

# Nonnegative Matrix Factorization with Local Similarity Learning

Chong Peng<sup>a,\*</sup>, Zhilu Zhang<sup>a,\*</sup>, Zhao Kang<sup>b</sup>, Chenglizhao Chen<sup>a,\*\*</sup>, Qiang Cheng<sup>c,d</sup>

<sup>a</sup>*College of Computer Science and Technology, Qingdao University*

<sup>b</sup>*School of Computer Science and Engineering, University of Science and Technology of China*

<sup>c</sup>*Department of Computer Science, University of Kentucky*

<sup>d</sup>*Institute of Biomedical Informatics, University of Kentucky*

---

## Abstract

Existing nonnegative matrix factorization methods usually focus on learning global structure of the data to construct basis and coefficient matrices, which ignores the local structure that commonly exists among data. To overcome this drawback, in this paper, we propose a new type of nonnegative matrix factorization method, which learns local similarity and clustering in a mutually enhanced way. The learned new representation is more representative in that it better reveals inherent geometric property of the data. Moreover, the new representation is performed in the kernel space, which enhances the capability of the proposed model in discovering nonlinear structures of data. Multiplicative updating rules are developed with theoretical convergence guarantees. Extensive experimental results have confirmed the effectiveness of the proposed model.

*Keywords:* nonnegative matrix factorization, clustering, local similarity

---

## 1. Introduction

High-dimensional data are ubiquitous in the learning community and it has become increasingly challenging to learn from such data [1]. For example, as

---

\*Chong Peng and Zhilu Zhang contribute equally to this paper.

\*\*Corresponding author: Chenglizhao Chen

Email address: cc1z123@163.com (Chenglizhao Chen)

one of the most important tasks in multimedia and data mining, information retrieval has drawn considerable attentions in recent years [2, 3, 4], where there is often a need to handle high-dimensional data. Often times, it is desirable and demanding to seek a data representation to reveal latent data structures of high-dimensional data, which is usually helpful for further data processing. It is thus a critical problem to find a suitable representation of the data in many learning tasks, such as image clustering and classification [5, 1], foreground-background separation in surveillance video [6, 7], matrix completion [8], community detection [9], link prediction [10], etc. To this end, a number of methods have been developed to seek proper representations of data, among which matrix factorization technique has been widely used to handle high-dimensional data. Matrix factorization seeks two or more low-dimensional matrices to approximate the original data such that the high-dimensional data can be represented with reduced dimensions [11, 12].

For some types of data, such as images and documents, the entries are naturally nonnegative. For such data, nonnegative matrix factorization (NMF) was proposed to seek two nonnegative factor matrices for approximation [13]. In fact, the non-negativity constraints of NMF naturally leads to learning parts-based representations of the data [13]. It is believed that parts-based representation commonly exist in human brain with psychological and physiological evidence [14, 15, 16]. It overcomes the drawback of latent semantic indexing (LSI) [17], for which the interpretation of basis vectors is difficult due to mixed signs. When the number of basis vectors is large, NMF has been proven to be NP-hard [18]; moreover, [19] has recently given some conditions, under which NMF is solvable. Recent studies have shown a close relationship between NMF and K-means [20], and further study has shown that both spectral clustering and kernel K-means [21] are particular cases of clustering with NMF under a doubly stochastic constraint [22]. This implies that NMF is especially suitable for clustering such data. In this paper, we will develop a novel NMF method, which focuses on the clustering capability.

In the past a few decades, various variants of NMF method have been de-

veloped, which can be mainly categorized into four types, including basic NMF [13], constrained NMF [23], structured NMF [24], and generalized NMF [25]. A fairly comprehensive review can be found in [26]. Among these methods, Semi-NMF [27] removes the nonnegative constraints on the data and basis vectors, which allows NMF-type methods to be used in more applications; convex NMF (CNMF) [27] restricts the basis vectors to lie in the feature space of the input data so that they can be represented as convex combinations of data vectors; orthogonal NMF (ONMF) [23] imposes orthogonality constraints on factor matrices, which leads to clearer clustering interpretation. Different from ONMF, [28] restricts orthogonality constraint on the basis factors, which renders the overlapped basis vectors as in other NMF models to be distinct. The classic NMF only considers the linear structures of the data and ignores the nonlinear ones, which may lead to less competitive performance since nonlinear structures are usually important for many applications such as clustering. To learn the latent nonlinear structures of the data, graph regularized nonnegative matrix factorization (GNMF) considers the intrinsic geometrical structures of the data on a manifold by incorporating a Laplacian regularization [29]. By modeling the data space as a manifold embedded in an ambient space, GNMF considers both linear and nonlinear relationships of the data points in the original instance space, and thus it is also more discriminating than ordinary NMF which only considers the Euclidean structure of the data [29, 28]. This renders GNMF more suitable for clustering purpose than the original NMF. Based on GNMF, robust manifold nonnegative matrix factorization (RMNMF) constructs a structured sparsity-inducing norm-based robust formulation [30]. With a  $\ell_{2,1}$ -norm, RMNMF is insensitive to the between-sample data outliers and improves the robustness of NMF [30]. Moreover, the relaxed requirement on signs of the data makes it a nonlinear version of Semi-NMF. Some recent works also restrict manifold structure on both factor matrices [24]. Other than seeking two factor matrices, some recent works seek the product of three or even more nonnegative matrices to approximate the original nonnegative data matrix, which leads to nonnegative matrix tri-factorization [31, 24] and deep NMF [32].

In recent years, the importance of preserving local manifold structure has drawn considerable attentions in research community of machine learning, data mining, and pattern recognition [33, 34, 35]. Besides pairwise sample similarity, it has been shown that local geometric structure of the data is also crucial in revealing underlying structure of the data [33]: 1) In the transformed low-dimensional space, it is important to maintain the intrinsic information of high-dimensional data [36]; 2) It may be insufficient to represent the underlying structures of the data with a single characterization and both global and local ones are necessary [37]; 3) In some ways, we can regard the local geometric structure of the data as data dependent regularization, which helps avoid over-fitting issues [33]. Despite the importance of local structure of data, it is yet to be exploited in NMF study. In this paper, we propose a new type of NMF method, which simultaneously learns both similarity and geometric structures of the data and clustering such that the learned basis and coefficient matrices well preserve discriminative information of the data. Recent studies reveal that high-dimensional data often reside in a union of low-dimensional subspaces and the data can be self-expressed by a low-dimensional representation [11, 38], which can be regarded as pairwise similarity of samples. Instead of simply using pairwise similarity of samples, in our method, we transform the pairwise similarity into the similarity between a score vector of a sample on basis and the representation of another sample in the same cluster, which integrates basis and coefficient learning into simultaneous similarity learning and clustering. To better reveal the underlying nonlinear structures among data, we incorporate the kernel method in our model such that both local and global nonlinear relationships of the data can be considered in a seamlessly integrated model.

The main contributions of this paper are outlined as follows:

- For the first time, in an effective yet simple way, local similarity learning is embedded into learning matrix factorization, which allows our method to learn global and local structures of the data. The learned basis and representations well preserve the inherent structures of the data and are

more representative;

- To our best knowledge, we are the first to integrate the orthogonality-constrained coefficient matrix into local similarity adaption, such that local similarity and clustering can mutually enhance each other and be learned simultaneously;
- Nonlinear structures of data are considered in kernel space, which allows our model to count both local and global nonlinear structures of data;
- Efficient multiplicative updating rules are constructed to solve the proposed model and comprehensive theoretical analysis is provided to guarantee the convergence;
- Lastly, extensive experimental results have verified the effectiveness of our method.

The rest of this paper is organized as follows: In Section 2, we briefly review some methods that are closely related with our research. Then we introduce our method in Section 3. We provide an efficient alternating optimization procedure in Section 4, and then theoretically analyze the convergent behavior of the optimization in Section 5. Next, we conduct extensive experiments in Section 6. Finally, we conclude the paper in Section 7.

**Notation:** For a matrix  $M$ ,  $M_{ij}$ ,  $M_i$ , and  $M_{\bar{j}}$  denote the  $ij$ -th element,  $i$ -th column, and  $j$ -th row of  $M$ .  $\text{Tr}(\cdot)$  is the trace operator,  $\|\cdot\|_F$  and  $\|\cdot\|_2$  are the Frobenius and  $\ell_2$  norms.  $I_k$  denotes the identity matrix of size  $k \times k$ ,  $\text{diag}(\cdot)$  is an operator that returns a diagonal matrix with identical diagonal elements to the input matrix.

## 2. Related Work

In this section, we briefly review some techniques that are closely related with our research, including NMF and graph Laplacian.

### 2.1. NMF

Given nonnegative data  $X = [x_1, \dots, x_n] \in \mathcal{R}^{p \times n}$  with  $p$  and  $n$  being the dimension and sample size, respectively, NMF is to factor the nonnegative matrix  $X$  into  $U \in \mathcal{R}^{p \times k}$  (basis) and  $G \in \mathcal{R}^{n \times k}$  (coefficients) with the following optimization problem:

$$\min_{U \geq 0, G \geq 0} \|X - UG^T\|_F^2, \quad (1)$$

where  $k \ll n$  enforces a low-rank approximation of the original data. Usually, multiplicative updating rules are designed for the optimization of NMF problems.

