Efficient Federated Multi-View Clustering with Integrated Matrix Factorization and K-Means

Wei Feng[1] , **Zhenwei Wu**[1] , **Qianqian Wang** [2*] , **Bo Dong** [3] , **Zhiqiang Tao**[4] and **Quanxue Gao**[2]
[1]School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, China
[2]School of Telecommunications Engineering, Xidian University, Xi'an, China
[3]School of Continuing Education, Xi'an Jiaotong University, Xi'an, China
[4]School of Information, Rochester Institute of Technology, NY, USA;
weifeng.ft@xjtu.edu.cn, wzw3213151007@stu.xjtu.edu.cn, qqwang@xidian.edu.cn*,
dong.bo@mail.xjtu.edu.cn, zqtaomail@gmail.com, qxgao@xidian.edu.cn

Abstract

Multi-view clustering is a popular unsupervised multi-view learning method. Real-world multi-view data are often distributed across multiple entities, presenting a challenge for performing multi-view clustering. Federated learning provides a solution by enabling multiple entities to collaboratively train a global model. However, existing federated multi-view clustering methods usually conduct feature extraction and clustering in separate steps, potentially leading to a degradation in clustering performance. To address this issue and for the sake of efficiency, we propose a novel **F**ederated **M**ulti-**V**iew **C**lustering method with **I**ntegrated **M**atrix Factorization and **K**-Means (FMVC-IMK), which integrates matrix factorization and multi-view K-means into one step. Additionally, an adaptive weight is employed to balance the influence of data from each view. FMVC-IMK further incorporates a graph-based regularizer to preserve the original data's geometric structure within the learned global clustering structure. We also develop a federated optimization approach to collaboratively learn a global clustering result without sharing any original data. Experimental results on multiple datasets demonstrate the effectiveness of FMVC-IMK.

1 Introduction

Multi-view data [Wang *et al.*, 2023] can provide a comprehensive description of an object from different perspectives, such as modalities and features, where each view provides consistent and complementary information [Xu *et al.*, 2013]. For instance, human activities can be captured through cameras, video recorders, and textual descriptions. Due to the high expense of reliable label acquisition, multi-view clustering has emerged as a popular unsupervised learning method within the field of multi-view learning [Sun *et al.*, 2020]. Existing multi-view clustering methods can be roughly divided into five categories [Yang and Wang, 2018]: Co-training style algorithms [Jiang *et al.*, 2013], Multi-view graph clustering [Huang *et al.*, 2019; Wen *et al.*, 2020], Multi-kernel learning [Tzortzis and Likas, 2012], Multi-view subspace clustering [Zheng *et al.*, 2023], and Multi-task multi-view clustering [Al-Stouhi and Reddy, 2014].

Although multi-view clustering methods have exhibited promising performance, they are mainly designed for centralized scenarios where multi-view data is located in a single party. In reality, however, multi-view data is often collected and maintained by different entities [Chen *et al.*, 2023; Huang *et al.*, 2020; Feng and Yu, 2020]. Due to data privacy concerns, these data holders are generally unwilling to share their data with others directly. To address this challenge, federated learning was introduced, enabling collaborative training of multi-view clustering models without the need for direct data sharing [McMahan *et al.*, 2017]. For example, [Chen *et al.*, 2023] proposed a federated deep multi-view clustering method with global self-supervision, which has demonstrated remarkable clustering performance. [Ren *et al.*, 2024] both consider unaligned and incomplete data in federated multi-view clustering.

The utilization of deep federated models incurs significant computation and communication costs, rendering them unsuitable for scenarios that show a requirement for efficiency. Consequently, several heuristic federated multi-view clustering methods have been developed based on non-negative matrix factorization (NMF) [Huang *et al.*, 2022] and K-means. Owing to their simplicity and efficiency, these methods can better satisfy the efficiency requirements in computation/communication-sensitive applications. Nonetheless, it is widely recognized that both NMF and K-means are unable to process linearly inseparable data and retain the local geometric structure of the original data. For another, existing heuristic federated multi-view clustering performs feature extraction and clustering in two separate procedures [Huang *et al.*, 2022; Hu *et al.*, 2023], and hence, the clustering results cannot guide the extraction process. These limitations lead to the potential risk of performance degradation and greatly restrict their application.

To overcome these weaknesses, we develop an efficient **F**ederated **M**ulti-**V**iew **C**lustering with **I**ntegrated **M**atrix Factorization and **K**-Means (FMVC-IMK). It integrates K-Means into Federated NMF, enabling itself to enhance the

*Corresponding Author

performance of each other for superior clustering results.

Besides, considering the heterogeneity of distributed data, we introduce an adaptive update weight that can be locally computed to quantify each client's contribution to the global model. To improve the performance on linearly inseparable multi-view data, a graph-based regularizer is introduced to retain the local geometric structure information. Finally, we design a federated optimization algorithm to optimize the model without sharing any original multi-view data. To summarize, the main contributions of the work are as follows:

- We propose a novel FMVC-IMK for federated multi-view clustering. It performs federated multi-view NMF and K-means in an integrated step by approximating the coefficient matrix of NMF with the indicator matrix and centroid matrix of K-means. Therefore, it can learn a better clustering structure from multi-view data located in different clients.
- To retain the geometric structure in the original data, we introduce a graph-based regularizer to constrain the learned indicator to be consistent with the original data, which helps enhance the clustering performance, particularly for data that exhibit non-linear separability
- We develop an optimization algorithm with adaptive weights to cooperatively optimize the objective function among the server and multiple clients. The adaptive weights dynamically adjust the contribution of each client's locally trained model to the global model.
- We conduct extensive experiments on eight multi-view datasets and compare FMVC-IMK with several methods. The experimental results demonstrate the effectiveness and superiority of our method.

2 Related Work

2.1 Heuristic Multi-View Clustering

Heuristic multi-view clustering [Huang *et al.*, 2021] shows higher computational efficiency and demonstrates its significance in many computation-sensitive applications. NMF becomes an effective method to build multi-view clustering methods because they could well exploit the information of different views. For example, [Liu *et al.*, 2013] formulated a multi-view clustering method via joint NMF to learn a common consensus; ONMF is an variant of NMF with orthogonal constraints and [Liang *et al.*, 2020] applied co-orthogonal constraints on representation matrices and basis matrices to further capture the diversity within views and learn the appropriate basis matrices.

Apart from NMF, K-means is another popular method to build multi-view clustering [Cai *et al.*, 2013]. To handle linearly inseparable data, [Gao *et al.*, 2019b] introduced a multi-manifold regularizer to learn the hypergraph weights. Similarly, [Zhu *et al.*, 2020] imposed the constraints on high-level manifold consensus, aiming to capture deeper underlying structures of the data. Besides, [Zheng *et al.*, 2023] proposed a novel one-pass method, which achieves better clustering performance than traditional NMF-based methods. However, all the methods mentioned above are designed for centralized applications and cannot be directly utilized in federated scenarios.

