Neural Computing and Applications (2023) 35:7891–7907
https://doi.org/10.1007/s00521-022-08136-x

ORIGINAL ARTICLE

An algorithm of nonnegative matrix factorization under structure constraints for image clustering

Mengxue Jia[1,2] · Xiangli Li[1,3,4] · Ying Zhang[1,2]

Received: 6 April 2022 / Accepted: 29 November 2022 / Published online: 20 December 2022

Abstract
Nonnegative matrix factorization (NMF) is a crucial method for image clustering. However, NMF may obtain low accurate clustering results because the factorization results contain no data structure information. In this paper, we propose an algorithm of nonnegative matrix factorization under structure constraints (SNMF). The factorization results of SNMF could maintain data global and local structure information simultaneously. In SNMF, the global structure information is captured by the cosine measure under the ℓ_2 norm constraints. Meanwhile, ℓ_2 norm constraints are utilized to get more discriminant data representations. A graph regularization term is employed to maintain the local structure. Effective updating rules are given in this paper. Moreover, the effects of different normalizations on similarities are investigated through experiments. On real datasets, the numerical results confirm the effectiveness of the SNMF.

Keywords Image clustering · Nonnegative matrix factorization · Cosine measure · ℓ_2 norm

1 Introduction

Clustering is a hot topic in machine learning [1], which groups similar data into a same cluster while different data are assigned distinct clusters. With the development of media technology, image clustering is a significant problem and has wide applications such as face recognition [2], image annotation [3], image retrieval [4], and image segmentation [5]. Many different clustering algorithms are applied in image clustering, like the partition-based method [6], the density-based method [7], and the hierarchical method [8]. However, the high dimension of a picture always causes the "curse of dimension" phenomenon. So, in image clustering, the dimension reduction method plays an essential role. Principal component analysis (PCA) [9] and linear discriminant analysis (LDA) [10] are traditional and representative algorithms in dimension reduction. However, the results may contain negative elements in PCA and LDA. This phenomenon weakens their interpretability because the input image data are nonnegative. For this phenomenon, nonnegative matrix factorization (NMF) [11] is a better dimension reduction method. The nonnegative property makes the NMF more comprehensible. NMF and its variants can be roughly divided into two categories.

The first category is unsupervised. Without supervised information, the intrinsic data structure is vital for performance improvements. However, lossing data inherent structure is a drawback of NMF. Many scholars proposed different methods to maintain the original structure in the reduced space. Cai et al. [12] proposed the graph regularized nonnegative matrix factorization (GNMF). In GNMF, if data are near in original space, their representations

✉ Xiangli Li
lixiangli@guet.edu.cn

Mengxue Jia
jmg8029@163.com

Ying Zhang
zhangying751009@163.com

1 School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin 541004, Guangxi, China

2 School of Mathematics and Statistics, Xidian university, Xi'an 710126, Shaanxi, China

3 Guangxi Colleges and Universities Key Laboratory of Data Analysisand Computation, Guilin University of Electronic Technology, Guilin 541004, Guangxi, China

4 Center for Applied Mathematics of Guangxi (GUET), Guilin 541004, Guangxi, China

would be close in the reduced space. Shang et al. [13] developed the graph dual regularization NMF (DNMF). More than the local structure of data space, DNMF maintains the local structure of feature space. Ding et al. [14] developed the convex nonnegative matrix factorization (CNMF). In CNMF, a factor matrix is reconstructed by the convex combinations of data. In other words, in CNMF, the center of a cluster should be represented by data. Hu et al. [15] developed the graph regularized and convex nonnegative matrix factorization (GCNMF). Compared with CNMF, a graph regularized term is employed in GCNMF. Cui et al. [16] proposed the subspace clustering guided convex nonnegative matrix factorization (SCCNMF). Subspace clustering can reconstruct a sample by other samples. The above methods investigate how to maintain the structure in the reduced space in various ways.

Another drawback of NMF is the error measure. The square of Frobenius norm is sensitive to noise and outliers. To address this problem, researchers have adopted different error measures. Kong et al. [17] used the $\ell_{2,1}$ norm ($\ell_{2,1}$NMF). Li et al [18] employed the $\ell_{2,p}$ norm ($\ell_{2,p}$NMF). Compared with the square of Frobenius norm, these two norms are more robust. Except for above methods, different norms can be applied to data representations. Zhang et al. [19] and Xing et al. [20] added a regularized term in the NMF. [19] used the $\ell_{1/2}$ norm on data representations while [20] employed the $\ell_{2,1}$ norm.

The second category is semisupervised. Here is a brief introduction. Because the cost of collecting a little supervised information is affordable, some semisupervised methods are developed. Generally, there are two types of supervised information. The first type is label constraints, and the second type is pairwise constraints. Label constraints mean the data belong to known classes, while the pairwise constraints restrict data relationships. Based on label constraints, Babaee et al. [21] proposed the discriminative nonnegative matrix factorization (DNMF). When data have the same label, DNMF tries to set them on the same axis. Different from [21], Liu et al. [22] proposed the constrained nonnegative matrix factorization (CNMF). CNMF lets samples in the same class to have the same representation. Based on pairwise constraints, Wang et al. [23] proposed the penalized matrix factorization (PMF). PMF adds a penalized term to mitigate the situation that the factorization violates constraints. Yang et al. [24] developed the pairwise constraints guided nonnegative matrix factorization (PCNMF). Must links and cannot links generate two different regularized terms in PCNMF.

In this paper, the main focus is the unsupervised NMF. Above unsupervised methods achieved notable improvements. However, there still exist some drawbacks. The first drawback is that the method of calculating similarities is always basing the Euclidean distance. However, the Euclidean distance is sensitive to outliers and noise. This phenomenon may lead to a low-accurate clustering result. Second, there is no direct way to keep the global similarity in data representations. Keeping the global similarity by neighbors always causes information loss. A new nonnegative matrix factorization under structure constraints (SNMF) is proposed to handle these drawbacks.

The main contributions are concluded as follows:

1. A direct way to keep the global structure similarity in data representations is introduced into SNMF. Meanwhile, combined with a graph regularized term to save the local structure information, SNMF eases the structure information loss.
2. To make data representations more discriminative, the ℓ_2 norm constraints are employed in SNMF. Based on ℓ_2 norm constraints, the dot measure, which is equivalent to cosine measure, is used in data representations to avoid the drawback of Euclidean distance.
3. New updating rules are given, and experiments confirm the effectiveness of updating rules.

The rest of this paper is organized as follows. In Sect. 2, a brief introduction to related work is given. Section 3 introduces the details of SNMF. Section 4 concludes the experiment results to show the effectiveness of SNMF on clustering. Section 5 makes conclusions about this paper.

2 Related work

2.1 NMF algorithm and GNMF algorithm

Lee et al. [11] proposed the nonnegative matrix factorization (NMF). Here is a brief introduction.

The model of NMF in the Euclidean distance is as follows:

$$\min_{U\geq 0, V\geq 0} \|X - UV^T\|_F^2. \tag{1}$$

In (1), $X \in R_+^{m\times n}$ is a dataset. $U \in R_+^{m\times k}$ and $V \in R_+^{n\times k}$ are factorization results, which are called basis matrix and data representations (or encoding matrix). Each row in V represents a sample in the reduced k-dimension space spanned by U. The number k is predetermined, which is the number of clusters expected in most situations.

The updating formulations of (1) are as follows:

$$U_{mk} \leftarrow U_{mk}\frac{(XV)_{mk}}{(UV^TV)_{mk}},$$
$$V_{nk} \leftarrow V_{nk}\frac{(X^TU)_{nk}}{(VU^TU)_{nk}}.$$

Cai et al. [12] pointed out that (1) focused on factorization

results while losing the local structure information. However, the local structure information is vital for clustering. To address this drawback, Cai et al. [12] proposed the graph regularized nonnegative matrix factorization (GNMF). The GNMF model is as follows:

$$\min_{U\geq 0, V\geq 0} \|X - UV^T\|_F^2 + \lambda tr(V^T L V). \tag{2}$$

λ is a balanced parameter. The second term in (2) is called graph regularized term, and the matrix L in this term is the Laplacian matrix. With the help of this graph regularized term, when samples in the original space are near, their representations in the reduced space will be close. The calculation of this Laplacian matrix L is the following equation:

$$L = D - W. \tag{3}$$

In (3), D is a diagonal matrix, and the element $D_{ii} = \sum_{j=1}^{n} W_{ij}$. W is the similarity matrix which is calculated from a graph $G_G = (V_G, E)$. V_G represents the set of samples in the dataset X, and E is the edge set. The formulation for W is as follows.

$$W_{ij} = \begin{cases} 1, & i \in \kappa(j) \ or \ j \in \kappa(i), \\ 0, & otherwise. \end{cases} \tag{4}$$

$\kappa(i)$ is a set, which contains the κ nearest neighbors of sample i and does not contain itself. These κ nearest neighbors are generated by Euclidean distances.

Updating rules of (2) are given as follows:

$$U_{mk} \leftarrow U_{mk} \frac{(XV)_{mk}}{(UV^TV)_{mk}},$$
$$V_{nk} \leftarrow V_{nk} \frac{(X^TU)_{nk} + \lambda(WV)_{nk}}{(VU^TU)_{nk} + \lambda(DV)_{nk}}.$$

2.2 NMF with ℓ_2 norm constraints

The ℓ_2 norm constraint has excellent potential to feature extraction and data representations [25]. Yang et al. [25] proposed an algorithm to solve the following problem:

$$\begin{aligned} \min \quad & \frac{1}{2}\|X - UV^T\|_F^2, \\ \text{subject to} \quad & U \geq 0, \qquad V \geq 0, \\ & \|U_{*i}\|_2^2 = 1, \quad i = 1, 2, 3, \ldots, k. \end{aligned} \tag{5}$$

U_{*i} represents the ith column in U.

