LFMVC)** [37]. LFMVC calculates all base partitions and fuses them to learn a consensus partition. **Robust multiple kernel k-means with min-max optimization (RMKKM)** [1]. Inspired by the adversarial learning, RMKKM gives a min-max paradigm for more robustness to perturbation. **Simple multiple kernel k-means (SMKKM)** [25]. SMKKM introduces a novel clustering paradigm by minimizing alignment with reject to the kernel weights and maximizing alignment with reject to. the assignment partition. **Multiple kernel clustering with neighbor-kernel subspace segmentation (NKSS)** [49]. NKSS linearly combines the neighbor kernels through an exact-rank-constrained subspace segmentation to extract a consensus affinity matrix. **Multiple Kernel Clustering with Global and Local Graph Structure Preserving (SPMKC)** [31]. SPMKC aims to preserve the global and local structure by introducing a self-expressiveness term and a local structure learning term. **One pass late fusion multi-view clustering (OPLFMVC)** [22]. OPLFMVC unifies consensus partition learning and cluster labels generation into a single optimization to directly obtain cluster labels. **Localized simple multiple kernel k-means (LSMKKM)** [23]. LSMKKM inherits the advantages of SMKKM and adopts a local alignment to fuse the information of base kernels.

The source codes of these algorithms are publicly available and we directly run them without revision in the experiment. Among them, ONKC [24], MKKM-MR [21], LKAM [18], LFMVC [37], NKSS [49] and LSMKKM [23] have hyper-parameters to be tuned. Following the corresponding literature, we take grid search method to tune the hyper-parameters and produce the best possible results on each dataset. We also list the optimal hyper-parameters for each algorithm (if have) in the appendix for reproducibiity.

4.2 Experimental Results

4.2.1 Overall Clustering Performance. We list the ACC, NMI and RI comparison of the aforementioned algorithms in Table 2 where boldface indicates the best one and underline means the second best. From this table, We can have the following observation:

- Avg-KKM, as a baseline algorithm, provides the reference clustering performance, and as seen, the clustering performance of our proposed SWMKKM completely passes the baseline. For example, our proposed SWMKKM exceeds Avg-KKM by 11.6%, 7.0%, 23.2%, 17.3%, 8.1%, 17.2%, 5.9% and 13.8% on all benchmark datasets in term of ACC. Meanwhile, SWMKKM exceeds another baseline algorithm SB-KKM by 10.5%, 5.6%, 15.5%, 13.8%, 8.1%, 21.8%, 14.3% and 12.1%. These results verify the possibility and the efficiency of our proposed SWMKKM.

[1] http://archive.ics.uci.edu/ml/datasets/

[2] http://mlg.ucd.ie/aggregation/

[3] http://www.vision.caltech.edu/Image_Datasets/Caltech101/

[4] http://archive.ics.uci.edu/ml/datasets/Multiple+Features

[5] https://kdd.ics.uci.edu/databases/reuters21578/

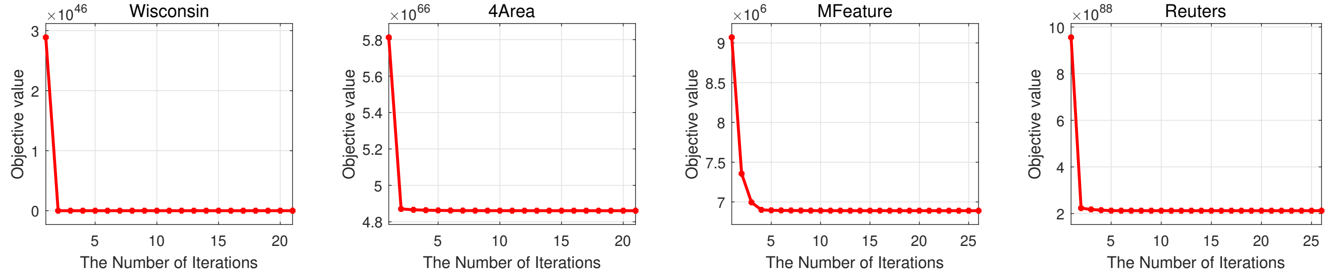

Figure 1: The objective values of SWMKKM's formulation varying with the number of iterations.