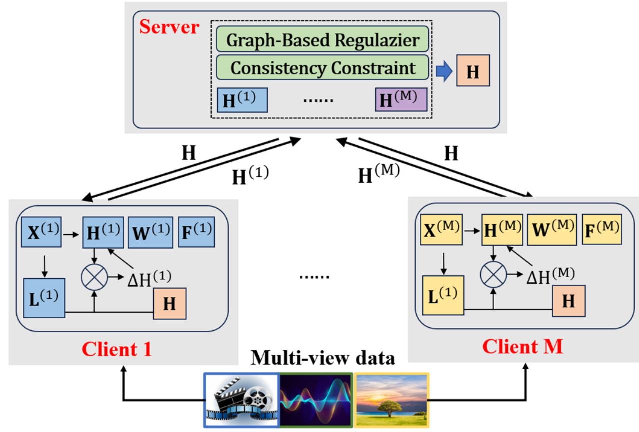

Figure 1: The framework of the proposed FMVC-IMK. G-R denotes the graph regularizer.

2.2 Federated Multi-View Clustering

Federated learning, as a distributed machine learning paradigm, aims to train models on multiple local clients without transferring raw data or other sensitive data, which can be roughly categorized into horizontal federated learning (HFL) [Gao *et al.*, 2019a; Zhao *et al.*, 2021], Vertical federated learning (VFL) [Sun *et al.*, 2021; Liu *et al.*, 2020], and Federated Transfer Learning (FTL) [Kevin *et al.*, 2021; Yang *et al.*, 2020]. The federated learning is quickly extended to federated multi-view learning [Che *et al.*, 2022]. In terms of federated multi-view clustering, [Chen *et al.*, 2023] developed federated deep multi-view clustering with global self-supervision. Effective as it is, the deep model it adopts is computationally expensive and thereby cannot satisfy the efficiency requirements in some scenarios. Differently, [Huang *et al.*, 2022] built a federated multi-view clustering method via NMF and K-means that helps reduce the computational cost. [Hu *et al.*, 2023] realized a federated multi-view fuzzy C-means (FedMVFCM). Nevertheless, these methods still confront the intrinsic weaknesses of NMF and K-means.

3 Method

3.1 Problem Statement

Suppose $\mathbf{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \cdots, \mathbf{X}^{(M)}\}$ denotes the multi-view data, where M is the number of views and $\mathbf{X}^{(m)} \in \mathbb{R}^{N \times d^{(m)}} (m = 1, 2, ..., M)$ is the data matrix of the m-th view; N is the number of samples and $d^{(m)}$ represents the feature dimension. Suppose that in a federated setting, there exists a centralized server $\boldsymbol{\mathcal{S}}$ and M local clients, with each client $\boldsymbol{\mathcal{C}}_m$ holding the data $\mathbf{X}^{(m)}$. Our design goal is to collaboratively learn a clustering model while considering the privacy of client data.

3.2 Objective Function

Given the $\mathbf{X}^{(m)}$ of the m-th view, we can factorize it into two matrices with lower dimensions via ONMF:

$$\begin{aligned} &||\mathbf{X}^{(m)} - \mathbf{G}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &s.t.\ \mathbf{G}^{(m)}, \mathbf{F}^{(m)} \geq 0, \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I} \end{aligned} \tag{1}$$

where $\mathbf{G}^{(m)} \in \mathbb{R}^{N\times c}$ and $\mathbf{F}^{(m)} \in \mathbb{R}^{c\times d^{(m)}}$ are the coefficient matrix and the basis matrix of the m-th device, respectively. Considering the m-view data held by the m-th local clients, we have the following objective function for Federated multi-view clustering:

$$\begin{aligned} &\min_{\mathbf{G}^{(m)},\mathbf{F}^{(m)}} \sum_{m=1}^{M} ||\mathbf{X}^{(m)} - \mathbf{G}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &s.t.\ \ \mathbf{G}^{(m)}, \mathbf{F}^{(m)} \geq 0, \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I} \end{aligned} \tag{2}$$

Since it has been proved that K-means is a matrix factorization problem [Bauckhage, 2015], by performing K-means on the coefficient matrix $\mathbf{G}^{(m)}$, we have:

$$\min_{\mathbf{H}^{(m)},\mathbf{W}^{(m)}} ||\mathbf{G}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}||_F^2 \tag{3}$$

where $\mathbf{H}^{(m)} \in \mathbb{R}^{N\times k}$ is the indicator matrix, of which the i-th row $\mathbf{H}_i^{(m)}$ is a **one-hot** vector and $\mathbf{H}_{i,j}^{(m)} = 1$ indicates that it assigns i-th sample of the m-view to the j-th cluster. By integrating (1) and (3), we have:

$$\begin{aligned} &\min_{\mathbf{H}^{(m)},\mathbf{W}^{(m)},\mathbf{F}^{(m)}} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &s.t. \qquad \mathbf{F}^{(m)} \geq 0, \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I} \end{aligned} \tag{4}$$

Similar to [Huang *et al.*, 2022], we remove the non-negative constraint of $\mathbf{F}^{(m)}$ to expand the application scope because the input data is also not constrained to be non-negative. We set an adaptive updating weight $\alpha^{(m)}(m = 1, 2, \cdots, M)$ for each local client based on their contribution to the global model to adjust the impact of each view on the clustering performance:

$$\begin{aligned} &\min_{\mathbf{H}^{(m)},\mathbf{W}^{(m)},\mathbf{F}^{(m)}} \sum_{m=1}^{M} \alpha^{(m)}||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &s.t.\ \ \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I}, \sum_{i=1}^{M} \alpha^{(m)} = 1 \end{aligned} \tag{5}$$

It should be noted that $\mathbf{H}_i^{(m)}$ is a discrete one-hot vector and thus difficult to optimize. Therefore, we introduce a regularization term $||{\mathbf{H}^{(m)}}^T\mathbf{H}^{(m)} - \mathbf{I}||_F^2$ to obtain a relaxed solution. For another, since we aim to learn a global clustering structure from multi-view data, we introduce a global indicator matrix $\mathbf{H}$ and constrain it with $\mathbf{H}^{(1)} = \mathbf{H}^{(2)} = \cdots = \mathbf{H}^{(M)} = \mathbf{H}$ to enforce the model to learn a consistent clustering structure:

$$\begin{aligned} &\min_{\mathbf{H}^{(m)},\mathbf{H}} \sum_{m=1}^{M} ||{\mathbf{H}^{(m)}}^T\mathbf{H} - \mathbf{I}||_F^2 \\ &s.t.\ \ \mathbf{H}^{(1)} = \cdots = \mathbf{H}^{(M)} = \mathbf{H} \end{aligned} \tag{6}$$

