Robust structured convex nonnegative matrix factorization for data representation

QING YANG¹, XUESONG YIN², SIMIN KOU², YIGANG WANG²
¹School of Computer Engineering, Nanjing Institute of Technology, Nanjing 211167, China
²Department of digital media technology, Hangzhou Dianzi University, Hangzhou 310012, China
Corresponding author: Xuesong Yin (e-mail: yinx@hdu.edu.cn).

This work was supported by Key R&D Program Projects of Zhejiang Province under Grant 2018C01030, and Industry & University Research Cooperation Project of Jiangsu Province under Grant DH20190144.

Abstract: Nonnegative Matrix Factorization (NMF) is a popular technique for machine learning. Its power is that it can decompose a nonnegative matrix into two nonnegative factors whose product well approximates the nonnegative matrix. However, the nonnegative constraint of the data matrix limits its application. Additionally, the representations learned by NMF fail to respect the intrinsic geometric structure of the data. In this paper, we propose a novel unsupervised matrix factorization method, called Robust Structured Convex Nonnegative Matrix Factorization (RSCNMF). RSCNMF not only achieves meaningful factorizations of the mixed-sign data, but also learns a discriminative representation by leveraging local and global structures of the data. Moreover, it introduces the L₂,₁-norm loss function to deal with noise and outliers, and exploits the L₂,₁-norm feature regularizer to select discriminative features across all the samples. We develop an alternate iterative scheme to solve such a new model. The convergence of RSCNMF is proven theoretically and verified empirically. The experimental results on eight real-world data sets show that our RSCNMF algorithm matches or outperforms the state-of-the-art methods.

Index Terms: Convex nonnegative matrix factorization, Global structure, L₂,₁ norm, Clustering

I. INTRODUCTION

Nonnegative matrix factorization (NMF) focuses on well approximating a high-dimensional nonnegative matrix as the product of two low-dimensional nonnegative factor matrices. The factor learned by NMF not only is an effective low-dimensional representation of the original data, but also has explicit physical meaning. NMF has been shown to be the optimal way to learn the parts of an object [1]-[3], and successfully applied in computer vision, chemometrics, and pattern recognition [4]-[7].

Given a nonnegative data matrix \( X \in \mathbb{R}^{m \times n} \) and a specified dimension \( l \), NMF seeks two nonnegative factors \( A \in \mathbb{R}^{m \times l} \) and \( B \in \mathbb{R}^{l \times n} \), whose product approximates \( X \) very well. It optimizes the following problem:

\[
\min_{A \geq 0, B \geq 0} \| X - AB \|_F^2. \tag{1}
\]

According to whether label information is used for matrix factorization, existing methods can be divided into three categories: supervised [8]-[11], semi-supervised [12]-[15], unsupervised NMFs [16]-[19]. Actually, due to the high cost of obtaining label information, the application of the supervised and semi-supervised NMF methods is usually limited. Recently, unsupervised NMF has attracted more and more attention. Graph regularized nonnegative matrix factorization (GNMF) [2] constructed a nearest neighbor graph to encode the local structure, which is merged into NMF as a regularizer. Due to the effectiveness of GNMF, a few improved methods have been proposed [19], [42], [48]. Robust manifold nonnegative matrix factorization (RMNMF) [19] and Manifold NMF with L₂,₁ norm (MNMF [21]) [48] replace the least square objective function of GNMF with the L₂,₁-norm loss function to tackle noise and outliers, respectively. Robust Graph Regularized Nonnegative Matrix Factorization (RGNMF) [42] defined the affinity graph to formulate the weight of the data and the features, respectively. Different from RMNMF and MNMF, RGNMF introduces the L₁-norm metric on an error matrix to capture corrupted data so that it can mitigate the effect of noise and outliers. Regularized nonnegative matrix factorization with adaptive local structure learning (NMFAN) [43] enhanced the learning performance by simultaneously carrying out local structure learning and...
Nonnegative matrix factorization with local similarity learning (KLS-NMF) [44] introduced self-expressiveness mechanism into matrix factorization to learn the local similarity of the data in the kernel space. Chen et al. [49] performed the multilayer concept factorization by exploiting the local structure. Qian et al. [50] combined sparse graph learning into NMF for solving the feature matrix. Tolic et al. [51] introduced manifold learning into kernel-based NMF to perform matrix factorization. Gligorijevic et al. [52] applied four different NMF algorithms to discover community structure in multiplex networks. Yang et al. [53] constrained the sparsity on the two factors to improve the applicability of NMF. To enhance the performance of NMF, Gao et al. [54] exploited multiple local centroids to capture the local structure. Chen et al. [55] focused on detecting stable topics by incorporating sparsity regularization and soft orthogonality into NMF. A novel semi-NMF method [57] is proposed to enhance clustering by explicitly preserving the spatial information and learning the manifold structure. Peng et al. [58] combined manifold learning and feature learning into the loss function of NMF to improve the performance of clustering.

Although NMF and its variants have been shown to achieve the satisfactory performance, their applications are limited by the nonnegative constraint on the original data. In real-world applications, one often handles data containing negative elements. To address the issue, convex NMF (CNMF) [20] can decompose the mixed-sign data into two nonnegative factor matrices, and thus extend the repertoire of NMF. Matrix factorization feature selection (MFFS) [21] added the orthogonality constraint imposed on the feature matrix to CNMF as a regularization term, which is applied to unsupervised feature selection scenario. Both CNMF and MFFS ignore the local structure of the data. Actually, it has been shown that the local structure can enhance the learning performance [2], [6], [22]. Graph regularized CNMF (GCNMF) [23] depicted the local structure by constructing an affinity graph and combined it into CNMF as a regularizer. Subspace learning-based graph regularized feature selection (SGFS) [24] integrated the local structure into the orthogonal CNMF. Global and local structures preserving sparse subspace learning (GLoSS) [25] exploited both the local structure and row-sparsity measure, and incorporated them into CNMF as two regularization terms. It is difficult for GLoSS to capture the global structure by imposing $L_2$-norm constraint on the feature matrix. Regularized matrix factorization feature selection (RMFFS) [26] imposed a combination of $L_1$-norm and $L_2$-norm on the feature matrix to select the feature subset. Subspace clustering guided CNMF (SC-CNMF) [27] dynamically captured the global structure in the process of matrix factorization to seek a proper representation of subspace. Robust unsupervised feature selection (NSSLFS) [28] enforced the $L_1$-norm and $L_2,1$-norm regularization on the feature matrix into the loss function of CNMF. Structure preserving unsupervised feature selection (SPFS) [29] only applied local or global structure to learn the feature space, instead of exploiting the combination of them. Local discriminative based sparse subspace learning (LDSSL) [30] used class labels of the data to preserve both the local and discriminative structure. Recent studies have shown that, if both local and global structures of the data are exploited, the learning performance will be significantly improved [31], [32]. Above-mentioned CNMF and its variants sought a compact representation by exploiting local or global structure only. In general, data structures from various real-world applications are complex, and a single formulation (either local or global) is not enough to find the most discriminative feature space [31]. Consequently, it is necessary to characterize both local and global structures of the data. Besides, those CNMF-based methods adopted the least square loss function to learn the latent representations, and are thus sensitive to noise and outliers.

![Diagram](image)

**FIGURE 1.** Differences between existing CNMF methods and our RSCNMF algorithm.

In this paper, we propose a novel unsupervised Robust Structured Convex Nonnegative Matrix Factorization (RSCNMF) to find a discriminative representation by leveraging local and global structures. We construct an affinity graph to characterize the local structure and simultaneously exploit the total variance of the data to encode the global structure. Compared with existing methods, the new model explicitly both minimizes the local scatter and maximizes the global scatter to learn local and global structures of the data. Hence, the learned representations can achieve local and global consistency. To address the noise problem in real-world applications, we introduce joint sparse learning into the new model. Specifically, we adopt the $L_{2,1}$-norm loss function to deal with noise and outliers. The $L_{2,1}$-norm minimization is constrained on the feature representation to select discriminative features. As a result,
we formulate the graph structure and joint sparse learning as a general framework of matrix factorization, as shown in Fig. 1. We develop a multiplicative scheme to iteratively solve such a general framework. The convergence of our optimization scheme is proved theoretically and experimentally.

