THE EQUIVALENCE OF MULTI-VIEW CLUSTERING VIA LATENT CO-ASSOCIATION MATRIX

Anonymous authors
Paper under double-blind review

ABSTRACT

Studying the relationship between multi-view clustering and ensemble clustering is of great practical and theoretical significance in achieving the equivalence of multi-view clustering. The problem of ensemble clustering is usually tackled by generating multiple base clusterings and then combining them to achieve the clustering result. Most existing ensemble clustering methods either linearly combine connective matrices to construct the co-association matrix or refine the co-association matrix with the built coherent-link matrix. However, directly casting ensemble clustering as a multi-view clustering problem becomes challenging. In this paper, we show that the equivalence of multi-view clustering and ensemble clustering can be established based on the formulation of objective function. To be specific, we have multiple co-association matrices by defining δ-sensitive feature descriptor and introduce a latent co-association matrix shared by different co-association matrices. The δ-sensitive feature descriptor can refine the original co-association matrix in a general manner and alleviate the impacts brought by poor base clusterings. We then define the δ-sensitive margin and prove that δ-sensitive feature descriptor generates the largest δ-sensitive margin, which helps lead to a discriminative co-association representation. These multiple co-association representations are collected into a tensor and the low-rank constraint is used to better explore the global relationship among entries in the tensor. Then the relationship between multi-view clustering and ensemble clustering is built in this way. Extensive experiments on several data sets validate the above analysis in terms of different metrics.

1 INTRODUCTION

Clustering is a basic but important task in the area of machine learning and data mining Jain (2010); Qin et al. (2022b). It aims to find the inherent structure of the data set and partition a set of data points into a certain number of groups. In the past decades, various algorithms for clustering have been presented by employing different techniques Johnson (1967); Qin et al. (2023); Moore (1998); von Luxburg (2007); Qin et al. (2022a), *i.e.*, spectral clustering von Luxburg (2007), Gaussian mixture models Moore (1998), hierarchical clustering Johnson (1967). Each clustering method owns its advantages and drawbacks, and may not perform well for all applications. Given a data set, even the same algorithm for clustering with different parameters or initializations may produce different clustering results. For a clustering task, it is hard to choose a proper algorithm as well as select appropriate parameters for the corresponding algorithm.

Since clusterings obtained by different algorithms characterize the data from different perspectives, the ensemble clustering techniques were presented by exploring the complementary information from different clusterings Tao et al. (2017). Compared with supervised ensemble learning, clustering ensemble appears more hard since the existing strategies for supervised ensemble learning are unable to be applied to clustering ensemble directly Tao et al. (2017). Ensemble clustering aims to achieve a more robust and better clustering result by combining different clusterings Kilmer et al. (2013). Each input clustering in ensemble clustering is termed the base clustering and the final clustering result is called the consensus clustering. With the base clusterings, the existing approaches generally learn a pairwise relationship matrix and then achieve the final clustering result by applying off-the-shelf clustering approaches to the obtained matrix. For base clustering, its quality plays an important

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

role in the consensus process. The base clusterings with low quality may influence the consensus results.

According to the way of generating the pairwise relationship matrix, the existing methods can be roughly divided into two categories. The first category aims to achieve the co-association matrix by summarizing the co-occurrence of data points belonging to the same cluster of the base clusterings. In recent years, various advanced methods for co-association matrix construction have been presented Huang et al. (2018; 2021); Zhou et al. (2021). Huang *et al.* presented a co-association matrix which is locally weighted and taken the uncertainty of each base clustering into account Huang et al. (2018). Zhou *et al.* learned the co-association matrix by proposing a self-paced strategy Zhou et al. (2021). Huang *et al.* enhanced the co-association matrix by employing the cluster-wise similarities Huang et al. (2021). Jia *et al.* presented a novel method based on the low-rank tensor approximation to seek for the solution of the formulated problem from a global perspective Jia et al. (2021). The second category regards the base clusterings as new representations for learning a pairwise relationship matrix. Gao *et al.* formulated the ensemble clustering as a problem of convex low-rank matrix representation Gao et al. (2016). Zhou *et al.* aimed at seeking for a dense affinity matrix by adopting a self-representation model regularized by the Frobenius norm Zhou et al. (2019). For the related work of ensemble clustering, we give the detailed presentation in the next section.

Motivated by the fact the relationship between multi-view clustering and ensemble clustering has never been studied. It is challenging to directly cast ensemble clustering as a multi-view clustering problem. To be specific, co-association matrices in the existing work for ensemble clustering are achieved by combining connective matrices from multiple base clusterings in a weighted linear manner. The quality of the co-association matrix obtained by linearly combining the connective matrices or its variants is easily affected by the poor performance of the adopted base clusterings, leading to the unexpected performance for clustering. The existing works also fail to extend the refinement of the co-association matrix to a more general case by employing the coherent-link, which can be regarded as the specific feature extracted from the co-association matrix.

To address the above limitations, in this article, we show that ensemble clustering is equivalent to multi-view clustering based on the formulation of objective function. We propose a novel ensemble clustering method to achieve a latent co-association matrix shared by multiple co-association matrices based on the introduced δ-sensitive feature descriptor. These co-association matrices can be regarded as co-association representations from different views. We define the δ-sensitive margin and assume that large δ-sensitive margin helps produce a discriminative feature representation for the co-association matrix. We also prove that δ-sensitive feature descriptor generates the largest δ-sensitive margin, which helps extract discriminative representations from the originally obtained co-association matrix. Then a latent co-association matrix can be obtained by combining co-association representations from different views. The latent co-association matrix is comprehensive since it incorporates multiple co-association matrices, leading to the expected clustering performance in ensemble clustering. As Xie et al. (2018) in multi-view clustering, we stack these multi-view co-association representations to form a tensor and globally explore relationship among entries in a tensor with the low-rank norm. Besides, the clustering result can be obtained by applying the off-the-shelf method for clustering to the latent co-association. Then the equivalence between ensemble clustering and multi-view clustering can be built based on the formulation of objective function. To the best of our knowledge, this is the first to study the relationship between multi-view clustering and ensemble clustering. We are also the first to introduce δ-sensitive feature descriptor to obtain multiple co-association matrices as different views and present the latent co-association matrix in ensemble clustering. The major contributions in this work are:

1. We establish the equivalence between ensemble clustering and multi-view clustering based on the formulation of objective function. We extend the traditional construction of the co-association matrix in ensemble clustering to multi-view cases by extracting multiple co-association representations as different views based on δ-sensitive feature descriptor.