### 2.2. Graph Laplacian

Graph Laplacian [39] is defined as

$$\begin{aligned} & \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \|G_i - G_j\|_2^2 S_{ij}^x \\ &= \sum_{j=1}^n D_{jj}^x G_j^T G_j - \sum_{i=1}^n \sum_{j=1}^n S_{ij}^x G_i^T G_j, \\ &= \mathbf{Tr}(G^T D^x G) - \mathbf{Tr}(G^T S^x G) = \mathbf{Tr}(G^T L^x G), \end{aligned} \quad (2)$$

where  $S^x$  is the weight matrix that measures the pair-wise similarities of original data points,  $D^x$  is a diagonal matrix with  $D_{ii}^x = \sum_j S_{ij}^x$ , and  $L^x = D^x - S^x$ . It is widely used to incorporate the geometrical structure of the data on manifold. In particular, the manifold enforces the smoothness of the data in linear and nonlinear spaces by minimizing (2), which leads to an effect that if two data points are close in the intrinsic geometry of the data distribution, then their new representations with respect to the new basis,  $G_i$  and  $G_j$ , are also close [29].

## 3. Proposed Method

As aforementioned, existing NMF methods do not fully exploit local geometric structures, nor do they exploit close interaction between local similarity and

clustering. In this section, we will propose an effective, yet simple, new method to overcome these two drawbacks.

CNMF restricts the basis of NMF to convex combinations of the columns of the data, i.e.,  $U = XW$ , which gives rise to the following:

$$\min_{W \geq 0, G \geq 0} \|X - XWG^T\|_F^2. \quad (3)$$

Here, (3) is also known as the concept factorization [40]. By restricting  $U = XW$ , (3) has the advantage that it could interpret the columns of  $U$  as weighted sums of certain data points and these columns correspond to centroids [27]. It is natural to see that  $W_{ij}$  reveals the importance of basis  $U_j$  to  $x_i$  by the value of  $W_{ij}$ .

It is noted that (3) is closely related to subspace clustering [11, 38, 41]. The observation is that high-dimensional data usually reside in low-dimensional subspaces and it usually has self-expressiveness assumption to recover such subspaces. The self-expressiveness assumption refers to that the data  $X$  can be approximately expressed as  $X \approx XZ$  with  $Z$  being a representation matrix. Local structures of the data are shown to be important [42] and it is necessary to take into consideration local similarity in learning tasks. A natural assumption is that if two data points  $x_i$  and  $x_j$  are close to each other, then their similarity,  $Z_{ij}$ , should be large; otherwise,  $Z_{ij}$  small. This assumption leads to the following minimization:

$$\min_Z \sum_{ij} \|x_i - x_j\|_2^2 Z_{ij} \Leftrightarrow \min_Z \text{Tr}(Z^T D), \quad (4)$$

where

$$D_{ij} = \|x_i - x_j\|_2^2,$$

or in matrix form,

$$D = \mathbf{1}_n \mathbf{1}_n^T \text{diag}(X^T X) + \text{diag}(X^T X) \mathbf{1}_n \mathbf{1}_n^T - 2X^T X,$$

with  $\mathbf{1}_n$  being a length- $n$  vector of 1s. It is noted that the minimization of (4) directly enforces  $Z_{ij}$  to reflect the pair-wise similarity information of the examples. Noticing that  $W$  and  $G$  are nonnegative and inspired by self-expressiveness

assumption, we take  $WG^T$  as the similarity matrix  $Z$ , such that  $Z_{ij} = W_i G_j^T$ . Here,  $W_i$  is the **score vector** of example  $x_i$  on the basis vectors, and  $G_j$  is the **coefficient vector** of the  $j$ -th sample with respect to the new basis. If  $x_i$  and  $x_j$  are close on data manifold or grouped into the same cluster, then it is natural that  $W_i$  and  $G_j$  have higher similarity; vice versa. This close relationship between the geometry of  $x_i$  and  $x_j$  on data manifold and the similarity of  $W_i$  and  $G_j$  suggests that using  $WG^T$  as  $Z$  in (4) is indeed meaningful. To encourage the interaction between similarity learning and clustering, we incorporate (4) into (3) with  $Z = WG^T$ , obtaining the following model:

$$\begin{aligned} \min_{W, G} \quad & \frac{1}{2} \|X - XWG^T\|_F^2 + \lambda \text{Tr}(W^T DG), \\ \text{s.t.} \quad & W \geq 0, G \geq 0. \end{aligned} \tag{5}$$

where  $\lambda \geq 0$  is a balancing parameter. Now, it is seen that the first term in above model captures global structure of the data by exploiting linear representation of each example with respect to the overall data, while the second term exploits local structure of the data by the connection between local geometric structure and pairwise similarity. It should be noted that there is remarkable difference between the proposed method and existing NMF methods, such as [29, 30, 43], in learning local similarity. In the literature, the graph Laplacian in (2) is often used to incorporate local structural information from data, which enforces the smoothness between the data and the coefficient vectors. By minimizing the distance of two new representation vectors whose corresponding examples have high similarity, the graph Laplacian enforces the intra-class similarity of the data on the new representation. The proposed method exploits local similarity from data with the second term of (5), which enforces the smoothness between the data and both score and coefficient vectors. Thus, the new method renders both score and coefficient vectors to consider local structures of the data. Different from the graph Laplacian, (5) tries to minimize the score and coefficient vectors-based pair-wise similarity of two examples if they have large distance, which renders the pair-wise similarity matrix, as well as the score and coefficient vectors, to have strong inter-class separability. To allow for immediate inter-



pretation of clustering from the coefficient matrix, we impose an orthogonality constraint of  $G$ , i.e.,  $G^T G = I_k$ , leading to the following model:

$$\begin{aligned} \min_{W, G} \quad & \frac{1}{2} \|X - XWG^T\|_F^2 + \lambda \text{Tr}(W^T DG), \\ \text{s.t.} \quad & W \geq 0, G \geq 0, G^T G = I_k. \end{aligned} \quad (6)$$

It should be noted that the constraint  $G^T G = I_k$  directly connects the problem of NMF with clustering in that  $G$  can be regarded as relaxed cluster indicators. More importantly, learning similarity and clustering are connected through such a  $G$  matrix and can be mutually promoted through an iterative optimization process.

Model (6) only learns linear relationships of the data and omits the nonlinear ones, which usually exist and are important. To take nonlinear relationships of the data into consideration, it is widely considered to seek data relationships in kernel space. We define a kernel mapping as  $\phi : \mathcal{R}^p \rightarrow \mathcal{R}^{\bar{p}}$ , which maps the data points  $x_i \in \mathcal{R}^p$  from the input space to  $\phi(x_i) \in \mathcal{R}^{\bar{p}}$  in a reproducing kernel Hilbert space  $\mathcal{H}$ , where  $\bar{p}$  is an arbitrary positive integer. After nonlinear mapping, we obtain the mapped sample matrix  $\phi(X) = [\phi(x_1), \dots, \phi(x_n)]$ . The similarity between each pair of data points is defined as the inner product of mapped data in the Hilbert space, i.e.,  $\mathbf{k}(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \phi(x_i)^T \phi(x_j)$ , where  $\mathbf{k}(\cdot, \cdot) : \mathcal{R}^{p \times p} \rightarrow \mathcal{R}$  is a reproducing kernel function. In the kernel space, (6) becomes

$$\begin{aligned} \min_{W, G} \quad & \frac{1}{2} \|\phi(X) - \phi(X)WG^T\|_F^2 + \lambda \text{Tr}(W^T D^\phi G), \\ \text{s.t.} \quad & W \geq 0, G \geq 0, G^T G = I_k, \end{aligned} \quad (7)$$

where  $D^\phi$  is extended  $D$  in (6) from instance space to kernel space, which is defined as

$$D^\phi = \mathbf{1}_n \mathbf{1}_n^T \text{diag} \left( \phi(X)^T \phi(X) \right) + \text{diag} \left( \phi(X)^T \phi(X) \right) \mathbf{1}_n \mathbf{1}_n^T - 2\phi(X)^T \phi(X). \quad (8)$$

We expand (7) and replace  $\phi(X)^T \phi(X)$  with  $K$ , the kernel matrix induced by kernel function associated with the mapping  $\phi(\cdot)$ , giving rise to the following

model, which is named Kernel Local Similarity NMF (KLS-NMF):

$$\begin{aligned} \min_{W, G} & \frac{1}{2} \text{Tr}(K - 2KWG^T + GW^T KWG^T) + \lambda \text{Tr}(W^T D^\phi G), \\ \text{s.t.} \quad & W \geq 0, G \geq 0, G^T G = I_k, \end{aligned} \quad (9)$$

where  $D^\phi = (D^\phi)^T = \mathbf{1}_n \mathbf{1}_n^T \text{diag}(K) + \text{diag}(K) \mathbf{1}_n \mathbf{1}_n^T - 2K$ . It should be noted that there is a stark difference between the KLS-NMF and existing NMF methods that learn local structure of data, such as [29, 30, 43]. Existing methods usually exploit local structure of data on manifold, which considers nonlinear structure of data if a nonlinear similarity scheme is used. However, when exploiting global information, they try to minimize the fitting error in Euclidean space, which rarely consider and thus do not fully extract nonlinear structure of data. It is seen that the KLS-NMF seeks structures of data in a starkly different way, which exploits both local and global structures of data in kernel space and thus explicitly considers nonlinear structures of data. For its optimization, we will develop and present an efficient algorithm in the next section.