Compared with (1), (5) has no other constraints on V. Therefore, the same updating rule is adopted for V. For U, Yang et al. [25] proposed a new updating rule which updated U by columns. Next is a description of updating a column of U [25].

x represents the ith column of U. The partial derivative of x is denoted as a. So, $a = (UV^TV)_{*i} - (XV)_{*i}$. Let $c_1 = (UV^TV)_{*i}$, and $c_2 = (XV)_{*i}$. Therefore, $a = c_1 - c_2$. To update x under the constraint $\|x\|_2^2 = 1$, following auxiliary variables need to be calculated [25].

$$\begin{aligned} T1 &= x^T x c_1^T x c_2 + x^t x c_2^T x c_1 + x^T x x^T c_1 c_1 + x^T x x^T c_2 c_2, \\ T2 &= x^T x c_1^T x c_1 + x^t x c_2^T x c_2 + x^T x x^T c_1 c_2 + x^T x x^T c_2 c_1, \\ P1 &= c_1^T x x^T c_2 x + c_2^T x x^T c_1 x + x^T x c_1^T c_1 x + x^T x c_2^T c_2 x, \\ P2 &= c_1^T x x^T c_1 x + c_2^T x x^T c_2 x + x^T x c_1^T c_2 x + x^T x c_2^T c_1 x, \\ M &= (a^T x)^2 - \|a\|_2^2 \|x\|_2^2, \\ B &= 2T1 + 2P1 + Mx, \\ C &= 4(x^T x c_1 + x^T c_2 x), \\ D &= -4x. \end{aligned} \tag{6}$$

Then, calculate τ and q.

$$\tau = min\left\{ \frac{\sqrt{C_i^2 - 4B_iD_i} - C_i}{2B_i}, \quad if \quad B_i \neq 0; \right.$$
$$\left. -\frac{D_i}{C_i}, \quad if \quad B_i = 0. \right\}, \quad i = 1, 2, 3, \ldots, m, \tag{7}$$

$$q = 1 - \left(\frac{\tau}{2}\right)^2 (a^T x)^2 + \left(\frac{\tau}{2}\right)^2 \|a\|_2^2 \|x\|_2^2. \tag{8}$$

Now the updating rule of the ith column of U is (9).

$$\begin{aligned} U_{*i} \leftarrow x &- \frac{\tau}{q}\left(x^T x c_1 + \frac{\tau}{2} T1 + x^T c_2 x + \frac{\tau}{2} P1\right) \\ &+ \frac{\tau}{q}\left(x^T x c_2 + \frac{\tau}{2} T2 + x^T c_1 x + \frac{\tau}{2} P2\right). \end{aligned} \tag{9}$$

There is a subtraction operation in (9). However, the theoretical analysis in [25] guarantees results' nonnegativity.

2.3 Cosine similarity

Suppose x_i and $x_j \in R^{m\times 1}$ are two samples. The cosine similarity between x_i and x_j is following:

$$cos(x_i, x_j) = \frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2}. \tag{10}$$

If $\|x_i\|_2 = 1$ and $\|x_j\|_2 = 1$, (10) can be reduced to (11).

$$cos(x_i, x_j) = x_i^T x_j. \tag{11}$$

Given a dataset $A \in R^{m\times n}$, each column $A_{*i}(i \in 1, 2, \ldots, n)$ represents a sample. If $\|A_{*i}\|_2 = 1(i \in 1, 2, \ldots, n)$, (12) saves cosine similarities between samples.

$$B = A^T A. \tag{12}$$

Clearly, B is a symmetric matrix, and B_{ij} is the cosine similarity between ith sample and jth sample. The matrix B is called cosine matrix of A.

3 Nonnegative matrix factorization under structure constraints

First, some symbols are given in Table 1.

3.1 Saving the global structure information in data representations

For dataset $X \in R_{+}^{m\times n}$, each column represents a sample. Normalization, which is used to eliminate some differences between samples, is a necessary operation in machine learning. Different normalizations have various influences on similarity, and the impacts of normalizations will be investigated in Sect. 4.3.

X^N represents the data after a normalization. The local similarity always is reflected by neighbors in (4) which discards similarities with other samples. This operation may cause information loss.

Define the global similarity matrix measured by the dot measure as follows:

$$G = X^{N^T} X^N. \tag{13}$$

G_{ij} reflects the global similarity between samples i and j, and a great value means a high similarity.

For data representations, the ℓ_2 norm could make them more discriminative [25]. Thus, the ℓ_2 norm constraint is added on the row of V. Meanwhile, the global structure should be kept in the coding matrix V. Thus, the same global similarity calculation is adopted on the coding matrix V. Because of ℓ_2 norm constraints, the dot measure on V could be seen as the cosine measure. Then a global structure regularized term is proposed as follows:

$$\begin{aligned} \min \quad & \|G - VV^T\|_F^2, \\ \text{subject to} \quad & \|V_{i*}\|_2^2 = 1, \quad i = 1, 2, 3\ldots, n. \end{aligned} \tag{14}$$

Although the objective function of (14) has a same form as [26] and [27], the purpose of (14) is different from these two works. Compared with [26], (14) has a more precise and distinct sense. Equation (14) represents that the global similarity matrices, which are based on dot measures, of original space and reduced space should be similar. In [26], G only captures the local information, and the purpose of VV^T does not have a clear explanation. In [27], G is a similarity matrix, while V is the cluster index matrix. Clustering results are determined by the biggest value of each row in V. However, in (14), V is a coding matrix, which is related to the factorization. Moreover, the ℓ_2 norm constraints of V in (14) do not exist in [26, 27].

The conventional method of exploring the structure information is always through neighbors of each sample. To find neighbors, calculating similarities and ranking similarities are all necessary. It is an indirect method to save similarities. Selecting neighbors will discard some similarities. Different from the conventional method, Eq. (14) is a direct way to keep the global structure. Equation (14) requires that the global similarity of data representations should reflect the original global similarity directly. That operation will capture the global structure information more effectively. In addition, compared with the conventional method, (13) reduces calculations because (13) does not need the similarity ranking. By minimizing this term, the global similarity in the reduced space will be approximately equal to the original global similarity.

Table 1 Definitions of symbols

Symbol	Dimension	Definition
A_{*i}	–	The ith column of matrix A
A_{j*}	–	The jth row of matrix A
A_{ij}	1	The element on the ith row and jth column of matrix A
X	$m \times n$	The input nonnegative data matrix
X^N	$m \times n$	The nonnegative matrix after normalization
U	$m \times k$	The output nonnegative matrix, basis matrix
V	$n \times k$	The output nonnegative matrix, encoding matrix
k	1	The predetermined number of clusters
G	$n \times n$	The global similarity matrix
L	$n \times n$	The Laplacian matrix
W	$n \times n$	The similarity matrix
D	$n \times n$	Diagonal matrix generated by W

3.2 Proposed method

This subsection describes details of the nonnegative matrix factorization under structure constraints (SNMF).

X is a given nonnegative dataset. After normalization, X^N is the dataset. Then calculate G and L on X^N. G is the global similarity matrix which is calculated by (13). L is the Laplacian matrix which stores local structure information, and is calculated by (3).

As paper [12], minimizing the following term will keep the local information in the reduced space.

$$\begin{aligned}\frac{1}{2}\sum_{i,j}^{n}\|V_{i*}-V_{j*}\|^2W_{ij}&=\sum_{i}^{n}V_{i*}V_{j*}^TD_{ii}-\sum_{i,j}^{n}V_{i*}V_{j*}^TW_{ij}\\&=tr(V^TLV).\end{aligned}\tag{15}$$

The reason is that the value of $\|V_{i*}-V_{j*}\|^2$ should be small when W_{ij} is 1. Thus, (15) requires data representations to be similar if the samples are neighbors in the original space. As the popular way [12], κ is set as 5 for W.

Therefore, to keep the global and local structure in the reduced space, the model of SNMF is as follows:

$$\begin{aligned}\min\quad&\frac{1}{2}\|X^N-UV^T\|_F^2+\frac{\alpha}{2}\|G-VV^T\|_F^2+\frac{\beta}{2}tr(V^TLV)\\\text{subject to}\quad&U\geq 0,\qquad V\geq 0,\\&\|V_{i*}\|_2^2=1,i=1,2,3\ldots,n.\end{aligned}\tag{16}$$

In (16), the global structure regularized term stores the global similarities information. Meanwhile, the graph regularized term is employed to keep the local structure in data representations V. ℓ_2 norm constraints are utilized to make data representations more discriminative. α and β are balanced parameters. α is related to the global data structure, and β has a relation to local data structure. These two regularized terms make SNMF to keep more data intrinsic structure information in the reduced space. They are beneficial to the clustering performance. Section 4.4 displays influences of α and β.

Different from (5), (16) pursues discriminative data representations. Meanwhile, the original structure information is maintained in the reduced space by these two regularized terms on V.

