Neurocomputing 514 (2022) 526–538

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Anchor-based incomplete multi-view spectral clustering

Jun Yin [a], Runcheng Cai [a], Shiliang Sun [b,*]

[a] College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China
[b] School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

ARTICLE INFO

Article history:
Received 18 January 2022
Revised 3 August 2022
Accepted 24 September 2022
Available online 29 September 2022
Communicated by Zidong Wang

Keywords:
Multi-view
Missing views
Anchor
Spectral clustering

ABSTRACT

In the past decade, multi-view clustering has become a research hot spot of machine learning. In traditional multi-view clustering methods, all views of the data points are assumed to be complete. However, in the practical applications, some views of data points may be missing and incomplete multi-view clustering methods are developed to handle these incomplete multi-view data. The existing incomplete multi-view clustering methods still have some defects such as insufficient use of missing information or neglecting the underlying relations among different views. To address these limitations, we propose an Anchor-based Incomplete Multi-view Spectral Clustering (AIMSC) approach. Specifically, AIMSC utilizes anchor points to connect all instances of each view and recover the missing information. Then, the similarities between all data points are derived from the similarities between data points and anchor points. Finally, anchor-based spectral clustering is executed to generate the clustering results. Experimental results on multiple benchmark datasets demonstrate the superiority of AIMSC.

1. Introduction

Clustering is a critical task in unsupervised learning. Over the past few decades, lots of clustering methods have been proposed, such as subspace clustering [1,2], density-based clustering [3,4], spectral clustering [5,6], etc. Nevertheless, these traditional algorithms can only tackle clustering problems with a single view.

As the booming of information technology, data can be gathered from a variety of sources in different domains or obtained through diverse property collectors [7,8]. For example, an image can be described by Fourier coefficients and pixel features. A web page can be decomposed into text, picture and video. Considering different views have potential connection and complementary information, researchers developed Multi-View Clustering (MVC) methods by leveraging information from all views to enhance the performance of clustering [9]. Multi-view Non-negative Matrix Factorization (MultiNMF) [10] constructs a joint NMF structure to promote clustering results from each view towards a uniform representation. Complex mapping multi-view clustering [11] investigates the connection between different views through the mapping relationship matrix and uses NMF to integrate information. Deep embedded multi-view clustering with collaborative training [12] introduces deep autoencoders to learn features from multiple views and optimizes the learning process by co-training. However, these MVC algorithms presuppose that the multi-view data are complete. This assumption does not always hold since some views of data points may be missing in the practical applications.

To tackle this barrier, researchers developed Incomplete Multi-view Clustering (IMC) methods. According to the mechanisms and principles, the present IMC methods can be divided into four categories [13], i.e., Kernel-based Incomplete Multi-view Clustering (KIMC), Matrix Factorization-based Incomplete Multi-view Clustering (MFIMC), Graph-based Incomplete Multi-view Clustering (GIMC) and Neural Networks-based Incomplete Multi-view Clustering (NNIMC). KIMC approaches investigate the relationships between data points by refining the kernel matrix. Kernel canonical correlation analysis based multi-view clustering [14] uses the Laplacian regularization to construct the complete kernel matrix. However, it requires that one view of the data must be complete. Multiple Kernel K-means with Incomplete Kernels (MKKM-IK) [15] integrates clustering and kernel repairing into one framework, and the kernel matrixs are optimized by iterative computation. Structure-preserving multiple kernel clustering [16] combines a kernel group self-expressiveness term and a kernel adaptive local structure learning term to preserve the global and local structure of the input data in kernel space. Consensus affinity graph Learning [17] learns multiple candidate affinity graphs from a kernel pool and synthesizes these graphs to generate a consensus affinity graph via an autoweighted fusion model.

* Corresponding author.
E-mail addresses: junyin@shmtu.edu.cn (J. Yin), 202030310093@stu.shmtu.edu.cn (R. Cai), slsun@cs.ecnu.edu.cn (S. Sun).

https://doi.org/10.1016/j.neucom.2022.09.142
0925-2312/© 2022 Elsevier B.V. All rights reserved.

MFIMC methods use matrix factorization to analyze the connection between multiple views. Partial Multi-View Clustering (PVC) [18] designs public and private latent spaces to extract information from aligned instances and unaligned instances respectively. Nevertheless, it cannot be applied directly to data with more than two views. Unified Embedding Alignment Framework (UEAF) [19] proposes a matrix factorization based model to reconstruct the missing data. It obtains the local structure by the reverse graph regularization and uses the automatic weighted mechanism to enhance the discriminability of the learned common representation. Doubly Aligned Incomplete Multi-view Clustering (DAIMC) [20] utilizes semi-NMF to learn the latent feature and introduces $L_{2,1}$-norm regularized regression to construct a consensus basis matrix. Generalized Incomplete Multiview Clustering with Flexible Locality Structure Diffusion (GIMC-FLSD) [21] extracts local geometric features from each view by a novel graph-regularized matrix factorization term, and introduces adaptively learned weights to control the balance between different views.

Different from MFIMC, GIMC combines information from each view to explore the similarities between all instances. Incomplete Multi-modality Grouping (IMG) [22] integrates PVC and graph Laplacian to adaptively capture global structures. However, like PVC, it can only handle two-view data. Simultaneous Representation Learning and Clustering (SRLC) [23] investigates the nonlinear information by constructing the similarity matrix and introduces the probabilistic spectral rotation matrices to deliver the clustering results. Incomplete multi-view clustering with cosine similarity [24] directly calculates the cosine similarity in the original multi-view space to strengthen the ability of preserving the manifold structure of the original multi-view data. It integrates the manifold structure preserving term with cosine similarity and the matrix factorization term into a unified objective function. Adaptive graph completion based incomplete multi-view clustering [25] combines data recovery and unified representation learning into one model. Meanwhile, it uses the adaptive scale vector to reduce the negative effect of the missing views.

NNIMC methods use neural networks to extract useful information from incomplete multi-view data. Deep partial multi-view learning [26] invents a Cross Partial Multi-view Networks (CPM-Nets) model, which can achieve the optimal tradeoff between consistency and complementarity across different views. A nonparametric classification loss is introduced to the model to prevent overfitting. Clustering-induced adaptive structure enhancing network [27] infers the missing data by multi-view autoencoder, and adopts graph learning and graph convolution to obtain effective structure.

In recent years, the anchor graph strategy has been used in clustering algorithms since it can effectively describe the relationships of data points and reduce the computational complexity. Wang et al. [28] proposed a systematic density-based clustering method with anchor points and achieved remarkable advances over traditional single-view clustering. Kang et al. [29] proposed the Large-scale Multi-View Subspace Clustering (LMVSC) approach, which constructs smaller graphs for each view by anchor points and finally integrates these graphs to extract a consistent representation. It obtains desired results on the complete multi-view dataset.

Although researchers have made significant progress in IMC problem, the existing IMC methods still have some limitations, such as insufficient use of missing information, neglecting the potential relations of different views and limited generalization capability. To address these limitations, this paper proposes an Anchor-based Incomplete Multi-view Spectral Clustering (AIMSC) approach. AIMSC selects data points with complete views as anchor points to connect all instances of each view. Then, it recovers the missing information and extracts the similarity matrix between all data points. Finally, anchor-based spectral clustering is executed to generate the clustering results.

The contributions of the proposed AIMSC are summarized as follows.

(1)The missing views are reconstructed by anchor points, not filled with the average values of existing instances of the corresponding view. More accurate missing information can be obtained in this way.
(2)Anchor-based spectral clustering is finally performed on a small size matrix, which can reduce the time complexity.
(3)To solve the objective function of AIMSC, we present an alternating iterative algorithm, and the convergence of the algorithm is demonstrated by theoretical analysis and experimental results.

The remaining of the paper is arranged as follows: Section 2 reviews spectral clustering and anchor graph strategy. Section 3 proposes AIMSC and its optimization algorithm. Section 4 illustrates the experimental results and the corresponding analyses. Section 5 summarizes the paper.