2. We define δ-sensitive margin and prove that δ-sensitive feature descriptor generates the largest δ-sensitive margin, which leads to a discriminative co-association representation.

3. We construct a specific example under the proposed novel ensemble clustering framework. Specifically, we stack co-association representations from different views to form a tensor with three dimensions and the tensor low-rank norm is flexibly applied to this constructed

tensor for investigating the relationship among different entries in the tensor, leading to an expected latent co-association matrix.

4. We formulate the proposed method as a constrained optimization problem and seek for the solution with an alternative iterative method. Extensive experiments performed on different data sets validate that better clustering performance can be achieved compared with the representative approaches.

2 EQUIVALENCE OF MULTI-VIEW CLUSTERING AND ENSEMBLE CLUSTERING

Given the co-association matrix A based on the the m-th connective matrix $A_m \in R^{N\times N}$, which is achieved by $\frac{1}{M}\sum_{m=1}^{M} A_m(i,j)$, we aim to extract multiple feature representations A_1, A_2,..., A_V from A with different strategies, where V is the total number of feature representations and M denotes the number of clusterings. Inspired by the merits of multi-view clustering Li et al. (2019), the extracted feature representations A_1, A_2,..., A_V can be regarded as co-association representations from different views and the multi-view learning can be adopted for ensemble clustering. Note that the original co-association matrix A is also employed as the representation in a view since it is able to describe the co-association information losslessly even with possible noise.

According to Wu et al. (2019) in multi-view learning, a tensor built by representations from different views can effectively describe the multi-view data set. We relate the ensemble clustering with multi-view clustering by employing the tensor to characterize co-association matrices from different views. $\Omega(.)$ is defined by stacking A, A_1, A_2,..., A_V into a co-association tensor $\mathcal{A}$, which is formulated as

$$\mathcal{A} = \Omega(A, A_1, A_2, ..., A_V). \tag{1}$$

For (i,j)-th entry in the co-association matrix A, we have

$$A(i,j) = \frac{1}{M}\sum_{m=1}^{M} A_m(i,j) \in [0,1]\,. \tag{2}$$

To achieve the feature representation A_v for the v-th view, we set $A_v(i,j)$ to be 0 or 1 with the guidance of the threshold δ_v, which is formulated as:

$$A_v(i,j) = \begin{cases} 1, & \text{if } A(i,j) \geq \delta_v, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where $\delta_v \in [0,1]$. Then, we can extract different feature representations A_1, A_2,..., A_V for the co-association matrix A based on thresholds $\delta_1, \delta_2, ..., \delta_V$ in this manner. Since $A_v(i,j)$ is influenced by the threshold δ_v, we call Eq. (3) as δ-sensitive feature descriptor. Obviously, a large δ_v is able to produce a desired co-association representation, which results in the satisfied clustering result. Next, we define the δ-sensitive margin and show that δ-sensitive feature descriptor can help lead to a discriminative representation of the co-association matrix A. We measure the relaxtion between $A_v(i,j)$ and $A(i,j)$, which is denoted by δ-sensitive margin $m = |A_v(i,j) - A(i,j)|$. In this work, we assume that a large δ-sensitive margin helps result in a discriminative feature representation for the co-association matrix A since large diversity among different representations is needed for improving the final performance from the perspective of ensemble learning. We also have the following observation:

Theorem 1. *If the feature representation A_v is extracted by δ-sensitive feature descriptor, then the largest δ-sensitive margin between $A_v(i,j)$ and $A(i,j)$ for the (i,j)-th entry is obtained.*

Proof: When $A_v(i,j) \geq \delta_v$, we have

$$A_v(i,j) \geq A(i,j). \tag{4}$$

We can achieve the δ-sensitive margin $m_1 = A_v(i,j) - A(i,j)$. When $A_v(i,j) < \delta_v$, we have

$$A_v(i,j) \leq A(i,j). \tag{5}$$

Likewise, the δ-sensitive margin $m_2 = A(i,j) - A_v(i,j)$ is obtained in this case. According to the definition of A_v, the maximal and minimal value of $A_v(i,j)$ is 1 and 0, respectively. If δ-sensitive feature descriptor $A'_v(i,j)$ is employed, which is formulated as:

$$A'_v(i,j) = \begin{cases} \xi_1, & \text{if } A(i,j) \geq \delta_v, \\ \xi_2, & \text{otherwise,} \end{cases} \tag{6}$$

where $0 < \xi_2 \leq \xi_1 < 1$. We can also obtain the δ-sensitive margin $m_1' = |\xi_1 - A(i,j)|$ and $m_2' = |\xi_2 - A(i,j)|$ for cases of $A_v'(i,j) \geq \delta_v$ and $A_v'(i,j) < \delta_v$, respectively. Due to $m_1 \geq m_1'$ and $m_2 \geq m_2'$, we can draw the conclusion that δ-sensitive feature descriptor formulated in Eq. (3) can guarantee the largest δ-sensitive margin. The proof ends here.