3.3 Optimization on (16)

For (16), an iterative algorithm is given to solve it. The Lagrange function of model (16) is as follows.

$$\begin{aligned}O=&\frac{1}{2}\|X^N-UV^T\|_F^2+\frac{\alpha}{2}\|G-VV^T\|_F^2\\&+\frac{\beta}{2}tr(V^TLV)+tr(\Psi U^T)+tr(\Phi V^T)\\&+\sum_{i=1}^{n}\gamma_i(V_{i*}V_{i*}^T-1)\\=&\frac{1}{2}tr(X^{NT}X^N-X^{NT}UV^T-VU^TX^N+VU^TUV^T)\\&+\frac{\alpha}{2}tr(G^TG-G^TVV^T-VV^TG+VV^TVV^T)\\&+\frac{\beta}{2}tr(V^TLV)+tr(\Psi U^T)+tr(\Phi V^T)\\&+\sum_{i=1}^{n}\gamma_i(V_{i*}V_{i*}^T-1)\\=&\frac{1}{2}(tr(X^{NT}X^N)-tr(X^{NT}UV^T)-tr(VU^TX^N)\\&+tr(VU^TUV^T))\\&+\frac{\alpha}{2}(tr(G^TG)-tr(G^TVV^T)-tr(VV^TG)\\&+tr(VV^TVV^T))\\&+\frac{\beta}{2}tr(V^TLV)+tr(\Psi U^T)+tr(\Phi V^T)\\&+\sum_{i=1}^{n}\gamma_i(V_{i*}V_{i*}^T-1)\\=&\frac{1}{2}(tr(X^{NT}X^N)-2tr(VU^TX^N)+tr(VU^TUV^T))\\&+\frac{\alpha}{2}(tr(G^TG)-2tr(G^TVV^T)+tr(VV^TVV^T))\\&+\frac{\beta}{2}tr(V^TLV)+tr(\Psi U^T)+tr(\Phi V^T)\\&+\sum_{i=1}^{n}\gamma_i(V_{i*}V_{i*}^T-1).\end{aligned}\tag{17}$$

In Eq. (17), Ψ_{mk}, Φ_{kn}, and γ_i are the Lagrange multipliers for $U_{mk}\geq 0$, $V_{kn}\geq 0$, and $\|V_{i*}\|_2^2=1$, respectively. $\|A\|_F^2=tr(A^TA)$, $tr(AB)=tr(BA)$ and $tr(A)=tr(A^T)$ are used in (17).

For O, the partial derivative respect to U_{mk} is as follows:

$$\frac{\partial O}{\partial U_{mk}}=-(X^NV)_{mk}+(UV^TV)_{mk}+\Psi_{mk}.$$

By the KKT condition $\Psi_{mk}U_{mk}=0$, when

$$\frac{\partial O}{\partial U_{mk}}=0,$$

we get the following equation:

$$(-X^N V + UV^T V)_{mk} U_{mk} = 0.$$

Then the updating rule for U is given.

$$U_{mk} \leftarrow U_{mk} \frac{(X^N V)_{mk}}{(UV^T V)_{mk}}, \tag{18}$$

For V, because of the ℓ_2 norm constraints, updating rules are given based Yang et al. [25]. The updating rules for V in (16) are one row by one row. The following are details.

In the tth iteration, x_r represents the ith row of V. In (16), let F represent the objective function and calculate the partial derivative on x_r which is the following.

$$\begin{aligned}\frac{\partial F}{\partial x_r} =& (-X^{N^T} U)_{i*} + (VU^T U)_{i*} - 2\alpha(GV)_{i*} \\ &+ 2\alpha(VV^T V)_{i*} + \beta(DV)_{i*} - \beta(WV)_{i*}.\end{aligned}$$

Note that B, C, D are vectors. So, the B_i, C_i and D_i are the ith element in vectors.

To get the updating rule, a transpose operation should be carried out on the obtained result. So, the updating rule of ith row in V is given as follows:

$$\begin{aligned}V_{i*} \leftarrow & \left(x - \frac{\tau}{q}\left(x^T x c_1 + \frac{\tau}{2} T1 + x^T c_2 x + \frac{\tau}{2} P1 \right) \right. \\ & \left. + \frac{\tau}{q}\left(x^T x c_2 + \frac{\tau}{2} T2 + x^T c_1 x + \frac{\tau}{2} P2 \right) \right)^T.\end{aligned} \tag{19}$$

Although there is a subtraction in Eq. (19), a theoretical analysis in the appendix guarantees the nonnegativity of V. After updating every row in V, the tth iteration finishes.

Algorithm 1 summarizes the SNMF algorithm.

Algorithm 1 SNMF

Require: $X, k, trynumber, \alpha, \beta, \varepsilon$.
Ensure: U, V.

```
Normalization process: normalize X and get X^N;
Calculate G, L. G is calculated by equation (13) while L is calculated by (3);
Initialize U and V randomly. Let V satisfy ℓ2 norm constraints;
t = 0;
while t < trynumber or the value of (16) ≥ ε do
    Update U by equation (18);
    for i=1 to n do
        Update the ith row V by equation (19);
    end for
    t = t + 1;
end while return U, V.
```

Denote

$$h = \frac{\partial F}{\partial x_r}.$$

Let

$$l_1 = (VU^T U)_{i*} + 2\alpha(VV^T V)_{i*} + \beta(DV)_{i*},$$
$$l_2 = (X^{N^T} U)_{i*} + 2\alpha(GV)_{i*} + \beta(WV)_{i*}.$$

Thus, $h = l_1 - l_2$.

Let

$$x = x_r^T, a = h^T, c_1 = l_1^T, c_2 = l_2^T.$$

Then the same method introduced in Sect. 2.2 is used. Calculate the other auxiliary variables by (6), (7) and (8).

3.4 Complexity analysis

The computational complexity of SNMF mainly depends on the updating of U and V. The complexities of updating U and V are $O(mnk^2)$ and $O(n^2k^2 + nk^2)$, respectively. Constructing the graph of X^N needs $O(n^2m)$, and calculating the global similarity matrix G occupies $O(n^2m)$. Therefore, the complexity of SNMF is $O(tmnk^2 + tn^2k^2 + tnk^2 + 2n^2m)$, where t is the number of iterations.

Table 2 Details of datasets

Dataset	Dimension	Classes	The number of samples in each class	The original dimension of each sample
USPS	256 × 2000	10	200	16 × 16
YaleB	1024 × 2242	38	59	32 × 32
ORL	10304 × 400	40	10	112 × 92

4 Experimental results

4.1 Data sets and comparison algorithms

Three image datasets are used to show the effectiveness of the SNMF, which are USPS[1], YaleB[2], and ORL[2] [28]. USPS contains hand-written number pictures, while human face pictures are the samples in YaleB and ORL. For USPS, its original dataset has too many instances. Thus, 200 samples are selected from each class randomly to form the USPS dataset in experiments. Each sample in these datasets is stretched to be a vector. Details of each dataset are shown in Table 2.

Comparison algorithms are:

1. NMF [11]
2. GNMF [12]
3. $\ell_{2,1}$NMF [17]
4. $\ell_{2,p}$NMF [18]
5. SCCNMF-1[16]
6. FR-NMF[29]

For these comparison algorithms, parameters are set according to the author's suggestion.

4.2 Evaluation metrics

Four widely used metrics are adopted to evaluate clustering results: cluster accuracy (ACC) [30], normalized mutual information (NMI) [30], F-measure (F*) [31], and adjusted rand index (ARI) [32].

ACC is calculated by Eq. (20):

$$ACC = \frac{\sum_{i=1}^{n} \delta(map(r_i), l_i)}{n}, \tag{20}$$

where r_i is the clustering label after a clustering algorithm. l_i is the true class label for ith sample. $map(\cdot)$ is a function to map a clustering label into the true class label set. $\delta(a, b)$ is a function that equals 1 when $a = b$ and 0 otherwise. n is the number of samples.

NMI is established on mutual information (MI). MI [30] is calculated by (21).

$$MI(C, C') = \sum_{c_i \in C, c'_j \in C'} p\left(c_i, c'_j\right) \cdot \log_2 \frac{p\left(c_i, c'_j\right)}{p(c_i) \cdot p\left(c'_j\right)}. \tag{21}$$

C denotes the set of clusters obtained from the ground truth and C' is obtained from a clustering algorithm. $p(c_i)$ and $p(c'_j)$ are the probabilities that a sample arbitrarily selected from the dataset belongs to the clusters c_i and c'_j, respectively. $p(c_i, c'_j)$ is the joint probability that the arbitrarily selected sample belongs to the clusters c_i as well as c'_j at the same time. [30].

Then NMI is calculated through Eq. (22).

$$NMI(C, C') = \frac{MI(C, C')}{\max(H(C), H(C'))}. \tag{22}$$

$H(C)$ and $H(C')$ are entropies of C and C' [30].

Let n_c represent the number of samples of the cth cluster. n_c^p represents the number of common samples between the cth cluster and the pth class. F* and ARI are defined as follows.

$$F* = \frac{2 \cdot \frac{n_c^p}{n_c} \cdot \frac{n_c^p}{n_p}}{\frac{n_c^p}{n_c} + \frac{n_c^p}{n_p}}, \tag{23}$$

$$ARI = \frac{\sum_{c,p} \binom{n_c^p}{2} - \sum_c \binom{q_c}{2} \sum_p \binom{s_p}{2} / \binom{n}{2}}{\frac{1}{2}\left(\sum_c \binom{q_c}{2} + \sum_p \binom{q_c}{2}\right) - \sum_c \binom{q_c}{2} \sum_p \binom{s_p}{2} / \binom{2}{2}}, \tag{24}$$

where $\binom{n}{k} = \frac{n!}{k!(n-k)!}$, $q_c = \sum_p n_c^p, s_p = \sum_c n_c^p$.