2. Related work

2.1. Spectral clustering

Spectral clustering [30,31] is a prevalent method for data analysis, which is based on graph theory. It applies the concept of cutting graphs to solve the clustering problem. Specifically, spectral clustering describes the relationships between data points by constructing a graph, which is presented as the similarity matrix. Then the new representation of data can be assembled with the eigenvectors of the graph Laplacian matrix. The objective function of spectral clustering is defined as

$$\min_{F^TF=I} tr\left(F^TLF\right), \tag{1}$$

where $L = D - S$ is the Laplacian matrix, $S \in R^{n\times n}$ is the similarity matrix which reflects the similarities between all data points, n is the number of all the data points, $D \in R^{n\times n}$ is a diagonal matrix whose diagonal element is defined by $d_{ii} = \sum_{j=1}^{n} s_{ij}$, and $F \in R^{n\times k}$ is the representation matrix. $k$is usually set as the number of clusters. The problem (1) can be solved by eigenvalue decomposition of L, and F is composed by eigenvectors of Lcorresponding to k smallest eigenvalues. K-means is performed on F to obtain the clustering results.

2.2. Anchor graph strategy

Anchor graph strategy [32–34] is an effective strategy of reducing the computational complexity of large-scale datasets. It uses a small number of anchor points to describe the whole dataset and achieves excellent results. Specifically, $m(m << n)$ data points are selected as anchor points by random sampling or K-means, which can approximately maintain the intrinsic properties of the original data. Then a similarity matrix $W \in R^{m\times n}$ is constructed which represents the similarities between data points and anchor points. Note that the matrix W is imposed the nonnegative constraint and the sum of elements in each column is one. The crucial step of anchor graph strategy is how to extract an appropriate W. Liu et al. [32] proposed a local anchor embedding algorithm to compute W. Chen and Cai [34] introduced sparse coding [35] to acquire W. With this similarity matrix, the original data can be easily and efficiently approximated. With W, the similarity matrix $S \in R^{n\times n}$ can be derived from

$$S = W^T\Lambda^{-1}W, \tag{2}$$

where $\Lambda = diag(W\mathbf{1}) \in R^{m\times m}$ is a diagonal matrix and $\mathbf{1}$ is an all-one vector. S is called anchor graph which can be fully represented by W.

The anchor graph S has the following crucial properties. Firstly, S is a nonnegative matrix since W is nonnegative. Secondly, S is a doubly-stochastic matrix that means the summation of each column and each row equal to one. Thus, we have

$$D = diag(S\mathbf{1}) = diag(\mathbf{1}) = I. \tag{3}$$

Then, the resulting graph Laplacian matrix L can be obtained by

$$L = I - S. \tag{4}$$

3. Anchor-based incomplete multi-view spectral clustering

3.1. Motivation

For incomplete multi-view data, at least one view of a data point is existing. Otherwise, this data point is nonexistent. In addition, all views of a data point are derived from a common latent representation. Therefore, the information of the missing views can be obtained by the existing views and cannot be ignored. Although each view is represented by different feature sets, the similarity between data points tends to be consistent. Furthermore, graph-based clustering methods (i.e., spectral clustering) can find internal relationships of data points and obtain high clustering accuracy. Based on the above analysis, we try to get a uniform similarity matrix which leverages the information of different views. Fortunately, the anchor graph strategy [32–34] provides us an inspiring perspective, which uses a small number of anchor points to connect other data points and represent the entire dataset.

According to this idea, we can randomly select some data points with complete views as anchor points to bridge all instances of each view. However, the method of constructing Win Section 2.2 can not be directly applied for incomplete multi-view data. To obtain the optimal similarity matrix, we learn it from the original data by solving an optimization problem. After constructing the similarity matrix W between anchor points and data points, we can build a unified similarity matrix Samong all data points. With this similarity matrix, spectral clustering can be performed to generate the clustering results.

Based on the above analysis, the objective function of AIMSC is defined as

$$\min_{X^{(v)},W}\sum_{v=1}^{u}\left\|X^{(v)} - A^{(v)}W\right\|_F^2 + \alpha\|W\|_F^2 \tag{5}$$
$$s.t.\ W^T\mathbf{1} = \mathbf{1},\ W \geqslant 0.$$

where $A^{(v)} \in R^{d^{(v)}\times m}$ is the anchor data matrix of view-$v, X^{(v)} \in R^{d^{(v)}\times n}$ is the complete data matrix of view-$v, d^{(v)}$ is the dimension of view-v, u is the number of views, $\alpha\|W\|_F^2$ is a regularization term and α is a tradeoff-parameter. In fact, anchor points can describe the entire dataset since they link all data points. Our target is to leverage information from all views to explore a consistent similarity matrix W by narrowing the discrepancies of different views.

3.2. Optimization

The objective function in Eq. (5) is not a convex problem, and we use the alternating iterative method to solve it. The iterative procedure can be divided into two parts, i.e. obtaining the complete data $X^{(v)}$ and constructing the similarity matrix W.

Step (1): Fix $W$and update $X^{(v)}$.

The first step is to obtain the complete data $X^{(v)}$. $X^{(v)}$ is consisted by available and missing instances which can be represented as

$$X^{(v)} = Z^{(v)} + Y^{(v)}, \tag{6}$$

where $Z^{(v)} \in R^{d^{(v)}\times n}$ is the missing data matrix of view-$v$which contains available instances filled with zeros and $Y^{(v)} \in R^{d^{(v)}\times n}$ is the available data matrix of view-$v$which contains missing instances filled with zeros. Since anchor points connect all views, the relationships between anchor points and data points in missing views can be complemented by other views. The missing data matrix can be acquired by

$$Z^{(v)} = A^{(v)}WQ^{(v)}. \tag{7}$$

As mentioned in Section 3.1, the instances which appear in all views can be used as anchors. From these instances, we randomly select $m(m << n)$ anchors to build the anchor matrix $A^{(v)}$. The number of anchor points m is set as r percent of the number of all the data points. The parameter r is empercically given, and the optimal range of r will be analyzed in the experiments. $Q^{(v)} \in R^{n\times n}$ is a diagonal matrix corresponding to view-v. It is defined as

$$q_{ii}^{(v)} = \begin{cases} 1 & \text{if the } i\text{th instance of view} - v \text{ is missing} \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

which can be used to mark the position of missing instances of view-v.

Step (2): Fix $X^{(v)}$ and update W Another part is to construct the similarity matrix W. The problem (5) is equivalent to

$$\min_{\mathbf{w}_i}\sum_{i=1}^{n}\left[\sum_{v=1}^{u}\left\|\mathbf{x}_i^{(v)} - A^{(v)}\mathbf{w}_i\right\|_2^2 + \alpha\|\mathbf{w}_i\|_2^2\right] \tag{9}$$
$$s.t.\ \mathbf{w}_i^T\mathbf{1} = 1,\ \mathbf{w}_i \geqslant 0,$$

where $\mathbf{x}_i^{(v)}$ and $\mathbf{w}_i$ are the ith column of $X^{(v)}$ and W, respectively. For the ith data point, we have

$$\sum_{v=1}^{u}\left\|\mathbf{x}_i^{(v)} - A^{(v)}\mathbf{w}_i\right\|_2^2 + \alpha\|\mathbf{w}_i\|_2^2$$
$$= \sum_{v=1}^{u}\left\|\mathbf{x}_i^{(v)}\mathbf{1}^T\mathbf{w}_i - A^{(v)}\mathbf{w}_i\right\|_2^2 + \alpha\|\mathbf{w}_i\|_2^2 \tag{10}$$
$$= \sum_{v=1}^{u}\mathbf{w}_i^T\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right)^T\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right)\mathbf{w}_i + \alpha\mathbf{w}_i^T\mathbf{w}_i$$
$$= \mathbf{w}_i^T\left[\sum_{v=1}^{u}\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right)^T\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right) + \alpha I\right]\mathbf{w}_i.$$

.

Thus Eq. (9) can be transformed to a quadratic programming problem as

$$\min_{\mathbf{w}_i}\sum_{i=1}^{n}\mathbf{w}_i^T\left[\sum_{v=1}^{u}\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right)^T\left(\mathbf{x}_i^{(v)}\mathbf{1}^T - A^{(v)}\right) + \alpha I\right]\mathbf{w}_i \tag{11}$$
$$s.t.\ \mathbf{w}_i^T\mathbf{1} = 1,\ \mathbf{w}_i \geqslant 0.$$

Then, we can use convex quadratic programming to solve it.