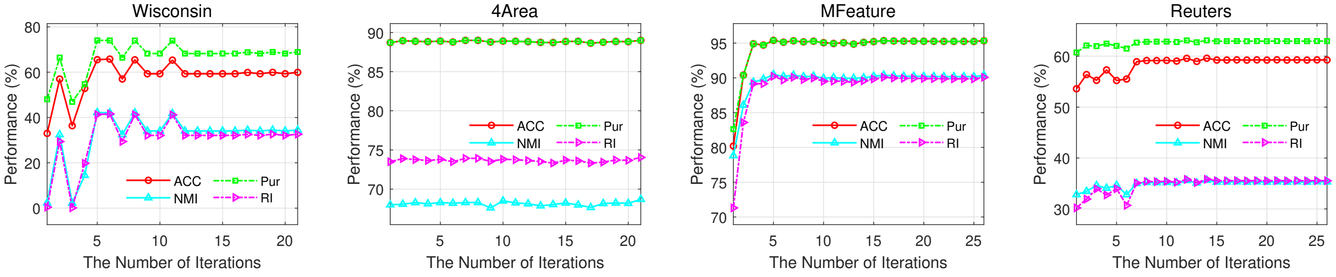

Figure 2: The evolution of SWMKKM's learning process varying with the number of iterations.

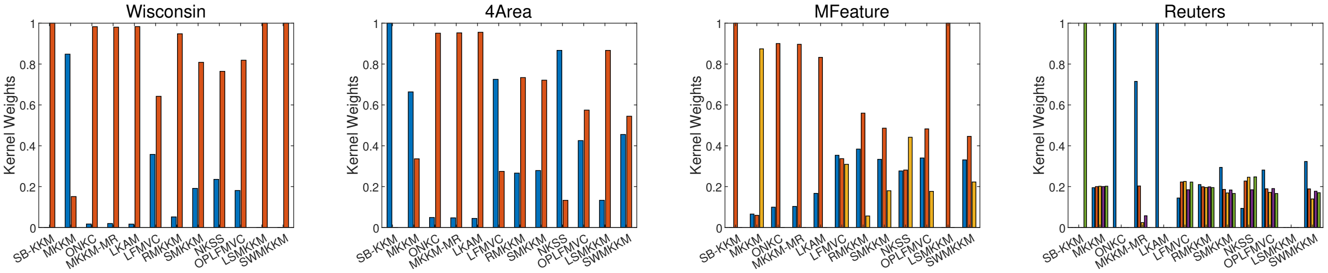

Figure 3: The kernel weights learned by various compared algorithms.

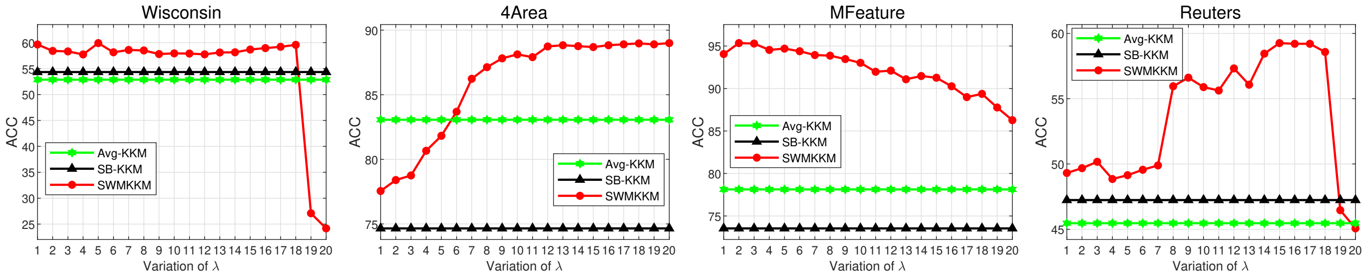

Figure 4: The sensitivity of SWMKKM with the variation of λ in term of ACC.

- Recently proposed LSMKKM does improve the clustering performance by its localized alignment criterion and min-max optimization paradigm. As seen, LSMKKM can be treated as the state-of-the-art comparison algorithm, since it achieves the most second best results on all datasets. However, our proposed SWMKKM still exceeds it by 4.4%, 1.8%, 24.8%, 2.2%, 9.2%, 0.4%, 17.2% and 12.2% on all benchmark datasets in term of ACC. This outperforming strongly demonstrates the superiority of our proposed sample weight criterion.
- As can be seen from the results, our proposed SWMKKM achieves the state-of-the-art performance on all datasets in terms of ACC and RI. For instance, SWMKKM exceeds the second best result on each dataset by 4.4%, 1.8%, 8.3%, 2.2%, 3.0%, 0.3%, 5.7% and 12.1% in term of ACC. Especially on Reuters dataset, while other algorithms can not achieve a discriminative effect and can not obtain a clustering accuracy of more than 50, our proposed SWMKKM achieve far ahead clustering performance.

Table 2: Aggregated clustering accuracy (ACC), normalized mutual information (NMI) and rand index (RI) comparison (mean±std) of different clustering algorithms on all benchmark datasets. Boldface indicates the best result and underline means the second best on each dataset. The results on purity (PUR) is similar and given in appendix due to space limit.