With the obtained representations for co-association matrix $A, A_1, A_2, ..., A_V$ based on δ-sensitive feature descriptor, we collect them into a tensor $\mathcal{A}$ as in Eq. (1). Since the tensor low-rank norm is able to describe the relationship among different entries in a tensor from the global perspective, we impose the tensor low-rank norm on the tensor $\mathcal{A}$. Due to the non-smooth and discrete properties, rank(.) is hard to solve. Then, we relax it by its convex hull, *i.e.*, the nuclear norm $||.||_*$. Based on $\mathcal{A}$, we attempt to remove the existing incorrect connections depicted by $\mathcal{E}$, resulting in a refined co-association tensor $\mathcal{Z}$. The procedure is formulated as follows:

$$\min_{\mathcal{Z}} ||\mathcal{Z}||_* + \lambda ||\mathcal{E}||_F^2 \;\; s.t. \;\; \mathcal{A} = \Omega(A, A_1, ..., A_V), \; \mathcal{A} = \mathcal{Z} + \mathcal{E}, \tag{7}$$

where $\lambda > 0$ is the parameter for balancing different terms and $||.||_F^2$ is the Frobenius norm of a tensor. After solving Eq. (7), we can achieve the final co-association matrix by averaging $\mathcal{Z}$ along the frontal direction. Then, the obtained co-association matrix can be regarded as the latent co-association matrix A_H and we can achieve the clustering result by performing spectral clustering on A_H.

Since the objective function in Eq. (7) for ensemble clustering is consistent with multi-view clustering in Xie et al. (2018), the relation between ensemble clustering and multi-view clustering can be built in this manner. We establish the equivalence between ensemble clustering and multi-view clustering based on the formulation of objective function. As aforementioned, the original co-association matrix A is able to characterize the co-association information losslessly even with noise. Then, the original co-association matrix A is also used as a feature representation from one view.

In this work, we extract different feature representations $A_1, A_2, ..., A_V$ for the co-association matrix A. We then collect the original co-association matrix A, the extracted $A_1, A_2, ..., A_V$ into a co-association tensor $\mathcal{A} \in \mathcal{R}^{N \times N \times V}$ with $\mathcal{A}(:,:,1) = A_1$, $\mathcal{A}(:,:,2) = A_2$, ..., and $\mathcal{A}(:,:,V) = A$. Since entries in the co-association tensor $\mathcal{A}$ consist of many error connections, we use the error tensor depicted by $\mathcal{E} \in \mathcal{R}^{N \times N \times V}$ to refine $\mathcal{A}$. Then, we divide $\mathcal{A}$ into two different parts: a refined co-association tensor $\mathcal{Z}$ and the error tensor $\mathcal{E}$, *i.e.*, $\mathcal{A} = \mathcal{Z} + \mathcal{E}$. Ideally, the refined co-association tensor $\mathcal{Z}$ is expected to be low-rank. The entries in $\mathcal{Z}$ should be in the range of $[0, 1]$. In addition, each frontal slice of the refined co-association tensor $\mathcal{Z}$ is expected to be symmetric. With these constraints, we mathematically formulate the proposed method as a problem of constrained optimization as follows:

$$\begin{aligned}
&\min_{\mathcal{Z},\mathcal{E}} ||\mathcal{Z}||_* + \lambda ||\mathcal{E}||_F^2 \; s.t. \; \mathcal{A} = \mathcal{Z} + \mathcal{E}, \\
&\quad \mathcal{Z}(i,j,1) = A_1(i,j), \\
&\qquad\qquad \vdots \\
&\quad \mathcal{Z}(i,j,V) = A_V(i,j), \\
&\quad \mathcal{Z}(:,:,1) = \mathcal{Z}(:,:,1)^T, \; \mathcal{Z}(i,j,1) \in [0,1], \; \forall i,j, \\
&\qquad\qquad \vdots \\
&\quad \mathcal{Z}(:,:,V) = \mathcal{Z}(:,:,V)^T, \; \mathcal{Z}(i,j,V) \in [0,1], \; \forall i,j,
\end{aligned} \tag{8}$$

where $||.||_F^2$ is imposed on $\mathcal{E}$ and $\lambda > 0$ is the parameter for balancing the error term. Then, we can achieve the optimal solution $\mathcal{Z}^*$ by solving the problem in Eq. (8) and the latent co-association matrix A_H can be obtained by averaging $\mathcal{Z}^*$ along the frontal direction. Based on A_H, we obtain the final clustering result by applying the off-the-shelf clustering method, *i.e.*, spectral clustering Ng et al. (2001).

3 Specific Example

Within the general framework of proposed novel ensemble clustering, we provide a specific example where the number of V is given, *i.e.*, $V = 3$. We set $\mathcal{Z}(i,j,1) = A_1(i,j)$ with $A_1(i,j) = 1$ and

$\mathcal{Z}(i,j,2) = A_2(i,j)$ with $A_2(i,j) = 0$, respectively. Then the above optimization problem is formulated as follows:

$$\begin{aligned}
&\min_{\mathcal{Z},\mathcal{E}} ||\mathcal{Z}||_* + \lambda||\mathcal{E}||_F^2 \; s.t. \; \mathcal{A} = \mathcal{Z} + \mathcal{E}, \; \mathcal{Z}(i,j,1) = A_1(i,j), \text{ if } A_1(i,j) = 1, \\
&\mathcal{Z}(i,j,2) = A_2(i,j), \text{ if } A_2(i,j) = 0, \; \mathcal{Z}(:,:,1) = \mathcal{Z}(:,:,1)^T, \; \mathcal{Z}(i,j,1) \in [0,1], \\
&\mathcal{Z}(:,:,2) = \mathcal{Z}(:,:,2)^T, \; \mathcal{Z}(i,j,2) \in [0,1], \; \mathcal{Z}(:,:,3) = \mathcal{Z}(:,:,3)^T, \; \mathcal{Z}(i,j,3) \in [0,1], \; \forall i,j.
\end{aligned} \tag{9}$$