*Remark.* In this paper, we aim at providing a new NMF method to take both local and global nonlinear relationships of the data into consideration. It has been discussed that the proposed KLS-NMF learns both local and global information in nonlinear space in a way starkly different from existing methods. It is also worth mentioning that our method can be extended to multiple-kernel scenario. Since the future extension is out of the scope of this paper, we do not further explore it here.

#### 4. Optimization

We solve (9) using an iterative update algorithm and element-wisely update  $W$  and  $G$  as follows:

$$W_{ik} \leftarrow W_{ik} \sqrt{\frac{(KG)_{ik}}{(KWG^T G)_{ik} + \lambda(D^\phi G)_{ik}}} \quad (10)$$

$$G_{ik} \leftarrow G_{ik} \sqrt{\frac{(KW)_{ik} + (\lambda G G^T D^\phi W)_{ik}}{\lambda(D^\phi W)_{ik} + (G G^T KW)_{ik}}} \quad (11)$$

By counting dominating multiplications, it is seen that the complexity of (10) and (11) per iteration is  $O(n^2p + n^2k)$ . The correctness and convergence proofs of the updates are provided in the following section.

## 5. Correctness and Convergence

In this section, we will present theoretical results regarding the updating rules given in (10) and (11), respectively.

### 5.1. Correctness and Convergence of (10)

For the updating rule of (10), we present the following two main results:

- 1) When convergent, the limiting solution of (10) satisfies the KKT condition.
- 2) The iteration of (10) converges. We formally establish the above results in Theorems 1 and 2, respectively.

**Theorem 1.** *Fixing  $G$ , the limiting solution of the updating rule in (10) satisfies the KKT condition.*

*Proof.* Fixing  $G$ , the subproblem for  $W$  is

$$\min_{W \geq 0} \frac{1}{2} \text{Tr}(-2KWG^T + GW^T KWG^T) + \lambda \text{Tr}(W^T D^\phi G), \quad (12)$$

Imposing the non-negativity constraint  $W_{ik} \geq 0$ , we introduce the Lagrangian multipliers  $\Psi = [\psi_{ij}]$  and the Lagrangian function

$$\begin{aligned} \mathcal{L}_W = & \frac{1}{2} \text{Tr}(-2KWG^T + GW^T KWG^T) \\ & + \lambda \text{Tr}(W^T D^\phi G) + \text{Tr}(\Psi W^T). \end{aligned} \quad (13)$$

The gradient of  $\mathcal{L}_W$  is

$$\frac{\partial \mathcal{L}_W}{\partial W} = -KG + \lambda D^\phi G + KWG^T G + \Psi. \quad (14)$$

For ease of notation, we denote  $\bar{A} = KG$ ,  $\bar{B} = D^\phi G$ ,  $\bar{C} = K$ , and  $\bar{D} = G^T G$ . By the complementary slackness condition, we obtain

$$(-\bar{A} + \lambda \bar{B} + \bar{C} W \bar{D})_{ik} W_{ik} = \psi_{ik} W_{ik} = 0. \quad (15)$$

Note that (15) provides the fixed point condition that the limiting solution should satisfy. It is easy to see that the limiting solution of (10) satisfies (15), which is described as follows. At convergence, (10) gives

$$W_{ik} = W_{ik} \sqrt{\frac{(\bar{A})_{ik}}{(\bar{C}W\bar{D})_{ik} + \lambda(\bar{B})_{ik}}}, \quad (16)$$

which is reduced to

$$(-\bar{A} + \lambda\bar{B} + \bar{C}W\bar{D})_{ik} W_{ik}^2 = 0 \quad (17)$$

by simple algebra. It is easy to see that (15) and (17) are identical in that both of them enforce either  $W_{ik} = 0$  or  $(-\bar{A} + \lambda\bar{B} + \bar{C}W\bar{D})_{ik} = 0$ .  $\square$

Next, we prove the convergence of the iterative updating rule as stated in Theorem 2.

**Theorem 2.** *For fixed  $G$ , (12), as well as (9), is monotonically decreasing under the updating rule in (10).*

To prove the above theorem, we use an auxiliary function approach [44] with relevant definition and propositions given below.

**Definition 1.** A function  $J(H, H')$  is called an auxiliary function of  $L(H)$  if for any  $H$  and  $H'$  the following are satisfied

$$J(H, H') \geq L(H), \quad J(H, H) = L(H). \quad (18)$$

**Proposition 1.** *Given a function  $L(H)$  and its auxiliary function  $J(H, H')$ , if we define a variable sequence  $\{H^{(t)}\}$  with*

$$H^{(t+1)} = \underset{H}{\operatorname{argmin}} J(H, H^{(t)}), \quad (19)$$

*then the value sequence,  $\{L(H^{(t)})\}$ , is decreasing due to the following chain of inequalities:*

$$L(H^{(t)}) = J(H^{(t)}, H^{(t)}) \geq J(H^{(t+1)}, H^{(t)}) \geq L(H^{(t+1)}).$$

**Proposition 2** ([27]). *For any matrices  $\Gamma \in \mathcal{R}_+^{n \times n}$ ,  $\Omega \in \mathcal{R}_+^{k \times k}$ ,  $S \in \mathcal{R}_+^{n \times k}$ , and  $S' \in \mathcal{R}_+^{n \times k}$ , with  $\Gamma$  and  $\Omega$  being symmetric, the following inequality holds:*

$$\sum_{i=1}^n \sum_{s=1}^k \frac{(\Gamma S' \Omega)_{is} S_{is}^2}{S'_{is}} \geq \mathbf{Tr}(S^T \Gamma S \Omega). \quad (20)$$

With the aid of Definition 1 and Propositions 1 and 2, we prove Theorem 2 in the following.

*Proof of Theorem 2.* For fixed  $G$ , the objective function for  $W$  in (12) can be equivalently rewritten as

$$P(W) = \mathbf{Tr}(-W^T \bar{A} + \frac{1}{2} W^T \bar{C} W \bar{D} + \lambda W^T \bar{B}) + \frac{1}{2} \mathbf{Tr}(\bar{C}).$$

First, we show that the function  $\bar{P}(W, W')$  defined in (21) is an auxiliary function of  $P(W)$ :

$$\begin{aligned} \bar{P}(W, W') &= \frac{1}{2} \mathbf{Tr}(\bar{C}) - \sum_{ik} \bar{A}_{ik} W'_{ik} \left(1 + \log \frac{W_{ik}}{W'_{ik}}\right) \\ &\quad + \frac{1}{2} \sum_{ik} \frac{(\bar{C} W' \bar{D})_{ik} W_{ik}^2}{W'_{ik}} + \lambda \sum_{ik} \bar{B}_{ik} \frac{W_{ik}^2 + W'^2_{ik}}{2W'_{ik}}. \end{aligned} \quad (21)$$

To show this, we find the upper-bounds and lower-bounds for the positive and negative terms in  $P(W)$ , respectively. For the positive terms, we use Proposition 2 and the inequality  $a \leq (a^2 + b^2)/2b$  for  $a, b \geq 0$  to get the following upper-bounds:

$$\begin{aligned} \mathbf{Tr}(W^T \bar{B}) &= \sum_{ik} \bar{B}_{ik} W_{ik} \leq \sum_{ik} \bar{B}_{ik} \frac{W_{ik}^2 + W'^2_{ik}}{2W'_{ik}}, \\ \mathbf{Tr}(W^T \bar{C} W \bar{D}) &\leq \sum_{ik} \frac{(\bar{C} W' \bar{D})_{ik} W_{ik}^2}{W'_{ik}}. \end{aligned} \quad (22)$$

With simple algebra, it is easy to seen that  $a \geq 1 + \log a$  for  $a \geq 0$ . Then, for the negative term, we use the above inequality to get the following lower-bound:

$$\begin{aligned} \mathbf{Tr}(W^T \bar{A}) &= \sum_{ik} \bar{A}_{ik} W_{ik} \\ &\geq \sum_{ik} \bar{A}_{ik} W'_{ik} \left(1 + \log \frac{W_{ik}}{W'_{ik}}\right). \end{aligned} \quad (23)$$

Combining these bounds, we get the auxiliary function  $\bar{P}(W, W')$  for  $P(W)$ . Next, we will show that the update of (10) essentially follows (19), then according to Proposition 1 we can conclude the proof. To show this, the remaining problem is to find the global minimum of (21). For this, we first prove that (21) is convex. The first-order derivative of  $\bar{P}(W, W')$  is

$$\frac{\partial \bar{P}(W, W')}{\partial W_{ik}} = -\frac{\bar{A}_{ik} W'_{ik}}{W_{ik}} + \frac{(\bar{C} W' \bar{D})_{ik} W_{ik}}{W'_{ik}} + \lambda \frac{\bar{B}_{ik} W_{ik}}{W'_{ik}}. \quad (24)$$