After updating $[\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n]$ in turn, a new W can be constructed. Then, return to step (1) and use the new $W$to calculate $X^{(v)}$. This process continues until the objective function in Eq. (5) converges. For incomplete multi-view data, we cannot learn

Wdirectly from the original data. Thus, in the beginning of the optimization, we initialize W with a random nonnegative matrix.

3.3. Algorithm

With the obtained W, the similarity matrix S between all data points can be derived from Eq. (2). Then we can perform spectral clustering with S. As mentioned in Section 2.2, S is a doubly-stochastic matrix and the Laplacian matrix L can be derived from Eq. (4). Therefore, Eq. (1) is equivalent to

$$\max_{F^TF=I} tr\left(F^TSF\right). \tag{12}$$

Eq. (12) can be solved by eigenvalue decomposition of S, and F is constituted by eigenvectors of Scorresponding to $k$largest eigenvalues. Since the size of matrix S is $n \times n$, it is time-consuming to directly decompose S for a large-scale dataset. We can skillfully perform fast eigenvalue decomposition on a small size matrix to solve the dilemma. Let $H = W^T\Lambda^{-\frac{1}{2}} \in R^{n\times m}$, we have

$$S = W^T\Lambda^{-\frac{1}{2}}\Lambda^{-\frac{1}{2}}W = HH^T. \tag{13}$$

We denote

$$\widehat{S} = \Lambda^{-\frac{1}{2}}WW^T\Lambda^{-\frac{1}{2}} = H^TH. \tag{14}$$

Through the eigenvalue decomposition of $\widehat{S} \in R^{m\times m}$, we can obtain $k(k < m)$ largest eigenvector-value pairs $\{(\mathbf{b}_i, t_i)\}_{i=1}^{k}$. In addition, we have the following algorithm.

Theorem 1. *If $(\mathbf{b}_i, t_i)$ is the ith eigenvector-value pair of $\widehat{S} = H^TH$, then the ith eigenvector of $S = HH^T$ is $\mathbf{f}_i = H\mathbf{b}_it_i^{-\frac{1}{2}}$.*

Proof. Suppose that the result of SVD on $H \in R^{n\times m}$ is $H = U\Sigma V^T$, where $U = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n] \in R^{n\times n}$ and $V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m] \in R^{m\times m}$ are orthogonal matrices, $\Sigma \in R^{n\times m}$ is a rectangular diagonal matrix with diagonal entries $\Sigma_{ii} = \sigma_i$. Then we have

$$S = U\Sigma V^TV\Sigma^TU^T = U\Sigma\Sigma^TU^T = U\Lambda_1U^T \tag{15}$$

and

$$\widehat{S} = V\Sigma^TU^TU\Sigma V^T = V\Sigma^T\Sigma V^T = V\Lambda_2V^T, \tag{16}$$

where $\Lambda_1 = \Sigma\Sigma^T \in R^{n\times n}$ and $\Lambda_2 = \Sigma^T\Sigma \in R^{m\times m}$ are diagonal matrices, and the diagonal entries of Λ_1 and Λ_2 are both σ_i^2. Eq. (15) and Eq. (16) are just the eigenvalue decomposition results of S and $\widehat{S}$, respectively. From Eq. (15) and Eq. (16), we know that the ith eigenvectors of S and $\widehat{S}$ are $\mathbf{u}_i$ and $\mathbf{v}_i$ respectively, and their corresponding eigenvalues are both σ_i^2. Then we have $\mathbf{f}_i = \mathbf{u}_i$, $\mathbf{b}_i = \mathbf{v}_i$ and $t_i = \sigma_i^2$. According to $H = U\Sigma V^T$, we have $U\Sigma = HV$. Since Σ is a rectangular diagonal matrix, there holds $\mathbf{u}_i\sigma_i = H\mathbf{v}_i$, which can be transformed to $\mathbf{u}_i = H\mathbf{v}_i\sigma_i^{-1}$. With $\mathbf{f}_i = \mathbf{u}_i, \mathbf{b}_i = \mathbf{v}_i$ and $t_i = \sigma_i^2$, we can obtain $\mathbf{f}_i = H\mathbf{b}_it_i^{-\frac{1}{2}}$. □

Suppose that $B = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k]$ and $T \in R^{k\times k}$ is a diagonal matrix containing k largest eigenvalues $\{t_i\}_{i=1}^{k}$ on the main diagonal. Because $F = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_k]$, with Theorem 1, we have

$$F = HBT^{-\frac{1}{2}}. \tag{17}$$

Finally, our desired clustering results are generated by conducting K-means on F. The AIMSC algorithm is presented in Algorithm 1.

Algorithm 1: AIMSC algorithm

```
Input: Incomplete multi-view data $Y^{(v)}$, anchor data $A^{(v)}$.
Output: The cluster indicators.
Randomly initialize $W$ with a nonnegative matrix.
while the objective function in Eq. (5) is not convergent do
    Compute $Z^{(v)}$ via Eq. (7);
    Update $X^{(v)}$ via Eq. (6);
    Update $W$ by solving Eq. (11);
end while
Compute $\widehat{S}$ via Eq. (14);
Decompose $\widehat{S}$ to obtain $B$and $T$;
Compute $F$ via Eq. (17);
Perform K-means on $F$.
```

3.4. Computational complexity

The computational complexity of AIMSC algorithm mainly consists of four steps. In the first step, we obtain $X^{(v)}$ and the computational complexity is $\mathcal{O}\left(\sum_{v=1}^{u} tn^2d^{(v)}\right)$, where t represents the number of iterations. The second step is computing W via convex quadratic programming and the computational complexity is $\mathcal{O}\left(\sum_{v=1}^{u} tnm^2d^{(v)} + tnm^3\right)$. In the third step, we decompose $\widehat{S}$ to obtain F with the computational complexity $\mathcal{O}(m^3)$. In the last step, the computational complexity of K-means is $\mathcal{O}\left(rnk^2\right)$, where r represents the iteration number of K-means. To sum up, the computational complexity of AIMSC is $\mathcal{O}\left(\sum_{v=1}^{u} tn^2d^{(v)} + \sum_{v=1}^{u} tnm^2d^{(v)} + tnm^3 + rnk^2\right)$.

3.5. Convergence

As mentioned in Section 3.2, the optimization procedure of AIMSC can be divided into two steps. For each step, we minimize the value of objective function. Specifically, in Step (1), with the fixed W, the value of objective function is minimized using $X^{(v)}$ calculated by Eq. (7) and Eq. (6). In Step (2), with the fixed $X^{(v)}$, it is minimized using Wobtained by solving the quadratic programming problem (11). Thus, the value of objective function is monotonically decreasing in the iteration. In addition, the objective function is the sum of $\sum_{v=1}^{u}\left\|X^{(v)} - A^{(v)}W\right\|_F^2$ and $\alpha\|W\|_F^2$, and both of them are nonnegative. Thus, the value of objective function is lower bounded. According to monotone convergence theorem, the optimization process of AIMSC is convergent.

4. Experimental results and analyses

In this section, we compare AIMSC with state-of-the-art multi-view clustering methods to verify its effectiveness. Four evaluation metrics ACC (Accuracy), NMI (Normalized Mutual Information), Precision and F-score are used to assess the clustering performance of different methods, which are widely adopted in previous researches. The definitions of these metrics can be found in [36]. For these evaluation metrics, the larger values represent the better results. The code of AIMSC is released at https: //github.com/yinjun8429/AIMSC.

4.1. Experimental setting

Our experiments were conducted on six multi-view datasets. The following is a brief description of the datasets.

- BBC [37]: It contains 2225 pieces of news from the BBC news website, which is classified into five categories, i.e., business, entertainment, politics, sport and tech. Two views were derived from splitting articles into related segments of text. The dimensions of two views are 6838 and 6790, respectively.
- BBCSport [37]: It contains 737 pieces of sports news from the BBC Sport website. Five classes of sports were captured in these news, i.e., cricket, football, tennis, athletics, and rugby. Two views were derived from splitting articles into related segments of text. The dimensions of two views are 3183 and 3203, respectively.
- Multiple features handwritten dataset (Digit) [38]: It consists of ten different handwritten digits, and each digit has 200 examples. Six different image features are extracted for each example. In our experiments, we choose morphological features with 6 dimensions and pixel features with 240 dimensions as two views.
- COIL20 [39]: It consists of 20 classes of object image and each class has 72 images. In our experiment, we choose three different image features, i.e., intensity feature, local binary patterns feature and Gabor feature as three views. The dimensions of three views are 1024, 3304 and 6750, respectively.
- One-hundred plant species leaves dataset (100Leaves) [40]: It contains one-hundred plant species and each species has sixteen examples. Three views of the leaves are shape descriptor, fine scale margin and texture histogram. The dimensions of three views are 64, 64 and 64, respectively.
- Animal [26]: It consists of 10158 animal images belonging to 50 classes. Each image has two views, which are deep features extracted by DECAF [41] and VGG19 [42] respectively. The dimensions of two views are both 4096.