Algrithmn	Sonar	Wisconsin	Politicsuk	Cal-5	Wpbc	MFeature	4Area	Reuters
				ACC				
Avg-KKM	57.5 ± 0.2	52.9 ± 0.4	51.9 ± 2.6	59.8 ± 3.4	59.3 ± 0.1	78.1 ± 1.0	83.1 ± 0.2	45.5 ± 1.5
SB-KKM	58.6 ± 2.1	54.3 ± 0.0	59.6 ± 3.8	64.3 ± 0.7	59.3 ± 0.0	73.5 ± 2.1	74.7 ± 0.0	47.2 ± 0.0
MKKM	57.5 ± 0.2	54.1 ± 2.8	50.6 ± 3.1	52.8 ± 4.0	59.3 ± 0.0	63.8 ± 1.5	74.5 ± 0.0	45.4 ± 1.5
LMKKM	57.5 ± 0.2	46.0 ± 0.7	45.3 ± 0.4	53.5 ± 1.1	59.3 ± 0.0	64.9 ± 1.2	73.7 ± 0.0	-
ONKC	61.8 ± 0.0	56.5 ± 0.3	62.7 ± 1.2	68.1 ± 3.4	58.1 ± 0.5	79.8 ± 1.9	71.1 ± 0.0	41.8 ± 1.2
MKKM-MR	57.0 ± 0.0	55.8 ± 0.6	62.4 ± 1.7	70.2 ± 0.2	56.2 ± 0.0	79.5 ± 2.5	71.7 ± 0.0	46.2 ± 1.4
LKAM	57.0 ± 0.0	56.7 ± 0.2	61.3 ± 2.2	68.7 ± 3.6	64.4 ± 0.0	90.7 ± 0.0	51.2 ± 3.2	45.5 ± 0.0
LFMVC	56.2 ± 0.6	53.6 ± 0.5	63.3 ± 0.1	71.3 ± 2.7	58.2 ± 0.0	82.6 ± 0.0	83.3 ± 0.3	45.7 ± 1.6
RMKKM	57.5 ± 0.1	53.7 ± 0.6	52.3 ± 1.9	61.8 ± 4.1	58.8 ± 0.0	93.1 ± 3.0	70.6 ± 0.0	45.5 ± 1.5
SMKKM	63.8 ± 0.0	53.7 ± 0.6	47.9 ± 0.9	69.5 ± 2.6	58.2 ± 0.0	95.0 ± 0.3	70.8 ± 0.0	45.5 ± 0.7
NKSS	62.3 ± 0.0	54.9 ± 0.8	55.6 ± 0.1	64.2 ± 1.8	61.9 ± 0.0	82.8 ± 0.8	61.6 ± 2.6	-
SPMKC	54.6 ± 0.0	41.2 ± 0.7	52.4 ± 0.1	55.1 ± 3.6	56.7 ± 0.0	63.1 ± 1.5	74.3 ± 0.0	26.8 ± 0.0
OPLFMVC	59.9 ± 0.0	51.8 ± 5.9	66.8 ± 3.6	70.0 ± 3.5	57.2 ± 0.0	80.3 ± 1.8	65.1 ± 1.3	43.9 ± 1.0
LSMKKM	64.7 ± 0.0	58.1 ± 0.4	50.3 ± 0.2	75.9 ± 4.0	58.2 ± 0.0	94.9 ± 0.2	71.8 ± 0.4	47.1 ± 1.0
SWMKKM	**69.1 ± 0.0**	**59.9 ± 0.6**	**75.1 ± 0.7**	**78.1 ± 0.2**	**67.4 ± 0.6**	**95.3 ± 0.0**	**89.0 ± 0.0**	**59.3 ± 0.0**
				NMI				
Avg-KKM	1.7 ± 0.1	34.2 ± 0.8	25.9 ± 1.3	59.7 ± 2.9	2.0 ± 0.0	73.1 ± 0.6	62.1 ± 0.5	27.4 ± 0.4
SB-KKM	2.2 ± 0.8	31.4 ± 0.0	52.1 ± 5.5	59.3 ± 0.9	2.0 ± 0.0	67.7 ± 1.5	53.9 ± 0.0	25.5 ± 0.0
MKKM	1.7 ± 0.1	28.2 ± 2.3	26.2 ± 0.7	53.1 ± 3.4	2.0 ± 0.0	64.0 ± 0.8	53.8 ± 0.0	27.3 ± 0.4
LMKKM	1.7 ± 0.1	15.1 ± 1.5	22.2 ± 0.2	51.0 ± 1.3	2.0 ± 0.0	65.0 ± 0.3	52.7 ± 0.0	-
ONKC	3.8 ± 0.0	31.9 ± 0.2	43.1 ± 3.0	62.3 ± 2.3	2.8 ± 0.1	71.8 ± 1.5	46.2 ± 0.0	22.3 ± 0.4
MKKM-MR	1.2 ± 0.0	32.7 ± 0.3	43.4 ± 2.7	65.4 ± 0.3	0.0 ± 0.0	71.6 ± 2.0	47.0 ± 0.0	25.3 ± 0.7
LKAM	1.2 ± 0.0	**36.2 ± 0.1**	43.5 ± 0.7	64.0 ± 2.5	1.8 ± 0.0	82.3 ± 0.1	22.3 ± 1.7	29.9 ± 0.0