Then the Hessian of  $H(W, W')$  can be obtained element-wisely as

$$\frac{\partial^2 \bar{P}(W, W')}{\partial W_{ik} \partial W_{jl}} = \delta_{ij} \delta_{jk} \left( \frac{\bar{A}_{ik} W'_{ik}}{W_{ik}^2} + \frac{(\bar{C} W' \bar{D})_{ik} + \lambda \bar{B}_{ik}}{W'_{ik}} \right), \quad (25)$$

where  $\delta_{ij}$  is delta function that returns 1 if  $i = j$  or 0 otherwise. It is seen that the Hessian matrix of  $\bar{P}(W, W')$  has zero elements off diagonal and nonnegative elements on diagonal, and thus is positive definite. Therefore,  $\bar{P}(W, W')$  is convex and achieves the global optimum by its first-order optimality condition, i.e., (24) = 0, which gives rise to

$$\frac{\bar{A}_{ik} W'_{ik}}{W_{ik}} = \frac{(\bar{C} W' \bar{D})_{ik} W_{ik}}{W'_{ik}} + \lambda \frac{\bar{B}_{ik} W_{ik}}{W'_{ik}}. \quad (26)$$

It is seen that (26) can be further reduced to

$$W_{ik} = W'_{ik} \sqrt{\frac{\bar{A}_{ik}}{(\bar{C} W' \bar{D})_{ik} + \lambda \bar{B}_{ik}}}. \quad (27)$$

Define  $W^{(t+1)} = W$ , and  $W^{(t)} = W'$ , then we can see that (12) is decreasing under the update of (27). Substituting  $\bar{A}$ ,  $\bar{B}$ ,  $\bar{C}$ ,  $\bar{D}$  into (27), we recover (10).  $\square$

### 5.2. Correctness and Convergence of (11)

Fixing  $W$ , we need to solve the following optimization problem for  $G$ :

$$\begin{aligned} \argmin_G &= \frac{1}{2} \text{Tr}(-2KWG^T + GW^T K WG^T) + \lambda \text{Tr}(W^T D^\phi G), \\ \text{s.t. } & G \geq 0, G^T G = \Lambda, \end{aligned} \quad (28)$$

where  $\Lambda$  is nonnegative and diagonal. We introduce the Lagrangian multipliers  $\Theta$ , which is symmetric and has size  $k \times k$ . Then the Lagrangian function to be

minimized gives rise to

$$\begin{aligned}
\mathcal{L}_G &= \frac{1}{2} \text{Tr}(-2KWG^T + GW^TKWG^T) + \lambda \text{Tr}(W^TD^\phi G) + \frac{1}{2} \text{Tr}(\Theta(G^TG - \Lambda)) \\
&= \frac{1}{2} \text{Tr}(-2KWG^T + GW^TKWG^T + 2\lambda W^TD^\phi G + G\Theta G^T) - \xi \\
&= \frac{1}{2} \text{Tr}(-2AG^T + GCG^T + 2\lambda BG^T + G\Theta G^T) - \xi \\
&= \frac{1}{2} \text{Tr}(-2AG^T + 2\lambda BG^T + G(C + \Theta)^+ G^T - G(C + \Theta)^- G^T) - \xi,
\end{aligned} \tag{29}$$

where we define  $\xi = \frac{1}{2} \text{Tr}(\Theta\Lambda)$ ,  $A = KW$ ,  $B = D^\phi W$ , and  $C = W^TKW$  for easier notation, and  $M^+$ ,  $M^-$  to be two nonnegative matrices for a nonnegative matrix  $M$  such that  $(M^+ - M^-) = M$ . The gradient of  $\mathcal{L}_G$  is

$$\frac{\partial \mathcal{L}_G}{\partial G} = -2A + 2GC + 2\lambda B + 2G\Theta. \tag{30}$$

Then the KKT complementarity condition gives

$$(-A + GC + \lambda B + G\Theta)_{ik} G_{ik} = 0, \tag{31}$$

which is a fixed point relation that the local minimum for  $G$  must hold. Noting that

$$C + \Theta = (C + \Theta)^+ - (C + \Theta)^-,$$

we follow the previous subsection and give an update as follows:

$$G_{ik} \leftarrow G_{ik} \sqrt{\frac{A_{ik} + (G(C + \Theta)^-)_{ik}}{\lambda B_{ik} + (G(C + \Theta)^+)_{ik}}}. \tag{32}$$

To show that the update of (32) will converge to a local minimum, we will show two results: the convergence of the update algorithm and the correctness of the converged solution.

From (32), it is easy to show that, at convergence, the solution satisfies the following condition:

$$(-A + GC + \lambda B + G\Theta)_{ik} G_{ik}^2 = 0, \tag{33}$$

which is the fixed point condition in (31). Hence, the correctness of the converged solution can be verified.

The convergence is assured by the following theorem.

**Theorem 3.** For fixed  $W$ , the Lagrangian function  $\mathcal{L}_G$  is monotonically decreasing under the updating rule in (32).

*Proof.* To prove Theorem 3, we use the auxiliary function approach. For ease of notation, we define  $E = C + \Theta$ .

First, we find upper-bounds for each positive term in  $\mathcal{L}_G$ . By inequality  $a \leq (a^2 + b^2)/2b$  for  $a, b \geq 0$ , we get

$$\text{Tr}(G^T B) = \sum_{ik} B_{ik} G^{ik} \leq \sum_{ik} B_{ik} \frac{G_{ik}^2 + G_{ik}'^2}{2G_{ik}'}. \quad (34)$$

Then, according to Proposition 2, by setting  $\Gamma$  or  $S$  to be identity matrices, we get the following two upper-bounds

$$\text{Tr}(GE^+ G^T) \leq \sum_{ik} \frac{(G' E^+)_{ik} G_{ik}^2}{G_{ik}'} \quad (35)$$

Then, by the inequalities  $a \geq 1 + \log a$  for  $a \geq 0$ , we get the following lower-bounds for negative terms:

$$\begin{aligned} \text{Tr}(G^T A) &\geq \sum_{ik} A_{ik} G_{ik}' \left(1 + \log \frac{G_{ik}}{G_{ik}'}\right) \\ \text{Tr}(GE^- G^T) &\geq \sum_{ikl} E_{kl}^- G_{ik}' G_{il}' \left(1 + \log \frac{G_{ik} G_{il}}{G_{ik}' G_{il}'}\right). \end{aligned} \quad (36)$$

Hence, combining the above bounds, we construct an auxiliary function for  $\mathcal{L}_G$ :

$$\begin{aligned} J(G, G') &= - \sum_{ik} A_{ik} G_{ik}' \left(1 + \log \frac{G_{ik}}{G_{ik}'}\right) \\ &\quad + \lambda \sum_{ik} B_{ik} \frac{G_{ik}^2 + G_{ik}'^2}{2G_{ik}'} + \frac{1}{2} \sum_{ik} \frac{(G' E^+)_{ik} G_{ik}^2}{G_{ik}'} \\ &\quad - \frac{1}{2} \sum_{ikl} E_{kl}^- G_{ik}' G_{il}' \left(1 + \log \frac{G_{ik} G_{il}}{G_{ik}' G_{il}'}\right). \end{aligned} \quad (37)$$

We take the first order derivative of (37), then we get

$$\frac{\partial J(G, G')}{\partial G_{ik}} = - \frac{A_{ik} G_{ik}'}{G_{ik}} + \lambda \frac{B_{ik}}{G_{ik}'} G_{ik} + \frac{(G' E^+)_{ik}}{G_{ik}'} G_{ik} - \frac{(G' E^-)_{ik} G_{ik}'}{G_{ik}}. \quad (38)$$

Further, we can get the Hessian of (37) by taking its second order derivative:

$$\frac{\partial^2 J(G, G')}{\partial G_{ik} \partial G_{jl}} = \delta_{ij} \delta_{kl} \left( \frac{A_{ik} G_{ik}'}{G_{ik}^2} + \lambda \frac{B_{ik}}{G_{ik}'^2} + \frac{(G' E^+)_{ik}}{G_{ik}'^2} + \frac{(G' E^-)_{ik} G_{ik}'}{G_{ik}^2} \right). \quad (39)$$



It is easy to verify that the Hessian matrix has zero elements off diagonal, and nonnegative values on diagonal. Therefore,  $J(G, G')$  is convex in  $G$  and its global minimum is obtained by its first order optimality condition,  $(38) = 0$ , which gives rise to

$$G_{ik} = G'_{ik} \sqrt{\frac{A_{ik} + (G'E^-)_{ik}}{\lambda B_{ik} + (G'E^+)_{ik}}}. \quad (40)$$

According to Proposition 1, by setting  $G^{(t+1)} = G$  and  $G^{(t)} = G'$ , we recover (32) and it is easy to see that  $\mathcal{L}_G(G)$  is decreasing under (32).  $\square$

It is seen that in (32), the multipliers  $\Theta$  is yet to be determined. By the first order optimality condition of  $\mathcal{L}_G$ , i.e.,  $(30) = 0$ , we can see that

$$\begin{aligned} & G^T(-A + GC + \lambda B + G\Theta) \\ &= -G^T A + G^T GC + \lambda G^T B + G^T G\Theta \\ &= -G^T A + C + \lambda G^T B + \Theta \\ &= 0, \end{aligned} \quad (41)$$

hence

$$E = G^T A - \lambda G^T B. \quad (42)$$

Note that by defining  $E^- = \lambda G^T B$ , and  $E^+ = G^T A$ , we have  $E^+ - E^- = E$  and  $E^+ \geq 0$ ,  $E^- \geq 0$ . Substituting  $E^+$  and  $E^-$  into (32), we get the updating rule in (11).