Table 1 summarizes six multi-view datasets.

BBC and BBCSport are natural incomplete multi-view datasets. It means that some views of examples are originally missing on these two datasets. For Digit, COIL20, 100Leaves and Animal, we randomly remove some instances from each view to create incomplete multi-view data with the missing rates of 10%, 30%, and 50%.

We compare AIMSC with ten methods introduced in Section 1, including MultiNMF, LMVSC, PVC, DAIMC, UEAF, GIMC-FLSD, IMG, SRLC, MKKM-IK, CPM-Nets and the following two methods.

- Best Single View (BSV): BSV performs K-means on each view independently and finally outputs the result of the view with the best clustering performance.
- Concat: Concat concatenates all views into one single view and then performs K-means.

For BSV, Concat, LMVSC and MultiNMF, which can not handle incomplete multi-view data directly, the missing views are filled by the average values of the available instances of corresponding views. As PVC and IMG can only deal with incomplete two view data, they are not conducted on COIL20 and 100Leaves datasets. All methods are performed 10 times and the average values are presented. In addition, the parameter α of AIMSC is set as 10^{-3}. The parameter analysis will be presented in Section 4.4.

4.2. Clustering performance

From Tables 2–11, we show the clustering results of different methods. From these results, we have the following findings.

(1)In most cases, AIMSC achieves varying degrees of superiority on six datasets. For example, compared with other IMC methods, AIMSC improves about 1% to 10% on BBC dataset in terms of ACC. Compared to another anchor-based multi-view clustering approach LMVSC, AIMSC is more competitive on incomplete multi-view datasets. These illustrate that AIMSC can effectively finish the incomplete multi-view clustering task.

(2)In most conditions, IMC methods perform better than the traditional multi-view clustering methods BSV, Concat, MultiNMF and LMVSC. A common operation of these methods is that they fill the missing instances with the average value of available instances of the corresponding view. It may make the data points with missing views be partitioned into the same cluster since they are filled with the same values. Meanwhile,

Table 1
Statistics of the datasets.

Dataset	Views	Clusters	Features	Instances	Domain
BBC	2	5	6838|6790	2225	Text
BBCSport	2	5	3183|3203	737	Text
Digit	2	10	6|240	2000	Image
COIL20	3	20	1024|3304|6750	1440	Image
100Leaves	3	100	64|64|64	1600	Image
Animal	2	50	4096|4096	10158	Image

Table 2
ACC (%), NMI (%), Precision (%) and F-score (%) of different methods on BBC dataset.

Method	ACC	NMI	Precision	F-score
BSV	61.42	50.27	47.18	56.05
Concat	58.27	49.53	45.15	53.88
MultiNMF	48.07	36.91	29.13	41.22
LMVSC	59.28	41.72	31.42	39.56
PVC	84.57	66.74	73.36	74.95
DAIMC	80.56	68.08	71.06	74.42
UEAF	84.64	69.16	71.14	75.16
GIMC-FLSD	90.11	74.91	81.54	**82.45**
IMG	86.19	67.09	73.77	75.33
SRLC	86.74	67.55	75.67	76.75
MKKM-IK	85.69	68.11	74.08	75.61
CPM-Nets	87.31	71.16	76.81	77.04
AIMSC	**91.39**	**77.89**	**84.02**	82.10

Table 3
ACC (%), NMI (%), Precision (%) and F-score (%) of different methods on BBCSports dataset.

Method	ACC	NMI	Precision	F-score
BSV	54.72	35.60	44.68	49.05
Concat	48.60	29.76	37.34	43.18
MultiNMF	52.71	43.57	42.44	49.77
LMVSC	67.03	44.55	46.96	56.12
PVC	72.66	59.52	65.16	67.12
DAIMC	73.09	54.34	66.35	64.59
UEAF	75.27	59.60	61.68	60.87
GIMC-FLSD	**82.80**	68.00	63.40	72.50
IMG	75.37	56.46	65.76	66.74
SRLC	80.57	67.43	74.80	75.75
MKKM-IK	76.49	63.49	64.27	68.41
CPM-Nets	78.56	65.39	69.48	70.71
AIMSC	82.09	**70.62**	**77.36**	**75.99**

Table 4
ACC (%) and NMI (%) of different methods on Digit dataset.

	ACC			NMI		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	66.61	60.15	51.64	65.73	57.66	49.54
Concat	66.90	59.56	51.27	66.18	57.53	49.27
MultiNMF	59.26	54.70	46.75	61.22	53.50	45.81
LMVSC	70.65	61.25	47.50	64.58	60.71	45.45
PVC	54.54	54.63	52.87	60.29	51.53	51.66
DAIMC	64.05	52.20	38.17	58.16	42.95	32.79
UEAF	59.59	47.34	40.26	59.44	44.22	36.56
GIMC-FLSD	67.09	42.45	35.55	68.02	39.04	32.41
IMG	69.30	64.36	55.13	70.45	63.02	54.02
SRLC	73.10	65.30	**57.70**	62.01	56.60	50.14
MKKM-IK	69.18	59.29	53.61	67.99	59.76	50.17
CPM-Nets	72.51	63.18	55.49	69.62	62.16	52.31
AIMSC	**78.02**	**66.73**	54.80	**79.11**	**69.75**	**56.24**

Table 5
Precision (%) and F-score (%) of different methods on Digit dataset.

	Precision			F-score		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	54.02	41.08	28.40	58.15	46.26	34.83
Concat	54.41	41.65	28.28	58.61	46.26	34.64
MultiNMF	46.67	37.37	28.44	51.86	42.28	33.43
LMVSC	41.67	30.67	25.27	49.40	40.62	30.88
PVC	43.22	40.57	40.29	50.91	44.81	45.59
DAIMC	51.42	36.94	23.88	53.96	38.27	26.81
UEAF	47.53	33.18	23.76	52.11	35.51	28.42
GIMC-FLSD	52.98	24.85	16.07	60.60	31.65	23.19
IMG	58.82	48.71	38.42	64.45	55.09	45.71
SRLC	56.26	50.27	**43.27**	56.46	50.66	43.38
MKKM-IK	55.79	41.35	37.38	58.27	45.43	40.52
CPM-Nets	58.21	49.13	41.54	61.84	52.11	44.64
AIMSC	**69.86**	**55.50**	40.61	**74.14**	**61.22**	**46.34**

Table 6
ACC (%) and NMI (%) of different methods on COIL20 dataset.

	ACC			NMI		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	60.61	54.21	55.67	75.48	69.37	69.76
Concat	59.36	58.54	59.91	76.96	75.12	76.19
MultiNMF	67.50	63.94	63.78	77.30	74.46	74.44
LMVSC	71.25	66.74	64.10	80.61	76.39	72.79
DAIMC	63.40	61.67	58.17	76.00	71.49	69.71
UEAF	65.11	64.21	63.26	78.82	77.97	76.79
GIMC-FLSD	70.73	67.75	67.91	80.11	79.07	75.63
SRLC	71.60	69.71	67.57	79.57	78.52	77.02
MKKM-IK	68.14	65.58	63.91	78.26	76.49	74.61
CPM-Nets	71.82	68.03	66.75	80.32	78.21	76.52
AIMSC	**72.14**	**70.18**	**69.92**	**81.93**	**82.15**	**80.48**

Table 7
Precision (%) and F-score (%) of different methods on COIL20 dataset.