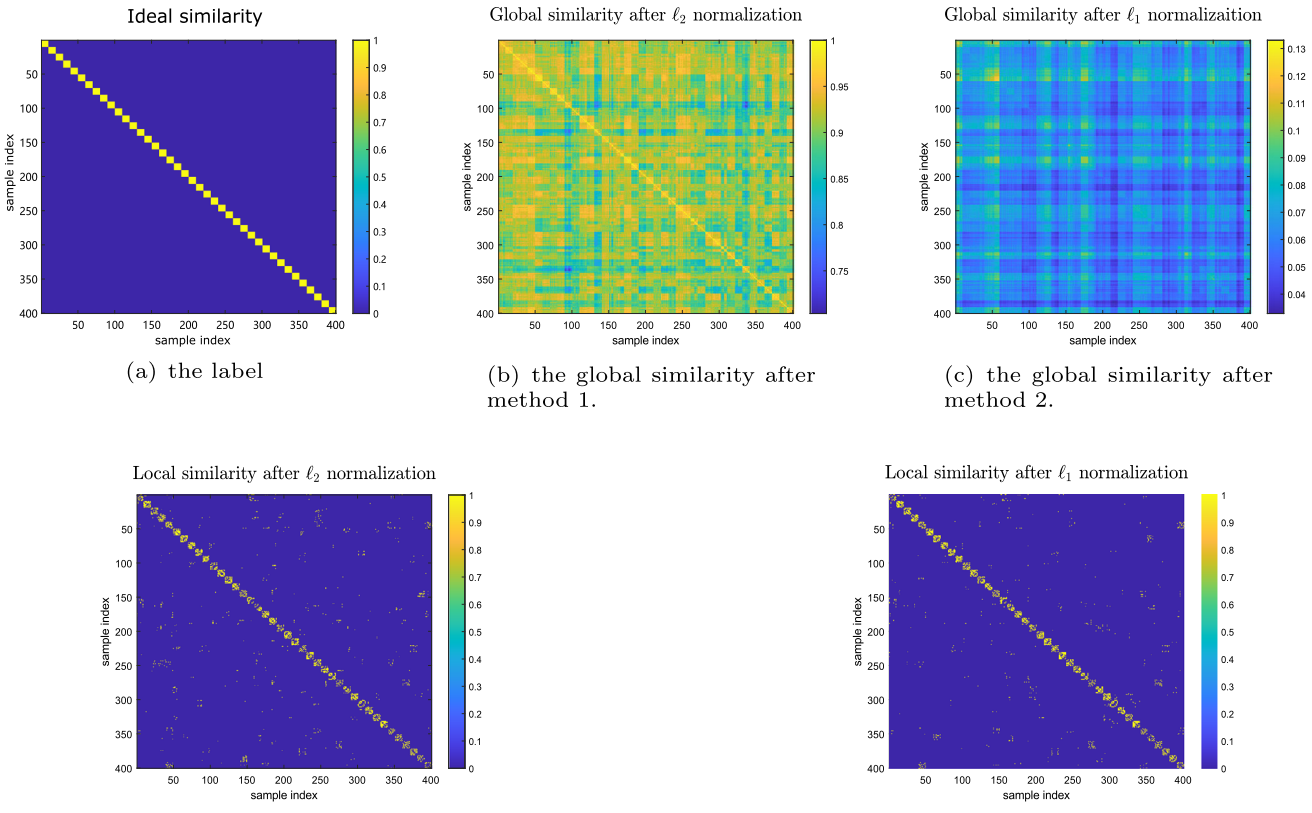
The value range is [0,1] for ACC, NMI, and F* while [−1,1] for ARI. High values of these metrics represent satisfactory clustering results.

4.3 Normalization influence on similarity

Different normalizations will have multiple effects on similarity matrices G and W. To show normalizations' various effects directly, an example is displayed in Fig. 1 on ORL [28].

[1] https://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/multiclass.html#usps.

[2] http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html.



(a) the label

(b) the global similarity after method 1.

(c) the global similarity after method 2.

(d) the local similarity after method 1.

(e) the local similarity after method 2.

Fig. 1 Different normalization methods and their effects on the similarity based ORL. Method 1 represents the ℓ_2 norm of each sample equals 1 and method 2 represents the ℓ_1 norm of each feature equals 1

Table 3 Proportion of same classes similarities in overall similarities

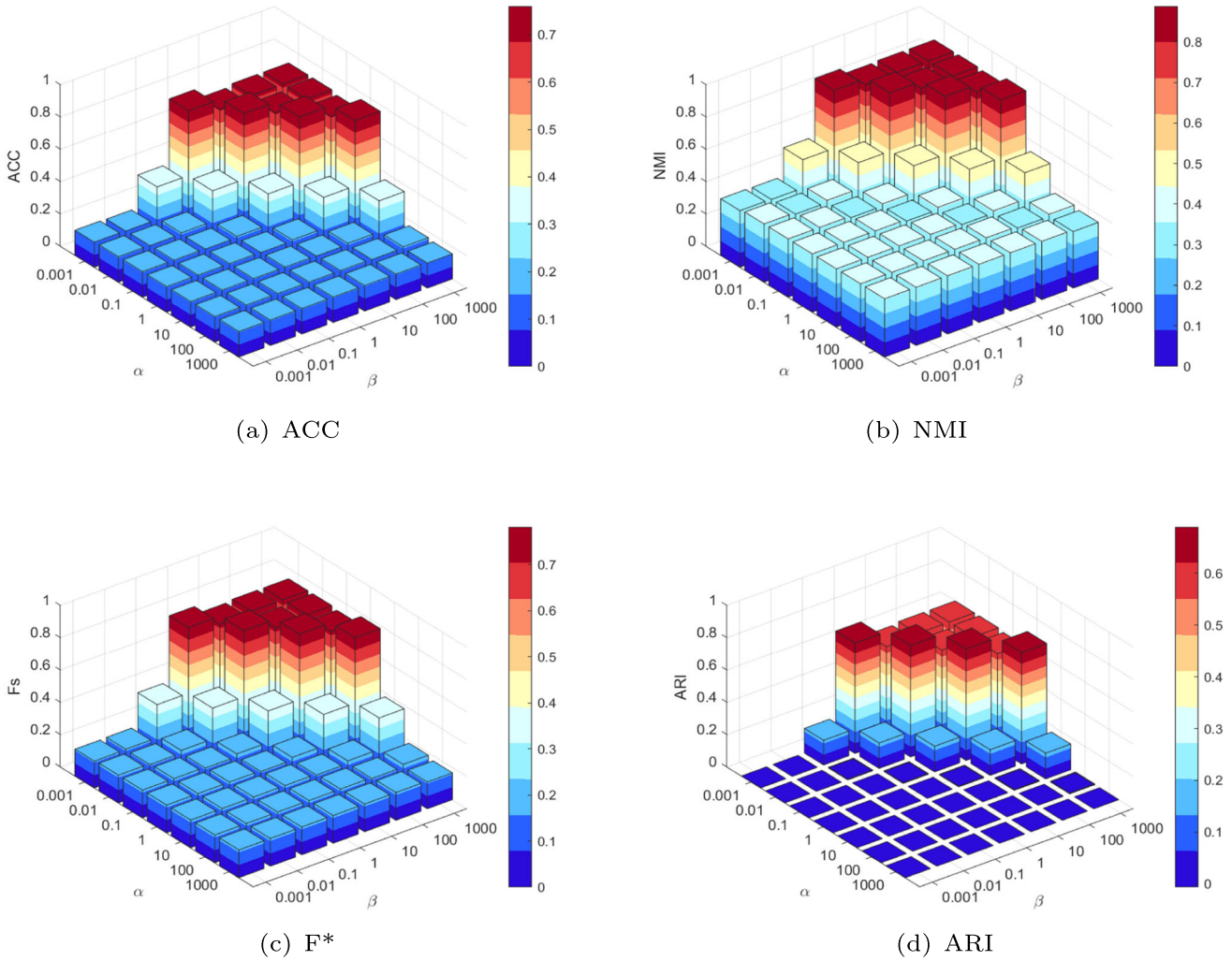
Proportion	Method 1	Method 2
Global similarity	0.0264	0.0272
Local similarity	0.7073	0.7710

Figure 1a is an ideal situation, which is generated by labels. When two samples are in a same class, the similarity for them is 1, and equals 0 otherwise. Basing the samples' indexes, only the submatrices on the main diagonal have values of 1, and each yellow square represents a class.

Two different normalizations are selected. The first normalization, denoted as method 1, will normalize the ℓ_2 norm of samples to be 1, which is widely used. The second normalization, marked as method 2, normalizes the ℓ_1 norm of each feature to be 1. For nonnegative data, these two normalizations will keep data nonnegative. Figure 1b–e displays the effects on global similarity G and local similarity W of these two methods. In an ideal situation, Fig. 1b–e should be similar to Fig. 1a.

It is clear that Fig. 1b–e has differences. From Fig. 1b, method 1 makes the elements on the main diagonal 1. However, method 1 also assigns high similarities between different classes, which is reflected by that too many yellow pixels are not in the submatrices on the main diagonal. In Fig. 1c, method 2 assigns high values on the submatrices on the main diagonal in most situations. Comparing Fig. 1d, e, which reflect the local similarity, we find that pixels in yellow are more concentrated in submatrices on the main diagonal in Fig. 1e. This phenomenon verifies that method 2 has advantages on the local similarity over method 1.

More than Fig. 1, Table 3 displays the proportion of same classes similarities in overall similarities. Values in Table 3 represents the proportion of the sum of similarity values in submatrices on the main diagonal to overall similarities for global similarity G and local similarity W. From Table 3, proportions of method 2 are higher than method 1 in both global similarities and local similarities. This phenomenon means that the correct information takes a high proportion in method 2, and confirms that method 2 is superior to method 1.



(a) ACC (b) NMI

(c) F* (d) ARI

Fig. 2 clustering results (ACC, NMI, F* and ARI) on dataset ORL

From the above analysis, method 2 could reflect the data similarity well, and this normalization is selected as the normalization in this paper.