*Remark.* We can see that the objective function in (9) is decreasing and the value sequence converges with the alternative updating rules of  $W$  and  $G$ . Next, we further investigate the convergence rate of the updating rules. We set  $\mathcal{Y} = [W^T, G^T]^T \in \mathcal{R}^{2n \times k}$ , and regard the updates of (10) and (11) as a mapping  $\mathcal{Y}^{(t+1)} = \mathcal{M}(\mathcal{Y}^{(t)})$ , then at convergence we have  $\mathcal{Y}^* = \mathcal{M}(\mathcal{Y}^*)$ . Following [27, 45], with non-negativity constraint enforced, we expand  $\mathcal{Y} \approx \mathcal{M}(\mathcal{Y}^*) + (\partial\mathcal{M}/\partial\mathcal{Y})(\mathcal{Y} - \mathcal{Y}^*)$ , which indicates that  $\|\mathcal{Y}^{(t+1)} - \mathcal{Y}^*\| \leq \|\partial\mathcal{M}/\partial\mathcal{Y}\| \cdot \|\mathcal{Y}^{(t)} - \mathcal{Y}^*\|$  under an appropriate matrix norm. In general,  $\|\partial\mathcal{M}/\partial\mathcal{Y}\| \neq 0$ , hence the updates of (10) and (11) roughly have a first-order convergence rate.

## 6. Experiments

In this section, we conduct extensive experiments to verify the effectiveness of the proposed KLS-NMF. We will present the evaluation metrics, benchmark datasets, algorithms in comparison, and experimental results in detail in rest of this section.

### 6.1. Evaluation Metrics

In our experiments, we use four evaluation metrics to testify the effectiveness of the proposed method, including accuracy, normalized mutual information (NMI), purity, and correntropy. Accuracy measures the extent to which each cluster contains data points from the same class. NMI measures the quality of the clusters. Purity measures the extent to which each cluster contains samples from primarily the same class. Higher values of these metrics indicate better clustering quality. Correntropy is an entropy-based criterion defined as a non-linear similarity measure. Different from the other metrics, smaller values of correntropy indicate better clustering results. More details of these metrics can be found in [30, 46, 47].

### 6.2. Benchmark Data Sets

Due to the nature of parts-based representation, NMF has been widely used for image data which are naturally nonnegative. Thus, in this paper, we evaluate the proposed KLS-NMF on five benchmark image data sets, including PIX, JAFFE, Alphadigit, Semeion, and Faces94. We briefly describe these data sets as follows:

- PIX contains 100 gray scale images collected from 10 objects, which has size  $100 \times 100$  pixes.
- JAFFE collects 213 images of 10 Japanese female models posed 7 facial expressions. These images are rated on 6 motion adjectives by 60 Japanese subjects. Each image has size of  $26 \times 26$  pixels.

Table 1: Performance of K-mean Clustering on Semeion

N	Accuracy (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	87.57±10.53	86.25±09.60	88.18±10.02	87.88±10.73	87.10±11.63	<b>88.86±10.54</b>	<b>88.86±10.54</b>
3	80.31±09.91	75.29±10.43	80.23±10.51	80.58±10.52	79.43±07.39	82.84±08.54	<b>82.88±08.53</b>
4	71.95±06.07	65.93±05.80	70.32±08.91	67.88±10.86	70.80±08.62	<b>75.32±11.16</b>	<b>75.32±11.16</b>
5	70.24±06.77	58.24±06.59	67.61±10.23	64.40±07.41	64.36±08.39	<b>75.26±07.33</b>	<b>75.26±07.33</b>
6	58.25±05.69	51.91±08.28	57.50±06.14	61.71±09.32	61.57±06.77	63.26±07.60	<b>64.91±08.69</b>
7	59.32±07.24	48.69±05.34	54.42±05.89	61.36±05.91	57.68±07.48	63.16±05.72	<b>64.66±05.42</b>
8	59.63±07.53	44.16±04.66	53.52±04.81	60.33±05.64	58.02±06.95	59.00±03.66	<b>67.15±06.74</b>
9	56.35±04.12	40.80±03.19	50.16±05.59	56.06±05.52	56.63±08.88	59.21±05.54	<b>59.25±02.74</b>
10	55.56	42.18	45.20	52.54	49.15	49.91	<b>60.58</b>
Average	66.57	57.05	63.01	65.86	64.97	68.53	<b>70.99</b>
N	NMI (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	56.16±28.34	50.15±25.10	56.20±28.69	57.41±28.32	55.51±30.48	<b>60.70±30.26</b>	<b>60.70±30.26</b>
3	54.01±13.67	45.73±13.57	53.90±14.75	55.95±12.58	50.22±11.18	<b>58.69±11.41</b>	58.68±11.41
4	50.68±04.88	43.50±05.45	49.93±06.20	50.52±07.34	49.02±05.37	<b>58.22±09.09</b>	<b>58.22±09.09</b>
5	52.28±06.09	39.10±07.97	51.08±08.22	54.32±03.32	49.88±07.96	<b>61.15±07.27</b>	<b>61.15±07.27</b>
6	45.58±04.75	36.22±06.83	45.25±06.11	51.11±05.01	47.46±05.93	53.98±07.86	<b>55.26±07.79</b>
7	46.55±06.27	35.22±03.55	44.05±04.81	51.57±04.88	46.56±06.12	<b>54.62±04.00</b>	54.07±04.08
8	48.18±04.90	34.07±04.02	44.36±03.54	52.49±02.81	46.70±04.29	54.32±03.41	<b>58.96±04.45</b>
9	47.18±03.78	31.63±02.37	42.75±04.51	49.29±03.99	45.75±04.75	52.26±03.21	<b>54.43±02.45</b>
10	44.82	33.65	37.96	47.38	43.12	48.26	<b>54.98</b>
Average	49.49	38.81	47.28	52.23	48.25	55.80	<b>57.38</b>
N	Purity (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	87.57±10.53	86.25±09.60	88.18±10.02	87.88±10.73	87.10±11.63	<b>88.86±10.54</b>	<b>88.86±10.54</b>
3	80.31±09.91	75.29±10.43	80.39±10.19	80.67±10.35	79.43±07.39	<b>82.88±08.53</b>	<b>82.88±08.53</b>
4	72.33±05.76	66.51±05.53	71.91±06.45	71.09±07.60	72.06±05.92	<b>76.51±08.74</b>	<b>76.51±08.74</b>
5	70.51±06.74	58.83±07.00	69.13±07.59	69.25±04.18	67.59±06.43	<b>76.10±06.40</b>	<b>76.10±06.40</b>
6	60.91±04.53	53.78±07.31	61.03±05.25	65.64±06.08	63.45±06.24	66.05±07.40	<b>67.83±07.45</b>
7	60.88±06.43	50.50±04.66	57.35±05.68	65.02±04.32	61.12±06.32	65.88±04.01	<b>67.11±03.74</b>
8	60.58±06.55	45.71±04.74	55.72±03.92	63.94±03.71	60.13±05.82	61.88±03.57	<b>68.84±04.47</b>
9	59.04±04.61	42.98±02.75	52.57±05.44	60.18±05.00	59.20±06.61	60.86±04.87	<b>64.10±02.78</b>
10	56.56	43.82	45.20	52.54	54.74	56.75	<b>61.83</b>
Average	67.63	58.18	64.61	68.47	67.20	70.64	<b>72.67</b>
N	Correntropy						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	0.4384±0.2833	0.4984±0.2510	0.4362±0.2902	0.4259±0.2832	0.4449±0.3048	<b>0.3929±0.3016</b>	<b>0.3929±0.3026</b>
3	0.7289±0.2166	0.8602±0.2151	0.7414±0.1997	0.6981±0.1993	0.7918±0.1773	<b>0.6549±0.1809</b>	<b>0.6549±0.1808</b>
4	0.9864±0.0975	1.1300±0.1089	1.0131±0.1461	0.9896±0.1468	1.0631±0.1340	<b>0.8356±0.1817</b>	<b>0.8356±0.1817</b>
5	1.1080±0.1414	1.4139±0.1850	1.1425±0.1205	1.0607±0.0770	1.1748±0.1702	<b>0.9021±0.1689</b>	<b>0.9021±0.1689</b>
6	1.4067±0.1227	1.6486±0.1765	1.3213±0.0867	1.2637±0.1294	1.3704±0.1589	1.1896±0.2033	<b>1.1567±0.2016</b>
7	1.5003±0.1760	1.8186±0.0997	1.4529±0.1277	1.3595±0.1369	1.4998±0.1818	1.2741±0.1124	<b>1.2546±0.1208</b>
8	1.5559±0.1471	1.9778±0.1207	1.5035±0.0939	1.4252±0.0842	1.6329±0.1154	1.3704±0.1022	<b>1.2281±0.1327</b>
9	1.6742±0.1197	2.1673±0.0750	1.6604±0.1231	1.6074±0.1266	1.7842±0.0934	1.5133±0.1016	<b>1.4443±0.0778</b>
10	1.8329	2.2039	1.7931	1.7480	1.9018	1.7186	<b>1.4920</b>
Average	1.2480	1.5243	1.2294	1.1753	1.2960	1.0946	<b>1.0401</b>

- Alphadigit is a binary data set, which collects handwritten digits 0-9 and letters A-Z. Totally, there are 36 classes and 39 samples for each class. Each image has size of 20×16 pixels.
- Semeion collects 1,593 handwritten digits that are written by around 80 persons. These images were scanned and stretched into size 16 ×16.
- Faces94 contains images of 153 individuals, each of whom has 20 images of size 200×180. We resize the images to 20×18 pixels in our experiments.