LFMVC	1.2 ± 0.2	32.4 ± 0.6	45.2 ± 0.0	65.4 ± 2.1	1.8 ± 0.0	78.2 ± 0.0	62.4 ± 1.2	27.4 ± 0.4
RMKKM	1.7 ± 0.0	31.1 ± 0.6	26.2 ± 1.3	61.8 ± 2.2	1.9 ± 0.0	87.2 ± 2.1	45.8 ± 0.0	27.4 ± 0.4
SMKKM	5.3 ± 0.0	31.2 ± 0.6	30.5 ± 0.9	64.2 ± 1.3	2.4 ± 0.0	89.7 ± 0.5	45.8 ± 0.0	27.7 ± 0.2
NKSS	4.4 ± 0.0	34.1 ± 0.6	31.2 ± 0.1	57.6 ± 1.2	0.1 ± 0.0	86.1 ± 1.4	38.8 ± 1.1	-
SPMKC	5.8 ± 0.0	3.9 ± 0.8	26.7 ± 0.1	51.9 ± 1.1	1.7 ± 0.0	65.3 ± 1.2	53.2 ± 0.0	0.6 ± 0.0
OPLFMVC	2.8 ± 0.0	31.4 ± 5.2	46.6 ± 3.7	66.2 ± 2.7	2.4 ± 0.0	76.6 ± 0.7	51.1 ± 3.8	24.8 ± 1.5
LSMKKM	6.1 ± 0.0	32.2 ± 0.4	20.5 ± 0.3	71.7 ± 2.8	2.4 ± 0.0	89.5 ± 0.3	44.6 ± 2.2	27.0 ± 0.6
SWMKKM	**10.6 ± 0.0**	34.5 ± 0.4	**53.6 ± 0.9**	**74.6 ± 1.0**	**3.7 ± 0.2**	**90.4 ± 0.1**	**68.7 ± 0.0**	**35.3 ± 0.0**
				RI				
Avg-KKM	1.8 ± 0.1	28.1 ± 0.7	27.9 ± 2.8	46.7 ± 4.6	2.9 ± 0.1	63.3 ± 1.1	62.5 ± 0.4	21.8 ± 1.4
SB-KKM	2.7 ± 1.2	24.5 ± 0.0	47.5 ± 6.0	49.5 ± 0.9	2.9 ± 0.0	57.4 ± 2.8	55.5 ± 0.0	23.6 ± 0.0
MKKM	1.8 ± 0.1	24.8 ± 2.2	27.6 ± 2.0	41.7 ± 4.4	2.9 ± 0.0	49.8 ± 1.2	55.4 ± 0.0	21.8 ± 1.4
LMKKM	1.8 ± 0.1	7.1 ± 0.5	20.7 ± 0.4	40.6 ± 0.8	2.9 ± 0.0	50.6 ± 0.6	52.0 ± 0.0	-
ONKC	5.1 ± 0.0	26.9 ± 0.4	40.8 ± 1.4	55.3 ± 3.8	2.3 ± 0.3	64.4 ± 2.4	47.1 ± 0.0	20.3 ± 0.3
MKKM-MR	1.5 ± 0.0	26.6 ± 0.4	40.9 ± 1.4	58.0 ± 0.3	0.0 ± 0.0	64.0 ± 3.2	48.0 ± 0.0	23.1 ± 0.6
LKAM	1.5 ± 0.0	31.8 ± 0.3	40.4 ± 1.5	57.2 ± 4.7	5.5 ± 0.0	80.5 ± 0.1	17.2 ± 1.7	24.1 ± 0.0
LFMVC	1.1 ± 0.4	27.9 ± 0.8	41.6 ± 0.1	63.4 ± 3.6	2.3 ± 0.0	72.1 ± 0.0	62.8 ± 0.8	22.1 ± 1.6
RMKKM	1.8 ± 0.1	23.6 ± 0.7	28.1 ± 2.1	49.2 ± 5.4	2.6 ± 0.0	86.3 ± 3.3	46.5 ± 0.0	21.8 ± 1.4
SMKKM	7.1 ± 0.0	23.6 ± 0.7	28.7 ± 1.2	57.4 ± 2.0	2.3 ± 0.0	89.3 ± 0.6	46.5 ± 0.0	22.1 ± 0.8
NKSS	5.6 ± 0.0	29.4 ± 0.9	30.0 ± 0.1	46.2 ± 2.5	0.9 ± 0.0	80.1 ± 2.2	35.9 ± 1.7	-
SPMKC	0.4 ± 0.0	-0.9 ± 0.5	26.8 ± 0.1	39.4 ± 4.0	1.4 ± 0.0	53.3 ± 1.4	43.4 ± 0.0	0.1 ± 0.0
OPLFMVC	3.5 ± 0.0	24.0 ± 5.1	45.6 ± 5.1	59.5 ± 5.5	1.7 ± 0.0	69.4 ± 0.5	50.7 ± 4.5	20.6 ± 0.5
LSMKKM	8.2 ± 0.0	29.5 ± 0.5	16.8 ± 0.2	72.0 ± 6.3	2.3 ± 0.0	89.2 ± 0.4	43.1 ± 0.4	21.6 ± 0.2
SWMKKM	**14.2 ± 0.0**	**32.6 ± 0.7**	**62.6 ± 0.9**	**72.6 ± 0.4**	**9.2 ± 0.3**	**90.1 ± 0.1**	**74.0 ± 0.0**	**35.5 ± 0.0**