Graph-Based Regularization: NMF-based methods usually cannot handle data that are not linearly separated. To address this issue, we employ a graph-based regularization term to retain the local geometric structure, such that similar samples should be assigned to the same clusters:

$$\frac{1}{2}\min_{\mathbf{H}} \sum_{m=1}^{M} \sum_{i,j=1}^{N} ||\mathbf{H}_i - \mathbf{H}_j||_2^2 \mathbf{S}_{ij}^{(m)} \tag{7}$$

where $\mathbf{S}^{(m)}$ is the similarity matrix of the m-th view and can be locally calculated by m-th client as follows:

$$\mathbf{S}_{ij}^{(m)} = \begin{cases} e^{-\frac{\|\mathbf{x}_i^{(m)} - \mathbf{x}_j^{(m)}\|^2}{2\theta^2}}, \mathbf{x}_i^{(m)} \in \boldsymbol{\mathcal{N}}_{p,j}^{(m)} \text{ or } \mathbf{x}_j^{(m)} \in \boldsymbol{\mathcal{N}}_{p,i}^{(m)} \\ 0 \quad \text{, otherwise} \end{cases} \tag{8}$$

where $\boldsymbol{\mathcal{N}}_{p,i}^{(m)}$ represents the p-nearest neighbors of $\mathbf{x}_i^{(m)}$ in the m-th view. It has been proved in [Yang *et al.*, 2022] that (7) is equal to:

$$\sum_{m=1}^{M} Tr(\mathbf{H}^T\mathbf{L}^{(m)}\mathbf{H}) \tag{9}$$

where $\mathbf{L}^{(m)} = \mathbf{D}^{(m)} - \mathbf{S}^{(m)}$ is the Laplacian matrix; $\mathbf{D}^{(m)}$ is a diagonal matrix and $\mathbf{D}_{ii}^{(m)} = \sum_{j=1}^{N} \mathbf{S}_{ij}^{(m)}$.

Introducing the above two regularization terms into (5), we get the final objective function:

$$\begin{aligned} &\min_{\mathbf{H}^{(m)},\mathbf{W}^{(m)},\mathbf{F}^{(m)},\mathbf{H}} \sum_{m=1}^{M} \alpha^{(m)}||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &+ \sum_{m=1}^{M}\left(\frac{\mu}{2}||{\mathbf{H}^{(m)}}^T\mathbf{H} - \mathbf{I}||_F^2 + \frac{\lambda}{2}Tr(\mathbf{H}^T\mathbf{L}^{(m)}\mathbf{H})\right) \\ &s.t.\ \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I},\ \sum_{i=1}^{M} \alpha^{(m)} = 1, \\ &\mathbf{H}^{(1)} = \cdots = \mathbf{H}^{(M)} = \mathbf{H} \end{aligned} \tag{10}$$

where μ and λ are penalty parameters.

3.3 Adaptive Update of $\alpha^{(m)}$

As aforementioned, we leverage $\alpha^{(m)}$ to adjust the influence of each client for better performance, which incurs a problem of how to decide the value of $\alpha^{(m)}$. By observing (10), we find that updating $\alpha^{(m)}$ is only related to the first term, *i.e.*, (5). According to Theorem 1, the weight of each device can be determined automatically.

Theorem 1. *If the weight of each client is fixed, solving the problem* (5) *is equivalent to solving the following problem*

$$\begin{aligned} &\min_{\mathbf{H}^{(m)},\mathbf{W}^{(m)},\mathbf{F}^{(m)}} \sum_{m=1}^{M} \sqrt{||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2} \\ &s.t.\ \ \mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T = \mathbf{I} \end{aligned} \tag{11}$$

Proof. Taking Γ and Λ as the Lagrange multiplier and the proxy for the constraint to $\mathbf{F}^{(m)}$, respectively, the Lagrange function is as follows:

$$\sum_{m=1}^{M} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F + \Gamma(\Lambda, \mathbf{F}^{(m)}) \tag{12}$$

We then take the partial derivative of (12) with respect to (w.r.t.) $\mathbf{F}^{(m)}$:

$$\begin{aligned} &\frac{\partial \sum_{m=1}^{M} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F}{\partial \mathbf{F}^{(m)}} + \frac{\Gamma(\Lambda, \mathbf{F}^{(m)})}{\partial \mathbf{F}^{(m)}} \\ =&\frac{\partial ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F}{\partial \mathbf{F}^{(m)}} + \frac{\Gamma(\Lambda, \mathbf{F}^{(m)})}{\partial \mathbf{F}^{(m)}} \end{aligned} \tag{13}$$

Introducing Γ and Λ to (5), we get the Lagrange function:

$$\sum_{m=1}^{M} \alpha^{(m)} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 + \Gamma(\Lambda, \mathbf{F}^{(m)}) \tag{14}$$

Similarly, taking the partial derivative of (14) w.r.t. $\mathbf{F}^{(m)}$:

$$\begin{aligned} &\sum_{m=1}^{M} \frac{\partial \alpha^{(m)} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2}{\partial \mathbf{F}^{(m)}} + \frac{\Gamma(\Lambda, \mathbf{F}^{(m)})}{\partial \mathbf{F}^{(m)}} \\ &= \alpha^{(m)} \frac{\partial ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2}{\partial \mathbf{F}^{(m)}} + \frac{\Gamma(\Lambda, \mathbf{F}^{(m)})}{\partial \mathbf{F}^{(m)}} \end{aligned} \tag{15}$$

If we fix all the weight $\alpha^{(m)}$ to the same, (5) is equivalent to (11). So the corresponding partial derivatives (13) and (15) should be equal. By solving this equation, we have:

$$\alpha^{(m)} = \frac{1}{2||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F} \tag{16}$$

Thus, $\alpha^{(m)}$ adaptively updates according to (16).

3.4 Optimization Algorithm

(10) can be easily optimized in centralized scenarios, but it becomes challenging in federated settings because data with different views is held by different clients. Therefore, we develop a collaborative optimization algorithm to learn the optimal solution of (10). We first obtain the following observations from (10):

- There are four parameters to be optimized: $\mathbf{H}^{(m)}$, $\mathbf{W}^{(m)}$, $\mathbf{F}^{(m)}$, and $\mathbf{H}$;
- The update of $\mathbf{H}$ must be conducted centrally because it is only related to parameter $\mathbf{H}^{(m)}$ and does not involve local data on the client;
- $\mathbf{F}^{(m)}$, $\mathbf{W}^{(m)}$, and $\alpha^{(m)}$ can be updated locally on each client because it is related to local private data;
- The update of $\mathbf{H}^{(m)}$ is related to centralized parameters $\mathbf{H}$, local parameters $\mathbf{F}^{(m)}$ and $\mathbf{H}^{(m)}$, and most importantly, it is related to local private data $\mathbf{X}^{(m)}$, its update can only be performed locally. Therefore, it is the only parameter that needs to be transferred between each local client and the centralized server. Moreover, since $\mathbf{H}^{(m)}$ is the cluster assignment of samples on each client, it will not result in any data leakage, which fulfills the privacy requirements of federated learning.