Table 2: Performance of K-mean Clustering on JAFFE

N	Accuracy (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	99.75±00.79	<b>100.0±00.00</b>	99.75±00.00	99.75±00.79	99.25±02.37	<b>100.00±00.00</b>	<b>100.00±00.00</b>
3	96.54±05.05	98.07±02.12	87.98±13.94	96.36±03.91	84.06±16.95	98.55±01.93	<b>99.72±01.47</b>
4	95.92±05.96	97.77±01.78	80.37±17.35	89.54±13.01	91.88±14.41	<b>99.07±02.04</b>	<b>99.07±02.04</b>
5	95.75±03.92	96.28±05.01	88.29±08.25	87.26±10.56	72.47±06.66	95.87±08.72	<b>98.39±02.23</b>
6	89.47±04.41	90.17±07.98	76.26±13.45	83.50±08.14	88.98±12.69	<b>97.80±01.14</b>	<b>97.80±01.14</b>
7	89.68±10.77	88.80±08.91	72.05±11.21	83.14±09.33	79.65±08.69	91.44±09.04	<b>96.79±02.35</b>
8	92.05±05.57	81.64±06.58	69.44±10.06	79.24±07.30	74.74±07.43	84.04±06.55	<b>96.52±01.61</b>
9	86.84±04.69	82.48±08.20	63.82±05.77	79.76±06.36	79.01±06.05	89.88±06.37	<b>95.51±01.23</b>
10	90.61	83.10	69.95	81.69	82.63	69.01	<b>96.24</b>
Average	92.96	90.92	78.66	86.69	83.63	91.74	<b>97.67</b>
N	NMI (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	98.55±04.59	<b>100.0±00.00</b>	98.55±04.59	98.55±04.59	96.79±10.16	<b>100.00±00.00</b>	<b>100.0±00.00</b>
3	91.29±10.58	93.73±06.54	78.83±18.03	89.92±10.13	78.25±17.32	95.46±06.09	<b>95.84±04.63</b>
4	91.48±10.86	94.15±04.48	75.52±17.36	86.39±15.37	92.30±09.66	<b>97.82±04.70</b>	<b>97.82±04.70</b>
5	92.94±05.56	93.79±04.99	84.42±08.55	85.72±08.72	73.86±05.60	95.30±07.01	<b>96.69±04.49</b>
6	85.58±05.96	86.50±09.24	73.17±13.80	83.17±06.95	88.91±10.75	<b>95.68±02.05</b>	<b>95.68±02.05</b>
7	88.18±09.17	85.55±08.77	69.79±11.35	85.46±05.05	81.43±08.65	91.93±05.90	<b>94.79±03.58</b>
8	91.22±04.86	82.72±06.04	66.10±11.38	82.18±04.27	81.33±06.17	88.18±03.46	<b>94.50±02.53</b>
9	87.20±03.18	83.21±06.77	62.37±05.03	83.03±04.25	82.49±04.57	91.39±02.26	<b>93.73±01.57</b>
10	89.44	92.32	70.65	82.38	84.46	81.34	<b>94.40</b>
Average	90.65	89.11	75.49	86.31	84.42	93.01	<b>95.94</b>
N	Purity (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	99.75±00.79	<b>100.0±00.00</b>	99.75±00.79	99.75±00.79	99.25±02.37	<b>100.00±00.00</b>	<b>100.0±00.00</b>
3	96.54±05.05	98.07±02.12	88.94±11.81	96.36±03.91	86.25±13.26	98.55±01.93	<b>98.72±01.47</b>
4	95.92±05.96	97.77±01.78	83.29±13.59	90.61±11.26	94.11±09.71	99.07±02.04	<b>99.09±02.04</b>
5	95.75±03.92	96.28±05.01	88.66±07.62	88.28±09.17	76.60±06.07	96.74±06.08	<b>98.39±02.23</b>
6	89.47±04.41	90.64±07.11	78.30±11.83	84.83±06.79	90.41±10.32	<b>97.80±01.14</b>	<b>97.80±01.14</b>
7	90.61±08.87	89.21±08.30	73.39±11.25	86.43±06.49	81.60±07.80	92.98±06.68	<b>96.79±02.35</b>
8	92.23±05.24	83.47±05.97	70.44±09.99	81.48±05.65	78.57±05.82	87.40±04.39	<b>96.52±01.61</b>
9	87.52±03.63	83.94±06.64	66.02±05.31	82.31±04.65	81.57±04.89	91.65±04.11	<b>95.51±01.23</b>
10	90.61	83.10	74.18	82.16	82.36	77.93	<b>96.24</b>
Average	93.16	91.39	80.33	88.02	85.66	93.57	<b>97.67</b>
N	Correntropy						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	0.0145±0.0459	<b>0.0000±0.000</b>	0.1003±0.2701	0.0145±0.0459	0.1318±0.3102	<b>0.0000±0.0000</b>	<b>0.0000±0.0000</b>
3	0.1378±0.1675	0.0990±0.1035	0.3058±0.2096	0.1596±0.1604	0.4214±0.3025	0.0719±0.0964	<b>0.0717±0.0962</b>
4	0.1702±0.2170	0.1165±0.0894	0.3842±0.3751	0.2721±0.3073	0.2117±0.2263	<b>0.0437±0.0940</b>	<b>0.0437±0.0940</b>
5	0.1640±0.1290	0.1440±0.1159	0.4196±0.2118	0.3314±0.2023	0.6217±0.1288	0.1090±0.1628	<b>0.0768±0.1043</b>
6	0.3727±0.1541	0.3488±0.2389	0.5968±0.2612	0.4347±0.1796	0.3003±0.2739	<b>0.1115±0.0531</b>	<b>0.1115±0.0531</b>
7	0.3316±0.2574	0.4056±0.2462	0.7196±0.1719	0.4080±0.1416	0.5948±0.2332	0.2266±0.1655	<b>0.1463±0.1004</b>
8	0.2633±0.1459	0.5182±0.1810	0.6979±0.0937	0.5345±0.1279	0.6524±0.1541	0.3543±0.1039	<b>0.1651±0.0758</b>
9	0.4057±0.1007	0.5320±0.2145	0.7824±0.1181	0.5378±0.1347	0.5921±0.2704	0.2727±0.0715	<b>0.1988±0.0498</b>
10	0.3507	0.5873	1.0282	0.5852	0.5381	0.6197	<b>0.1860</b>
Average	0.2456	0.3057	0.5594	0.3642	0.4516	0.2010	<b>0.1011</b>

### 6.3. Algorithms in Comparison

To illustrate the effectiveness of KLS-NMF, we compare them with several state-of-the-art NMF methods, including weighted NMF (WNMF) [48], ONMF [23], CNMF [27], Kernel NMF (KNMF) [27], Multi-kernel NMF (MKNMF) [49], and RMNMF [30]. We briefly describe these methods as follows:

- **WNMF.** It extends the results of the original NMF to a weighted case.
- **ONMF.** It has different variants that imposes orthogonality constraint on different factor matrices. In our experiment, we adopt the matrix tri-factorization model that imposes orthogonal constraints on the left and right factor matrices.