We adopt the inexact augmented Lagrangian multiplier Jia et al. (2018) to solve Eq. (9). Specifically, three auxiliary matrices $H, L, J \in R^{N\times N}$ are introduced to handle the symmetric and bounded constraints on $\mathcal{Z}(:,:,1)$, $\mathcal{Z}(:,:,2)$ and $\mathcal{Z}(:,:,3)$, respectively. To deal with the equality constraints, we introduce four Lagrangian multipliers $\Theta_1, \Theta_2, \Theta_3 \in R^{N\times N}$ and $\Theta_4 \in \mathcal{R}^{N\times N\times 3}$. Then the augmented Lagrangian form of Eq. (9) is formulated as

$$\begin{aligned}
&\min_{\mathcal{Z},\mathcal{E},H,L,J} ||\mathcal{Z}||_* + \lambda||\mathcal{E}||_F^2 + \frac{\mu}{2}||\mathcal{Z}(:,:,1) - H + \frac{\Theta_1}{\mu}||_F^2 + \frac{\mu}{2}||\mathcal{Z}(:,:,2) - L + \frac{\Theta_2}{\mu}||_F^2 \\
&+ \frac{\mu}{2}||\mathcal{Z}(:,:,3) - J + \frac{\Theta_3}{\mu}||_F^2 + \frac{\mu}{2}||\mathcal{Z} + \mathcal{E} - \mathcal{A} + \frac{\Theta_4}{\mu}||_F^2 \\
&s.t. \; H(i,j) = A_1(i,j), \text{ if } A_1(i,j) = 1, \; H = H^T, \; 0 \le H(i,j) \le 1, \; \forall i,j, \\
&\quad L(i,j) = A_2(i,j), \text{ if } A_2(i,j) = 0, \; L = L^T, \\
&\quad 0 \le L(i,j) \le 1, \; \forall i,j, \; J = J^T, \; 0 \le J(i,j) \le 1, \; \forall i,j,
\end{aligned} \tag{10}$$

where $\mu > 0$ is a parameter. We then optimize Eq. (10) by solving different subproblems alternately in the following.

$\mathcal{Z}$ sub-problem:

$$\min_{\mathcal{Z}} \frac{1}{\mu}||\mathcal{Z}||_* + \frac{1}{2}||\mathcal{Z} - (\mathcal{A} + \zeta - \mathcal{E} - \frac{\Theta_4}{\mu})||_F^2, \tag{11}$$

where

$$\begin{cases}
\zeta(:,:,1) = H - \frac{\Theta_1}{\mu} \\
\zeta(:,:,2) = L - \frac{\Theta_2}{\mu} \\
\zeta(:,:,3) = J - \frac{\Theta_3}{\mu}.
\end{cases}, \tag{12}$$

Based on Zhang et al. (2014), we can obtain solution of Eq. (11) in a closed form.

$\mathcal{E}$ sub-problem:

$$\min_{\mathcal{E}} \lambda||\mathcal{E}||_F^2 + \frac{\mu}{2}||\mathcal{Z} + \mathcal{E} - \mathcal{A} + \frac{\Theta_4}{\mu}||_F^2, \tag{13}$$

which is a quadratic function regarding $\mathcal{E}$. We then set the derivative of $\mathcal{E}$ to 0 and obtain the global minimum solution as follows:

$$\mathcal{E} = \frac{\mu\mathcal{A} - \Theta_4 - \mu\mathcal{Z}}{2\lambda + \mu}. \tag{14}$$

H sub-problem:

$$\begin{aligned}
&\min_{H} \frac{\mu}{2}||\mathcal{Z}(:,:,1) - H + \frac{\Theta_1}{\mu}||_F^2 \\
&s.t. \; H(i,j) = A_1(i,j), \text{ if } A_1(i,j) = 1, \; H = H^T, \; 0 \le H(i,j) \le 1, \; \forall i,j.
\end{aligned} \tag{15}$$

Since Eq. (15) is a bounded and symmetric constrained least squares problem, we can obtain the optimal solution in elementwise Jia et al. (2020b) as follows:

$$H(i,j) = \begin{cases}
A_1(i,j) & \text{if } A_1(i,j) = 1, \\
\kappa_1(i,j) & \text{if } \kappa_1(i,j) \in [0,1] \text{ and } A_1(i,j) \ne 1, \\
0 & \text{if } \kappa_1(i,j) \le 0 \text{ and } A_1(i,j) \ne 1, \\
1 & \text{if } \kappa_1(i,j) \ge 1 \text{ and } A_1(i,j) \ne 1,
\end{cases} \tag{16}$$

where

$$\kappa_1 = \frac{1}{2}(\mathcal{Z}(:,:,1) + \mathcal{Z}(:,:,1)^T + \frac{\Theta_1 + \Theta_1^T}{\mu}). \tag{17}$$

L sub-problem:

$$\min_L ||\frac{\mu}{2}||\mathcal{Z}(:,:,2) - L + \frac{\Theta_2}{\mu}||_F^2 \ \ s.t.\ L(i,j) = A_2(i,j),\ \text{if}\ A_2(i,j) = 0,\ L = L^T, \\ 0 \le L(i,j) \le 1,\ \forall i,j. \tag{18}$$

Likewise, the optimal solution can be obtained as follows:

$$L(i,j) = \begin{cases} A_2(i,j) & \text{if}\ \ A_2(i,j) = 0, \\ \kappa_2(i,j) & \text{if}\ \ \kappa_2(i,j) \in [0,1]\ \ \text{and}\ A_2(i,j) \neq 0, \\ 0 & \text{if}\ \ \kappa_2(i,j) \le 0\ \ \text{and}\ A_2(i,j) \neq 0, \\ 1 & \text{if}\ \ \kappa_2(i,j) \ge 1\ \ \text{and}\ A_2(i,j) \neq 0, \end{cases} \tag{19}$$

where

$$\kappa_2 = \frac{1}{2}(\mathcal{Z}(:,:,2) + \mathcal{Z}(:,:,2)^T + \frac{\Theta_2 + \Theta_2^T}{\mu}). \tag{20}$$

J sub-problem:

$$\min_J \frac{\mu}{2}||\mathcal{Z}(:,:,3) - J + \frac{\Theta_3}{\mu}||_F^2 \ \ s.t.\ J = J^T,\ 0 \le J(i,j) \le 1,\ \forall i,j. \tag{21}$$