4.4 Parameter selection

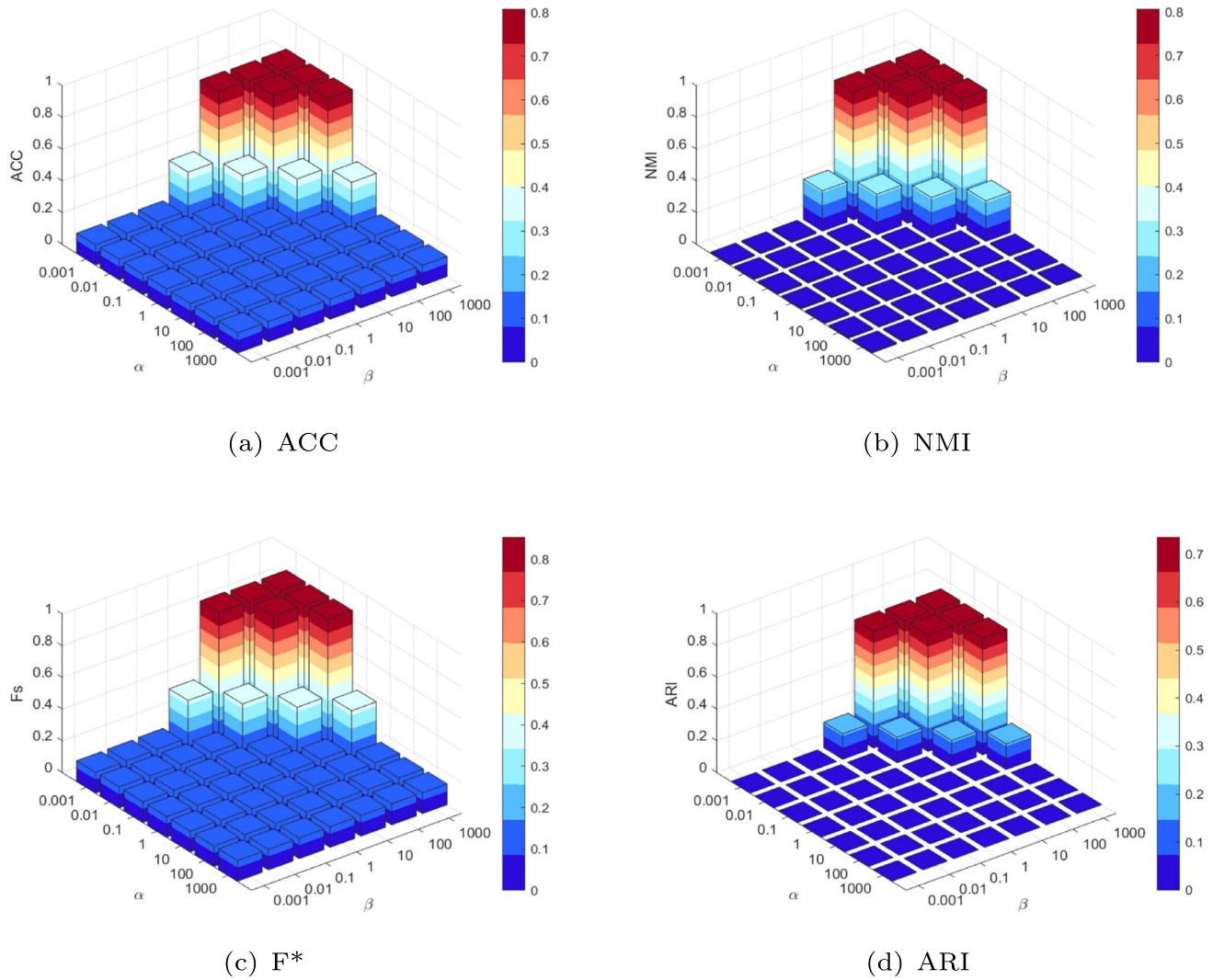
The *trynumber* and ε for SNMF are 200 and 10^{-6}, respectively. The value of k equals the number of classes in the ground truth. Now, only α and β are not determined. Obviously, α and β have influences on clustering results at the same time. Thus, a same changing set [0.001, 0.01, 0.1, 1.0, 10, 100, 1000] is adopted for α and β. Clustering results of tuning α and β are displayed in Figs. 2, 3 and 4.

Figure 2 shows that 4 combinations, which are (1.0, 1000), (0.1, 100), (0.01, 10), (0.001, 1.0), have the satisfactory results on ORL. Basing Figs. 3 and 4, satisfactory combinations are (0.1, 1000), (0.01, 100), (0.001, 10) on USPS and YaleB. The combinations of ORL have poor performance on USPS and YaleB. However, combinations of USPS and YaleB have adequate performance on ORL. So, in the next experiments, we set $\alpha = 0.01$ and $\beta = 100$.

From these results, when the ratio of $\frac{\beta}{\alpha}$ between 1000 and 10,000, the performance of SNMF is satisfactory. When emphasizing local structure too much (which leads to the ratio being too big), SNMF has poor performance on ORL and YaleB. When emphasizing global structure too much (which leads to the ratio being too small), SNMF has worse performance through all datasets. This phenomenon means that the ratio of local structure information and global structure information should be suitable. SNMF only employs local and global structure information effectively when the ratio is proper.

4.5 Experimental results and analysis

The clustering results of SNMF and other comparison algorithms are shown in Table 4. Kmeans is used on data representations to get final clustering results. To eliminate



(a) ACC (b) NMI

(c) F* (d) ARI

Fig. 3 clustering results (ACC, NMI, F* and ARI) on dataset USPS

accidents' impacts, each algorithm is carried out 20 times. Because all these algorithms need random initializations, the same random seed is adopted for each algorithm at each time. Mean values and standard deviations are reported in Table 4. Bold numbers represent the best results. '–' represents that the time of this algorithm is too long to get results. SNMF-A represents $\beta = 0$ in SNMF.

From Table 4, we have the following conclusions.

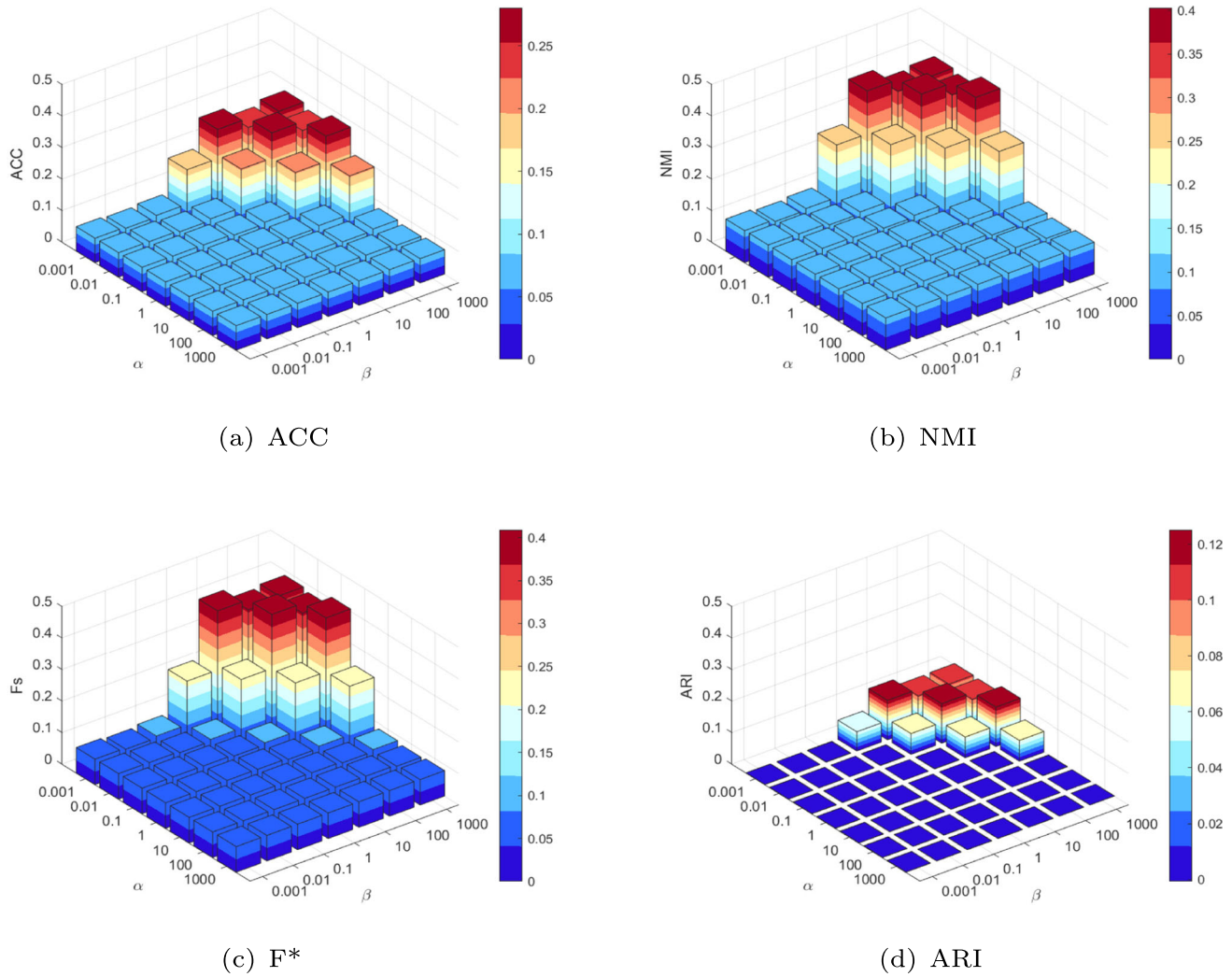
First, SNMF has the best results. These results confirm that data intrinsic structure information plays a crucial role in clustering. Sufficiently using data structure information is beneficial for clustering.

Second, using global structure information solely does not provide enough structure information for clustering. In Table 4, the performance of SNMF-A is the worst. This phenomenon shows that the global structure information is not enough to promote clustering without local structure information.

Third, the performance of SNMF is better than GNMF and SNMF-A. This result means that using local and global structure information simultaneously is beneficial to clustering. Although only using the global structure is not enough to elevate clustering results, there is still some valuable information for clustering. The absence of either of these two kinds of structure information will weaken clustering results, which are confirmed by the performance of NMF, GNMF and SNMF-A.

4.6 Time cost analysis

In this section, the time costs of one iteration for NMF, GNMF, $\ell_{2,1}$NMF, $\ell_{2,p}$NMF, SCCNMF-1, FR-NMF, and SNMF are displayed in Table 5. The notion '–' in Table 5 represents that the time is too long to finish one iteration. From this table, it is clear that time costs of SNMF are not the least. For SNMF, maintaining data structure



(a) ACC (b) NMI

(c) F* (d) ARI

Fig. 4 clustering results (ACC, NMI, F* and ARI) on dataset YaleB

information in the reduced space will augment the computational burden.