- Comparing all results on kernel dataset of Reuters, it can be found that there is a obvious bottleneck of clustering performance. Taking ACC as an example, in recent years, various studies on multiple kernel k-means have not significantly improved it, and ACC is always maintained at about 45 to 47. However, the improvement of our proposed SWMKKM

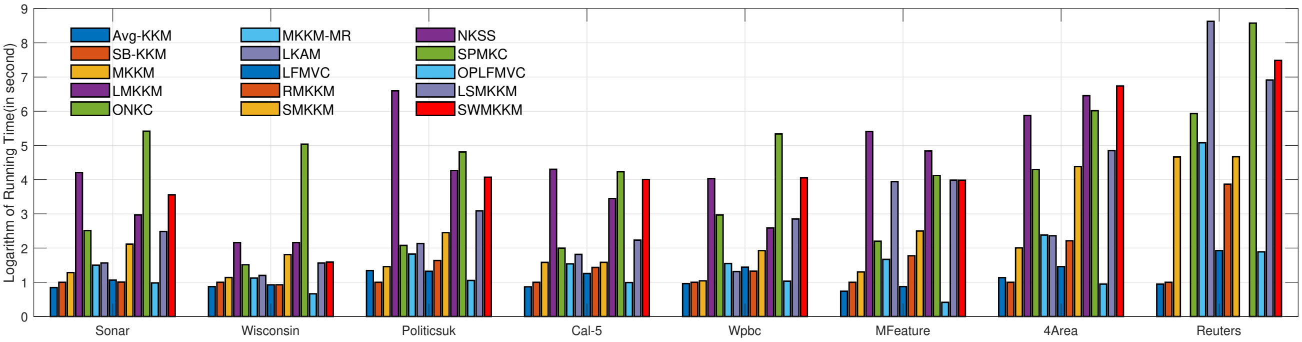

Figure 5: Running time comparison of different algorithms on eight benchmark datasets (logarithm in seconds).

is dramatically significant and breakthrough, which outperforms all comparison method by one-fifth.

- Most multi-view clustering algorithms, such as MKKM, ONKC, LFMVC, SPMKC and so on, adopt alternative optimization method to learn the optimal variables, which can not be guaranteed to achieve the convergence of a global optimum. This leads to their results are not stable enough, and the effect of the model can not be fully mined. Instead, our proposed SWMKKM adopts a novel paradigm with the reduced gradient descent method, which can make the learning process smoother and ensure convergence to the global optimization.

To sum up, our proposed SWMKKM strongly shows superior clustering performance compared with comparisons on all datasets, validating the effectiveness of the proposed sample weighted multiple kernel k-means. We expect that its ingenious idea and dramatic performance will attract intensive attention and inspire more general study in community. In addition, note that '-' in Table 2 indicates that the corresponding results can not be obtained because of out-of-memory error which is caused by unbearable memory complexity.

4.2.2 Convergence and Evolution. As discussed in Section 3.4, our proposed SWMKKM theoretically guarantees to achieve the convergence of a global optimum. To verify this point, we plot the curves of SWMKKM's objective value with reject to the number of iterations on all datasets, as shown in Figure 1. It can be observed that the objective value decreases monotonically and the algorithm usually converges quickly. Furthermore, to show the evolution of the learning process of SWMKKM, we calculate clustering performance at each iteration with learned $\mathbf{H}$, and report them in Figure 2. As observed, the clustering performance of SWMKKM usually rises rapidly on small fluctuations and then keep relatively stable in most cases, which sufficiently demonstrates effectiveness and necessity of our learning process. The results on the other datasets are similar and thus omitted due to space limit.