We can achieve the optimal solution as follows:

$$J(i,j) = \begin{cases} \kappa_3(i,j) & \text{if}\ \ \kappa_3(i,j) \in [0,1], \\ 0 & \text{if}\ \ \kappa_3(i,j) \le 0, \\ 1 & \text{if}\ \ \kappa_3(i,j) \ge 1, \end{cases} \tag{22}$$

where

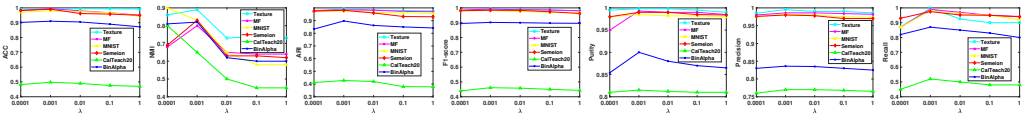
$$\kappa_3 = \frac{1}{2}(\mathcal{Z}(:,:,3) + \mathcal{Z}(:,:,3)^T + \frac{\Theta_3 + \Theta_3^T}{\mu}). \tag{23}$$

Update μ, Θ_1, Θ_2 and Θ_3

$$\begin{cases} \mu = \min(1.1\mu, \mu_{max}) \\ \Theta_1 = \Theta_1 + \mu(\mathcal{Z}(:,:,1) - H) \\ \Theta_2 = \Theta_2 + \mu(\mathcal{Z}(:,:,2) - L) \\ \Theta_3 = \Theta_2 + \mu(\mathcal{Z}(:,:,3) - J) \end{cases}, \tag{24}$$

where the parameter μ is set to be 0.0001 as initialization and μ_{max} denotes the upper-bound of μ. We summarize all procedure in Algorithm 1 for Appendix.

4 Experiments and Analysis



(a) (b) (c) (d) (e) (f) (g)

Figure 1: Clustering performance of our method on different data sets against different λ.

We perform a series of experiments to demonstrate the effectiveness of the proposed method on different data sets in terms of different metrics.

Data sets: Six commonly used data sets Huang et al. (2018); Zhou et al. (2019); Jia et al. (2021) in ensemble clustering are adopted in the experiment, including BinAlpha, MNIST, Multiple features (MF), Semeion, Texture and CalTech20.

Table 1: Clustering performance (Mean) on BinAlpha, MF, MNIST, Semeion, CalTech20, and Texture.

BinAlpha	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.186	0.429	0.403	0.424	0.431	0.375	0.454	0.375	0.298	0.443	0.858	**0.910**
NMI	0.300	0.574	0.553	0.570	0.575	0.537	0.592	0.518	0.541	0.585	0.916	**0.942**
ARI	0.081	0.291	0.289	0.284	0.287	0.269	0.300	0.248	0.227	0.291	0.817	**0.897**
F1-score	0.126	0.313	0.313	0.306	0.308	0.295	0.320	0.271	0.302	0.311	0.822	**0.902**
MF	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.507	0.648	0.609	0.671	0.649	0.589	0.652	0.362	0.581	0.592	0.990	**0.995**
NMI	0.536	0.654	0.650	0.655	0.655	0.618	0.652	0.347	0.621	0.602	0.979	**0.986**
ARI	0.3710	0.523	0.514	0.533	0.530	0.481	0.526	0.257	0.459	0.472	0.979	**0.985**
F1-score	0.554	0.575	0.567	0.583	0.582	0.541	0.576	0.370	0.527	0.528	0.981	**0.992**
MNIST	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.207	0.665	0.658	0.613	0.573	0.609	0.656	0.480	0.543	0.539	0.977	**0.980**
NMI	0.133	0.622	0.635	0.612	0.594	0.608	0.635	0.434	0.482	0.521	0.979	**0.989**
ARI	0.051	0.522	0.531	0.483	0.460	0.495	0.524	0.342	0.429	0.384	0.969	**0.976**
F1-score	0.557	0.572	0.582	0.540	0.522	0.558	0.574	0.427	0.445	0.450	0.972	**0.982**
Semeion	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.425	0.692	0.739	0.682	0.620	0.638	0.679	0.450	0.571	0.594	0.983	**0.991**
NMI	0.418	0.631	0.656	0.630	0.598	0.601	0.635	0.386	0.571	0.569	0.962	**0.980**
ARI	0.248	0.507	0.540	0.507	0.465	0.480	0.508	0.290	0.401	0.418	0.962	**0.981**
F1-score	0.360	0.560	0.588	0.559	0.525	0.540	0.560	0.391	0.477	0.481	0.966	**0.984**
CalTech20	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.421	0.345	0.423	0.324	0.336	0.450	0.363	0.340	0.495	0.297	0.418	**0.498**
NMI	0.269	0.401	0.454	0.396	0.406	0.455	0.428	0.350	0.452	0.381	0.621	**0.710**
ARI	0.184	0.267	0.359	0.222	0.224	0.351	0.258	0.225	0.395	0.202	0.328	**0.428**
F1-score	0.363	0.338	0.432	0.291	0.298	0.437	0.332	0.316	0.457	0.269	0.384	**0.461**
Texture	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
ACC	0.410	0.732	0.793	0.719	0.686	0.675	0.675	0.416	0.634	0.614	0.993	**0.995**
NMI	0.438	0.731	0.782	0.742	0.739	0.703	0.718	0.419	0.693	0.638	0.995	**0.997**
ARI	0.237	0.619	0.696	0.628	0.609	0.569	0.585	0.298	0.534	0.486	0.993	**0.995**
F1-score	0.350	0.656	0.724	0.663	0.648	0.614	0.626	0.397	0.590	0.537	0.993	**0.994**

Table 2: Clustering performance (Mean) on BinAlpha, MF, MNIST, Semeion, CalTech20, and Texture.