In addition to the cost of one iteration, the number of iterations to get a convergence result is a crucial factor for time costs. Figure 5 displays the convergence curves on USPS, and other convergence curves on the remaining datasets have similar tendencies as USPS. Figure 5 shows that almost all methods will be stable after 100 iterations, except FR-NMF. For FR-NMF, it will be stable after 200 iterations. Figure 5 reveals that SNMF does not need more iterations to get a stable result compared with other methods.

The above analysis confirms that SNMF takes a little more time to get a stable result. However, SNMF makes satisfactory promotions compared with the extra time costs. Thus, the time cost of SNMF is acceptable.

4.7 Robustness analysis

To evaluate the robustness of SNMF, Gaussian noise and Poisson noise are added to the data, respectively. For Gaussian noise, the mean and standard deviation are 0 and 0.1, respectively. If some elements become negative after adding Gaussian noise, they are mapped to be 0. Some noisy data under these two different noises are shown in Fig. 6. Clustering results on noisy datasets are shown in Tables 6 and 7.

From Tables 6 and 7, results show that SNMF has the best results on noisy datasets. The performance of SNMF confirms that global and local structure information is beneficial for clustering when noise exists. Only considering local structure information is not enough, which is demonstrated by the performance of GNMF and SNMF.

Table 4 Results (mean(std)) on datasets

Index	Algorithm	USPS	YaleB	ORL
ACC	NMF	0.6635(0.0340)	0.2280(0.0100)	0.6592(0.0191)
	GNMF	0.6681(0.0433)	0.2649(0.0132)	0.6465(0.0234)
	$\ell_{2,1}$NMF	0.6432(0.0214)	0.2279(0.0092)	0.6666(0.0286)
	$\ell_{2,p}$NMF	0.6418(0.0324)	0.2613(0.0120)	0.6476(0.0379)
	SCCNMF-1	0.4696(0.0270)	0.0852(0.0033)	0.6261(0.0344)
	FR-NMF	0.1046(0.0615)	0.0242(0.0065)	–
	SNMF-A	0.1291(0.0024)	0.0750(0.0018)	0.1623(0.0053)
	SNMF	**0.7526(0.0215)**	**0.2939(0.0115)**	**0.703(0.0191)**
NMI	NMF	0.5788(0.0139)	0.3605(0.0106)	0.8222(0.0108)
	GNMF	0.6687(0.0148)	0.4014(0.0181)	0.8068(0.0120)
	$\ell_{2,1}$NMF	0.5728(0.0160)	0.3617(0.0092)	0.8201(0.0138)
	$\ell_{2,p}$NMF	0.5831(0.0222)	0.3907(0.0133)	0.8207(0.0183)
	SCCNMF-1	0.4357(0.0149)	0.1165(0.0056)	0.8039(0.0147)
	FR-NMF	0.5149(0.0343)	0.3657(0.0143)	–
	SNMF-A	0.0087(0.0013)	0.0975(0.0026)	0.3750(0.0093)
	SNMF	**0.7193(0.0143)**	**0.4066(0.0099)**	**0.8560(0.0106)**
F*	NMF	0.6879(0.0240)	0.3384(0.0165)	0.6936(0.0164)
	GNMF	0.715(0.0252)	0.4005(0.0172)	0.6818(0.0191)
	$\ell_{2,1}$NMF	0.6727(0.0181)	0.3408(0.0144)	0.6952(0.0240)
	$\ell_{2,p}$NMF	0.6768(0.0278)	0.3848(0.0185)	0.6835(0.0305)
	SCCNMF-1	0.4919(0.0255)	0.1046(0.0048)	0.6669(0.0292)
	FR-NMF	0.0937(0.0555)	0.0202(0.0063)	–
	SNMF-A	0.1330(0.0033)	0.0.0797(0.0019)	0.1766(0.0066)
	SNMF	**0.7832(0.0183)**	**0.4224(0.0127)**	**0.7374(0.0169)**
ARI	NMF	0.4920(0.0229)	0.1051(0.0098)	0.6170(0.0284)
	GNMF	0.5502(0.0239)	0.1122(0.0126)	0.5242(0.0245)
	$\ell_{2,1}$NMF	0.4739(0.0195)	0.1073(0.0047)	0.5470(0.0341)
	$\ell_{2,p}$NMF	0.4771(0.0349)	0.1223(0.0071)	0.5415(0.0432)
	SCCNMF-1	0.2904(0.0176)	0.0045(0.0014)	0.5110(0.0347)
	FR-NMF	0.4080(0.0469)	0.0904(0.0088)	–
	SNMF-A	0(0.0007)	0(0.0005)	0.0011(0.0026)
	SNMF	**0.6212(0.0245)**	**0.1238(0.0094)**	**0.6167(0.0282)**

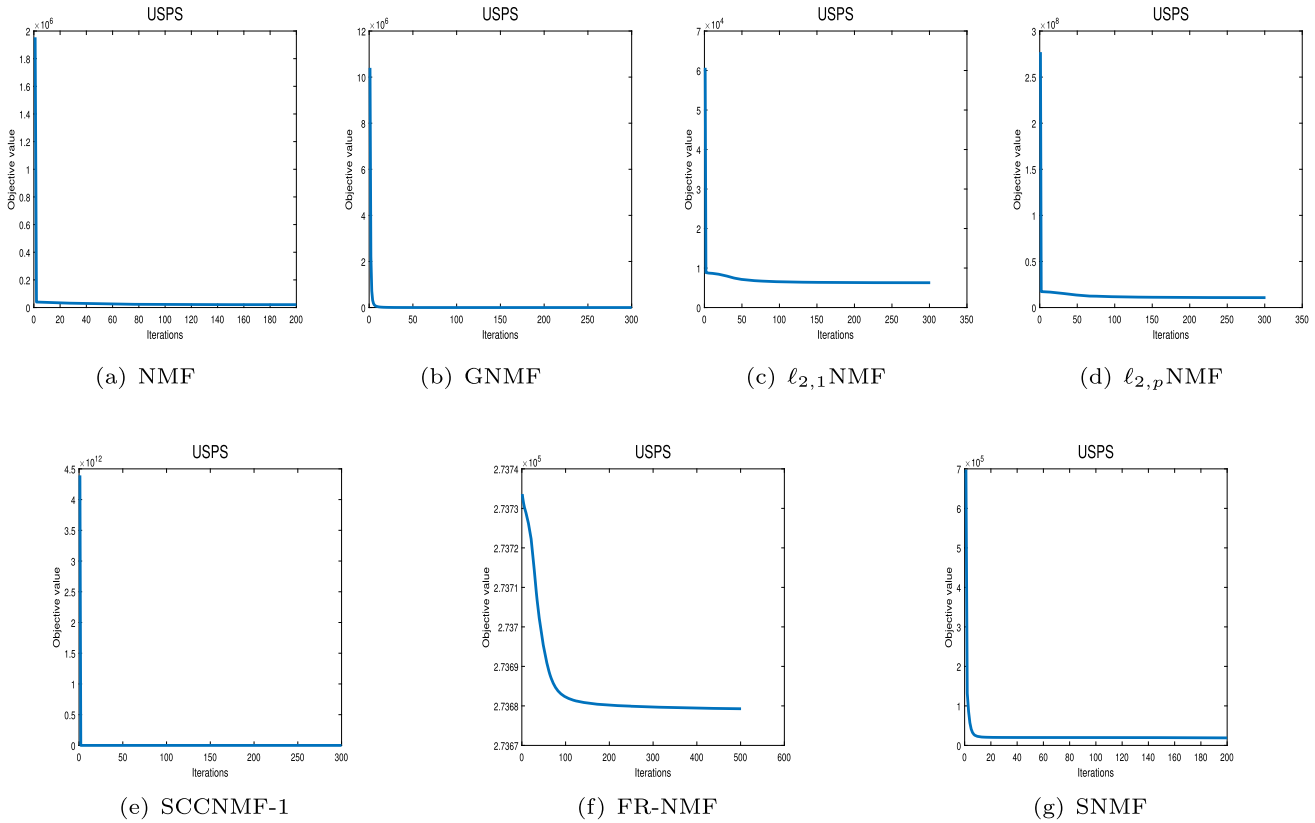
Table 5 Time costs (seconds) of one iteration on three data sets

Time(s)	USPS	YaleB	ORL
NMF	0.0034	0.0184	0.0275
GNMF	0.0018	0.0060	0.0095
$\ell_{2,1}$NMF	0.0417	0.1599	0.1072
$\ell_{2,p}$NMF	0.0441	0.1614	0.1169
SCCNMF-1	1.0333	4.0027	0.3665
FR-NMF	0.0240	0.1780	–
SNMF	0.0279	0.0971	0.0334

5 Conclusion

In this paper, a nonnegative matrix factorization under structure constraints is proposed. Different from previous works, the global structure is considered in SNMF. To accomplish this aim, a new global structure regulation term is presented. The ℓ_2 norm constraints are applied to get discriminative data representations. In addition, SNMF employs a graph Laplacian term to keep the local structure. Effective updating rules, which guarantee SNMF non-increasing, are given. The effects of different normalizations on similarity are investigated by experiments.