The appealing characteristics of this paper are highlighted as follows:

1) We encode the global structure by exploiting the overall variance of the mixed-sign data so that the learned representations with the same structure have the same clustering label. We also construct an affinity graph to model the local structure, which makes nearby representations have the same clustering label. Hence, the representations learned by the proposed algorithm can achieve local and global consistency.

2) Our algorithm integrates matrix factorization and joint sparse learning into a novel unsupervised framework. In such a framework, we apply the $L_{2,1}$-norm based loss function not only to characterize a general model for feature learning methods based on matrix factorization, but also to improve the robustness of the proposed model in real-world applications. Furthermore, we impose the $L_{2,1}$-norm minimization on the feature representation to select discriminative features.

3) The proposed algorithm devises a new matrix factorization model by reconciling local and global structures, the $L_{2,1}$-norm based loss function and the $L_{2,1}$-norm feature regularization. We formulate this new model as an optimization problem solved by a developed iterative multiplicative scheme. The optimization algorithm is proven to be effective and efficient.

4) Our algorithm can achieve encouraging clustering results compared with state-of-the-art algorithms. It is a general framework that can take a few methods based on matrix decomposition as its special cases. Moreover, the proposed algorithm can be naturally extended to semi-supervised scenario.

II. RELATED WORK

This section briefly reviews CNMF and other representative works closely related to our proposed algorithm.

A. CONVEX NONNEGATIVE MATRIX FACTORIZATION (CNMF)

Given an any real matrix $X = [x_1, x_2, \ldots, x_n]_{i \times m}$, CNMF aims to find a mixed-sign basis matrix $W \in \mathbb{R}_{+}^{m \times l}$ and a nonnegative coefficient matrix $H \in \mathbb{R}_{+}^{l \times n}$, which achieves $X \approx WH$. The matrix $W$ can be expressed as the product of $X$ and $W$, where $W \in \mathbb{R}_{+}^{n \times l}$ is called the feature matrix in feature selection scenario. The objective function of CNMF is defined as

$$\min_{W,H,20} \| X - XWH \|_F^2.$$  \hspace{1cm} (2)

CNMF can handle data matrices with nonnegative entries. Thus, it extends the application of NMF. Obviously, if the input matrix $X$ is required to be nonnegative, CNMF degrades to NMF.

B. GRAPH REGULARIZED AND CONVEX NONNEGATIVE MATRIX FACTORIZATION (GCNMF)

Although NMF and CNMF are two simple and effective dimensionality reduction methods, they ignore the local structure of the data. It has been shown that the performance of the learning model will be greatly improved when considering the local structure [2]. GCNMF [23] builds a similar graph to model the local structure and merges it into CNMF as a regularization term. Specifically, GCNMF optimizes the following loss function:

$$\min_{W,H,20} \| X - XWH \|_F^2 + \lambda \text{Tr}(HLH^T).$$ \hspace{1cm} (3)

where $\text{Tr}(A)$ denotes the trace of the matrix $A$ and $L = D - S$ is a Laplacian matrix. The diagonal entry of the diagonal matrix $D$ is column (or row) sums of $S$. The weight matrix $S$ is used to measure the nearest neighbors between sample points.

C. SUBSPACE CLUSTERING GUIDED CONVEX NONNEGATIVE MATRIX FACTORIZATION (SC-CNMF)

Unlike other graph-based methods that use pairwise distance to build a similarity graph, SC-CNMF applies subspace clustering to construct the similarity graph and captures the multi-subspace structure [27]. Moreover, the similarity graph is not established in advance, but generated in the process of matrix factorization. The objective function of SC-CNMF is defined as

$$\min_{W,H,20} \| X - XWH \|_F^2 + \| X - XZ \|_F^2 + \alpha \text{Tr}(HLH^T)$$

$$+ \beta \text{Tr}(H^T eH) + \gamma \|WH\|_2^2,$$ \hspace{1cm} (4)

where the similarity matrix is computed as $S = (Z + Z^T) / 2$ and accordingly the Laplacian matrix is expressed as $L = D - S$. $\text{Tr}(H^T eH)$ is used to reduce the relevance among the rows of $H$. In order to improve the performance, SC-CNMF iteratively implements subspace clustering and CNMF. In fact, the computational cost of subspace clustering is more than $O(n^4)$ [33]. The high computational complexity limits its applications.

D. UNSUPERVISED FEATURE SELECTION VIA MATRIX FACTORIZATION

MFSS is an unsupervised feature selection algorithm that is regarded as a matrix factorization problem [21]. Specifically, the technique of matrix factorization is used to learn features under the orthogonal constraint. MFSS optimizes the following objective function:

$$\min_{W,H,20} \| X - XWH \|_F^2 + \frac{d}{2} \|W^TW - I\|_F^2.$$ \hspace{1cm} (5)

MFSS is essentially an orthogonal CNMF algorithm. As a variant of MFSS, RMFFS formulates a combination of the
L1-norm and L2-norm on the feature matrix and merges them into CNMF as two regularization terms [26]. The objective function of RMFFS is defined as follows:

$$\min_{W,\alpha,\beta} \|X - XWH\|_F^2 + \alpha \|W\|^2_F - \|V\|^2_F \cdot$$ (6)

Both MFSS and RMFFS ignore the local structure. Thus, it is difficult for them to find a compact representation. Especially, since RMFFS uses the regularization parameter \( \alpha \) to balance two different regularizers, it cannot guarantee the existence of the optimal solution [11]. GLoSS models the structure information to guide sparse subspace learning [25]. It adds the \( L_1 \)-norm regularization to both control the row sparsity of the feature matrix and to select representative features. GLoSS solves the following problem:

$$\min_{W,\alpha,\beta} \|X - XWH\|_F^2 + \alpha \|W\|^2_F + \beta \|W\|_1.$$ (7)

GLoSS expects to preserve local and global structure of the data. The Laplacian matrix \( L \), however, is solved by local linear embedding (LLE) or locality preserving projections (LPP) only. Thus, GLoSS is still a matrix factorization algorithm based on the local structure. SPFS extends the solution of the matrix \( L \) in GLoSS [29]. It defines the sparsity regularizer to preserve the global structure. SPFS can respect the local structure or the global structure only, but it can’t consider both local and global structures. The objective function of SPFS is formulated as

$$\min_{W,\alpha,\beta} \|X - XWH\|_F^2 + \alpha \|W\|^2_F + \beta \|W\|_1,$$ (8)

where the Laplacian matrix \( L \) is solved by one of LLE, LPP and sparsity regularizer.

In the above-described algorithms, \( W \) is regarded as a feature matrix of selected features. Its elements are either set to 0 or 1, which is an NP-hard optimization problem [28]. To obtain the optimal solution, the current models relax \( W \) into the nonnegative space to implement feature selection [28],[34]. Clearly, these methods mentioned above promote the learning performance of CNMF from different perspectives. However, they decompose the data matrix into the two factors by exploiting local or global structure only. In other words, they fail to simultaneously consider local and global structures. Many algorithms adopt the least square loss function to solve the two factor matrices, which is sensitive to noise and outliers. Moreover, NMF-based approaches cannot tackle data matrices with negative elements. To address these issues, we propose robust structured convex nonnegative matrix factorization (RSCNMF) to learn a robust discriminative representation.

### 3. ROBUST STRUCTURED CONVEX NONNEGATIVE MATRIX FACTORIZATION

In this section, we introduce the objective function of our RSCNMF, and develop an optimization scheme based on iterative updates of the two factor matrices to solve this objective function. Then, we provide the convergence proof of RSCNMF.