Table 3: Performance of K-mean Clustering on PIX

N	Accuracy (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	94.00±10.22	<b>99.50±01.58</b>	96.50±06.26	94.50±10.39	89.00±12.20	94.50±10.39	94.50±10.39
3	96.00±05.84	<b>97.33±02.11</b>	96.33±04.83	96.00±05.84	82.67±21.36	96.00±05.84	96.00±05.84
4	92.75±07.77	97.00±03.87	88.00±12.68	89.75±13.36	83.25±14.24	96.25±04.60	<b>97.25±03.81</b>
5	86.40±12.75	88.20±09.59	82.20±09.21	86.00±09.57	82.80±09.10	88.20±10.93	<b>88.60±11.16</b>
6	85.00±11.63	88.50±09.01	77.50±09.24	86.33±09.84	78.50±10.93	87.67±09.53	<b>90.17±09.51</b>
7	86.43±08.97	85.14±07.17	81.57±08.48	89.29±06.50	79.14±08.52	87.57±07.13	<b>92.00±06.32</b>
8	80.88±04.04	78.25±06.54	78.50±04.56	83.25±08.60	81.25±06.85	83.75±05.68	<b>91.00±01.84</b>
9	88.22±05.06	85.11±03.15	73.89±04.39	82.78±03.93	79.33±07.84	87.11±05.03	<b>91.00±04.81</b>
10	74.00	81.00	80.00	69.00	<b>89.00</b>	78.00	<b>89.00</b>
Average	87.08	88.89	83.83	86.32	82.77	88.78	<b>92.17</b>
N	NMI (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	81.39±28.27	<b>97.58±07.64</b>	87.45±21.45	83.81±28.77	67.62±32.75	83.81±28.77	83.81±28.77
3	89.87±11.64	<b>91.93±06.30</b>	91.00±08.50	89.87±11.64	79.34±16.44	89.87±11.64	89.87±11.64
4	89.41±09.06	94.10±06.12	84.84±11.15	88.21±11.69	82.36±11.02	93.39±07.13	<b>94.67±05.78</b>
5	87.90±09.29	85.66±08.03	79.03±09.09	83.93±07.46	84.46±04.78	88.72±07.04	<b>88.84±07.68</b>
6	86.02±08.30	85.07±09.63	75.43±08.46	87.75±06.03	81.94±07.25	88.12±05.90	<b>89.98±06.53</b>
7	88.64±04.80	84.45±07.20	82.52±05.68	88.39±05.28	83.33±05.91	89.06±05.13	<b>91.43±05.16</b>
8	85.16±02.09	77.32±05.59	80.95±03.37	87.80±04.71	84.36±05.58	85.75±02.92	<b>90.18±02.26</b>
9	89.22±01.67	85.85±03.55	78.50±03.96	85.59±01.77	84.60±04.81	88.75±02.23	<b>91.37±04.05</b>
10	83.91	83.95	82.97	80.90	89.31	82.92	<b>89.35</b>
Average	86.84	87.32	82.52	86.25	81.92	87.82	<b>89.94</b>
N	Purity (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	94.00±10.22	<b>99.50±01.58</b>	96.50±06.26	94.50±10.39	89.00±12.20	94.50±10.39	94.50±10.39
3	96.00±05.84	<b>97.33±02.11</b>	96.33±04.83	96.00±05.84	87.00±14.44	96.00±05.84	96.00±05.84
4	92.75±07.77	97.00±03.87	89.00±10.94	91.50±09.87	86.00±10.62	96.25±04.60	<b>97.25±03.81</b>
5	89.20±08.75	89.20±07.55	82.80±08.70	86.80±07.44	85.40±06.11	90.20±07.21	<b>90.60±07.43</b>
6	87.33±08.90	88.83±08.46	79.00±08.47	88.50±07.00	82.17±08.32	89.67±06.56	<b>91.50±07.00</b>
7	88.71±06.19	86.29±06.97	83.71±06.50	89.71±05.34	82.14±06.50	89.14±05.23	<b>92.57±05.25</b>
8	83.88±02.66	79.63±04.97	81.12±03.30	86.25±06.01	82.87±06.18	85.38±04.25	<b>91.00±01.84</b>
9	89.44±03.24	86.67±03.47	77.22±04.57	84.78±02.03	82.22±05.93	88.11±03.63	<b>91.33±04.31</b>
10	79.00	82.00	83.00	74.00	<b>89.00</b>	80.00	<b>89.00</b>
Average	88.92	89.60	85.41	88.00	85.09	89.92	<b>92.64</b>
N	Correntropy						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
2	0.1861±0.2827	<b>0.0242±0.0764</b>	0.1619±0.2877	0.1619±0.2877	0.3238±0.3275	0.1619±0.2877	0.1450±0.2535
3	0.1606±0.1845	<b>0.1280±0.0999</b>	0.2778±0.2698	0.1606±0.1845	0.3820±0.2928	0.1606±0.1845	0.1606±0.1845
4	0.2118±0.1812	<b>0.1180±0.1225</b>	0.2582±0.2143	0.2357±0.2337	0.3371±0.2055	0.1322±0.1425	0.1322±0.1425
5	0.2810±0.2158	0.3330±0.1865	0.4076±0.1770	0.3732±0.1732	0.4045±0.1450	0.2618±0.1634	<b>0.2519±0.1646</b>
6	0.3614±0.2145	0.3859±0.2490	0.3648±0.1637	0.3167±0.1559	0.4890±0.1609	0.2071±0.1525	<b>0.2602±0.1210</b>
7	0.3190±0.1347	0.4366±0.2022	0.3433±0.1448	0.3260±0.1481	0.5285±0.1524	0.3073±0.1441	<b>0.2476±0.1334</b>
8	0.4451±0.0627	0.6803±0.1676	0.3864±0.1224	0.3661±0.1414	0.5011±0.1414	0.4276±0.0877	<b>0.2904±0.0703</b>
9	0.3417±0.0530	0.4485±0.1125	0.4100±0.0721	0.4568±0.0561	0.6194±0.2018	0.3565±0.0706	<b>0.2947±0.1046</b>
10	0.5344	0.5332	0.5333	0.6346	0.6625	0.5675	<b>0.4161</b>
Average	0.3157	0.3431	0.3493	0.3368	0.4720	0.2981	<b>0.2443</b>

- **CNMF**. It restricts the learned basis to lie within the column space of the input data, such that the basis vectors can be represented as a convex combination of the inputs.
- **KNMF**. Based on CNMF, KNMF exploits latent nonlinear structures of the data in kernel space. In our experiment, we use rbf kernel with radial ranging in the set  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ .
- **RMNMF**. It relaxes the data and basis matrix to have mixed signs, and adopts robust  $\ell_{2,1}$  norm to measure the fitting errors. Moreover, nonlinear structures of the data are exploited on manifold. We use the binary weighting to construct the graph Laplacian, with the default 5

Table 4: Performance of K-mean Clustering on Alphadigit

N	Accuracy (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
5	70.72±09.62	70.67±09.43	73.54±09.91	73.79±09.37	70.82±12.35	81.03±13.00	<b>81.38±12.89</b>
10	56.08±06.26	46.49±04.14	56.18±04.38	63.49±06.67	60.69±07.56	56.82±02.94	<b>65.46±07.03</b>
15	47.62±03.20	34.97±01.82	46.70±03.83	54.41±02.82	49.86±04.09	48.70±02.91	<b>54.50±03.89</b>
20	45.55±01.87	30.13±02.20	40.21±02.37	51.18±03.77	48.38±03.51	45.00±03.34	<b>52.88±03.56</b>
25	43.67±01.94	26.01±01.44	31.04±01.82	45.12±02.37	42.07±01.70	39.26±03.23	<b>48.61±02.94</b>
30	39.38±01.76	23.83±01.71	28.50±01.36	41.30±02.26	40.53±02.32	36.73±02.72	<b>45.88±02.84</b>
35	37.96±01.51	21.66±01.24	23.63±00.85	39.93±01.15	38.22±02.10	34.83±01.27	<b>44.29±01.60</b>
36	36.67	22.36	27.75	41.45	34.97	33.12	<b>41.74</b>
Average	47.21	34.51	40.94	51.33	48.19	46.94	<b>54.34</b>
N	NMI (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
5	58.21±07.83	55.43±09.30	61.31±09.11	63.88±10.05	59.50±10.57	69.48±11.60	<b>69.82±11.61</b>
10	55.20±04.62	43.44±04.15	56.99±03.24	62.16±04.01	60.47±04.68	59.03±04.03	<b>64.13±04.87</b>
15	52.91±04.18	37.73±03.81	51.73±03.41	58.49±03.16	55.36±02.74	53.88±01.58	<b>58.42±02.02</b>
20	54.20±01.49	35.18±02.48	49.91±01.95	58.35±02.46	55.91±02.43	54.15±03.23	<b>61.55±02.44</b>
25	54.47±02.67	34.06±02.45	43.32±01.77	55.94±01.15	53.96±01.91	50.31±02.89	<b>59.78±01.84</b>
30	53.34±01.13	33.09±01.44	42.26±01.39	54.24±01.52	53.73±01.73	48.65±01.32	<b>58.65±01.49</b>
35	54.03±01.10	32.77±01.48	36.61±01.02	54.42±00.98	53.54±00.77	47.54±01.57	<b>58.81±00.94</b>
36	53.48	35.14	39.61	56.21	52.56	45.34	<b>56.64</b>
Average	54.48	38.36	47.72	57.96	55.64	53.55	<b>60.98</b>
N	Purity (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
5	72.97±07.60	71.18±08.98	75.18±08.29	75.85±07.86	72.92±10.62	82.00±10.93	<b>82.36±10.84</b>
10	58.28±05.78	48.69±05.13	59.92±04.08	66.62±05.80	63.59±07.16	60.74±03.71	<b>67.87±06.47</b>
15	51.03±03.18	37.44±02.39	49.86±03.78	<b>59.70±03.07</b>	53.09±03.36	51.28±02.15	56.91±03.27
20	48.85±02.35	32.06±02.25	43.42±01.98	54.74±02.88	51.42±03.20	48.91±03.12	<b>56.29±03.48</b>
25	46.27±02.18	27.68±01.50	33.82±01.95	49.26±02.15	45.14±01.74	42.73±03.04	<b>51.63±02.80</b>
30	42.53±01.64	25.48±01.56	30.46±01.27	45.48±01.87	43.63±02.01	40.08±02.07	<b>49.52±02.78</b>
35	41.00±01.32	23.33±01.07	25.46±00.94	43.49±00.88	41.17±01.58	37.87±01.70	<b>47.41±01.54</b>
36	39.51	24.15	29.26	<b>45.37</b>	38.68	35.75	44.73
Average	50.05	36.25	43.42	54.84	51.21	49.92	<b>60.98</b>
N	Correntropy						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
5	0.9370±0.1900	1.0348±0.2160	0.9115±0.2234	0.8387±0.2334	0.9340±0.2415	0.7086±0.2692	<b>0.7086±0.2692</b>
10	1.4883±0.1534	1.8787±0.1380	1.3223±0.1502	1.2572±0.1333	1.3132±0.1556	1.3611±0.1337	<b>1.1333±0.1570</b>
15	1.8413±0.1653	2.4328±0.1489	1.7484±0.0974	1.6219±0.1234	1.7441±0.1069	1.8018±0.0618	<b>1.5627±0.0989</b>
20	1.9793±0.0645	2.8013±0.1071	1.8991±0.0851	1.8000±0.1064	1.9057±0.1050	1.9814±0.1396	<b>1.6574±0.1072</b>
25	2.1145±0.1240	3.0620±0.1137	2.1630±0.0881	2.0462±0.0533	2.1379±0.0887	2.3076±0.1340	<b>1.8609±0.0802</b>
30	2.2897±0.0555	3.2834±0.0709	2.3231±0.0960	2.2456±0.0745	2.2703±0.0850	2.5198±0.0646	<b>2.0292±0.0729</b>
35	2.3577±0.0565	3.4484±0.0759	2.4508±0.0495	2.3379±0.0501	2.3829±0.0397	2.6907±0.0807	<b>2.1161±0.0520</b>
36	2.3400	3.3534	2.4454	2.2638	2.4192	2.8258	<b>2.1925</b>
Average	1.9185	2.6619	1.9079	1.8014	1.8884	2.0246	<b>1.6576</b>