BinAlpha	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.197	0.446	0.413	0.444	0.457	0.383	0.478	0.396	0.285	0.470	0.876	**0.900**
Precision	0.071	0.277	0.248	0.272	0.277	0.220	0.305	0.238	0.294	0.296	0.801	**0.836**
Recall	0.635	0.361	0.426	0.349	0.348	0.451	0.337	0.323	0.314	0.327	0.845	**0.870**
MF	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.533	0.677	0.650	0.690	0.673	0.616	0.676	0.387	0.615	0.623	0.990	**0.993**
Precision	0.344	0.534	0.517	0.551	0.530	0.472	0.541	0.311	0.424	0.496	0.981	**0.986**
Recall	0.712	0.627	0.628	0.619	0.647	0.637	0.618	0.739	0.713	0.566	0.981	**0.990**
MNIST	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.209	0.685	0.676	0.663	0.626	0.624	0.691	0.498	0.557	0.585	0.980	**0.985**
Precision	0.124	0.541	0.536	0.490	0.459	0.448	0.543	0.373	0.316	0.420	0.968	**0.982**
Recall	0.952	0.607	0.641	0.603	0.609	0.745	0.610	0.576	0.831	0.485	0.977	**0.980**
Semeion	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.449	0.703	0.739	0.702	0.651	0.645	0.705	0.460	0.607	0.634	0.983	**0.990**
Precision	0.246	0.522	0.552	0.523	0.466	0.468	0.527	0.329	0.381	0.448	0.966	**0.980**
Recall	0.712	0.606	0.631	0.601	0.603	0.644	0.599	0.664	0.660	0.524	0.966	**0.975**
CalTech20	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.520	0.637	0.665	0.642	0.646	0.645	0.660	0.590	0.664	0.633	0.788	**0.815**
Precision	0.284	0.563	0.612	0.529	0.510	0.538	0.543	0.479	0.503	0.525	0.743	**0.770**
Recall	0.562	0.243	0.335	0.201	0.211	0.373	0.239	0.253	0.449	0.181	0.259	**0.520**
Texture	PTA-SL	PTGP	LWEA	LWSC	LWGP	ECPCSHC	ECPCSMC	DREC	SPCE	SEC	CELTA	Ours
Purity	0.427	0.746	0.798	0.744	0.728	0.685	0.698	0.441	0.658	0.647	0.995	**0.998**
Precision	0.228	0.627	0.700	0.631	0.592	0.543	0.582	0.337	0.465	0.500	0.991	**0.995**
Recall	0.897	0.689	0.752	0.700	0.719	0.710	0.677	0.752	0.827	0.586	0.996	**0.998**

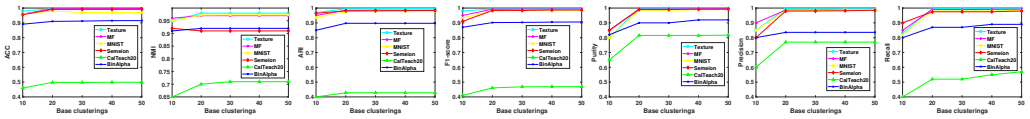
Evaluation metrics: We adopt seven widely used metrics to study the performance for clustering, *i.e.*, accuracy (ACC), adjust rank index (ARI), normalized mutual information (NMI), F1-score, Purity, Precision and Recall. The definitions of these metrics can be referred to Jia et al. (2020a); Zhang et al. (2020).

Table 3: Clustering performance on BinAlpha with different $A_v(i,j)$.

Metrics	ACC	NMI	ARI	F1-score	Purity	Precision
(0.6,0.4)	0.800	0.840	0.820	0.832	0.820	0.750
(0.7,0.3)	0.820	0.860	0.835	0.845	0.853	0.770
(0.8,0.2)	0.879	0.892	0.842	0.870	0.876	0.800
(0.9,0.1)	0.900	0.920	0.880	0.894	0.880	0.815
(1,0)	0.910	0.942	0.897	0.902	0.900	0.836

Table 4: Clustering performance on MF with different $A_v(i,j)$.

Metrics	ACC	NMI	ARI	F1-score	Purity	Precision
(0.6,0.4)	0.900	0.912	0.920	0.925	0.940	0.923
(0.7,0.3)	0.925	0.934	0.943	0.950	0.945	0.935
(0.8,0.2)	0.947	0.956	0.960	0.965	0.960	0.941
(0.9,0.1)	0.980	0.975	0.978	0.980	0.973	0.968
(1,0)	0.995	0.986	0.985	0.992	0.993	0.986



(a) (b) (c) (d) (e) (f) (g)

Figure 2: Clustering performance of our method on different data sets against different number of base clusterings.

Compared methods: We employ 11 representative ensemble clustering methods as the compared methods in the experiment as: 1. PTGP and PTA-SL in Huang et al. (2016); 2. LWSC, LWEA, and LWGP in Huang et al. (2018); 3. ECPCSMC and ECPCSHC in Huang et al. (2021); 4. SEC, DREC, and SPCE in Liu et al. (2017); 5. CELTA in Jia et al. (2021).

Generation for base clusterings: We generate 100 different candidate base clusterings as a pool by using the K-means algorithm for all data sets, which is consistent with Jia et al. (2021). We randomly vary K, which is in $\left[2, \sqrt{N}\right]$ with N being the total number of data points.

Experimental settings: In the experiment, 20 base clusterings are randomly chosen from the candidate base clustering pool for each data set. We then use these base clusterings to conduct different methods for ensemble clustering. The parameters of the compared methods are set the same as the original papers. For all data sets, we set the parameter λ to 0.001. We report clustering results of our method and compared methods in terms of the average performance over 20 repetitions.

4.1 Parameter Selection

In this part, we study how the parameter λ and the number of base clusterings influence the clustering performance, respectively. We show the clustering performance in terms of multiple metrics with different λ for the proposed method in Fig. 1, from which we can gain some insights. The values of λ are nearly consistent for all data sets when the best clustering performance occurs, which demonstrates the robustness of our method for all used data sets in terms of different metrics. A proper λ is able to produce satisfied clustering performance for all data sets, which shows that modeling error based on the co-association tensor plays an important role in improving the final clustering performance.

We also study how the number of base clusterings affects the clustering performance based on different metrics for all data sets in Fig. 2, from which some observations can be obtained. With the increasing number of base clusterings, the clustering performance of the proposed method can be improved. The performance improvements become smaller with more base clusterings for different data sets in terms of different metrics when the number of base clusterings is larger than 20, indicating that sufficient base clusterings are able to achieve satisfied clustering performance.

Table 5: Clustering performance on MNIST with different $A_v(i,j)$.