4.2.3 Kernel Coefficients Analysis. We also study the kernel weight coefficients learned by comparison algorithms on all datasets. The results on MFeatures and Reuters are plotted in Figure 3. As seen, the kernel weights learned by ONKC, MKKM-MR, LKAM and LSMKKM on each benchmark dataset are considerably sparse. This kind of sparsity would cause that the algorithm pays more attention on a certain kernel matrix and lacks sufficient mining of information in different kernels, resulting in unsatisfying performance. However, the kernel weights learned by our proposed SWMKKM are relatively denser, which promotes the full use of information in different kernels. The figures on the other datasets are similar and thus omitted due to space limit.

4.2.4 Parameter Sensitivity Analysis. To further study the influence of hyper-parameters λ on our proposed SWMKKM, we carry out corresponding experiments and plot the change of ACC with the variation of λ, as reported in Figure 4. Note that the results of Avg-KKM and SB-KKM are also given as baseline references. From the observation, our proposed SWMKKM achieves advanced clustering performance across a wide range of λ. The figures on the other datasets are similar and thus omitted due to space limit.

4.2.5 Running Time Comparison. Finally, we also report the running time of all comparison algorithms on all benchmark datasets in our experiment in Figure 5. Note that, we scale the values and set the execution time of SB-KKM be reference for clearer comparison. As observed, our proposed SWMKKM is at a medium level and holds a running time being match for that of LSMKKM. What's more, SWMKKM does not greatly increase the time cost while significantly improving the clustering performance.

5 CONCLUSION

We observed that the existing algorithms still have a performance bottleneck due to the lack of considering different contribution of samples. As expected, they ignore the relationship among the importance of different samples and thus the "ideal" similarity structure cannot be effectively generated. To address this issue, we propose a novel sample weighted multiple kernel k-means (SWMKKM) in this paper, which, for the first time, develops a sample weighted criterion for clustering. We inherit the min-max optimization paradigm from SMKKM and introduce the reduced gradient descent method to solve the resultant optimization problem. Comprehensive experimental results show the leading performance and the effectiveness of SWMKKM. Though empirically observing that SWMKKM achieves exciting improvement in this work, we have not found a satisfying method to learn adaptive sample weights and further improve the performance. In the future, we plan to explore the general effect of sample weight criterion and seek for a better manner to study the importance of different samples.

REFERENCES

[1] Seojin Bang, Yaoliang Yu, and Wei Wu. 2018. Robust Multiple Kernel k-means Clustering using Min-Max Optimization. arXiv:1803.02458 [cs.LG]

[2] Francesco Camastra and Alessandro Verri. 2005. A novel kernel method for clustering. *IEEE transactions on pattern analysis and machine intelligence* 27, 5 (2005), 801–805.

[3] John M. Danskin. 1966. The Theory of Max-Min, with Applications. In *SIAM Journal on Applied Mathematics*. 641–664.

[4] Liang Du, Peng Zhou, Lei Shi, Hanmo Wang, Mingyu Fan, Wenjian Wang, and Yi-Dong Shen. 2015. Robust multiple kernel k-means using l21-norm. In *Twenty-fourth international joint conference on artificial intelligence*.

[5] Mehmet Gönen and Ethem Alpaydın. 2011. Multiple kernel learning algorithms. *The Journal of Machine Learning Research* 12 (2011), 2211–2268.

[6] Mehmet Gönen and Adam A Margolin. 2014. Localized data fusion for kernel k-means clustering with application to cancer biology. In *Advances in Neural Information Processing Systems*. 1305–1313.

[7] Mehmet Gönen and Adam A. Margolin. 2014. Localized Data Fusion for Kernel k-Means Clustering with Application to Cancer Biology. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 1305–1313.

[8] Yina Han, Kunde Yang, Yuanliang Ma, and Guizhong Liu. 2013. Localized multiple kernel learning via sample-wise alternating optimization. *IEEE transactions on cybernetics* 44, 1 (2013), 137–148.

[9] Yina Han, Kunde Yang, Yixin Yang, and Yuanliang Ma. 2016. Localized multiple kernel learning with dynamical clustering and matrix regularization. *IEEE transactions on neural networks and learning systems* 29, 2 (2016), 486–499.

[10] Hsin-Chien Huang, Yung-Yu Chuang, and Chu-Song Chen. 2012. Multiple Kernel Fuzzy Clustering. *IEEE Trans. Fuzzy Systems* 20, 1 (2012), 120–134.