### A. PROBLEM FORMULATION

It is more natural and reasonable to assume that data are often collected from both single manifold and multiple manifolds. In the application of object recognition, for example, although images sampled by an object lying on a single manifold, those taken by different objects locate on various manifolds. Generally, the more the images from different manifolds are separated in the low-dimensional representation space, the better the recognition is. Actually, the global structure of the data plays an important role in separating various objects [35]. It is well-known that the global structure is defined by the inter-class scatter with available class labels in supervised or semi-supervised learning scenario. Due to the lack of labels, unsupervised feature learning methods based on matrix factorization do not pay attention to the global structure. Recent studies have shown that principal component analysis (PCA) can depict the global information in unsupervised scenario through maximizing the representation variance [32],[36]. Following this line, we build a principal component graph to model the global structure and introduce it into matrix factorization. Assuming that the vector \( h_i \) is the low-dimensional representation of the sample \( x_i \) about the new basis, we can formulate the global scatter imposed on the low-dimensional representations as

$$J_G = \sum_{i=1}^{n} \|h_i - m\|^2,$$ (9)

where \( m = (1/n)\sum_{i=1}^{n} h_i \) is defined as the global mean vector.

Simplifying Equation (8), we get

$$J_G = \sum_{i=1}^{n} h_i^T h_i - \sum_{i=1}^{n} (h_i^T m) = Tr(HH^T - \frac{1}{n} Hee^T H^T)$$ (10)

where \( R = I - E \) is a principal component matrix and \( I \) denotes an \( n \times n \) identity matrix. \( E = (1/n)ee^T \) and all elements of the n-dimensional column vector \( e \) are 1. By maximizing \( J_G \) in (9), the learned representations are globally consistent. We set \( M = E - I \), and thus \( Tr(HH^T) = Tr(HMH^T) \). It is easy to obtain that maximizing \( Tr(HH^T) \) is equivalent to minimizing \( Tr(HMH^T) \).

Generally, it is insufficient to uncover the hidden semantics if only a single characterization (global or local) is taken into account [37]. Thus, when both local and global structures are exploited, the learning performance can be substantially improved [35],[38]. In order to explore the local geometric structure, manifold learning theory assumes that local invariance plays an essential role in the development of various graph-based methods. Specifically, local invariance indicates that if two data points \( x_i \) and \( x_j \) are close in the original data space, their corresponding representations \( h_i \) and \( h_j \) in the latent subspace are also close to each other. Recent studies have demonstrated that local invariance can be used to promote matrix factorization [2],[6],[19]. Similar to [2] and [22], we define a local scatter by building an affinity
graph to formulate the consistency of nearby representations, which is expressed as
\[
J_c = \frac{1}{2} \sum_{i,j=1}^{n} \| h_i - h_j \|_2^2 \quad S_{ij} = \sum_{i,j=1}^{n} h_i^T h_j D_{ij} - \sum_{i,j=1}^{n} h_i^T h_j S_{ij} \quad (11)
\]

where \(L=D-S\) denotes the Laplacian matrix. The diagonal entry of the diagonal matrix \(D\) is column (or row) sums of the weight edge matrix \(S\) whose value is computed by
\[
S_{ij} = \begin{cases} 
\exp(\|x_i - x_j\|_2^2 / \sigma^2) & x_i \in N_p(x_j) \text{ or } x_j \in N_p(x_i) \\
0 & \text{otherwise}
\end{cases}
\]
where \(N_p(x_i)\) is a set of \(p\)-nearest neighbors of \(x_i\).

Data is often corrupted by noise and outliers in real-world applications. In contrast to the model learned by using the least square loss function, dimensionality reduction methods via sparse learning are insensitive to noise and outliers [19],[39]. Moreover, sparse learning has been demonstrated to be a powerful technique in feature learning and clustering tasks [40],[41]. Hence, we introduce sparse learning into matrix factorization and feature learning, and define the following loss function:
\[
\min_{W \geq 0, H \geq 0} \| X - XWH \|_{F,1} + \lambda \| W \|_{F,1},
\]
where \(\lambda \geq 0\) is a trade-off parameter. As observed in (12), we apply the L_{2,1}-norm based loss function to enhance the robustness of the proposed model. Actually, as mentioned above, few feature learning methods based on CNMF apply sparse learning to seek the latent representation and to avoid the interference of noise and outliers. Additionally, we exploit the L_{2,1}-norm constraint on the feature matrix to guarantee its row sparsity and to select discriminative features.

According to the above analysis, we define a novel CNMF framework based on sparse learning as follows:
\[
\min_{W \geq 0, H \geq 0} \| X - XWH \|_{F,1} + \alpha \text{Tr}(HLH^T) + \beta \text{Tr}(HMH^T) + \lambda \| W \|_{F,1},
\]
\[
\text{s.t. } W \geq 0, H \geq 0,
\]
where \(\alpha\) and \(\beta\) are two regularization parameters. The second and third terms in the objective function are used to formulate local and global structures. The first and last ones are used to enhance the model robustness and to select discriminative features.

As observed in (13) and Fig.1, there are three differences between state-of-the-art methods and our proposed RSCNMF.

1) Our algorithm seeks a discriminative representation by leveraging local and global structures. We build an affinity graph to characterize the local structure and simultaneously exploit the total variance of the raw data to encode the global structure. MFFS and RMFFS ignored the local structure. Hence, it is difficult for them to learn a good data representation. RGNMF and NMFAN focused on the local structure and thus ignored the global structure in learning the discriminative representation. KLS-NMF captured the local structure in the kernel space by encoding the self-expressiveness property. GLoSS and SPFS cannot consider both local and global structures and only adopt one of them.

2) As formulated in the first term of (13), our algorithm makes use of convex NMF to find two nonnegative factor matrices so that it is competent for matrix factorization of the mixed-sign data. KLS-NMF applies an RBF kernel to map the original data containing negative elements into the high-dimensional feature space for matrix factorization. As pointed out in [45,46], however, there is no explicit reason why such a kernel should correspond to the feature space which is very suitable for self-expressiveness. Although NMF and its variants have been shown to achieve the satisfactory performance, their applications are limited by the nonnegativity constraint on the original data. In other words, these NMF-based algorithms, such as RGNMF and NMFAN, can only deal with the nonnegative data.

3) Our algorithm introduces the L_{2,1}-norm based loss function into CNMF to make the new model robust to noise and outliers. CNMF-based methods are sensitive to noise and outliers, since they apply the least square loss function to learn a representation.

4) A L_{2,1}-norm regularization \(\| W \|_{2,1}\) is applied to select discriminative features across all data points in our model. Other methods do not implement feature learning.

B. OPTIMIZING SCHEME

Similar to other methods, the objective function of RSCNMF is convex in matrix \(W\) only or \(H\) only. In this section, we develop an iterative optimization algorithm to obtain a close-form solution of the two factor matrices. To facilitate the later optimization, we first rewrite our objective function in (13) as:
\[
O = \text{Tr}((X - XWH)Q(X - XWH)^T) + \alpha \text{Tr}(HLH^T)
\]
\[
+ \beta \text{Tr}(HMH^T) + \lambda \text{Tr}(WPW^T)
\]
\[
= \text{Tr}(XQX^T) - 2\text{Tr}(XQW^TW^TX^T) + \text{Tr}(XWHQH^TW^TX^T) + \alpha \text{Tr}(HLH^T)
\]
\[
+ \beta \text{Tr}(HMH^T) + \lambda \text{Tr}(WPW^T),
\]
where the matrix properties \(\text{Tr}(AB) = \text{Tr}(BA)\) and \(\text{Tr}(A) = \text{Tr}(A^T)\) are used in the second equality. \(Q\) and \(P\) are two diagonal matrices with \(Q_{ii} = 1/(2\|X-XWH\|_2^2)\) and \(P_{ii} = 1/(2\|W\|_2^2)\), respectively. If \(\|X-XWH\|_2^2\) or \(\|W\|_2^2\) is close to zero, we use \(Q_{ii} = 1/(2\|X-XWH\|_2^2 + \epsilon)\) or \(P_{ii} = 1/(2\|W\|_2^2 + \epsilon)\) to compute their diagonal entries accordingly, where \(\epsilon\) is a very small positive number.