	Precision			F-score		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	47.15	35.81	38.31	54.56	43.52	45.31
Concat	50.24	46.20	48.95	56.95	53.56	55.75
MultiNMF	58.07	50.35	50.40	61.10	54.91	55.17
LMVSC	60.67	53.18	47.25	64.16	58.37	52.18
DAIMC	55.54	48.84	46.34	59.83	52.45	50.66
UEAF	57.39	56.10	54.78	62.58	61.35	59.45
GIMC-FLSD	63.51	60.77	59.11	66.62	64.23	61.80
SRLC	63.57	**62.46**	59.38	66.78	64.18	62.55
MKKM-IK	60.58	56.86	54.04	64.60	62.93	60.74
CPM-Nets	64.14	61.50	58.22	67.20	64.31	61.04
AIMSC	**64.55**	59.72	**59.64**	**67.57**	**65.06**	**64.00**

Table 8
ACC (%) and NMI (%) of different methods on 100Leaves dataset.

	ACC			NMI		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	55.50	51.80	47.22	78.23	73.91	69.55
Concat	60.20	52.91	46.87	80.73	75.52	71.67
MultiNMF	60.27	54.21	47.76	82.49	76.90	72.21
LMVSC	61.31	55.50	49.75	80.99	76.47	73.91
DAIMC	62.45	51.17	42.92	82.10	72.53	66.46
UEAF	68.16	58.58	50.81	85.51	78.67	73.52
GIMC-FLSD	67.60	53.63	43.98	**85.69**	78.29	72.44
SRLC	66.03	60.41	58.00	81.39	78.15	76.57
MKKM-IK	66.62	60.15	56.27	82.42	79.20	77.82
CPM-Nets	67.24	59.20	55.42	82.06	77.61	75.36
AIMSC	**69.25**	**63.89**	**58.02**	85.53	**81.74**	**77.91**

Table 9
Precision (%) and F-score (%) of different methods on 100Leaves dataset.

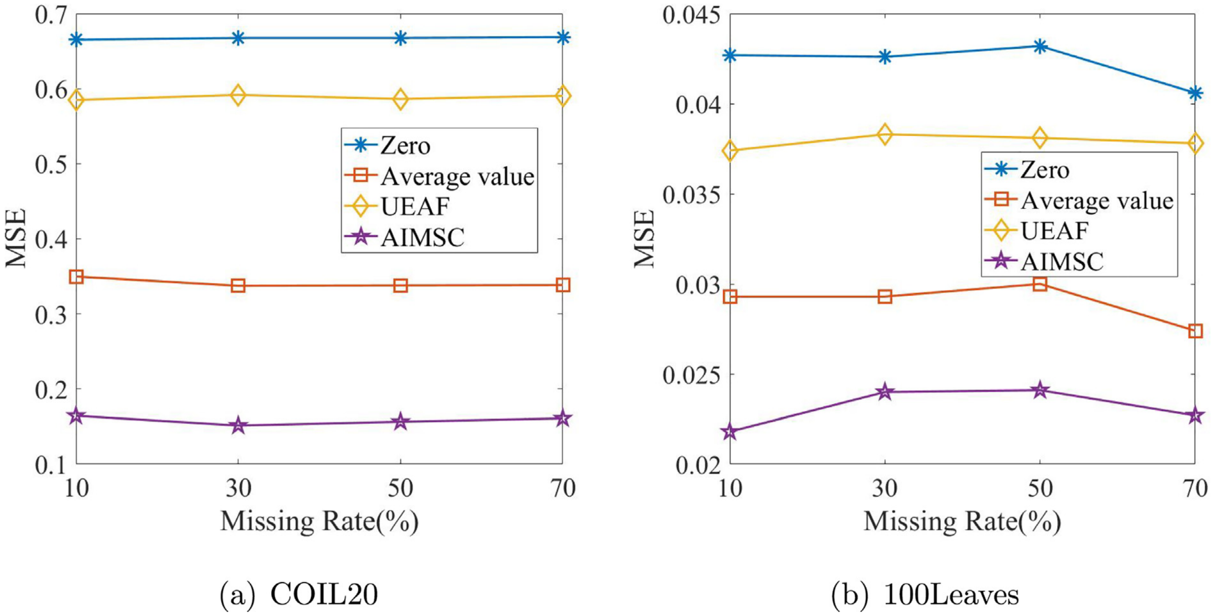
	Precision			F-score		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	36.68	20.35	9.90	42.28	27.81	15.77
Concat	44.53	34.04	27.75	49.05	38.26	31.16
MultiNMF	41.75	31.75	24.15	49.58	38.89	29.99
LMVSC	39.39	32.96	30.51	46.57	38.83	34.55
DAIMC	45.51	20.14	8.69	51.10	27.22	13.81
UEAF	54.07	39.14	29.55	59.13	44.17	33.83
GIMC-FLSD	51.15	35.13	25.27	58.31	42.35	31.10
SRLC	50.42	43.04	40.57	54.54	47.06	44.00
MKKM-IK	54.05	45.70	39.88	57.34	46.85	43.72
CPM-Nets	52.16	41.60	37.52	57.84	45.34	42.45
AIMSC	**56.33**	**49.26**	**41.33**	**60.04**	**52.31**	**44.14**

Table 10
ACC (%) and NMI (%) of different methods on Animal dataset.

	ACC			NMI		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	62.57	55.89	49.75	72.74	66.45	60.52
Concat	57.40	47.01	38.52	67.39	58.67	53.44
MultiNMF	53.14	48.99	40.17	62.95	58.71	49.82
LMVSC	56.02	50.42	43.81	68.32	63.08	58.42
PVC	61.83	58.36	54.81	69.54	64.70	59.69
DAIMC	57.60	46.82	36.36	65.64	54.15	44.38
UEAF	56.41	50.75	46.11	66.82	61.67	57.70
GIMC-FLSD	63.69	57.76	54.87	71.87	**67.82**	60.31
IMG	59.70	57.74	55.67	70.29	66.86	63.78
SRLC	60.15	58.27	54.69	68.72	65.21	61.24
MKKM-IK	59.86	55.73	52.29	68.49	64.88	59.16
CPM-Nets	63.79	56.64	53.85	70.65	63.97	60.92
AIMSC	**65.60**	**59.37**	**56.60**	**73.79**	67.48	**64.30**

Table 11
Precision (%) and F-score (%) of different methods on Animal dataset.

	Precision			F-score		
Method|Missing rate	0.1	0.3	0.5	0.1	0.3	0.5
BSV	52.75	26.44	12.13	53.04	33.30	18.55
Concat	47.25	30.65	24.77	47.23	32.87	26.43
MultiNMF	36.86	25.08	15.47	38.69	29.59	19.76
LMVSC	38.91	25.02	15.76	42.67	30.78	21.53
PVC	55.37	**50.32**	44.91	54.23	48.95	43.72
DAIMC	47.29	20.10	8.16	46.56	25.00	12.44
UEAF	44.40	32.68	27.59	44.60	35.20	30.18
GIMC-FLSD	**56.32**	44.12	44.16	54.73	46.07	43.37
IMG	53.32	49.90	46.44	51.81	48.19	44.76
SRLC	51.47	48.56	43.91	51.14	47.61	43.06
MKKM-IK	51.30	43.12	40.06	50.35	44.60	41.12
CPM-Nets	55.43	46.69	45.50	53.32	47.29	42.58
AIMSC	54.54	49.26	**46.87**	**55.50**	**49.38**	**45.52**



(a) COIL20

(b) 100Leaves

Fig. 1. MSE of different completion methods on COIL20 and 100Leaves datasets.