[11] Hsin-Chien Huang, Yung-Yu Chuang, and Chu-Song Chen. 2011. Multiple kernel fuzzy clustering. *IEEE Transactions on Fuzzy Systems* 20, 1 (2011), 120–134.

[12] Mahdi M Kalayeh, Haroon Idrees, and Mubarak Shah. 2014. NMF-KNN: Image annotation using weighted multi-view non-negative matrix factorization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 184–191.

[13] Aparajita Khan and Pradipta Maji. 2019. Approximate graph Laplacians for multimodal data clustering. *IEEE transactions on pattern analysis and machine intelligence* 43, 3 (2019), 798–813.

[14] Marius Kloft, Ulf Brefeld, Sören Sonnenburg, and Alexander Zien. 2011. Lp-norm multiple kernel learning. *The Journal of Machine Learning Research* 12 (2011), 953–997.

[15] Marius Kloft, Ulrich Rückert, and Peter L Bartlett. 2010. A unifying view of multiple kernel learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 66–81.

[16] Ritwik Kumar, Ting Chen, Moritz Hardt, David Beymer, Karen Brannon, and Tanveer Syeda-Mahmood. 2013. Multiple kernel completion and its application to cardiac disease discrimination. In *2013 IEEE 10th International Symposium on Biomedical Imaging*. IEEE, 764–767.

[17] Liang Li, Siwei Wang, Xinwang Liu, En Zhu, Li Shen, Kenli Li, and Keqin Li. 2022. Local Sample-weighted Multiple Kernel Clustering with Consensus Discriminative Graph. *arXiv preprint arXiv:2207.02846* (2022).

[18] Miaomiao Li, Xinwang Liu, Lei Wang, Yong Dou, Jianping Yin, and En Zhu. 2016. Multiple Kernel Clustering with Local Kernel Alignment Maximization. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. 1704–1710.

[19] Teng Li, Yong Dou, Xinwang Liu, Yang Zhao, and Qi Lv. 2017. Multiple kernel clustering with corrupted kernels. *Neurocomputing* 267 (2017), 447–454.

[20] Weixuan Liang, Sihang Zhou, Jian Xiong, Xinwang Liu, Siwei Wang, En Zhu, Zhiping Cai, and Xin Xu. 2020. Multi-view spectral clustering with high-order optimal neighborhood laplacian matrix. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[21] Xinwang Liu, Yong Dou, Jianping Yin, Lei Wang, and En Zhu. 2016. Multiple Kernel *k*-Means Clustering with Matrix-Induced Regularization. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. 1888–1894.

[22] Xinwang Liu, Li Liu, Qing Liao, Siwei Wang, Yi Zhang, Wenxuan Tu, Chang Tang, Jiyuan Liu, and En Zhu. 2021. One Pass Late Fusion Multi-view Clustering. In *International Conference on Machine Learning*. PMLR, 6850–6859.

[23] Xinwang Liu, Sihang Zhou, Li Liu, Chang Tang, Siwei Wang, Jiyuan Liu, and Yi Zhang. 2021. Localized simple multiple kernel k-means. In *International Conference on Compute Vision*. 6850–6859.

[24] Xinwang Liu, Sihang Zhou, Yueqing Wang, Miaomiao Li, Yong Dou, En Zhu, and Jianping Yin. 2017. Optimal Neighborhood Kernel Clustering with Multiple Kernels. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. 2266–2272.

[25] Xinwang Liu, En Zhu, and Jiyuan Liu. 2020. SimpleMKKM: Simple Multiple Kernel K-means. *arXiv preprint arXiv:2005.04975* (2020).

[26] Xinwang Liu, En Zhu, Jiyuan Liu, Timothy M. Hospedales, Yang Wang, and Meng Wang. 2020. SimpleMKKM: Simple Multiple Kernel K-means. *CoRR* abs/2005.04975 (2020). arXiv:2005.04975 https://arxiv.org/abs/2005.04975

[27] Supratim Manna, Jessy Rimaya Khonglah, Anirban Mukherjee, and Goutam Saha. 2021. Robust kernelized graph-based learning. *Pattern Recognition* 110 (2021), 107628.

[28] Vishal M Patel and René Vidal. 2014. Kernel sparse subspace clustering. In *2014 ieee international conference on image processing (icip)*. IEEE, 2849–2853.

[29] Bryan Perozzi, Leman Akoglu, Patricia Iglesias Sánchez, and Emmanuel Müller. 2014. Focused Clustering and Outlier Detection in Large Attributed Graphs. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, New York, USA) *(KDD '14)*. ACM, New York, NY, USA, 1346–1355. https://doi.org/10.1145/2623330.2623682

[30] Alain Rakotomamonjy, Francis R. Bach, Stéphane Canu, and Yves Grandvalet. 2008. SimpleMKL. *JMLR* 9 (2008), 2491–2521.