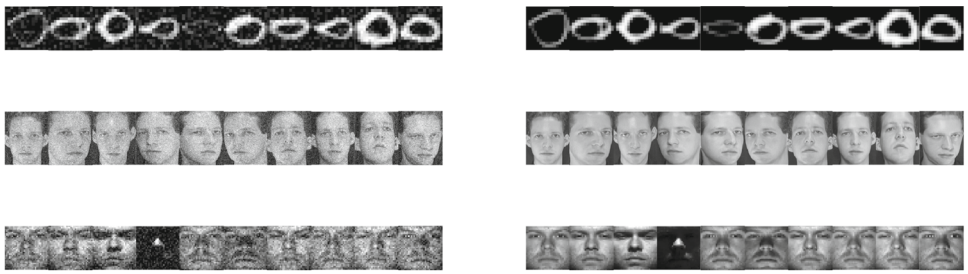
However, there are still some problems. First, capturing the global structure is still a problem. In Table 3, we find that the proportion of correct information on global similarity is low. There should be some better methods for



(a) NMF (b) GNMF (c) $\ell_{2,1}$NMF (d) $\ell_{2,p}$NMF

(e) SCCNMF-1 (f) FR-NMF (g) SNMF

Fig. 5 convergence curves on USPS



(a) Gaussian noise (b) Poisson noise

Fig. 6 Noisy data: first column is USPS dataset, second column is ORL dataset, and third column is YaleB dataset

generating global similarity. Second, there is no relationship between global similarities and local similarities in this work. However, if an algorithm could establish a framework to connect global similarities and local similarities to refine the quality of these similarities, it may be beneficial for clustering. All these problems need deeper investigations.

Appendix

Because (18) and (19) are two gradient descent methods, (16) is non-increasing. Here is the proof.

Denote the objective function of (16) as F. The partial derivative of U_{mk} in F is

$$\frac{\partial F}{\partial U_{mk}} = (-X^N V + UV^T V)_{mk}. \tag{25}$$

The formulation of updating U_{mk} through the gradient descent method is the following:

$$U_{mk} = U_{mk} + \tau_u (X^N V - UV^T V)_{mk}. \tag{26}$$

When $\tau_u = U_{mk}/(UV^T V)_{mk}$, the nonnegative constraints on U hold, and (26) is (18).

For V, the updating rule (19) is also a gradient descent method. The proof is similar to [25].

Table 6 Results (mean(std)) on Gaussian noisy datasets

Gaussian noise				
Index	Algorithm	USPS	YaleB	ORL
ACC	NMF	0.6664(0.0365)	0.2230(0.0106)	0.6588(0.0247)
	GNMF	0.6178(0.0265)	0.1012(0.0061)	0.5639(0.0229)
	$\ell_{2,1}$NMF	0.6303(0.0228)	0.2243(0.0099)	0.6469(0.0310)
	$\ell_{2,p}$NMF	0.6361(0.0330)	0.2532(0.0091)	0.6378(0.0274)
	SCCNMF-1	0.5250(0.0326)	0.0643(0.0025)	0.6528(0.0304)
	FR-NMF	0.0981(0.0660)	0.0246(0.0070)	–
	SNMF	**0.7406(0.0204)**	**0.2788(0.0117)**	**0.7058(0.0206)**
NMI	NMF	0.5766(0.0158)	0.3510(0.0099)	0.8174(0.0090)
	GNMF	0.6217(0.0193)	0.1486(0.0106)	0.7419(0.0135)
	$\ell_{2,1}$NMF	0.5642(0.0130)	0.3549(0.0102)	0.8116(0.0148)
	$\ell_{2,p}$NMF	0.5744(0.0229)	0.3810(0.0109)	0.8079(0.0142)
	SCCNMF-1	0.4754(0.0236)	0.0657(0.0045)	0.8174(0.0154)
	FR-NMF	0.5125(0.0357)	0.3622(0.0072)	–
	SNMF	**0.7048(0.0146)**	**0.3952(0.0112)**	**0.8597(0.0072)**
F*	NMF	0.6889(0.0266)	0.3248(0.0153)	0.6892(0.0195)
	GNMF	0.6711(0.0231)	0.1323(0.0088)	0.6012(0.0209)
	$\ell_{2,1}$NMF	0.6640(0.0218)	0.3337(0.0146)	0.6807(0.0270)
	$\ell_{2,p}$NMF	0.6697(0.0294)	0.3648(0.0136)	0.6745(0.0233)
	SCCNMF-1	0.5467(0.0327)	0.0738(0.0035)	0.6947(0.0233)
	FR-NMF	0.0882(0.0595)	0.0220(0.0074)	–
	SNMF	**0.7723(0.0165)**	**0.4060(0.0157)**	**0.7430(0.0161)**
ARI	NMF	0.4919(0.0243)	0.0991(0.0055)	0.5409(0.0205)
	GNMF	0.5064(0.0330)	0.0141(0.0030)	0.4191(0.0225)
	$\ell_{2,1}$NMF	0.4650(0.0184)	0.1015(0.0049)	0.5243(0.0343)
	$\ell_{2,p}$NMF	0.4696(0.0294)	0.1134(0.0065)	0.5193(0.0306)
	SCCNMF-1	0.3380(0.0302)	-0.0036(0.0008)	0.5342(0.0344)
	FR-NMF	0.4081(0.0502)	0.0946(0.0049)	–
	SNMF	**0.6077(0.0318)**	**0.1133(0.0079)**	**0.6294(0.0192)**

Let x_r and h represent the ith row of V and $\frac{\partial F}{\partial x_r}$, respectively. Denote $F_1(x_r)$ as the related part of x_r in (16). When omitting the nonnegative constraints, the Lagrange function of x_r is the following:

$$L(x_r, \lambda) = F_1(x_r) + \frac{\lambda}{2}(x_r x_r^T - 1). \tag{27}$$

Denote

$$x = x_r^T, a = h^T, c_1 = l_1^T, c_2 = l_2^T.$$

Rewrite (27) as follows:

$$L(x, \lambda) = F_1(x^T) + \frac{\lambda}{2}(x^T x - 1). \tag{28}$$

When$\nabla_x L(x, \lambda) = 0$, we have

$$a - x\lambda = 0.$$

Through the constraint $x^T x = 1$, we obtain

$$\lambda = x^T a = a^T x.$$

Thus, rewrite $\nabla_x L(x, \lambda)$ as follows:

$$\begin{aligned} \nabla_x L(x, \lambda) &= a - x\lambda \\ &= a - x^T a x \\ &= a x^T x - x a^T x \\ &= (a x^T - x^T a)x \end{aligned} \tag{29}$$

Let A represents $ax^T - xa^T$. Therefore, A is a skew-symmetric matrix. Since Ax is the gradient of (28) the updating

Table 7 Results (mean(std)) on Poisson noisy datasets

Poisson noise				
Index	Algorithm	USPS	YaleB	ORL
ACC	NMF	0.6635(0.0340)	0.2281(0.0100)	0.6593(0.0192)
	GNMF	0.6681(0.0433)	0.2649(0.0132)	0.6456(0.0206)
	$\ell_{2,1}$NMF	0.6432(0.0214)	0.2279(0.0093)	0.6666(0.0287)
	$\ell_{2,p}$NMF	0.6418(0.0324)	0.2614(0.0121)	0.6476(0.0379)
	SCCNMF-1	0.5009(0.0290)	0.0694(0.0025)	0.6631(0.0264)
	FR-NMF	0.0830(0.0290)	0.0256(0.0041)	–
	SNMF	**0.7526(0.0215)**	**0.2939(0.0115)**	**0.703(0.0191)**
NMI	NMF	0.5788(0.0139)	0.3605(0.0107)	0.8223(0.0108)
	GNMF	0.6687(0.0148)	0.4014(0.0181)	0.8066(0.0105)
	$\ell_{2,1}$NMF	0.5728(0.0160)	0.3617(0.0093)	0.8202(0.0138)
	$\ell_{2,p}$NMF	0.5831(0.0222)	0.3908(0.0133)	0.8207(0.0183)
	SCCNMF-1	0.4581(0.0213)	0.0786(0.0052)	0.8299(0.0132)
	FR-NMF	0.1587(0.0119)	0.1099(0.0029)	–
	SNMF	**0.7193(0.0143)**	**0.4066(0.0099)**	**0.8560(0.0106)**
F*	NMF	0.6879(0.0240)	0.3384(0.0165)	0.6936(0.0164)
	GNMF	0.715(0.0252)	0.4005(0.0172)	0.6813(0.0174)
	$\ell_{2,1}$NMF	0.6727(0.0181)	0.3408(0.0144)	0.6952(0.0240)
	$\ell_{2,p}$NMF	0.6768(0.0278)	0.3848(0.0185)	0.6835(0.0305)
	SCCNMF-1	0.5173(0.0320)	0.0825(0.0038)	0.7064(0.0212)
	FR-NMF	0.0753(0.0258)	0.0248(0.0039)	–
	SNMF	**0.7832(0.0183)**	**0.4224(0.0127)**	**0.7374(0.0169)**
ARI	NMF	0.4920(0.0229)	0.1051(0.0062)	0.5489(0.0236)
	GNMF	0.5502(0.0239)	0.1122(0.0126)	0.5247(0.0219)
	$\ell_{2,1}$NMF	0.4739(0.0195)	0.1073(0.0047)	0.5470(0.0341)
	$\ell_{2,p}$NMF	0.4771(0.0349)	0.1224(0.0072)	0.5415(0.0432)
	SCCNMF-1	0.3186(0.0266)	-0.0019(0.0009)	0.5607(0.0329)
	FR-NMF	0.0966(0.0118)	0.0026(0.0006)	–
	SNMF	**0.6212(0.0245)**	**0.1238(0.0094)**	**0.6167(0.0282)**

rule in the gradient descent method of x should be the following:

$$y = x - \tau_v Ax.$$

However, it is difficult to satisfy the constraint $y^T y = 1$. From [33, 34], a modified method (30) is used.

$$y(\tau) = x - \tau A\left(\frac{x + y(\tau)}{2}\right). \tag{30}$$

(30) could satisfy that $y^T y = x^T x = 1$ for any skew-symmetric matrix A and τ.