Based on the observations, the federated optimization algorithm is as follows:

(1) Solving $\mathbf{H}$ with fixed $\mathbf{H}^{(m)}$, $\mathbf{W}^{(m)}$, $\mathbf{F}^{(m)}$ by $\boldsymbol{\mathcal{S}}$: In this case, solving $\mathbf{H}$ in function (10) is equivalent to solving the following objective with a fixed $\mathbf{H}^{(m)}$:

$$\begin{aligned} \min_{\mathbf{H}} \sum_{m=1}^{M} &\left(\frac{\mu}{2} ||{\mathbf{H}^{(m)}}^T \mathbf{H} - \mathbf{I}||_F^2 + \frac{\lambda}{2} Tr(\mathbf{H}^T \mathbf{L}^{(m)} \mathbf{H}) \right) \\ s.t. \quad &\mathbf{H}^{(1)} = \cdots = \mathbf{H}^{(M)} = \mathbf{H} \end{aligned} \tag{17}$$

To solve (17), we introduce a new term in the function derived from the constraint and Augmented Lagrangian function of (17) concerning $\mathbf{H}$:

$$\begin{aligned} \boldsymbol{\mathcal{L}}_{\mathbf{H}} = \min_{\mathbf{H}} \sum_{m=1}^{M} &\Bigg(\frac{\mu}{2} ||{\mathbf{H}^{(m)}}^T \mathbf{H} - \mathbf{I}||_F^2 + \frac{\rho}{2} ||\mathbf{H} - \mathbf{H}^{(m)}||_F^2 \\ &+ \left\langle \Phi^{(m)}, \mathbf{H}^{(m)} - \mathbf{H} \right\rangle + \frac{\lambda}{2} Tr(\mathbf{H}^T \mathbf{L}^{(m)} \mathbf{H}) \Bigg) \end{aligned} \tag{18}$$

where $\Phi^{(m)}$ is the Lagrangian multiplier of client $\boldsymbol{\mathcal{C}}_m$, ρ is the penalty parameter, and $\langle \cdot, \cdot \rangle$ is the inner product operation. We take the partial derivative of $\boldsymbol{\mathcal{L}}_{\mathbf{H}}$ w.r.t. $\mathbf{H}$ and set it to 0, we have:

$$\begin{aligned} \frac{\partial \boldsymbol{\mathcal{L}}_{\mathbf{H}}}{\partial \mathbf{H}} = \mu \sum_{m=1}^{M} &\Big(\mathbf{H}^{(m)} {\mathbf{H}^{(m)}}^T \mathbf{H} - \mathbf{H}^{(m)} - \Phi^{(m)} \\ &+ \rho(\mathbf{H} - \mathbf{H}^{(m)}) + \lambda \mathbf{L}^{(m)} \mathbf{H} \Big) = 0 \end{aligned} \tag{19}$$

By solving (19), we can get

$$\mathbf{H} = \mathbf{A}^{-1} \mathbf{B} \tag{20}$$

where

$$\begin{aligned} \mathbf{A} &= \sum_{m=1}^{M} \left(\mu \mathbf{H}^{(m)} {\mathbf{H}^{(m)}}^T + \lambda \mathbf{L}^{(m)} \right) + M\rho \mathbf{I} \\ \mathbf{B} &= (\rho + \mu) \sum_{i=1}^{M} \mathbf{H}^{(m)} + \sum_{i=1}^{M} \Phi^{(m)} \end{aligned} \tag{21}$$

Clearly, the optimization of $\mathbf{H}$ can be performed centrally with $\mathbf{H}^{(m)}$ from all clients.

(2) Solving $\mathbf{F}^{(m)}$ with fixed $\mathbf{H}^{(m)}$, $\mathbf{W}^{(m)}$, and $\mathbf{H}$ by $\boldsymbol{\mathcal{C}}_m$: Because the second term of (10) is independent of $\mathbf{F}^{(m)}$, we only focus on the first term and the function becomes:

$$\begin{aligned} &\min_{\mathbf{F}^{(m)}} \sum_{m=1}^{M} \alpha^{(m)} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &s.t. \ \mathbf{F}^{(m)} {\mathbf{F}^{(m)}}^T = \mathbf{I}, \ \sum_{i=1}^{M} \alpha^{(m)} = 1 \end{aligned} \tag{22}$$

By taking its partial derivative w.r.t. $\mathbf{F}^{(m)}$ and set it to 0:

$$\frac{\partial \sum_{m=1}^{M} \alpha^{(m)} ||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2}{\partial \mathbf{F}^{(m)}} = 0 \tag{23}$$

By simple algebra, we obtain its solution as follows:

$$\mathbf{F}^{(m)} = {\mathbf{Z}^{(m)}}^{-1} {\mathbf{W}^{(m)}}^T {\mathbf{H}^{(m)}}^T \mathbf{X}^{(m)} \tag{24}$$

where $\mathbf{Z}^{(\mathbf{m})} = {\mathbf{W}^{(m)}}^T {\mathbf{H}^{(m)}}^T \mathbf{H}^{(m)} \mathbf{W}^{(m)}$.

(3) Solving $\mathbf{W}^{(m)}$ with fixed $\mathbf{H}^{(m)}$ and $\mathbf{F}^{(m)}$, $\mathbf{H}$ by $\boldsymbol{\mathcal{C}}_m$: Similar to $\mathbf{F}^{(m)}$, by taking partial derivatives w.r.t. $\mathbf{W}^{(\mathbf{m})}$ and setting it to 0, we can obtain the solution of $\mathbf{W}^{(m)}$:

$$\mathbf{W}^{(m)} = ({\mathbf{H}^{(m)}}^T\mathbf{H}^{(m)})^{-1}{\mathbf{H}^{(m)}}^T\mathbf{X}^{(m)}{\mathbf{F}^{(m)}}^T(\mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T) \tag{25}$$

(4) Solving $\mathbf{H}^{(m)}$ with fixed $\mathbf{F}^{(m)}$, $\mathbf{W}^{(m)}$, and $\mathbf{H}$ by $\boldsymbol{\mathcal{C}}_m$: To solve $\mathbf{H}^{(m)}$, we follow the work [Smith *et al.*, 2018] and similar to the update of $\mathbf{H}$, we introduce a new term transformed from the constraint and Augmented Lagrangian function of (10) w.r.t. $\mathbf{H}^{(m)}$, then we get:

$$\begin{aligned}\boldsymbol{\mathcal{L}}_{\mathbf{H}^{(m)}} = \sum_{m=1}^{M}\Big(&\alpha^{(m)}||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &+ \frac{\mu}{2}||{\mathbf{H}^{(m)}}^T\mathbf{H} - \mathbf{I}||_F^2 + \left\langle \Phi^{(m)}, \mathbf{H}^{(m)} - \mathbf{H}\right\rangle \\ &+ \frac{\rho}{2}||\mathbf{H} - \mathbf{H}^{(m)}||_F^2\Big)\end{aligned} \tag{26}$$

If we have $\Delta\mathbf{H}^{(m)}$, then $\mathbf{H}^{(m)} \leftarrow \mathbf{H}^{(m)} + \Delta\mathbf{H}^{(m)}$.The problem is changed to solving $\Delta\mathbf{H}^{(m)}$ on $\boldsymbol{\mathcal{C}}_m$. We define the m-th sub-problem on m-th client to solve $\Delta\mathbf{H}^{(m)}$:

$$\begin{aligned}&\min_{\Delta\mathbf{H}^{(m)}} \boldsymbol{\mathcal{G}}_m^{\sigma_1,\sigma_2}(\Delta\mathbf{H}^{(m)}; \mathbf{H}^{(m)}, \mathbf{W}^{(m)}, \mathbf{F}^{(m)}, \mathbf{H}) \\ &= F_1(\Delta\mathbf{H}^{(m)}; \mathbf{H}^{(m)}, \mathbf{W}^{(m)}, \mathbf{F}^{(m)}) \\ &+ \frac{\mu}{2}F_2(\Delta\mathbf{H}^{(m)}; \mathbf{H}^{(m)}, \mathbf{H}) + g_m(\mathbf{H}^{(m)} + \Delta\mathbf{H}^{(m)})\end{aligned} \tag{27}$$