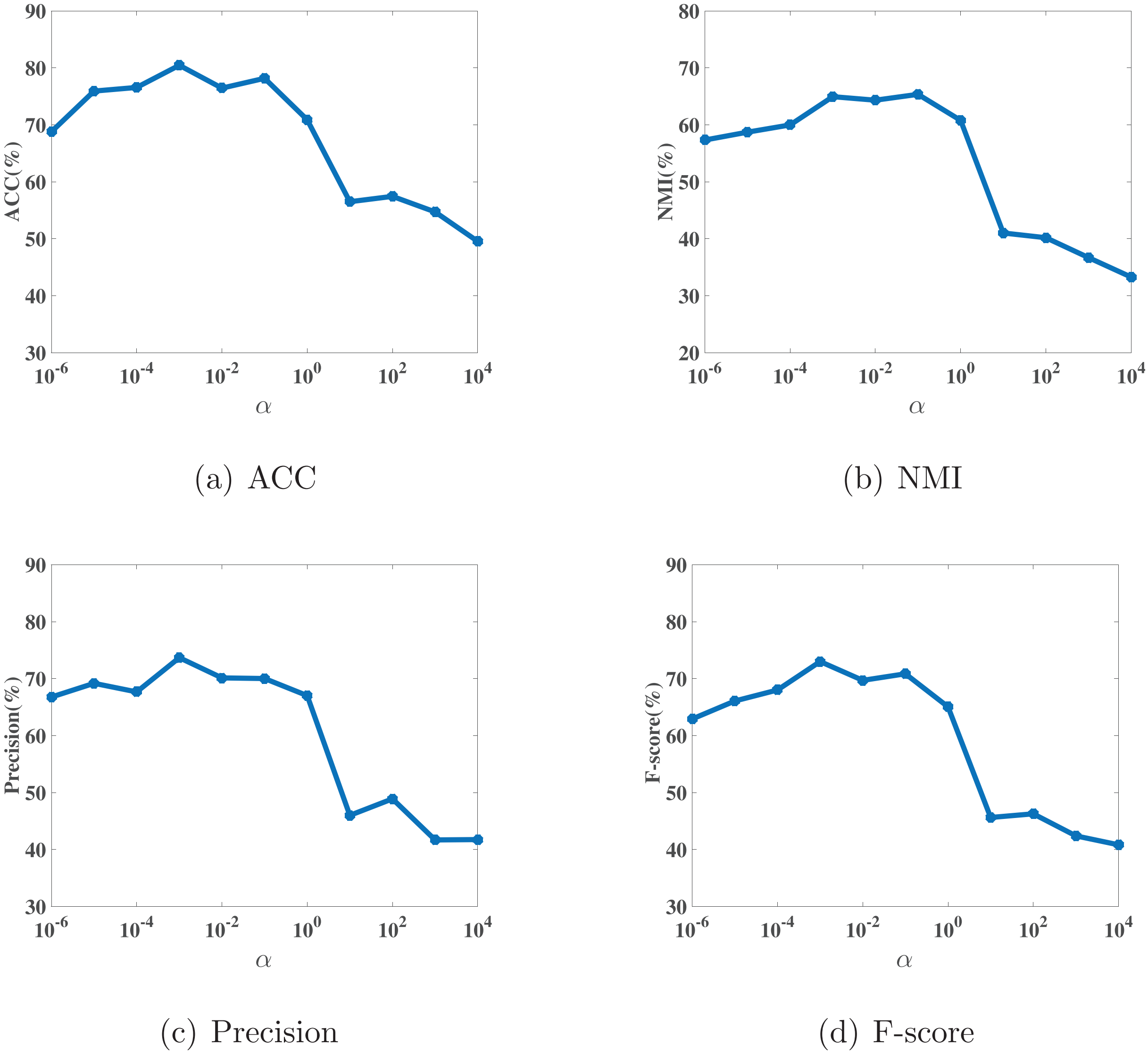
the clustering results are getting worse with the increase of missing rate. Thus, this strategy is inefficient for handling the IMC problem. In contrast, the results show that the strategy of reconstructing missing information in AIMSC can effectively improve the clustering performance.

(3)Compared to PVC, DAIMC, UEAF, GIMC-FLSD and MKKM-IK, the GIMC methods SRLC, IMG and AIMSC generally perform better. It indicates that the GIMC method is a promising strategy to mine the potential relationships of different views of incomplete multi-view data, which can generate more stable clustering results. As a deep learning method, CPM-Nets also has good performances. However, basically, AIMSC performs better than it.

(4)On COIL20 and 100Leaves datasets, AIMSC combines information from three views to generate clustering results without additional adjustments, which demonstrates its capability to extend to more than two views.

4.3. Completion performance

In order to evaluate the completion ability of AIMSC, we calculate the Mean Square Error (MSE) of the reconstructed data and the real data on the COIL20 and 100Leaves datasets. Zero and average value of the existing data are often employed for filling the missing data. UEAF can also recover the missing data. AIMSC is compared with these three completion methods. Fig. 1 shows the MSE of four completion methods with different missing rates. It can be seen from Fig. 1 that, on both two datasets, AIMSC obtains the lowest MSE, no matter which missing rate is adopted. This demonstrates that AIMSC is effective for missing data completion.



(a) ACC

(b) NMI

(c) Precision

(d) F-score

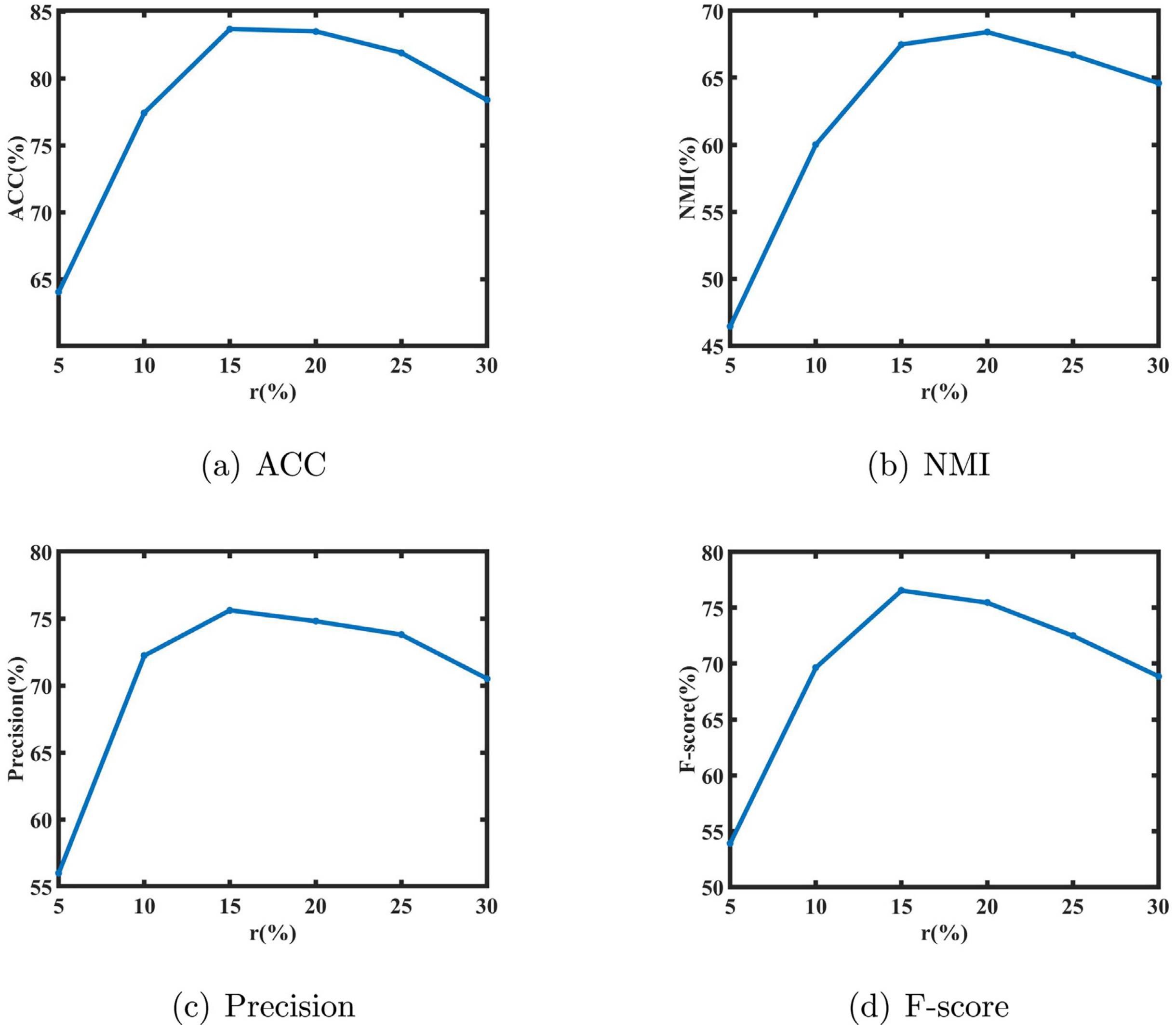
Fig. 2. The performance of AIMSC with different α on BBCSport dataset.