We express the two multiplier matrices with constraints \(W \geq 0\) and \(H \geq 0\) as \(Y = [y_{ii}]\) and \(\Phi = [\phi_{ii}]\), respectively. The corresponding Lagrange function \(L\) is defined as follows:
\[
L = \text{Tr}(XQX^T) - 2\text{Tr}(XQW^TW^TX^T) + \text{Tr}(XWHQH^TW^TX^T) + \alpha \text{Tr}(HLH^T)
\]
\[
+ \beta \text{Tr}(HMH^T) + \lambda \text{Tr}(WPW^T)
\]
\[
+ \text{Tr}(\Phi H^T).\]
We take the partial derivatives of \( L \) with respect to \( W \) and \( H \), respectively, and get
\[
\frac{\partial L}{\partial W} = -2X^T X Q H^T + 2X^T W X W H Q H^T + 2\lambda W P + \Psi, \tag{16}
\]
\[
\frac{\partial L}{\partial H} = -2W^T X Q + 2W^T X W H Q H^T + 2\alpha H L + 2\beta H M + \Phi. \tag{17}
\]
According to the Karush-Kuhn-Tucker (KKT) conditions, we have \( w_{ik}W_{ik} = 0 \) and \( h_{ij}H_{ij} = 0 \). The equations for \( W \) and \( H \) can be represented as:
\[
(W^T X W H Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij} = 0 \tag{18}
\]
\[
(W^T X W H Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij} = 0 \tag{19}
\]
It is known if \( W_{ik} = 0 \), then \( W_{ik} = 0 \), and vice versa.

Therefore, if (18) and (19) hold, (20) and (21) also hold and vice versa.
\[
(W^T X W H Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij} = 0 \tag{20}
\]
\[
(W^T X W H Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij} = 0 \tag{21}
\]
Finally, we arrive at the following rules for two factor matrices \( W \) and \( H \):
\[
W_{ik} \leftarrow W_{ik} \frac{(X^T X)QH^T + (X^T X)W H Q H^T\delta_{ij}}{(X^T X)QH^T + (X^T X)W H Q H^T + \lambda WP\delta_{ij}}, \tag{24}
\]
\[
H_{ij} \leftarrow H_{ij} \frac{(W^T X Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij}}{(W^T X Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij}}. \tag{25}
\]

With (24) and (25), we can easily obtain the values of two factor \( W \) and \( H \) through an iterative scheme. Such an iterative scheme is developed to solve the objective function of the proposed method, which is presented in Algorithm 1. In our experiments, if \( t > 300 \) or \( |O_{i1} - O_{i|} / O_{i1}| < 0.001 \), where \( t \) is the iteration number and denotes the value of the cost function in the \( t \)-th iteration, then Algorithm 1 stops.

### Algorithm 1 RSCNMF

**Input:**
- Data matrix \( X \in \mathbb{R}^{m \times n} \).
- Regularization parameters \( \alpha, \beta, \lambda \).

**Output:**
- \( W \in \mathbb{R}^{m \times l} \), \( H \in \mathbb{R}^{l \times n} \).
1. Set \( t = 0 \) and \( Z = X \cdot WH; \) Initialize \( W^{(0)}, H^{(0)} \).
2. Compute the weight matrices \( S \) and \( M \).
3. Repeat:
4. \( W^{(t+1)} \leftarrow W^{(t)} \frac{(X^T X)QH^T + (X^T X)W H Q H^T\delta_{ij}}{(X^T X)QH^T + (X^T X)W H Q H^T + \lambda WP\delta_{ij}}, \)
5. \( H^{(t+1)} \leftarrow H^{(t)} \frac{(W^T X Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij}}{(W^T X Q + \alpha H L + \beta H M)H_{ij} - (W^T X Q)\delta_{ij}}.
6. \( Z^{(t+1)} = X - XWH^{(t+1)}QH^{(t+1)} \).
7. \( Q^{(t+1)} = \left[ \begin{array}{c} 1/(2)w_{i1}^{(t+1)} \\ 0 \\ \vdots \\ 1/(2)w_{ln}^{(t+1)} \end{array} \right], \)
8. \( p^{(t+1)} = \left[ \begin{array}{c} 1/(2)w_{i1}^{(t+1)} \\ 0 \\ \vdots \\ 1/(2)w_{ln}^{(t+1)} \end{array} \right] \).
9. \( t \leftarrow t + 1 \).
10. Until stopping criteria are satisfied.

### C. CONVERGENCE ANALYSIS

Following NMF and CNMF, we need to demonstrate the monotone property of the objective function in (14) under the update rules (24) and (25). To achieve this, we have to introduce an auxiliary function and a lemma.

**Definition 1.** If the conditions \( G(a, a') \geq G(a) \) and \( G(a, a') = G(a) \) are met, then \( G(a, a') \) is an auxiliary function for \( L(a) \).

According to Definition 1, we can deduce the following lemma which plays an important role in analyzing the convergence of RSCNMF.

**Lemma 1.** If \( G \) is an auxiliary function for \( L \), then \( L \) is nonincreasing under the following updating rule:
\[
a^{(t+1)} = \arg \min_a G(a,a^{(t)}) . \tag{26}
\]

**Proof.** \( L(a^{(t+1)}) \leq G(a^{(t+1)}, a^{(t)}) \leq G(a^{(t)}, a^{(t)}) = L(a^{(t)}) \).

As can be seen from the updated rules (24) and (25), RSCNMF involves two variables \( W \) and \( H \). The solution of \( Q \) requires \( W \) and \( H \), and \( P \) is only relevant to \( W \). According to Definition 1 and Lemma 1, we first prove that the update rule for \( W \) in (24) cannot increase the objective function \( O \) when \( H, Q, \) and \( P \) are fixed. We define \( F(W) \) to denote the part of \( O \) with respect to \( W \):
\[
F(W) = \text{Tr}(-2X^T X Q H^T W^T + W^T X W H Q H^T + \lambda WP W^T).
\]

In order to facilitate the following form, we rewrite \( F(W) \) as
\[
F(W) = \text{Tr}(-2W^T X W H Q H^T + W^T X W H Q H^T + \lambda WP W^T).
\]

**Lemma 2.**

\[
G(W,W^{(t)}) = -\sum_{a}^n B_{a}^{(t)} W_{a}^{(t)} (1 + \log W_{a}^{(t)}) + \sum_{a}^n B_{a}^{(t)} W_{a}^{(t)} + \lambda \sum_{a}^n x_{a} W_{a}^{(t)} \log W_{a}^{(t)}.
\]

is an auxiliary function for \( F(W) \).

Proof. From (29), we can easily validate \( G(W,W) = F(W) \), and thus only verify \( G(W,W^{(t)}) \geq F(W) \).

According to Proposition 5 in (Ding, Li, & Jordan, 2010), we can obtain
\[ \text{Tr}(W^T A^\top W C) \leq \sum_{a} (A^\top W^{(i)} C)_a W_{a}^2, \]  
\[ \quad \text{and} \]
\[ \lambda \text{Tr}(W P W^T) \leq \lambda \sum_{a} (W^{(i)} P)_a W_{a}^2, \]
where \( \lambda \) is the largest singular value of \( W \).

For any \( c, d > 0 \), the inequality \( c \leq (c^2+d^2)/2d \) holds. Thus, the second term of \( F(W) \) in (28) is restrained by
\[ \text{Tr}(B^T W^T) = \sum_{a} B_{a}^T W_{a}^2 \leq \sum_{a} B_{a}^T W_{a}^2 + (W_{a}^2)^2. \]

Since the inequality \( y \geq 1+\log y \) holds for any \( y > 0 \), we get
\[ W_{a}^2 \geq 1 + \log W_{a}, \]
and
\[ W_{a}^2 \geq 1 + \log W_{a}. \]

From (33), we observe that the first term in \( F(W) \) can be expressed as
\[ \text{Tr}(B^T W^T) = \sum_{a} B_{a}^T W_{a}^2 \geq \sum_{a} B_{a}^T W_{a}^2 (1 + \log W_{a}^2). \]

From (34), the fourth term in \( F(W) \) is limited to:
\[ \text{Tr}(W^T A^\top W C) \geq \sum_{a} (A^\top W^{(i)} C)_a W_{a}^2 (1 + \log W_{a}). \]

By combining (30), (31), (32), (35), and (36), we obtain \( G(W) \) in (25). Lemma 2 is proven.