where

$$\begin{aligned}&F_1(\Delta\mathbf{H}^{(m)}; \mathbf{H}^{(m)}, \mathbf{W}^{(m)}, \mathbf{F}^{(m)}) \\ =&\frac{1}{M}\alpha^{(m)}||\mathbf{X}^{(m)} - \mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &-2\sum_{i=1}^{N}\alpha^{(m)}x_i^{(m)} - {\mathbf{F}^{(m)}}^T{\mathbf{W}^{(m)}}^T\Delta {h_i^{(m)}}^T \\ &+2\sum_{i=1}^{N}\alpha^{(m)}h_i^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}{\mathbf{F}^{(m)}}^T{\mathbf{W}^{(m)}}^T\Delta {h_i^{(m)}}^T \\ &+\frac{\sigma_1}{2}||\Delta\mathbf{H}^{(m)}\mathbf{W}^{(m)}\mathbf{F}^{(m)}||_F^2 \\ &F_2(\Delta\mathbf{H}^{(m)}; \mathbf{H}^{(m)}, \mathbf{H}) = \frac{1}{M}||{\mathbf{H}^{(m)}}^T\mathbf{H} - \mathbf{I}||_F^2 \\ &- 2\sum_{i=1}^{N}(\mathbf{H}_i^{(m)}\mathbf{H}^T - I_i)\mathbf{H}\Delta {h_i^{(m)}}^T + \frac{\sigma_2}{2}||{\Delta\mathbf{H}^{(m)}}^T\mathbf{H}||_F^2 \\ &g_m(\mathbf{H}^{(m)}) = \left\langle \Phi^{(m)}, \mathbf{H}^{(m)} - \mathbf{H}\right\rangle + \frac{\rho}{2}||\mathbf{H} - \mathbf{H}^{(m)}||_F^2\end{aligned} \tag{28}$$

By solving the above sub-problem locally according to [Smith *et al.*, 2018], we can update $\mathbf{H}^{(m)}$ on m-th client locally by $\mathbf{H}^{(m)} \leftarrow \mathbf{H}^{(m)} + \beta\Delta\mathbf{H}^{(m)}$ as long as the m-th client has parameter $\mathbf{H}$.

3.5 Communication Rounds

As shown in **Fig.** 1, FMVC-IMK requires transmitting some parameters to support collaborative and privacy-preserving model training. To better illustrate how FMVC-IMK works, we herein provide a brief introduction to its workflow.

In the initialization stage, each client $\boldsymbol{\mathcal{C}}_m$ locally computes the similarity matrix $S^{(m)}$ and generates the Laplacian matrix $\mathbf{L}^{(m)}$. Then, $\boldsymbol{\mathcal{C}}^{(m)}$ transmits $\mathbf{L}^{(m)}$ to server $\boldsymbol{\mathcal{S}}$, which aggregates all the $\mathbf{L}^{(m)}$ to obtain $\sum_{m=1}^{M}\mathbf{L}^{(m)}$. Considering that $\mathbf{L}^{(m)}$ may reveal local data distribution of $\boldsymbol{\mathcal{C}}_m$, we adopt partially homomorphic encryption(PHE) to ensure that $\boldsymbol{\mathcal{S}}$ can only obtain the aggregated result rather than the concrete $\mathbf{L}^{(m)}$. Since PHE achieves better efficiency and this process is only executed once at the beginning, it will not introduce much computation and communication overhead.

In the optimization stage, $\boldsymbol{\mathcal{C}}_m$ locally updates $\mathbf{F}^{(m)}$ and $\mathbf{W}^{(m)}$ with (24) and (25), respectively. Then, $\boldsymbol{\mathcal{C}}_m$ gets the $\Delta\mathbf{H}^{(m)}$ and updated $\alpha^{(m)}$ with (27) and(16) and updated $\mathbf{H}^{(m)} \leftarrow \mathbf{H}^{(m)} + \beta\Delta\mathbf{H}^{(m)}$. The updated $\mathbf{H}^{(m)}$ is transmitted to $\boldsymbol{\mathcal{S}}$, who subsequently leverage (20) to update $\mathbf{H}$ with all the $\mathbf{H}^{(m)}$ collected. This process repeats until the model converges. Finally, we summarize the workflow in **Alg.** 1.

Algorithm 1 FMVC-IMK

Input: The data $\mathbf{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, ..., \mathbf{X}^{(M)}\}$ in M local clients; the number of cluster k;
Penalty parameter μ, λ, ρ, θ
Output: Global cluster result $\mathbf{H}$

```
1:  Each client C_m initializes W^(m), F^(m), H^(m), and α^(m)=1/M;
2:  S aggregates L^(m) into Σ_{m=1}^M L^(m) via PHE;
3:  while not converged do
4:    for m = 1 to M do
5:                                        ▷ on m-th client C_m
6:      Update F^(m) according to (24)
7:      Update W^(m) according to (25)
8:      Get ΔH^(m) by solving (27), and update H^(m) by
        H^(m) ← H^(m) + βΔH^(m)
9:      Update α^(m) according to (16)
10:     Send H^(m) to S
11:   end for
12:                                       ▷ on the Server S
13:   Update H according to (20)
14:   Resend H to all clients
15: end while
16: return H
```

3.6 Complexity Analysis

The computational cost of our method consists of two parts: the client side and the global server side. Suppose N, E, C, and $d^{(m)}$ denote the sample number, iteration number, cluster number, and data dimension of m-th view, and assume $C \ll N$ and $C \ll d^{(m)}$.

For client $\boldsymbol{\mathcal{C}}_m$, the complexity of initialization, Laplacian matrix construction, and model update is $\boldsymbol{\mathcal{O}}\left(d^{(m)} + N\right)$, $\boldsymbol{\mathcal{O}}\left(N^2\right)$, and $\boldsymbol{\mathcal{O}}\left(N^2 d^{(m)} E\right)$;

For the global server $\boldsymbol{\mathcal{S}}$, the complexity of model aggregation process is $\boldsymbol{\mathcal{O}}\left(N^2 E\right)$.

Datasets	3-sources			BBCSport			ORL			Sonar		
Metrics	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
DiMSC	<u>70.81</u>	<u>63.81</u>	<u>76.13</u>	82.17	64.07	82.17	79.00	<u>91.34</u>	82.42	56.41	1.63	56.41
MvLRSSC	54.67	44.92	63.31	63.37	40.92	65.07	64.31	80.62	68.11	50.48	0.01	53.37
RMSL	34.91	14.43	42.60	76.63	72.36	76.63	<u>86.00</u>	**94.48**	<u>89.75</u>	50.48	1.76	53.37
GMC	69.23	62.16	74.56	80.70	**76.00**	79.43	63.25	85.71	71.50	50.48	4.50	53.37
MvDGNMF	66.27	48.77	70.41	<u>85.11</u>	70.07	<u>85.11</u>	71.50	84.23	76.75	63.94	6.00	63.94
UDBGL	34.91	5.60	35.50	36.40	2.43	36.58	59.25	77.36	62.50	57.21	1.61	57.21
FastMICE	55.62	50.25	71.01	43.93	11.16	45.40	78.75	90.46	82.25	58.17	3.23	58.17
FedMVL	56.21	45.88	68.05	62.13	42.28	71.14	51.50	69.47	56.75	<u>64.90</u>	<u>8.71</u>	<u>64.90</u>
FMVC-IMK	**78.70**	**70.50**	**84.62**	**90.26**	<u>74.95</u>	**90.26**	**93.25**	89.66	**93.05**	**74.52**	**18.03**	**74.52**

Table 1: Clustering performance comparison in terms of ACC(%), NMI(%), and PUR(%) on 3-sources, BBCSport, ORL and Sonar datasets.