Metrics	ACC	NMI	ARI	F1-score	Purity	Precision
(0.6,0.4)	0.900	0.921	0.930	0.910	0.925	0.927
(0.7,0.3)	0.913	0.930	0.942	0.928	0.937	0.945
(0.8,0.2)	0.932	0.945	0.950	0.942	0.950	0.956
(0.9,0.1)	0.960	0.952	0.968	0.957	0.970	0.978
(1,0)	0.980	0.989	0.976	0.982	0.985	0.982

Table 6: Clustering performance on Semeion with different $A_v(i,j)$.

Metrics	ACC	NMI	ARI	F1-score	Purity	Precision
(0.6,0.4)	0.900	0.912	0.920	0.900	0.925	0.942
(0.7,0.3)	0.917	0.930	0.937	0.940	0.950	0.959
(0.8,0.2)	0.968	0.952	0.960	0.955	0.974	0.970
(0.9,0.1)	0.980	0.975	0.974	0.978	0.985	0.974
(1,0)	0.991	0.980	0.981	0.984	0.990	0.980

4.2 Experimental Results and Analysis

We list the clustering performance for different approaches in terms of seven metrics over six data sets on Tables 1-2. We repeat each experiment for 30 times and report the mean. We can have the conclusions in the following.

- The proposed method always produces better clustering performance than the compared methods on different data sets under seven metrics, which demonstrates the advantage of extracting representations for co-association matrix based on δ-sensitive feature descriptor as different views and stacking them into a tensor for the latent co-association matrix.
- Moreover, our method consistently achieves satisfied clustering performance on six data sets, indicating that the proposed method is robust to all used data sets.
- In addition, significant improvements can be obtained by our method on some data sets. For example, ACC of our method is 0.980 while it is 0.539 for SEC on MNIST data set.

We also show the visualization of the obtained latent co-association matrix on data sets and give the corresponding analysis in the appendix.

4.3 Study of $A_v(i,j)$

We then explore how the value of the feature representation $A_v(i,j)$ influences the final clustering results in this part. We set $A_v(i,j) = \{0.6, 0.7, 0.8, 0.9, 1.0\}$ when $A(i,j) \geq \delta_v$ and $A_v(i,j) = \{0.4, 0.3, 0.2, 0.1, 0.0\}$ when $A(i,j) < \delta_v$ in the experiment. For simplicity, we can use five cases to study the desired value of $A_v(i,j)$, *i.e.*, $A_v(i,j) = \begin{cases} 0.6, & \text{if } A(i,j) \geq \delta_v, \\ 0.4, & \text{otherwise,} \end{cases}$. According to Tables 3-6, we can find that better clustering performance can be obtained when $A_v(i,j) = \begin{cases} 1, & \text{if } A(i,j) \geq \delta_v, \\ 0, & \text{otherwise,} \end{cases}$. That is, the value of the feature representation $A_v(i,j)$ is properly choosed in this work, which is demonstrated by the experiment.

5 Conclusion

We show the equivalence between ensemble clustering and multi-view clustering based on the formulation of objective function. To be specific, we propose the multi-view co-association matrices and introduce the latent co-association matrix shared by co-association matrices from different views based on δ-sensitive feature descriptor. These multi-view co-association matrices are collected into a tensor and the relation between ensemble clustering and multi-view clustering can be built in this manner. Experiments on six data sets demonstrate the effectiveness of our method under seven metrics compared with 11 representative methods.

STATEMENT ON THE USE OF AI TOOLS

We used AI tools to assist with English language editing and LaTeX formatting. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the accuracy, originality, and integrity of the manuscript.

REFERENCES

Junning Gao, Makoto Yamada, Samuel Kaski, Hiroshi Mamitsuka, Shanfeng Zhu, and Subbarao Kambhampati. A robust convex formulation for ensemble clustering. In *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 1476–1482, 2016.

Dong Huang, Jian-Huang Lai, and Chang-Dong Wang. Robust ensemble clustering using probability trajectories. *IEEE Trans. Knowl. Data Eng.*, 28(5):1312–1326, 2016.

Dong Huang, Chang-Dong Wang, and Jian-Huang Lai. Locally weighted ensemble clustering. *IEEE Trans. Cybern.*, 48(5):1460–1473, 2018.

Dong Huang, Chang-Dong Wang, Hongxing Peng, Jianhuang Lai, and Chee-Keong Kwoh. Enhanced ensemble clustering via fast propagation of cluster-wise similarities. *IEEE Trans. Syst. Man Cybern. Syst.*, 51(1):508–520, 2021.

Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognit. Lett.*, 31(8):651–666, 2010.

Yuheng Jia, Sam Kwong, and Junhui Hou. Semi-supervised spectral clustering with structured sparsity regularization. *IEEE Signal Process. Lett.*, 25(3):403–407, 2018.

Yuheng Jia, Hui Liu, Junhui Hou, and Sam Kwong. Clustering-aware graph construction: A joint learning perspective. *IEEE Trans. Signal Inf. Process. over Networks*, 6:357–370, 2020a.

Yuheng Jia, Wenhui Wu, Ran Wang, Junhui Hou, and Sam Kwong. Joint optimization for pairwise constraint propagation. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2020b.

Yuheng Jia, Hui Liu, Junhui Hou, and Qingfu Zhang. Clustering ensemble meets low-rank tensor approximation. pp. 7970–7978, 2021.

Stephen Johnson. Hierarchial clustering schemes. *Psychometrika*, 32:241–254, 02 1967.

Misha Elena Kilmer, Karen S. Braman, Ning Hao, and Randy C. Hoover. Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM J. Matrix Anal. Appl.*, 34(1):148–172, 2013.

Ruihuang Li, Changqing Zhang, Huazhu Fu, Xi Peng, Joey Tianyi Zhou, and Qinghua Hu. Reciprocal multi-layer subspace learning for multi-view clustering. In *Proceedings of International Conference on Computer Vision*, pp. 8171–8179, 2019.