**Theorem 1.** When the update rule in (24) is used to update \( W \), the value of the objective function in (14) is monotonically decreasing.

Proof. According to Lemmas 1 and 2, we replace (26) with \( G(W, W^{(i)}) \) in (29) and obtain the following update equation:
\[ W^{(i+1)} = W^{(i)} \sqrt{((X^T X)^\top Q H^{(i)} T + (X^T X)^\top W^{(i)} H^{(i)} Q H^{(i)} T + 2W^{(i)} P)} \]

Obviously, we can see that (37) is actually the update rule in (24). Since (29) is an auxiliary function, \( F(W) \) is monotonically decreasing under this update rule.

Similar to the above proof of (24), we also define an auxiliary function for \( H \), which does not increase the objective function of RSCNMF under the update rule in (25). Let \( F(H) \) denote the part of (14) in regard to \( H \) and fix \( W, Q \) and \( P \).
\[ F(H) = \text{Tr}(-2W^T X^\top Q H^{(i)} T + W^T X^\top W H^{(i)} Q H^{(i)} T) \]
\[ + \alpha \text{Tr}(H L H^{(i)} T) + \beta \text{Tr}(H H^{(i)} T). \]

For the convenience of proof, we can rewrite (38) as
\[ F(H) = \text{Tr}(-2B^T H^{(i)} T + 2B^T H^{(i)} T + A^\top H^{(i)} Q H^{(i)} T - A^\top H^{(i)} Q H^{(i)} T) \]
\[ + \alpha \text{Tr}(H D H^{(i)} T) - \alpha \text{Tr}(H S H^{(i)} T) + \beta \text{Tr}(H H^{(i)} T) - \beta \text{Tr}(H H^{(i)} T). \]

where \( A = W^T X^T X W, B = W^T X^T X, C = W^T X^T X, Q, \) and \( P = W^T X^T X Q. \)

**Lemma 3.** Function

\[ G(H, H^{(i)}) = \frac{1}{2} \sum_{i} 2B_{i}^T H_{i}^{(i)} (1 + \log H_{i}^{(i)}) + \sum_{i} B_{i}^T H_{i}^{(i)} + (H_{i}^{(i)})^2 \]
\[ + \alpha \sum_{i} (H_{i}^{(i)})^D_{i} - \alpha \sum_{i} S_{i} H_{i}^{(i)} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ + \beta \sum_{i} (H_{i}^{(i)})^E_{i} - \beta \sum_{i} H_{i}^{(i)} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]

is an auxiliary function for \( F(H) \).

Proof. We can easily verify that \( G(H, H) = F(H) \). To show that \( G(H, H^{(i)}) \) is an auxiliary function for \( F(H) \), we only need to prove \( G(H, H^{(i)}) \geq F(H) \).

Similar to the proof of Lemma 2, the following inequalities hold by means of Proposition 5 in [20]:
\[ \text{Tr}(A^\top H Q H^{(i)} T) \leq \sum_{i} (A^\top H Q)_{i} H_{i}^{(i)} \]
\[ \alpha \text{Tr}(H D H^{(i)} T) \leq \alpha \sum_{i} (H^{(i)})^D_{i} H_{i}^{(i)}, \]
and
\[ \beta \text{Tr}(H H^{(i)} T) \leq \beta \sum_{i} (H^{(i)})^E_{i} H_{i}^{(i)}. \]

Since the inequality \( c \leq (c^2+d^2)/2d \) holds for any \( c, d > 0 \), we have
\[ \text{Tr}(B^T H^{(i)} T) = \sum_{i} B_{i}^T H_{i}^{(i)} \leq \sum_{i} B_{i}^T H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ + \sum_{i} (A^\top H^{(i)} Q)_{i} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ + \alpha \sum_{i} (H^{(i)})^D_{i} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ + \beta \sum_{i} (H^{(i)})^E_{i} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]

According to (33) and (34), the following four inequalities hold:
\[ \text{Tr}(B^T H^{(i)} T) = \sum_{i} B_{i}^T H_{i}^{(i)} \leq \sum_{i} B_{i}^T H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ \text{Tr}(A^\top H Q H^{(i)} T) \geq \sum_{i} (A^\top H Q)_{i} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ \alpha \text{Tr}(H S H^{(i)} T) \geq \alpha \sum_{i} S_{i} H_{i}^{(i)} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]
\[ \beta \text{Tr}(H H^{(i)} T) \geq \beta \sum_{i} (H^{(i)})^E_{i} H_{i}^{(i)} (1 + \log H_{i}^{(i)}) \]

By modeling (41)-(48), we can obtain \( G(H, H^{(i)}) \geq F(H) \).

**Theorem 2.** When the update rule in (25) is used to update \( H \), the value of the objective function in (14) is monotonically decreasing.

Proof. In the light of Lemmas 1 and 3, we replace (26) with \( G(H, H^{(i)}) \) in (40) and thus get
\[ H_{i}^{(i+1)} = H_{i}^{(i)} \sqrt{(B^* + A^\top H^{(i)} Q + \alpha H^{(i)} S + \beta H^{(i)} T)_{i}} \]
\[ \sqrt{(B^* + A^\top H^{(i)} Q + \alpha H^{(i)} D + \beta H^{(i)} E)_{i}} \]
\[ \text{where} A = W^T(X^T X) W, A^* = W^T(X^T X) W, B = W^T (X^T X) Q \]
\[ \text{and} B^* = W^T (X^T X) Q. \]

It is easy to see that (49) is actually the update rule of (25). Since \( G(H, H^{(i)}) \) in (40) is the auxiliary function, \( F(H) \) does not increase under the update rule of (25).
According to the above analysis, we can deduce the following Theorem 3 with Theorems 1-2.

**Theorem 3.** The objective function in (14) decreases with the update rules in (24) and (25). The objective function is stable under the two rules if and only if \( W \) and \( H \) are at a fixed value.

Proof. After the \( t \)-th iteration, we can get \( W^{(t)} \) and \( H^{(t)} \), and easily calculate \( Q^{(t)} \) and \( P^{(t)} \). According to Theorem 1, we can arrive at the following inequality for a fixed \( H^{(t)} \):

\[
O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}) \leq O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}).
\]

(50)

Similarly, given a fixed \( W^{(t)} \), in the light of Theorem 2, the following inequality holds:

\[
O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}) \leq O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}).
\]

(51)

From (50) and (51), we have

\[
O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}) \leq O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}),
\]

(52)

which is equivalent to

\[
\sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2} \leq \sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2}.
\]

(53)

According to Lemma 1 in [40], we get

\[
\|z^{(t)}_{r}\|_{2}^{2} - \|z^{(t)}_{r}\|_{2}^{2} \leq \|z^{(t)}_{r}\|_{2}^{2} - \|z^{(t)}_{r}\|_{2}^{2}.
\]

(54)

Consequently,

\[
\sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2} \leq \sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2}.
\]

(55)

which \( Z = X - WH \).

By combining (53) and (55), we get the following inequality:

\[
\sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2} \leq \sum_{r=1}^{m} \|z^{(t)}_{r}\|_{2}^{2}.
\]

(56)

The inequality of (56) can be formulated as

\[
\|z^{(t)}_{r}\|_{2}^{2} \leq \|z^{(t)}_{r}\|_{2}^{2}.
\]

(57)

Thus, we obtain

\[
O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}) \leq O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}).
\]

(58)

Similar to (57), we can obtain the following inequality:

\[
\|w^{(t)}_{r}\|_{2}^{2} \leq \|w^{(t)}_{r}\|_{2}^{2}.
\]

(59)

Therefore,

\[
O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}) \leq O( W^{(t)}, H^{(t)}, Q^{(t)}, P^{(t)}).
\]

(60)

Obviously, the objective function of RSCNMF is decreasing. \( W \) and \( H \) are invariant under the update rules in (24) and (25) if and only if they are at a fixed value. The convergence of Algorithm 1 is proven.