Datasets	Yale			Vehicle Sensor			HAR			RGB-D		
Metrics	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
DiMSC	48.28	51.85	49.09	<u>76.06</u>	<u>29.47</u>	<u>76.06</u>	51.79	32.14	25.69	40.72	32.57	50.10
MvLRSSC	45.85	50.16	46.97	56.78	6.12	56.78	49.38	53.56	53.40	<u>43.95</u>	<u>37.29</u>	43.29
RMSL	<u>67.27</u>	<u>74.02</u>	<u>68.48</u>	68.07	12.34	68.07	48.64	52.99	55.38	13.80	3.06	26.43
GMC	54.55	62.44	54.55	64.68	19.55	64.68	48.04	<u>57.40</u>	48.60	40.23	33.06	46.51
MvDGNMF	47.27	52.24	50.91	52.63	0.20	52.63	46.36	35.21	46.36	26.57	0.78	26.98
UDBGL	52.73	65.94	54.55	51.26	0.05	51.26	47.78	46.20	50.45	43.89	35.96	<u>53.55</u>
FastMICE	62.42	57.01	65.46	51.49	0.09	51.69	<u>56.79</u>	49.58	<u>56.79</u>	41.81	32.61	49.53
FedMVL	46.67	51.50	47.27	74.03	17.39	74.03	53.68	54.70	43.71	32.51	23.65	45.89
FMVC-IMK	**78.79**	**77.90**	**79.39**	**83.06**	**33.24**	**83.06**	**69.36**	**59.42**	**69.36**	**46.42**	39.85	**58.04**

Table 2: Clustering performance comparison in terms of ACC(%), NMI(%), and PUR(%) on Yale, Vehicle Sensor, HAR, and RGB-D datasets.

4 Experiment

4.1 Experiment Settings

We compare our method with eight multi-view clustering methods on eight multi-view datasets. For the federating settings, our experiment includes a server and multiple clients, and each client holds the data with one view.

Datasets: We evaluate our method on eight public multi-view datasets: (1)**3-sources** is a three-view text dataset sourced from three reputable news outlets: BBC, Reuters, and the Guardian with 169 samples. (2)**BBCSport** [Greene and Cunningham, 2006] is a two-view dataset consisting of 544 samples of five categories sourced from BBC Sport; (3)**ORL** [Samaria and Harter, 1994] is a three-view dataset of 400 facial images, categorized into 40 classes. (4)**Sonar** [Sejnowski and Gorman,] includes three views and extracts its multi-view features from 208 patterns(samples). Then the 60 features are divided into three views equally. (5)**Yale** is a two-view dataset of 165 facial images of 11 people. (6)**Vehicle Sensor** [Duarte and Hu, 2004] is a four-view dataset whose features are gathered from distributed sensors. (7)**Human Activity Recognition(HAR)** [Reyes-Ortiz and Parra, 2012] is a four-view dataset with 10299 samples that documents six daily activities; (8)**SentencesNYU v2(RGB-D)** [Silberman *et al.*, 2012] includes images and descriptionsof indoor scenes . We process this dataset following [Trosten *et al.*, 2021].

Compared Methods: We compared FMVC-IMK with: (1)**DiMSC** [Cao *et al.*, 2015];(2)**MvLRSSC** [Brbić and Kopriva, 2018];(3)**RMSL** [Li *et al.*, 2019];(4)**GMC** [Wang *et al.*, 2019];(5)**MvDGNMF** [Li *et al.*, 2020];(6)**UDBGL**[Fang *et al.*, 2023];(7)**FastMICE**[Huang *et al.*, 2023] (8)**FedMVL** [Huang *et al.*, 2022].(1)-(7) are centralized methods, and (8) is federated method.

4.2 Experiment Results and Analysis

Table 1 and **Table** 2 illustrate the experimental results, from which we can observe that our method achieves better clustering than FedMVL because FMVC-IMK integrates NMF and K-means into a single step and leverages a graph-based regularizer to retain geometric structure information of the original data. Even compared with centralized multi-view clustering methods, our method shows comparative performance and achieves the best performance on most datasets. Especially, on the Sonar dataset, FMVC-IMK obtains 74.52% ACC, 18.03% NMI, and 74.52% PUR, achieving 9.62%, 9.32%, and 9.62% higher than the sub-optimal method. This demonstrates the superiority of FMVC-IMK.

Convergence analysis: We record the value of the objective function at each iteration on four datasets to verify the convergence property, as shown in **Fig.** 2. It can be seen that the objective value decreases rapidly and converges within 100 iterations on all four datasets. ACC increases rapidly, but the whole process fluctuates. The reason is that the global

Variants	TS-FMVC-IMK w/o G-R			FMVC-IMK w/o G-R			FMVC-IMK		
Dataset	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
3-sources	56.21	45.88	68.05	73.96	66.52	81.07	78.70	70.50	84.62
BBCSport	62.13	42.28	71.14	88.42	73.86	88.48	90.26	74.95	90.26
ORL	51.50	69.47	56.75	91.23	87.63	93.26	93.25	89.66	93.05
Sonar	64.90	8.71	64.90	72.60	15.08	72.60	74.52	18.03	18.03
Yale	46.67	51.50	47.27	74.33	75.49	75.15	78.79	77.90	79.39
Vehicle Sensor	74.03	17.39	74.03	81.43	32.33	81.43	83.06	33.24	83.06
HAR	53.68	54.70	43.71	68.56	57.49	68.56	69.36	59.42	69.36
RGB-D	32.51	23.65	45.89	45.62	38.14	56.45	46.42	39.85	58.04

Table 3: Results of ablation studies on eight multi-view datasets.

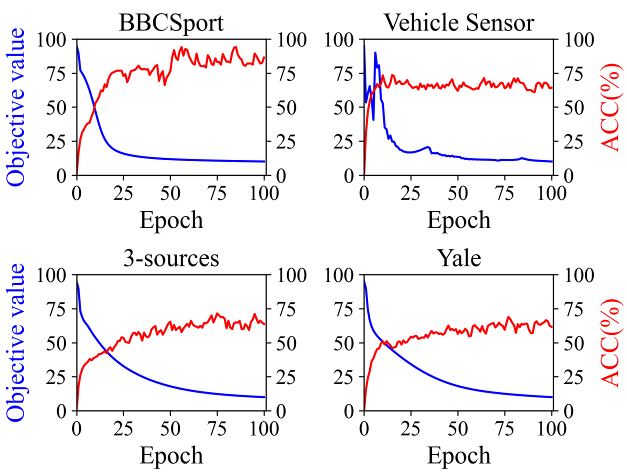

Figure 2: The convergence curves of FMVC-IMK on BBCSport, Vehicle Sensor, 3-sources, and Yale.