4.4. Parameter sensitivity

In AIMSC, there are two hyper-paramters, i.e. the tradeoff parameter α and the ratio parameter r. In our experiment, we test AIMSC on BBCSport dataset with different hyper-parameters. First, we fix $r = 15$ and let α vary in the range of $[10^{-6}, 10^{4}]$. As shown in Fig. 2, with different α, four clustering evaluation metrics have similar variation trend. When α is bigger than 10^{0} or smaller than 10^{-5} , the evaluation metrics decline rapidly. In addition, the fluctuation is not obvious in the range of $[10^{-4}, 10^{-1}]$, and the better clustering results can be obtained in this range. Then, we fix α= 10^{-3} and let $r$vary in the range of [5,30]. As shown in Fig. 3, for four evaluation metrics, AIMSC all obtain good performances with $r$in the range of [15,20]. Furthermore, the performance of AIMSC is bad when $r$is too small. It is probably caused by that too little anchor points can hardly obtain the exact connection between data points. When $r$is too big, the performance of AIMSC is also degraded. Too many anchor points increase the difficulty of estimating the similarities between data points and anchor points, and may introduce more errors, which weakens the ability of the final similarity matrix S.

4.5. Convergence analysis

The objective function of AIMSC in Eq. (5) is a non-convex problem, which is solved by alternating iteration. The iterative process of AIMSC on six datasets are shown in Fig. 4. Here, the missing rates of Digit, COIL20, 100Leaves and Animal are set as 30%. It can be seen that the objective values of AIMSC decrease fast within the first four iterations and the algorithm converges after about 10 to 20 iterations. Overall, AIMSC does not require too many iterations.



(a) ACC

(b) NMI

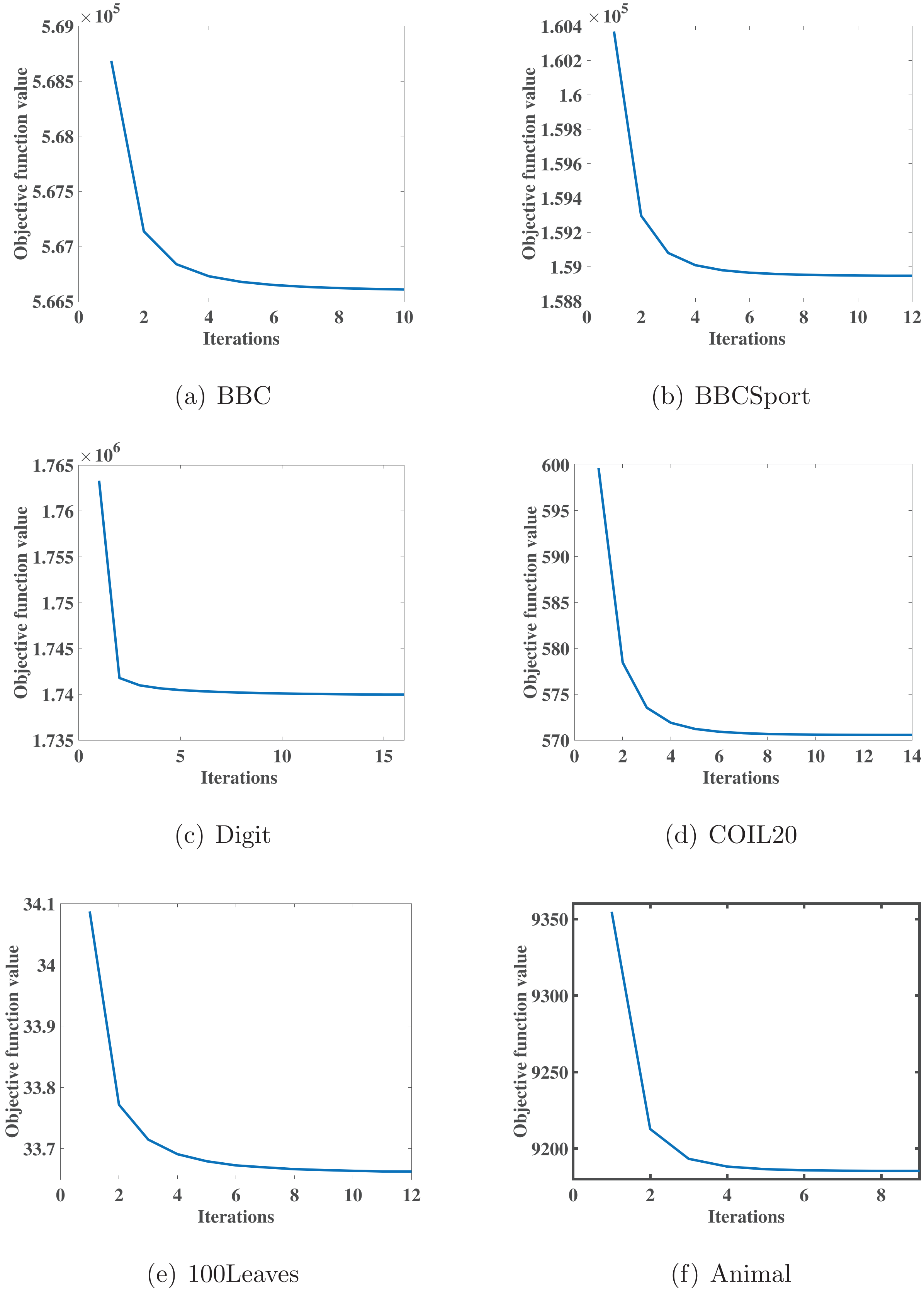
(c) Precision

(d) F-score

Fig. 3. The performance of AIMSC with different *r* on BBCSport dataset.

4.6. Speed test

In AIMSC, anchor-based spectral clustering (ASC) is performed by fast eigenvalue decomposition to improve the computational speed, which has the time complexity $\mathcal{O}(m^3)$. In contrast, traditional spectral clustering (SC) has the time complexity $\mathcal{O}(n^3)$. After deriving *W*, we use two spectral clustering methods separately to calculate the final clustering results and record the time. Table 12 shows the computational time of anchor-based spectral clustering and traditional spectral clustering on six datasets. Here, the missing rates of Digit, COIL20, 100Leaves and Animal are set as 30%. From this table, we can see that anchor-based spectral clustering is faster than traditional spectral clustering on six datasets, especially for the large dataset Animal. On Animal dataset, the computational time of SC is approximately 100 times ASC. This demonstrates that AIMSC can reduce the time complexity of spectral clustering when faced with large-scale datasets.



(a) BBC

(b) BBCSport

(c) Digit

(d) COIL20

(e) 100Leaves

(f) Animal

Fig. 4. The objective values of AIMSC versus the iterations.

Table 12
The computational time (second) of two spectral clustering methods on six datasets.

Method	BBC	BBCSport	Digit	COIL20	100Leaves	Animal
ASC	0.36	0.11	0.33	0.27	1.71	7.15
SC	7.06	0.34	5.47	1.86	3.98	644.93

5. Conclusion and future work

In this paper, we propose a novel method, namely AIMSC, for incomplete multi-view clustering. Specifically, AIMSC utilizes anchor points to connect all instances of each view. Then the missing data is reconstructed and the similarities between all data points are derived from the similarities between anchor points and data points. Finally, anchor-based spectral clustering is performed by fast eigenvalue decomposition which can decrease the computational complexity. Experimental results show that AIMSC is effective for filling the missing data and partitioning incomplete multi-view data. AIMSC selects the real data points with complete views as anchor points. In the future work, we will try to construct the virtual anchor points instead of choosing from real data points. It might yield more suitable anchor points for AIMSC.

CRediT authorship contribution statement

Jun Yin: Conceptualization, Methodology, Software, Writing – review & editing. **Runcheng Cai:** Data curation, Writing – original draft. **Shiliang Sun:** Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is supported by National Natural Science Foundation of China (Grants No. 62076096).