**D. COMPUTATIONAL COMPLEXITY**

We now analyze the complexity of Algorithm 1 and apply a big \( O \) notation to denote its computational complexity.

From Steps 4-5 in Algorithm 1, the cost for computing \( W \) and \( H \) is \( O((p+n)(m+n)+l^2) \) in each iteration. Actually, we have \( l = n \) and \( l = m \). Thus, the computational cost of \( W \) and \( H \) is approximately \( O(mn^2) \). Since \( Q \) and \( P \) are two diagonal matrices, they cost \( O(mn+n) \) and \( O(nl+l) \) in each iteration, respectively. Besides, we need \( O(mn^2) \) to construct the affinity graph. Consequently, when Algorithm 1 stops after \( t \) iterations, the overall cost for our RSCNMF is \( O(mn^2+tn(mn+n)+(n+l)n+mn) \).

**E. \( L_2 \)-NORM-BASED STRUCTURED CONVEX NONNEGATIVE MATRIX FACTORIZATION**

In this section, we propose a least squares formulation for Structured Convex Nonnegative Matrix Factorization, called SCNMF. Similar to RSCNMF, SCNMF integrates both local and global structures into the process of matrix factorization. It defines the following cost function:

\[
\min_{W,H} \|X - WH\|_F^2 + \alpha\text{Tr}(HLH^T) + \beta\text{Tr}(HMH^T) + \lambda\|W\|_F^2
\]

s.t. \( W \succeq 0, H \succeq 0 \).

(61)

The optimization scheme to solve two factors \( W \) and \( H \) in (61) is same as that of RSCNMF and thus is omitted. According to RSCNMF, we get two updating rules as follows:

\[
W_u \leftarrow W_u - \frac{(X^T X)^T H^T + (X^T X)^T WHH^T + \lambda W)}{(X^T X)^T H^T + (X^T X)^T WHH^T + \lambda W)}
\]

(62)

\[
H_u \leftarrow H_u - \frac{(W^T X)^T X(W^T X)^T WH + \alpha HS + \beta H)}{(W^T X)^T X(W^T X)^T WH + \alpha HS + \beta H)}
\]

(63)

The iterative algorithm for obtaining \( W \) and \( H \) is summarized in Algorithm 2. Since the convergence and the complexity of Algorithm 2 are similar to those of Algorithm 1, we omit them.

**Algorithm 2 SCNMF**

**Input:**
- Data matrix \( X \in \mathbb{R}^{m \times n} \).
- Regularization parameters \( \alpha, \beta, \lambda \).

**Output:**
- \( W \in \mathbb{R}^{m \times n} \), \( H \in \mathbb{R}^{n \times n} \).

1. Initialize \( W, H \).
2. Compute the weight matrices \( S \) and \( M \).
3. Repeat
   4. \( W_u \leftarrow W_u - \frac{(X^T X)^T H^T + (X^T X)^T WHH^T + \lambda W)}{(X^T X)^T H^T + (X^T X)^T WHH^T + \lambda W)} \)
   5. \( H_u \leftarrow H_u - \frac{(W^T X)^T X(W^T X)^T WH + \alpha HS + \beta H)}{(W^T X)^T X(W^T X)^T WH + \alpha HS + \beta H)} \)
6. Until stopping criteria are satisfied.

**IV. EXPERIMENTS**

To verify the effectiveness of the proposed RSCNMF algorithm, we perform extensive experiments on publicly available data sets. As described in state-of-the-art methods, clustering results are used to measure the performance of the algorithm.

**A. DATA SETS**

In the experiments, we select eight available data sets from different fields for a comprehensive evaluation, including

---

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3128975, IEEE Access

Q. YANG et al.: Robust structured convex nonnegative matrix factorization for data representation
two face image data sets\textsuperscript{1,2}, one digital image data sets\textsuperscript{3} and five biological data sets\textsuperscript{2}. The detailed statistics of these data sets are shown in Table 1. It is worth noting that three biological data sets, i.e., Lymphoma, MLLML, and MALDIML, contain negative elements, where NMF and its variants fail to work effectively.

![Image](37x33)\textsuperscript{2}

| Data sets     | Samples | Features | Classes |
|---------------|---------|----------|---------|
| ORL           | 400     | 1024     | 40      |
| SRBTCML       | 83      | 2308     | 4       |
| COL120        | 1440    | 1024     | 20      |
| Face96        | 3016    | 4096     | 151     |
| Lymphoma      | 96      | 4026     | 9       |
| MLLML         | 72      | 12582    | 3       |
| GIL\_85       | 85      | 22283    | 2       |
| MALDIML       | 112     | 16331    | 2       |

### B. COMPARED ALGORITHMS

To comprehensively verify the effectiveness of our RSCNMF, we compare it not only with current CNMF-based methods, but also with that of state-of-the-art algorithms based on NMF. These compared approaches are listed as follows.

1) **Baseline**: K-means algorithm is used to cluster the raw data.

2) **CNMF**: Convex NMF (CNMF) can apply both nonnegative and mixed-sign data matrices to solve two nonnegative factor matrices [20].

3) **GCNMF**: Graph regularized CNMF (GCNMF) respects the local geometric structure and merges it into CNMF as a regularization term for learning a discriminative representation [23].

4) **RMFFS**: Regularized matrix factorization feature selection (RMFFS) defines a combination of $L_1$-norm and $L_2$-norm on the feature matrix and adds it to the loss function of CNMF [26].

5) **SGFS**: Subspace learning-based graph regularized feature selection (SGFS) presents a feature selection framework, which incorporates both the graph regularization and the $L_2$-norm on the feature matrix into CNMF [24].

6) **SC-CNMF**: Subspace clustering guided CNMF (SC-CNMF) conducts subspace clustering to capture the global structure of the data to guide CNMF, and thus simultaneously optimize subspace clustering and CNMF in a unified framework [27].

7) **NSSLFS**: Robust unsupervised feature selection (NSSLFS) introduces the $L_{2,1}$-norm and the nonnegative constraints on the feature matrix into CNMF not only to remove the irrelevant feature, but also to capture the low-dimensional structure of the data space [28].

8) **SPFS**: Structure preserving unsupervised feature selection (SPFS) extends MFFS by incorporating global or local structures of the input data, which is formulated as a matrix factorization optimization issue [29].

9) **MNMF2L\_\_**: It constructs a nearest neighbor graph to formulate the local structure, and exploits $L_{2,1}$-norm cost function to improve the robustness [47].

10) **NMF-LCAG**: It merges the locality graph construction into the process of matrix factorization [48].

11) **RGNMF**: It defines a nearest neighbor graph to formulate the weight of the data and the features, and introduces the $L_1$-norm metric on an error matrix to mitigate the effect of noise and outliers [42].

12) **NMFAN**: It constructs an adaptive neighbor graph to respect the local structure, which carries out both local structure learning and matrix factorization [43].

13) **KLS-NMF**: It introduces self-expressiveness mechanism into matrix factorization to learn the local similarity of data in the kernel space [44].

14) **TS-NMF**: It focuses on enhancing clustering by seeking the optimal projection and jointly constructing a new representation of the data [57].

15) **NMF2L\_\_**: It combines manifold learning and feature learning into NMF to improve the performance of clustering, and presents two algorithms NMF2L\_\_ and NMF2L\_\_. Because NMF2L\_\_ outperforms NMF2L\_\_, NMF2L\_\_ is chosen to carry out the experimental comparison.

### C. EXPERIMENTAL SETTING

For simplicity, we apply a random strategy to initialize two low-rank nonnegative matrix factors $W$ and $H$ in our experiments. We tune the regularization parameters within $[10^0, 10^3, 10^4, ..., 10^4, 10^5, 10^6]$ for all the compared algorithms. The number of the nearest neighbors $p$ is set to 5 for GCNMF, SGFS, SPFS, and RSCNMF. Since the compared methods in this paper are unsupervised, we use all the samples in the data set as test ones. $K$-means algorithm is adopted to cluster the learned representations. For a given number of clusters $K$, we perform 20 runs of experiments independently and report the average clustering results of these 20 runs.