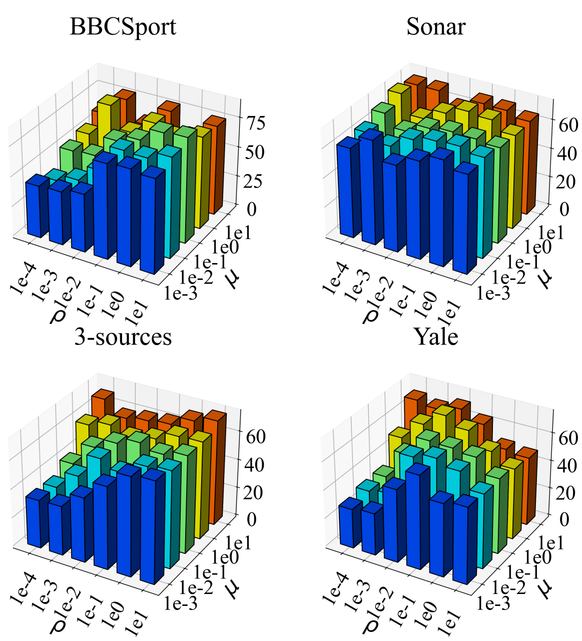

Figure 3: ACC w.r.t. λ and μ on 3-sources, RGBD, Sonar, and Yale.

cluster assignment is obtained by aggregating all local cluster assignments in each iteration, which inevitably affects ACC. Nevertheless, ACC still converges within 100 iterations.

Parameter Analysis: (10) indicates that our objective function involves two hyperparameters: λ and μ, and **Fig.**3 depicts the ACC when λ and μ take values on the interval of $[0.0001, 0.001, 0.01, 0.1, 1, 10]$ on four datasets. We can observe that: (1) When λ is too small, the accuracy is low because the local geometric structure is not well retained. However, when λ is too large, the accuracy is low due to the excessive influence of the local geometric structure; (2) Smaller μ reduces accuracy because it hinders learning a consistent cluster assignment, but bigger μ also reduces accuracy because local data heterogeneity is ignored. Proper values of λ and μ help improve the cluster performance of FMVC-IMK.

Ablation Experiments : We conduct the ablation studies and summarize them in **Table** 3. We test the performance of FMVC-IMK in three cases: (1)two-step FMVC-IMK without Graph Regularizer(***case 1***); (2)FMVC-IMK without Graph Regularizer(***case 2***); (3)the complete FMVC-IMK (***case 3***). From **Table** 3, the ACC, NMI, and PUR on BBCSport in ***case 2*** outperforms that in ***case 1*** by 26.29%, 31.58%, and 17.34%, which means that integrating NMF and K-means effectively improves the clustering performance. Besides, on the same dataset, the three metrics in ***case 3*** are increased by 4.74%, 3.98%, and 3.55% when compared with ***case 2***, indicating the graph regularizer helps to improve the clustering performance by enforcing the clustering to be consistent with the original data.

5 Conclusion

The paper presents a novel federated multi-view clustering method named FMVC-IMK to solve the multi-view clustering problem in the federated setting. By integrating matrix factorization and K-means clustering into a single step and introducing graph-based regularization, FMVC-IMK enhances clustering performance and preserves data privacy simultaneously. Additionally, we introduce an adaptive weight for all clients and establish the update strategy. Furthermore, we design a collaborative optimization algorithm to facilitate the application of our method in federated scenarios. Extensive experiments demonstrate the superiority of FMVC-IMK.

Acknowledgements

This research was partially supported by the National Key Research and Development Project of China No. 2021ZD011070, National Science Foundation of China under Grant No. 62102306, Shaanxi Continuing Higher Education Teaching Reform Research Project No. 21XJZ014, the Natural Science Basic Research Program of Shaanxi Province under Grant No. 2023-JC-YB-534, the Science and technology project of Xian under Grant No. 2022JH-JSYF-0009, Initiative Postdocs Supporting Program under Grant No. BX20190262.

References

[Al-Stouhi and Reddy, 2014] Samir Al-Stouhi and Chandan K Reddy. Multi-task clustering using constrained symmetric non-negative matrix factorization. In *Proceedings of the 2014 SIAM international conference on data mining*, pages 785–793. SIAM, 2014.

[Bauckhage, 2015] Christian Bauckhage. K-means clustering is matrix factorization. *arXiv preprint arXiv:1512.07548*, 2015.

[Brbić and Kopriva, 2018] Maria Brbić and Ivica Kopriva. Multi-view low-rank sparse subspace clustering. *Pattern Recognition*, 73:247–258, 2018.

[Cai *et al.*, 2013] Xiao Cai, Feiping Nie, and Heng Huang. Multi-view k-means clustering on big data. In *Twenty-Third International Joint conference on artificial intelligence*, 2013.

[Cao *et al.*, 2015] Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. Diversity-induced multi-view subspace clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–594, 2015.

[Che *et al.*, 2022] Sicong Che, Zhaoming Kong, Hao Peng, Lichao Sun, Alex Leow, Yong Chen, and Lifang He. Federated multi-view learning for private medical data integration and analysis. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(4):1–23, 2022.

[Chen *et al.*, 2023] Xinyue Chen, Jie Xu, Yazhou Ren, Xiaorong Pu, Ce Zhu, Xiaofeng Zhu, Zhifeng Hao, and Lifang He. Federated deep multi-view clustering with global self-supervision. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3498–3506, 2023.

[Duarte and Hu, 2004] Marco F Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2004.

[Fang *et al.*, 2023] Si-Guo Fang, Dong Huang, Xiao-Sha Cai, Chang-Dong Wang, Chaobo He, and Yong Tang. Efficient multi-view clustering via unified and discrete bipartite graph learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[Feng and Yu, 2020] Siwei Feng and Han Yu. Multi-participant multi-class vertical federated learning. *arXiv preprint arXiv:2001.11154*, 2020.

[Gao *et al.*, 2019a] Dashan Gao, Ce Ju, Xiguang Wei, Yang Liu, Tianjian Chen, and Qiang Yang. Hhhfl: Hierarchical heterogeneous horizontal federated learning for electroencephalography. *arXiv preprint arXiv:1909.05784*, 2019.

[Gao *et al.*, 2019b] Shengxiang Gao, Zhengtao Yu, Taisong Jin, and Ming Yin. Multi-view low-rank matrix factorization using multiple manifold regularization. *Neurocomputing*, 335:143–152, 2019.

[Greene and Cunningham, 2006] Derek Greene and Pádraig Cunningham. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd international conference on Machine learning*, pages 377–384, 2006.

[Hu *et al.*, 2023] Xingchen Hu, Jindong Qin, Yinghua Shen, Witold Pedrycz, Xinwang Liu, and Jiyuan Liu. An efficient federated multi-view fuzzy c-means clustering method. *IEEE Transactions on Fuzzy Systems*, 2023.

[Huang *et al.*, 2019] Shudong Huang, Zhao Kang, Ivor W Tsang, and Zenglin Xu. Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition*, 88:174–184, 2019.