References

[1] R. Vidal, Subspace clustering, IEEE Signal Process. Mag. 28 (2) (2011) 52–68.
[2] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.
[3] H.-P. Kriegel, P. Kröger, J. Sander, A. Zimek, Density-based clustering, Wiley Interdiscip. Rev.: Data Min. Knowl. Disc. 1 (3) (2011) 231–240.
[4] R.J. Campello, D. Moulavi, J. Sander, Density-based clustering based on hierarchical density estimates, in: Pacific-Asia conference on knowledge discovery and data mining (PAKDD), 2013, pp. 160–172.
[5] D. Yan, L. Huang, M.I. Jordan, Fast approximate spectral clustering, in: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 2009, pp. 907–916.
[6] U. Von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
[7] Y. Yang, H. Wang, Multi-view clustering: A survey, Big Data Min. Anal. 1 (2) (2018) 83–107.
[8] J. Yin, S. Sun, Multiview uncorrelated locality preserving projection, IEEE Trans. Neural Networks Learn. Syst. 31 (9) (2020) 3442–3455.
[9] P. Zhao, Y. Jiang, Z.-H. Zhou, Multi-view matrix completion for clustering with side information, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), 2017, pp. 403–415.
[10] J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in: Proceedings of the Society for Industrial and Applied Mathematics International Conference on Data Mining(SIAM), 2013, pp. 252–260.
[11] H. Yu, J. Xiong, X. Zhang, Multi-view clustering by exploring complex mapping relationship between views, Pattern Recogn. Lett. 138 (2020) 230–236.
[12] J. Xu, Y. Ren, G. Li, L. Pan, C. Zhu, Z. Xu, Deep embedded multi-view clustering with collaborative training, Inf. Sci. 573 (2021) 279–290.
[13] J. Liu, S. Teng, L. Fei, W. Zhang, X. Fang, Z. Zhang, N. Wu, A novel consensus learning approach to incomplete multi-view clustering, Pattern Recogn. 115 (2021).
[14] P. Rai, A. Trivedi, H. Daumé III, S.L. DuVall, Multiview clustering with incomplete views, in: Proceedings of the Neural Information Processing Systems Workshop on Machine Learning for Social Computing, 2010, pp. 1–7.
[15] X. Liu, X. Zhu, M. Li, L. Wang, E. Zhu, T. Liu, M. Kloft, D. Shen, J. Yin, W. Gao, Multiple kernel k-means with incomplete kernels, IEEE Trans. Pattern Anal. Mach. Intell. 42 (5) (2019) 1191–1204.
[16] Z. Ren, Q. Sun, Simultaneous global and local graph structure preserving for multiple kernel clustering, IEEE Trans. Neural Networks Learn. Syst. 32 (5) (2021) 1839–1851.
[17] Z. Ren, S.X. Yang, Q. Sun, T. Wang, Consensus affinity graph learning for multiple kernel clustering, IEEE Trans. Cybern. 51 (6) (2021) 3273–3284.
[18] S. Li, Y. Jiang, Z. Zhou, Partial multi-view clustering, in: Proceedings of the AAAI conference on Artificial Intelligence(AAAI), 2014, pp. 1968–1974.
[19] J. Wen, Z. Zhang, Y. Xu, B. Zhang, L. Fei, H. Liu, Unified embedding alignment with missing views inferring for incomplete multi-view clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2019, pp. 5393–5400.
[20] M. Hu, S. Chen, Doubly aligned incomplete multi-view clustering, International Joint Conference on Artificial Intelligence(IJCAI) (2018) 2262–2268.
[21] J. Wen, Z. Zhang, Z. Zhang, L. Fei, M. Wang, Generalized incomplete multiview clustering with flexible locality structure diffusion, IEEE Trans. Cybern. 51 (1) (2021) 101–114.
[22] H. Zhao, H. Liu, Y. Fu, Incomplete multi-modal visual data grouping, International Joint Conference on Artificial Intelligence(IJCAI) (2016) 2392–2398.
[23] W. Zhuge, C. Hou, X. Liu, H. Tao, D. Yi, Simultaneous representation learning and clustering for incomplete multi-view data, in: International Joint Conference on Artificial Intelligence (IJCAI), 2019, pp. 4482–4488.
[24] J. Yin, S. Sun, Incomplete multi-view clustering with cosine similarity, Pattern Recogn. 123 (2022).
[25] J. Wen, K. Yan, Z. Zhang, Y. Xu, J. Wang, L. Fei, B. Zhang, Adaptive graph completion based incomplete multi-view clustering, IEEE Trans. Multimedia 23 (2021) 2493–2504.
[26] C. Zhang, Y. Cui, Z. Han, J.T. Zhou, H. Fu, Q. Hu, Deep partial multi-view learning, IEEE Trans. Pattern Anal. Mach. Intell. 44 (5) (2022) 2402–2415.
[27] Z. Xue, J. Du, C. Zheng, J. Song, W. Ren, M. Liang, Clustering-induced adaptive structure enhancing network for incomplete multi-view data, in: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI), 2021, pp. 3235–3241.
[28] Y. Wang, D. Wang, W. Pang, C. Miao, A.-H. Tan, Y. Zhou, A systematic density-based clustering method using anchor points, Neurocomputing 400 (2020) 352–370.
[29] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, Z. Xu, Large-scale multi-view subspace clustering in linear time, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2020, pp. 4412–4419.
[30] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[31] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: Analysis and an algorithm, in: Advances in Neural Information Processing Systems (NIPS), 2002, pp. 849–856.
[32] W. Liu, J. He, S.-F. Chang, Large graph construction for scalable semi-supervised learning, in: International Conference on Machine Learning (ICML), 2010, pp. 679–686.
[33] W. Liu, J. Wang, S. Kumar, S.-F. Chang, Hashing with graphs, in: International Conference on Machine Learning (ICML), 2011, pp. 1–8.
[34] X. Chen, D. Cai, Large scale spectral clustering with landmark-based representation, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2011, pp. 313–318.
[35] H. Lee, A. Battle, R. Raina, A.Y. Ng, Efficient sparse coding algorithms, in: Advances in neural information processing systems, 2007, pp. 801–808.
[36] K. Zhan, F. Nie, J. Wang, Y. Yang, Multiview consensus graph clustering, IEEE Trans. Image Process. 28 (3) (2018) 1261–1270.
[37] D. Greene, P. Cunningham, Practical solutions to the problem of diagonal dominance in kernel document clustering, in: International Conference on Machine Learning (ICML), 2006, pp. 377–384.
[38] M. van Breukelen, R.P. Duin, D.M. Tax, J. Den Hartog, Handwritten digit recognition by combined classifiers, Kybernetika 34 (4) (1998) 381–386.
[39] S. Nene, S. Nayar, H. Murase, et al., Columbia object image library (coil-20) (1996).
[40] C. Mallah, J. Cope, J. Orwell, et al., Plant leaf classification using probabilistic integration of shape, texture and margin features, Signal Process. Pattern Recogn. Appl. 5 (1) (2013) 45–54.

[41] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.

[42] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations (ICLR), 2015.

Jun Yin received the B.S. degree in Mathematics and the Ph.D. degree in Pattern Recognition and Intelligence System from Nanjing University of Science and Technology, Nanjing, China in 2006 and 2011 respectively. He is an associate professor at the College of Information Engineering, Shanghai Maritime University. From 2016 to 2020, he was a postdoctoral fellow at the School of Computer Science and Technology, East China Normal University. From 2019 to 2020, he was a visiting scholar with the Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, USA. His research interests include multi-view learning, dimension reduction, manifold learning, etc. Dr. Yin is on the Editorial Board of the journal of Neural Processing Letters.

Runcheng Cai received the B.S. degree in Electronic Information Science and Technology from Dezhou University, Shandong, China in 2019. He is currently pursuing the M.S. degree in computer science at the College of Information Engineering, Shanghai Maritime University, Shanghai, China. His current research interest is multi-view clustering.

Shiliang Sun received the Ph.D. degree in pattern recognition and intelligent systems from Tsinghua University, Beijing, China, in 2007. He is a Professor with the School of Computer Science and Technology and the Head of the Pattern Recognition and Machine Learning Research Group, East China Normal University, Shanghai, China. From 2009 to 2010, he was a Visiting Researcher with the Department of Computer Science, Centre for Computational Statistics and Machine Learning, University College London, London, U.K. In 2014, he was a Visiting Researcher with the Department of Electrical Engineering, Columbia University, New York, NY, USA. His current research interests include kernel methods, multi-view learning, learning theory, approximate inference, sequential modeling, deep learning, and their applications. His research results have expounded in 100 + publications at peer-reviewed journals and conferences. Prof. Sun is on the Editorial Board of multiple international journals, including Neurocomputing, Pattern Recognition and IEEE Transactions on Neural Networks and Learning Systems.