Following [2],[6],[21],[23],[24], two widely used metrics, accuracy (ACC) normalized mutual information (NMI), are used to measure the effectiveness of the compared methods. The larger ACC and NMI are, the better the performances are. ACC is defined as follows:

\[
ACC = \frac{1}{n} \sum_{i=1}^{n} \delta(g_{i}, map(f_{i})),
\]

where $g_i$ is the ground-truth label provided by the data set, and $f_i$ denotes the clustering label. In (61), $\text{map}(f_i)$ is an optimal permutation function that maps the clustering labels to the ground-truth labels. The indicator function $\delta(c,d)$ is 1 if $c = d$, and 0 otherwise. NMI is formulated as

\[1\text{https://cswww.essex.ac.uk/mv/allfaces/index.html}
\[2\text{http://featureselection.asu.edu/datasets.php}
\[3\text{http://www.cad.zju.edu.cn/home/dengcai/Data/data.html}
where $n_i$ is the number of overlaps between the cluster $C_i$ and the $j$-th class, $n_i$ is the number of samples in cluster $C_i$ ($1 \leq i \leq K$) obtained by the clustering algorithm, and $\bar{n}_j$ denotes the number of samples belonging to the $j$-th ground-truth class ($1 \leq j \leq K$).

### D. EXPERIMENTAL RESULTS

Tables 2 and 3 show ACC and NMI obtained by nine compared algorithms, respectively. We report the mean and standard error of the clustering performance on the eight data sets. The best performance is highlighted in bold. We can get a number of interesting points from Tables 2 and 3.

1) It can be observed that RSCNMF performs better than all the compared algorithms on eight data sets. Especially, on the SRBCTML data sets, the NMI value of our algorithm is about 13% higher than that of other methods. This indicates that RSCNMF can learn more discriminative representations from mixed-sign data. Although the NMI value of SC-CNMF is close to that of RSCNMF on the COIL20 data set, our algorithm has superior performance on other data sets. For example, on SRBCTML, Lymphoma and GLI_85, the NMI of RSCNMF is improved by at least 10% compared with other algorithms. The corresponding ACC value is higher than that of state-of-the-art algorithms. Intuitively, this is due to the fact that the global and local structures of the data play an essential role in discovering a discriminative representation subspace.

2) SCNMF performs as well as RSCNMF on the MALDIDML data set. Its performance, however, is relatively worse than RSCNMF on the remaining seven datasets. This conclusion is consistent with that of [19],[47],[48], in which L_2,1-norm-based NMFs are more effective than L_2-norm-based NMFs. It is worth noting that SCNMF achieves better results than other compared methods on ORL, COIL20, Face96, Lymphoma, MLLML, and GLI_85. Although ACC of SCNMF is lower than that of MNMF-L_2,1 on the SRBCTML data set, its NMI is higher than that of RSCNMF. SCNMF degrades to GCNMF when the global structure is ignored. As we can see, SCNMF outperforms GCNMF on all eight data sets. This indicates that the global structure can play an important role in matrix factorization. SCNMF degenerates into CNMF when both global and local structures are not taken into account. Obviously, CNMF is inferior to SCNMF and GCNMF. This is consistent with our motivation to integrate both local and global structures into CNMF to enhance clustering.

3) TS-NMF is inferior to RSCNMF. It performs slightly better than SCNMF on the SRBCTML data set. However, its performance is lower than that of SCNMF on the other data sets. Although TS-NMF is inferior to the proposed algorithm, it outperforms other compared methods on ORL, SRBCTML and COIL20. TS-NMF also provides the comparable performance on the remaining data sets. Hence, it is a relatively effective approach. NMF2L_2,1 is inferior to the proposed algorithm. It is superior to other NMF-based methods on the ORL data set.

4) Although KLS-NMF performs as well as the proposed RSCNMF on the MALDIDML data sets, its performance is poor on other data sets. This indicates that KLS-NMF cannot effectively capture the global structure in kernel space by encoding self-expression attributes. Although the performance of SC-CNMF is relatively poor on the GLI_85 data set, it is superior to baseline and other six algorithms on ORL, COIL20, Lymphoma and MLLML data sets. SC-CNMF also achieves relatively satisfactory performance on SRBCTML, MALDIDML and Face96 data sets. Clearly, its learning performance can be evidently improved if the global structure is respected. However, SC-CNMF is inferior to RSCNMF. The reason is that SC-CNMF ignores the local geometric structure and thus fails to obtain more discriminative representations.

5) GCNMF, SGFS and SPFS are matrix factorization methods based on the local structure. They construct the nearest neighbor graph to learn the representations and thus achieve better performance than CNMF. This shows that it is both useful and necessary to directly consider the local geometric structure in learning hidden factors. Besides, although RMFSS and NSSLFS do not exploit the structure of the data, their performances perform better than that of CNMF. The reason is that the latent low-dimensional structure can be captured by using a combination of $L_1$-norm and $L_2,1$-norm imposed on the feature matrix.

6) Since Lymphoma, MLLML and MALDIDML contain negative entries, NMF-based methods, such as, MNMF-L_2,1, NMF-SCAG, RGNMF and NMFAN, fail to tackle these data. Hence, we can only give their experimental results on other five data sets. It can be observed that the proposed RSCNMF consistently outperforms MNMF-L_2,1, NMF-SCAG, RGNMF and NMFAN. In essence, the four algorithms focus on the local structure and ignore the global structure. Therefore, the learning performance is limited. This shows that the performance of the algorithm can be improved if the global structure is taken into consideration.
E. PARAMETER SENSITIVENESS

The proposed algorithm needs to set three essential parameters $\alpha$, $\beta$ and $\lambda$ in advance. In fact, when three parameters are set to 0, our RSCNMF degrades to the $L_{2,1}$-norm CNMF. When $\beta = 0$ and $\lambda = 0$, the proposed algorithm reduces to the $L_{2,1}$-norm GCNMF. Consequently, the proposed framework is a general one that can leverage the power of CNMF, $L_{2,1}$-norm, and structure regularization. Figs. 2-5 show how the clustering performance of RSCNMF varies with three parameters $\alpha$, $\beta$ and $\lambda$ on ORL, SRBCMTL, Lymphoma and MLLML, respectively. We can obtain similar observations on the remaining data sets and therefore omit those.
Different regularization parameters have different effects on an approach. Three parameters in the proposed algorithm also act out different effects, as shown in Figs. 2-5. Generally, RSCNMF achieves consistently better performance when the local regularization parameter $\alpha$ is in the interval of $[10^3, 10^6]$. If $\alpha$ is given a small value, it is difficult for RSCNMF to provide a satisfied performance. If $\alpha$ is greater than 1000, RSNMF is superior to other compared methods. On the SRBCTML data set, for example, the ACC and NMI values of RSNMF are relatively low when $\alpha < 1000$. However, when $\alpha > 1000$, RSNMF obtains significantly good performance. The ACC and NMI values of RSCNMF increase with the global regularization parameter $\beta$ varying from 0.001 to 1. When $\beta > 1$, RSNMF trends to be stable and outperform the compared methods. From the experiments, we find that the optimal value of $\beta$ is in $[10^0, 1000]$. The change of the $L_{2,1}$-norm parameter $\lambda$ has little effect on the performance of RSNMF. We set $\lambda$ to the range of $[1, 100]$ in the experiments.
We have presented an iterative multiplicative scheme to solve the new model. The update rules have been proven to be convergent. In this section, we experimentally verify the convergence of this scheme. The convergence curves of RSCNMF are shown in Fig. 6. For each subfigure, the x-axis represents the iteration number in Algorithm 1, and the y-axis denotes the value of the cost function. From the experiments, we can observe that the multiplicative updates of our proposed RSCNMF converge within 200 iterations for eight data sets. The fast convergence speed demonstrates that the proposed optimization method is effective. Obviously, the objective function of RSCNMF is convex when one variable is fixed and the other variable is solved. Fig. 7 shows the convergence curves of each variable on ORL and Lymphoma data sets. The similar results can be obtained on the remaining data sets and are thus omitted. We can observe that one variable can converge quickly when the other variable is fixed. Next,
we discuss the convergence rate of our algorithm. Following [56], we analyze the convergence rate of our algorithm by experiments. We set the solution of $W^*$ and $H^*$ to be $W^{(t)}$ and $H^{(t)}$, respectively, when $|O_{i, 1} - O_i| / O_i \times 10^6$ is met. To clearly demonstrate the convergence of $\{W^{(t)}\}$ and $\{H^{(t)}\}$, we test the sequences of $\{U_i = \|W^{(t)} W^*\|_F\}$ and $\{V_i = \|H^{(t)} H^*\|_F\}$, respectively. We further verify the convergence rate by displaying the curves of $\{A_i = \|W^{(t+1)} W^*\|_F\}$ and $\{B_i = \|H^{(t+1)} H^*\|_F\}$. The curves of the convergence rate are shown in Fig. 8 on COIL20 and SRBCTML data sets. The similar results can be obtained from the other data sets and thus omitted. We can observe that both $A_i$ and $B_i$ are less than 1 and tend to go down. This shows that the convergence rate of the proposed algorithm is fast.