[Huang *et al.*, 2020] Mingkai Huang, Hao Li, Bing Bai, Chang Wang, Kun Bai, and Fei Wang. A federated multi-view deep learning framework for privacy-preserving recommendations. *arXiv preprint arXiv:2008.10808*, 2020.

[Huang *et al.*, 2021] Shudong Huang, Ivor W Tsang, Zenglin Xu, and Jiancheng Lv. Measuring diversity in graph learning: A unified framework for structured multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5869–5883, 2021.

[Huang *et al.*, 2022] Shudong Huang, Wei Shi, Zenglin Xu, Ivor W Tsang, and Jiancheng Lv. Efficient federated multi-view learning. *Pattern Recognition*, 131:108817, 2022.

[Huang *et al.*, 2023] Dong Huang, Chang-Dong Wang, and Jian-Huang Lai. Fast multi-view clustering via ensembles: Towards scalability, superiority, and simplicity. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[Jiang *et al.*, 2013] Yu Jiang, Jing Liu, Zechao Li, Peng Li, and Hanqing Lu. Co-regularized plsa for multi-view clustering. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part II 11*, pages 202–213. Springer, 2013.

[Kevin *et al.*, 2021] I Kevin, Kai Wang, Xiaokang Zhou, Wei Liang, Zheng Yan, and Jinhua She. Federated transfer learning based cross-domain prediction for smart manufacturing. *IEEE Transactions on Industrial Informatics*, 18(6):4088–4096, 2021.

[Li *et al.*, 2019] Ruihuang Li, Changqing Zhang, Huazhu Fu, Xi Peng, Tianyi Zhou, and Qinghua Hu. Reciprocal multi-layer subspace learning for multi-view clustering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8172–8180, 2019.

[Li *et al.*, 2020] Jianqiang Li, Guoxu Zhou, Yuning Qiu, Yanjiao Wang, Yu Zhang, and Shengli Xie. Deep graph regularized non-negative matrix factorization for multi-view clustering. *Neurocomputing*, 390:108–116, 2020.

[Liang *et al.*, 2020] Naiyao Liang, Zuyuan Yang, Zhenni Li, Weijun Sun, and Shengli Xie. Multi-view clustering by non-negative matrix factorization with co-orthogonal constraints. *Knowledge-Based Systems*, 194:105582, 2020.

[Liu *et al.*, 2013] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 SIAM international conference on data mining*, pages 252–260. SIAM, 2013.

[Liu *et al.*, 2020] Yang Liu, Yan Kang, Chaoping Xing, Tianjian Chen, and Qiang Yang. A secure federated transfer learning framework. *IEEE Intelligent Systems*, 35(4):70–82, 2020.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[Ren *et al.*, 2024] Yazhou Ren, Xinyue Chen, Jie Xu, Jingyu Pu, Yonghao Huang, Xiaorong Pu, Ce Zhu, Xiaofeng Zhu, Zhifeng Hao, and Lifang He. A novel federated multi-view clustering method for unaligned and incomplete data fusion. *Information Fusion*, 108:102357, 2024.

[Reyes-Ortiz and Parra, 2012] Anguita Davide Ghio Alessandro Oneto Luca Reyes-Ortiz, Jorge and Xavier Parra. Human Activity Recognition Using Smartphones. UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C54S4K.

[Samaria and Harter, 1994] Ferdinando S Samaria and Andy C Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of 1994 IEEE workshop on applications of computer vision*, pages 138–142. IEEE, 1994.

[Sejnowski and Gorman,] Terry Sejnowski and R. Gorman. Connectionist Bench (Sonar, Mines vs. Rocks). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5T01Q.

[Silberman *et al.*, 2012] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012.

[Smith *et al.*, 2018] Virginia Smith, Simone Forte, Chenxin Ma, Martin Takáč, Michael I Jordan, and Martin Jaggi. Cocoa: A general framework for communication-efficient distributed optimization. *Journal of Machine Learning Research*, 18(230):1–49, 2018.

[Sun *et al.*, 2020] Gan Sun, Yang Cong, Yulun Zhang, Guoshuai Zhao, and Yun Fu. Continual multiview task learning via deep matrix factorization. *IEEE transactions on neural networks and learning systems*, 32(1):139–150, 2020.

[Sun *et al.*, 2021] Jiankai Sun, Xin Yang, Yuanshun Yao, Aonan Zhang, Weihao Gao, Junyuan Xie, and Chong Wang. Vertical federated learning without revealing intersection membership. *arXiv preprint arXiv:2106.05508*, 2021.

[Trosten *et al.*, 2021] Daniel J Trosten, Sigurd Lokse, Robert Jenssen, and Michael Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1255–1265, 2021.

[Tzortzis and Likas, 2012] Grigorios Tzortzis and Aristidis Likas. Kernel-based weighted multi-view clustering. In *2012 IEEE 12th international conference on data mining*, pages 675–684. IEEE, 2012.

[Wang *et al.*, 2019] Hao Wang, Yan Yang, and Bing Liu. Gmc: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6):1116–1129, 2019.

[Wang *et al.*, 2023] Xu Wang, Dezhong Peng, Ming Yan, and Peng Hu. Correspondence-free domain alignment for unsupervised cross-domain image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10200–10208, 2023.

[Wen *et al.*, 2020] Jie Wen, Ke Yan, Zheng Zhang, Yong Xu, Junqian Wang, Lunke Fei, and Bob Zhang. Adaptive graph completion based incomplete multi-view clustering. *IEEE Transactions on Multimedia*, 23:2493–2504, 2020.

[Xu *et al.*, 2013] Chang Xu, Dacheng Tao, and Chao Xu. A survey on multi-view learning. *arXiv preprint arXiv:1304.5634*, 2013.

[Yang and Wang, 2018] Yan Yang and Hao Wang. Multi-view clustering: A survey. *Big Data Mining and Analytics*, 1(2):83–107, 2018.

[Yang *et al.*, 2020] Hongwei Yang, Hui He, Weizhe Zhang, and Xiaochun Cao. Fedsteg: A federated transfer learning framework for secure image steganalysis. *IEEE Transactions on Network Science and Engineering*, 8(2):1084–1094, 2020.

[Yang *et al.*, 2022] Ben Yang, Xuetao Zhang, Zhiping Lin, Feiping Nie, Badong Chen, and Fei Wang. Efficient and robust multiview clustering with anchor graph regularization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):6200–6213, 2022.

[Zhao *et al.*, 2021] Jie Zhao, Xinghua Zhu, Jianzong Wang, and Jing Xiao. Efficient client contribution evaluation for horizontal federated learning. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3060–3064. IEEE, 2021.

[Zheng *et al.*, 2023] Xiao Zheng, Chang Tang, Xinwang Liu, and En Zhu. Multi-view clustering via matrix factorization assisted k-means. *Neurocomputing*, 534:45–54, 2023.

[Zhu *et al.*, 2020] Xiaofei Zhu, Jiafeng Guo, Wolfgang Nejdl, Xiangwen Liao, and Stefan Dietze. Multi-view image clustering based on sparse coding and manifold consensus. *Neurocomputing*, 403:53–62, 2020.