From Lemma 1 (2) in [25], (30) could be expressed as:

$$y(\tau) = \left(I + \frac{\tau}{2}A\right)^{-1}\left(I - \frac{\tau}{2}A\right)x. \tag{31}$$

Then from Lemma 2 in [25], (31) could be rewritten as

$$y(\tau) = x - \beta_1(\tau)a - \beta_2(\tau)x, \tag{32}$$

where

$$\beta_1(\tau) = \tau\frac{x^T x - \frac{\tau}{2}((a^T x)(x^T x) - (x^T a)(x^T x))}{1 - \left(\frac{\tau}{2}\right)^2(a^T x)^2 + \left(\frac{\tau}{2}\right)^2\|a\|^2\|x\|^2},$$

$$\beta_2(\tau) = -\tau\frac{x^T a - \frac{\tau}{2}((a^T x)(x^T a) - (a^T a)(x^T x))}{1 - \left(\frac{\tau}{2}\right)^2(a^T x)^2 + \left(\frac{\tau}{2}\right)^2\|a\|^2\|x\|^2}.$$

The nonnegative constraints on y should be handled next.

Note $q = 1 - \left(\frac{\tau}{2}\right)^2(a^T x)^2 + \left(\frac{\tau}{2}\right)^2\|a\|^2\|x\|^2$. Rewrite (32) as follows:

$$y(\tau) = x - \frac{\tau}{q}(\beta_1'(\tau)(c_1 - c_2) + \beta_2' x), \tag{33}$$

where

$$\beta_1^{'}(\tau) = x^T x - \frac{\tau}{2}((a^T x)(x^T x) - (x^T a)(x^T x)),$$
$$\beta_2^{'}(\tau) = x^T a - \frac{\tau}{2}((a^T x)(x^T a) - (a^T a)(x^T x)).$$

Expanding $\beta_1^{'}(\tau)(c_1 - c_2)$ and $\beta_2^{'}x$ in (33) through auxiliary variables (6), we get

$$\begin{aligned} y(\tau) = x - \frac{\tau}{q}\left(x^T x c_1 + \frac{\tau}{2}T1 + x^T c_2 x + \frac{\tau}{2}P1\right) \\ + \frac{\tau}{q}\left(x^T x c_2 + \frac{\tau}{2}T2 + x^T c_1 x + \frac{\tau}{2}P2\right). \end{aligned} \tag{34}$$

Because $y(\tau)$ has nonnegative constraints, the τ should satisfy

$$x - \frac{\tau}{q}\left(x^T x c_1 + \frac{\tau}{2}T1 + x^T c_2 x + \frac{\tau}{2}P1\right) \geq 0. \tag{35}$$

Denote $(a^T x)^2 - \|a\|^2\|x\|^2$ as M. Then $q = 1 - (\frac{\tau}{2})^2 M$. Therefore, we obtain

$$\begin{aligned} qx - \tau\left(x^T x c_1 + \frac{\tau}{2}T1 + x^T c_2 x + \frac{\tau}{2}P1\right) \geq 0 \\ \Rightarrow (2T1 + 2P1 + Mx)\tau^2 + 4(x^T x c_1 + x^T c_2 x)\tau - 4x \leq 0. \end{aligned}$$

Utilizing the auxiliary variables (6), a series of equations are obtained as follows:

$$B_i\tau^2 + C_i\tau + D_i \leq 0, \quad i = 1, 2, \ldots, k. \tag{36}$$

(6) confirms that $C_i \geq 0$ and $D_i \leq 0$. Thus, when τ satisfies (36) and $\tau > 0$, there exist three situations.

1. when $B_i > 0$,
$$\tau = \frac{\sqrt{C_i^2 - 4B_iD_i} - C_i}{2B_i}.$$
2. when $B_i = 0$,
$$\tau = -\frac{D_i}{C_i}.$$
3. when $B_i < 0$,
$$\tau = \frac{\sqrt{C_i^2 - 4B_iD_i} - C_i}{2B_i}.$$

Thus, we obtain

$$\begin{aligned} \tau = min\{\frac{\sqrt{C_i^2 - 4B_iD_i} - C_i}{2B_i}, \quad when\ B_i \neq 0; -\frac{D_i}{C_i},\ when\ B_i = 0\}, \\ i = 1, 2, \ldots, k. \end{aligned} \tag{37}$$

(37) guarantees (35) hold. Thus, the updating rule (19) guarantees all constraints hold.

Now it has been proved that (18) and (19) are gradient descent methods for (16). Therefore, (16) is non-increasing under (18) and (19).

Acknowledgements This work was supported by the National Natural Science Foundation of China (11961010, 61967004).

Data availability All datasets analyzed in this study are available in the homepage of Deng Cai (http://www.cad.zju.edu.cn/home/dengcai/).

Declarations

Conflict of interest All authors disclosed no relevant relationships.

References

1. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444. https://doi.org/10.1038/nature14539
2. Guan Y, Fang J, Wu X (2020) Multi-pose face recognition using cascade alignment network and incremental clustering. Signal Image Video Process 1:1–9
3. Ren Y, Kamath U, Domeniconi C, Xu Z (2019) Parallel boosted clustering. Neurocomputing 351:87–100
4. Xie P, Xing EP (2015) Integrating image clustering and codebook learning. In: AAAI, pp 1903–1909
5. Chang J, Chen Y, Qi L, Yan H (2020) Hypergraph clustering using a new laplacian tensor with applications in image processing. SIAM J Imag Sci 13(3):1157–1178
6. Song K, Yao X, Nie F, Li X, Xu M (2021) Weighted bilateral k-means algorithm for fast co-clustering and fast spectral clustering. Pattern Recognit 109:107560
7. Ren Y, Wang N, Li M, Xu Z (2020) Deep density-based image clustering. Knowl-Based Syst 1:105841
8. Kumar N, Uppala P, Duddu K, Sreedhar H, Varma V, Guzman G, Walsh M, Sethi A (2018) Hyperspectral tissue image segmentation using semi-supervised NMF and hierarchical clustering. IEEE Trans Med Imaging 38(5):1304–1313
9. Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720
10. Ji S, Ye J (2008) Generalized linear discriminant analysis: a unified framework and efficient model selection. IEEE Trans Neural Networks 19(10):1768–1782
11. Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. Nature 401(6755):788
12. Cai D, He X, Han J, Huang TS (2010) Graph regularized non-negative matrix factorization for data representation. IEEE Trans Pattern Anal Mach Intell 33(8):1548–1560
13. Shang F, Jiao L, Wang F (2012) Graph dual regularization non-negative matrix factorization for co-clustering. Pattern Recogn 45(6):2237–2250
14. Ding CH, Li T, Jordan MI (2008) Convex and semi-nonnegative matrix factorizations. IEEE Trans Pattern Anal Mach Intell 32(1):45–55
15. Hu W, Choi K-S, Wang P, Jiang Y, Wang S (2015) Convex nonnegative matrix factorization with manifold regularization. Neural Netw 63:94–103
16. Cui G, Li X, Dong Y (2018) Subspace clustering guided convex nonnegative matrix factorization. Neurocomputing 292:38–48
17. Kong D, Ding C, Huang H (2011) Robust nonnegative matrix factorization using l21-norm. In: Proceedings of the 20th ACM international conference on information and knowledge management, pp 673–682
18. Li Z, Tang J, He X (2017) Robust structured nonnegative matrix factorization for image representation. IEEE Trans Neural Netw Learn Syst 29(5):1947–1960

19. Zhang Z, Liao S, Zhang H, Wang S, Hua C (2018) Improvements in sparse non-negative matrix factorization for hyperspectral unmixing algorithms. J Appl Remote Sens 12(4):045015
20. Xing L, Dong H, Jiang W, Tang K (2018) Nonnegative matrix factorization by joint locality-constrained and l2, 1-norm regularization. Multimed Tools Appl 77(3):3029–3048
21. Babaee M, Tsoukalas S, Babaee M, Rigoll G, Datcu M (2016) Discriminative nonnegative matrix factorization for dimensionality reduction. Neurocomputing 173:212–223
22. Liu H, Wu Z, Li X, Cai D, Huang TS (2011) Constrained nonnegative matrix factorization for image representation. IEEE Trans Pattern Anal Mach Intell 34(7):1299–1311
23. Fei W, Tao L, Changshui Z (2008) Semi-supervised clustering via matrix factorization. In: Proceedings of 2008 SIAM International Conference on Data Mining (SDM 2008), pp 1–12
24. Yang Y-J, Hu B-G (2007) Pairwise constraints-guided non-negative matrix factorization for document clustering. In: IEEE/WIC/ACM International Conference on Web Intelligence (WI'07). IEEE, pp 250–256
25. Yang Z, Hu Y, Liang N, Lv J (2019) Nonnegative matrix factorization with fixed l2-norm constraint. Circuits Syst Signal Process 38(7):3211–3226
26. Ahmed I, Hu XB, Acharya MP, Ding Y (2021) Neighborhood structure assisted non-negative matrix factorization and its application in unsupervised point-wise anomaly detection. J Mach Learn Res 22(34):1–32
27. Kuang D, Ding C, Park H (2012) Symmetric Nonnegative Matrix Factorization for Graph Clustering, pp 106–117. https://doi.org/10.1137/1.9781611972825.10
28. Samaria FS, Harter AC (1994) Parameterisation of a stochastic model for human face identification. In: Proceedings of 1994 IEEE workshop on applications of computer vision, pp 138–142. https://doi.org/10.1109/ACV.1994.341300
29. Hedjam R, Abdesselam A, Melgani F (2021) NMF with feature relationship preservation penalty term for clustering problems. Pattern Recogn 112:107814
30. Cai D, He X, Han J (2005) Document clustering using locality preserving indexing. IEEE Trans Knowl Data Eng 17(12):1624–1637
31. Wang Y, Chen L, Mei J-P (2014) Stochastic gradient descent based fuzzy clustering for large data. In: 2014 IEEE international conference on fuzzy systems (FUZZ-IEEE). IEEE, pp 2511–2518
32. Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218
33. Goldfarb D, Wen Z, Yin W (2009) A curvilinear search method for p-harmonic flows on spheres. SIAM J Imag Sci 2(1):84–109. https://doi.org/10.1137/080726926
34. Vese LA, Osher SJ (2002) Numerical methods for p-harmonic flows and applications to image processing. SIAM J Numer Anal 40(6):2085–2104. https://doi.org/10.1137/S0036142901396715