V. CONCLUSION AND FUTURE WORK

We proposed a novel unsupervised matrix factorization algorithm, called RSCNMF, which extends the CNMF method by explicitly respecting local and global structures of the data. Specifically, RSCNMF is able to learn more discriminative representations via investigating the underlying structures of the data, and is robust to noise and outliers via introducing the $L_{2,1}$-norm based loss function. Consequently, the proposed algorithm can leverage the intrinsic geometric structure and the $L_{2,1}$-norm minimization imposed on both the loss function and the feature representation. Moreover, we characterize local and global structures and $L_{2,1}$-norm as a general framework of matrix factorization, which is solved by a developed optimization scheme. The convergence of the optimization scheme is also proven. Our empirical study on eight real-world data sets demonstrates that the performance of the proposed algorithm outperforms or matches state-of-the-art methods. In the future, we will integrate representation learning and adversarial learning into RSCNMF to extend its application.

REFERENCES

[1] D.D. Lee and H.S. Seung, “Learning the parts of objects by nonnegative matrix factorization,” Nature, vol. 401, pp. 788–791, Oct. 1999.
[2] D. Cai, X. He, J. Han, and T. S. Huang, “Graph regularized nonnegative matrix factorization for data representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8, pp. 1548–1560, Aug. 2011.
[3] M. Asteris, D. Papailiopoulos, A. G. Dimakis, “Orthogonal NMF through Subspace Exploration,” Advances in Neural Information Processing Systems, Dec. 2015, pp. 343-351.
[4] X. Pei, T. Wu, and C. Chen, “Automated graph regularized projective nonnegative matrix factorization for document clustering,” IEEE Trans. Cybern., vol. 44, no. 10, pp. 1821–1831, Oct. 2014.
[5] X. Teng, L. Lan, X. Zhang, G. Dong, and Z. Luo, “Transductive nonnegative matrix tri-factorization,” IEEE Access, vol. 8, pp. 81331–81347, May, 2020.
[6] H. Xiong, D. Kong, “Elastic nonnegative matrix factorization,” Pattern Recognition, vol. 90, pp. 464–475, 2019.
[7] W. D. Wang, F. Nie, and H. Huang, “Fast robust non-negative matrix factorization for largescale human action data clustering,” In: Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI), Jul. 2016, pp. 2104–2110.
[8] S. Zafeiriou, A. Tefas, I. Buciuc, and I. Pitas, “Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification,” IEEE Trans. Neural Netw., vol. 17, no. 3, pp. 683–695, May 2006.
[9] S. An, J. Yoo, and S. Choi, “Manifold-respecting discriminant nonnegative matrix factorization, Pattern Recognit. Lett.” vol. 32, pp. 832–837, 2011.
[10] N. Guan, D. Tao, Z. Luo, and B. Yuan, “Manifold regularized discriminant nonnegative matrix factorization with fast gradient descent,” IEEE Trans. Image Process., vol. 20, no. 7, pp. 2030–2048, Jul. 2011.
[11] X. Li, G. Cui, and Y. Dong, “Discriminative and orthogonal subspace constraints-based nonnegative matrix factorization,” ACM Trans. Intell. Syst. Technol., vol.9, no.6, pp.65-84, Nov. 2018.
[12] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, “Constrained nonnegative matrix factorization for image representation,” IEEE Trans. Pattern Anal.Mach. Intell., vol.34, no.7, pp.1299–1311, Jul. 2012.
[13] X. Zhang, L. Zong, X. Liu, and J. Luo, “Constrained clustering with nonnegative matrix factorization,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 7, pp. 1514–1526, Jul. 2016.
[14] D. Wang, X. Gao, and X. Wang, “Semi-Supervised nonnegative matrix factorization via constraint propagation,” IEEE Trans. Cybern., vol. 46, pp. 233-244, Jan. 2016.
[15] Z. Li, J. Tang, and X. He, “Robust structured nonnegative matrix factorization for image representation,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 5, pp. 1947-1960, May. 2018.
[16] X. Li, G. Cui, and Y. Dong, “Graph regularized non-negative low-rank matrix factorization for image clustering,” IEEE Trans. Cybern., vol. 47, no. 11, pp. 3840–3853, Nov. 2017.
[17] L. Zhang, Q. Zhang, B. Du, J. You, D. Tao, “Adaptive manifold regularized matrix factorization for data clustering,” in Proc. 26th Int. Joint Conf. Artif. Intell., pp. 3399-3405, Aug. 2017.
[18] Y. Yi, J. Wang, W. Zhou, C. Zheng, J. Kong and S. Qiao, “Non-negative matrix factorization with locality constrained adaptive graph,” IEEE Trans. Circuits Syst. Video Techn., vol. 30, no.2, pp. 427-441, Feb. 2020.
[19] J. Huang, F. Nie, H. Huang, and C. Ding, “Robust manifold nonnegative matrix factorization,” ACM Trans. Know. Disc. from Data, vol. 8, no.3, pp. 11-31, Jun. 2014.
[20] D C. Ding, T. Li, and M. J. Jordan, “Convex and semi-nonnegative matrix factorizations,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 1, pp. 45–55, Jan. 2010.
[21] B. Wang, W. Pedrycz, Q. Zhu, W. Zhu, “Subspace learning for unsupervised feature selection via matrix factorization, Pattern Recognition, vol. 48, pp.10-19, Jan. 2015.
[22] Z. Wang, F. Nie, R. Wang, H. Yang, & X. Li, “Local structured feature learning with dynamic maximum entropy graph,” Pattern Recognition, vol. 111, pp.107673, Mar. 2021.
[23] W. Hu, K. Choi, P. Wang, Y. Jiang, S. Wang, “Convex nonnegative matrix factorization with manifold regularization,” Neural Networks, vol. 63, pp. 94–103, Mar. 2015.
[24] R. Shang, W. Wang, R. Stolkin, L. Jiao, “Subspace learning-based graph regularized feature selection,” Knowledge-Based Systems, vol. 112, pp. 152–165, Nov. 2016.
[25] N. Zhou, Y. Xu, H. Cheng, J. Fang, W. Pedrycz, “Global and local structure preserving sparse subspace learning: An iterative approach to unsupervised feature selection,” Pattern Recognition, Vol. 53, pp. 87–101, May. 2016.
[26] M. Qi, T. Wang, F. Liu, B. Zhang, J. Wang, Y. Yi, “Unsupervised feature selection by regularized matrix factorization,” Neurocomputing, vol. 273, pp. 593-610, Jan. 2018.
[27] G. Cui, X. Li, Y. Dong, “Subspace clustering guided convex nonnegative matrix factorization,” Neurocomputing, vol. 292, pp. 38–48, May. 2018.
[28] W. Zheng, H. Yan, J. Yang, “Robust unsupervised feature selection by nonnegative sparse subspace learning, Neurocomputing,” vol. 334, pp. 156–171, Mar. 2019.
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3128975, IEEE Access