[31] Zhenwen Ren and Quansen Sun. 2020. Simultaneous global and local graph structure preserving for multiple kernel clustering. *IEEE transactions on neural networks and learning systems* 32, 5 (2020), 1839–1851.

[32] Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20 (1987), 53–65.

[33] Bernhard Scholkopf and Alexander J Smola. 2001. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press.

[34] Grigorios Tzortzis and Aristidis Likas. 2008. The global kernel k-means clustering algorithm. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. IEEE, 1977–1984.

[35] H. Wang, Y. Yang, and B. Liu. 2020. GMC: Graph-Based Multi-View Clustering. *IEEE Transactions on Knowledge and Data Engineering* 32, 6 (2020), 1116–1129.

[36] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. 2021. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4690–4699.

[37] Siwei Wang, Xinwang Liu, En Zhu, Chang Tang, Jiyuan Liu, Jingtao Hu, Jingyuan Xia, and Jianping Yin. 2019. Multi-view Clustering via Late Fusion Alignment Maximization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*. 3778–3784.

[38] Yueqing Wang, Xinwang Liu, Yong Dou, and Rongchun Li. 2017. Approximate Large-scale Multiple Kernel k-means Using Deep Neural Network.. In *IJCAI*. 3006–3012.

[39] Yueqing Wang, Xinwang Liu, Yong Dou, and Rongchun Li. 2017. Multiple kernel clustering framework with improved kernels. *Discover* 1, 2 (2017), 3–4.

[40] Shan Zeng, Zhiyong Wang, Rui Huang, Ling Chen, and David Feng. 2019. A study on multi-kernel intuitionistic fuzzy C-means clustering with multiple attributes. *Neurocomputing* 335 (2019), 59–71.

[41] Changqing Zhang, Huazhu Fu, Qinghua Hu, Xiaochun Cao, Yuan Xie, Dacheng Tao, and Dong Xu. 2018. Generalized latent multi-view subspace clustering. *IEEE transactions on pattern analysis and machine intelligence* 42, 1 (2018), 86–99.

[42] Lei Zhang and Qixin Cao. 2011. A novel ant-based clustering algorithm using the kernel method. *Information Sciences* 181, 20 (2011), 4658–4672.

[43] Tiejian Zhang, Xinwang Liu, Lei Gong, Siwei Wang, Xin Niu, and Li Shen. 2021. Late Fusion Multiple Kernel Clustering with Local Kernel Alignment Maximization. *IEEE Transactions on Multimedia* (2021), 1–1. https://doi.org/10.1109/TMM.2021.3136094

[44] Xiaoqian Zhang, Xuqian Xue, Huaijiang Sun, Zhigui Liu, Li Guo, and Xin Guo. 2021. Robust multiple kernel subspace clustering with block diagonal representation and low-rank consensus kernel. *Knowledge-Based Systems* (2021), 107243.

[45] Yi Zhang, Xinwang Liu, Siwei Wang, Jiyuan Liu, Sisi Dai, and En Zhu. 2021. One-Stage Incomplete Multi-view Clustering via Late Fusion. In *Proceedings of the 29th ACM International Conference on Multimedia*. 2717–2725.

[46] Bin Zhao, James T. Kwok, and Changshui Zhang. 2009. Multiple Kernel Clustering. In *SDM*. 638–649.

[47] Liang Zheng, Shengjin Wang, Lu Tian, Fei He, Ziqiong Liu, and Qi Tian. 2015. Query-adaptive late fusion for image search and person re-identification. In *Computer Vision and Pattern Recognition*. 1741–1750.

[48] Caiming Zhong, Xiaodong Yue, Zehua Zhang, and Jingsheng Lei. 2015. A clustering ensemble: Two-level-refined co-association matrix with path-based transformation. *Pattern Recognition* 48, 8 (2015), 2699–2709.

[49] Sihang Zhou, Xinwang Liu, Miaomiao Li, En Zhu, Li Liu, Changwang Zhang, and Jianping Yin. 2019. Multiple kernel clustering with neighbor-kernel subspace segmentation. *IEEE transactions on neural networks and learning systems* 31, 4 (2019), 1351–1362.

[50] Sihang Zhou, Qiyuan Ou, Xinwang Liu, Siqi Wang, Luyan Liu, Siwei Wang, En Zhu, Jianping Yin, and Xin Xu. 2021. Multiple Kernel Clustering With Compressed Subspace Alignment. *IEEE Transactions on Neural Networks and Learning Systems* (2021).