Hongfu Liu, Junjie Wu, Tongliang Liu, Dacheng Tao, and Yun Fu. Spectral ensemble clustering via weighted k-means: Theoretical and practical evidence. *IEEE Trans. Knowl. Data Eng.*, 29(5): 1129–1143, 2017.

Andrew W. Moore. Very fast em-based mixture model clustering using multiresolution kd-trees. In *Proceedings of Neural Information Processing Systems*, pp. 543–549, 1998.

Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of Neural Information Processing Systems*, pp. 849–856, 2001.

Yalan Qin, Hanzhou Wu, Xinpeng Zhang, and Guorui Feng. Semi-supervised structured subspace learning for multi-view clustering. *IEEE Transactions on Image Processing*, 31:1–14, 2022a.

Yalan Qin, Xinpeng Zhang, Liquan Shen, and Guorui Feng. Maximum block energy guided robust subspace clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2022b.

Yalan Qin, Chuan Qin, Xinpeng Zhang, Donglian Qi, and Guorui Feng. Nim-nets: Noise-aware incomplete multi-view learning networks. *IEEE Transactions on Image Processing*, 32:175–189, 2023.

Zhiqiang Tao, Hongfu Liu, Sheng Li, Zhengming Ding, and Yun Fu. From ensemble clustering to multi-view clustering. In *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 2843–2849, 2017.

Ulrike von Luxburg. A tutorial on spectral clustering. *Stat. Comput.*, 17(4):395–416, 2007.

Jianlong Wu, Zhouchen Lin, and Hongbin Zha. Essential tensor learning for multi-view spectral clustering. *IEEE Transactions on Image Processing*, 28(12):5910–5922, 2019.

Yuan Xie, Dacheng Tao, Wensheng Zhang, Yan Liu, Lei Zhang, and Yanyun Qu. On unifying multi-view self-representations for clustering by tensor multi-rank minimization. *International Journal of Computer Vision*, 126(11):1157–1179, 2018.

Changqing Zhang, Huazhu Fu, Qinghua Hu, Xiaochun Cao, Yuan Xie, Dacheng Tao, and Dong Xu. Generalized latent multi-view subspace clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42 (1):86–99, 2020.

Zemin Zhang, Gregory Ely, Shuchin Aeron, Ning Hao, and Misha Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3842–3849, 2014.

Jie Zhou, Hongchan Zheng, and Lulu Pan. Ensemble clustering based on dense representation. *Neurocomputing*, 357:66–76, 2019.

Peng Zhou, Liang Du, Xinwang Liu, Yi-Dong Shen, Mingyu Fan, and Xuejun Li. Self-paced clustering ensemble. *IEEE Trans. Neural Networks Learn. Syst.*, 32(4):1497–1511, 2021.

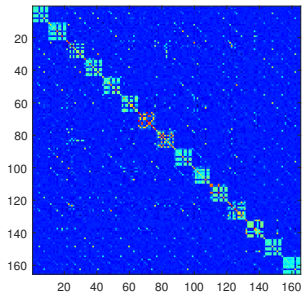
A APPENDIX

The algorithm of the specific example is shown in the following.

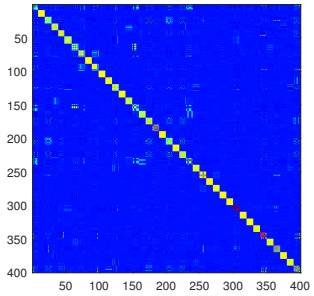
Algorithm 1: Algorithm of the specific example

Input: Base clusterings Π and the number of clusters k in the dataset;
Output: The latent co-association matrix A_H.
Initialize: Randomly initialize $\mathcal{Z}, \mathcal{E}, H, L, J$ and $\mu_{max} = 10^6$;
Compute A, A_1 and A_2 by Eq. (3);
repeat
  Update $\mathcal{Z}$ by solving Eq. (11);
  Update $\mathcal{E}$ by Eq. (14);
  Update H by Eq. (16);
  Update L by Eq. (19);
  Update J by Eq. (22);
  Update $\Theta_1, \Theta_2, \Theta_3$ and μ by Eq. (24);
until *convergence*;

We also present the visualization of the obtained latent co-association matrix on Yale and ORL data set for further analysis in this section. According to Fig. 3, it can be observed that our method is able to achieve satisfied performance since the diagonal blocks are equipped with large values.



(a) Yale data set



(b) ORL data set

Figure 3: Visualization of the latent co-association matrix on Yale and ORL data set.

We then give the summary of frequently used notations in the following.

Table 7: Summary of frequently used notations

a	A scalar
A	A matrix
$\mathcal{A}$	A tensor
$\|\|A\|\|_1$	$\|\|A\|\|_1 = \sum_{ij} \|A_{ij}\|$
$\|\|A\|\|_F$	$\|\|A\|\|_F = \sqrt{\sum_{ij} A_{ij}^2}$
$\|\|A\|\|_{2,1}$	$\|\|A\|\|_{2,1} = \sum_j \|\|A(:,j)\|\|_2$
$\|\|A\|\|_*$	Sum of the singular values
$\mathcal{A}_{ijk}$	The (i,j,k)-th entry of $\mathcal{A}$
$\mathcal{A}_{\circledast}$	The tensor nuclear norm based on t-SVD
$\mathcal{A}(i,:,:)$	The i-th horizontal slice of $\mathcal{A}$
$\mathcal{A}(:,i,:)$	The i-th lateral slice of $\mathcal{A}$
$\mathcal{A}(:,:,i)$	The i-th frontal slice of $\mathcal{A}$
$\mathcal{A}^T$	The transpose of $\mathcal{A}$
$\mathcal{A}_F$	$\mathcal{A}_F = \sqrt{\sum_{ijk} \|\mathcal{A}_{ijk}\|^2}$
$\mathcal{A}_1$	$\mathcal{A}_1 = \sum_{ijk} \|\mathcal{A}_{ijk}\|$
$\mathcal{A}_{2,1}$	$\mathcal{A}_{2,1} = \sum_{ij} \|\|\mathcal{A}(i,j,:)\|\|_2$