neighbors selected. The regularization parameter is also selected from  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ .

- **MKNMF**. It extends the KNMF to multi kernel model. A combining coefficient vector is learned for a set of kernels, such that the data is learned with linearly combined kernel. To be fair, we use a set of RBF kernels for MKNMF where the radial values are chosen from  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ .
- **KLS-NMF**. To be consistent with KNMF, we use rbf kernel with the same range and radial in  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ . Moreover, we select the parameter  $\lambda$  from the set  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ .

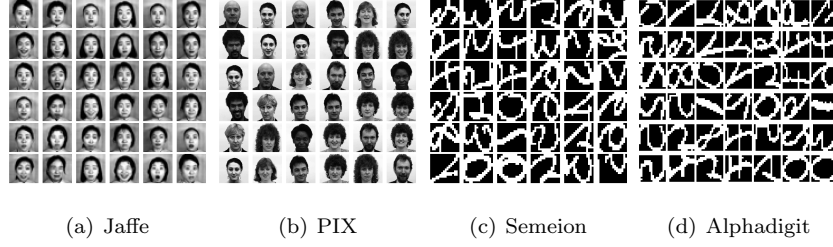


Figure 1: Examples selected images from Jaffe, PIX, Semeion, and Alphadigit data sets.

#### 6.4. Experimental Settings

In this subsection, we present the detailed experimental settings in our paper. To be fair and follow a common strategy, we provide the number of clusters  $K$  for all NMF methods. It should be noted that the NMF methods provide a new representation of the data and a final clustering step is still required on the sought representation matrix. To obtain the final clustering result, we apply K-means to the new representation of the data, where we use the `litekmeans`<sup>1</sup> package as a fast implementation in our experiments. We conduct 200 trials for K-means, where the clustering with the smallest objective value is regarded as the optimal solution.

For a given data set, we denote the total number of clusters by  $\bar{N}$ , e.g.,  $\bar{N} = 36$  in Alphadigit data. To better investigate the clustering performance of different methods on this data, we randomly select subsets of this data to conduct more detailed experiments. In particular, we randomly select a subset with  $N$  out of  $\bar{N}$  classes to conduct experiments. In our experiments, we vary the value of  $N$  such that subsets of different sizes are tested for better comparison. It is noted that a specific  $N$  value correspond to  $\bar{N}!/((\bar{N} - N)!N!)$  different subsets, from which we randomly choose 10. Experiments are conducted on the selected 10 subsets and the best average performance is reported by enumerating

<sup>1</sup><http://www.cad.zju.edu.cn/home/dengcai/Data/code/litekmeans.m>

Table 5: Performance of K-mean Clustering on Faces94

N	Accuracy (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
10	90.85±07.15	95.95±05.61	82.25±06.77	87.85±07.77	79.75±09.66	91.05±06.25	<b>100.00±00.00</b>
20	83.98±03.24	85.45±05.14	79.78±05.39	78.90±04.99	76.00±03.42	80.60±03.26	<b>88.89±03.47</b>
30	80.25±05.28	80.87±04.52	75.78±04.21	72.15±04.50	72.12±03.34	72.15±05.09	<b>83.97±04.12</b>
40	76.48±02.14	78.75±03.25	72.59±04.58	70.36±04.68	69.74±02.59	68.81±02.64	<b>83.47±02.62</b>
50	76.74±02.73	76.31±01.96	72.45±02.36	68.24±03.46	67.27±02.99	67.77±02.36	<b>82.13±03.12</b>
60	73.55±02.91	73.23±03.95	71.16±02.72	66.14±02.44	68.11±03.29	65.54±01.89	<b>81.72±03.25</b>
70	73.64±02.37	71.91±02.50	72.30±03.57	67.26±03.22	69.01±02.66	67.81±01.67	<b>80.14±03.33</b>
80	74.29±02.42	69.41±02.76	69.42±02.99	65.70±02.32	68.19±03.43	63.80±02.46	<b>78.45±03.26</b>
90	72.13±03.38	67.78±02.80	67.92±02.47	63.44±01.78	68.26±03.79	64.47±02.95	<b>79.49±02.73</b>
100	72.28±02.74	65.02±02.54	67.09±02.63	65.20±01.74	69.07±02.75	64.36±02.04	<b>79.72±01.83</b>
110	70.55±02.30	63.05±03.21	66.85±02.76	63.90±02.41	70.42±02.06	64.37±02.86	<b>77.82±02.85</b>
113	73.98	64.25	66.55	65.93	68.50	64.12	<b>77.26</b>
Average	76.56	74.33	76.35	69.59	70.54	69.57	<b>82.76</b>
N	NMI (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
10	95.22±03.71	97.47±02.87	90.87±03.68	94.00±03.75	89.85±04.90	95.45±03.23	<b>100.00±00.00</b>
20	92.43±01.47	93.72±02.50	89.49±02.06	89.84±02.21	88.35±02.48	91.04±02.38	<b>95.72±01.19</b>
30	91.98±02.46	92.12±02.03	88.46±01.88	87.44±01.80	88.20±01.68	86.71±03.36	<b>94.41±01.41</b>
40	91.11±01.71	91.78±01.79	86.48±02.33	85.84±03.52	87.12±01.34	83.98±03.01	<b>94.86±00.74</b>
50	91.42±01.22	91.10±00.84	86.79±01.67	85.52±01.94	85.83±01.44	85.09±02.64	<b>94.20±00.88</b>
60	90.34±01.67	90.37±01.80	86.70±01.26	84.55±01.72	87.37±01.93	82.89±01.73	<b>94.30±01.07</b>
70	90.95±00.98	89.24±01.43	87.74±01.76	85.56±02.52	88.69±01.32	85.64±00.88	<b>94.11±01.08</b>
80	91.42±01.22	88.71±01.12	86.26±01.44	84.41±01.98	88.83±01.48	82.04±02.49	<b>93.92±01.16</b>
90	90.25±01.40	87.40±01.82	85.63±00.91	82.87±01.59	88.19±01.77	83.05±03.15	<b>94.11±00.97</b>
100	90.69±01.13	86.75±02.03	85.39±01.00	84.34±01.94	88.90±01.06	83.58±02.29	<b>94.20±00.51</b>
110	89.87±01.28	85.32±01.68	85.37±01.16	83.58±01.82	89.40±00.88	83.31±02.06	<b>93.91±00.88</b>
113	90.69	85.14	85.43	83.74	87.85	85.60	<b>94.01</b>
Average	91.36	89.93	87.05	85.97	88.21	85.70	<b>94.81</b>
N	Purity (%)						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
10	93.05±05.25	97.75±04.08	86.45±05.13	90.30±06.00	84.85±07.22	92.75±05.04	<b>100.0±00.00</b>
20	87.50±02.01	<b>89.35±03.86</b>	83.52±03.67	83.03±03.76	80.18±02.91	84.52±03.05	88.90±03.47
30	84.95±03.57	<b>85.68±03.27</b>	80.05±03.16	78.22±03.31	77.30±02.44	77.35±04.13	83.97±04.12
40	82.20±01.94	<b>83.64±02.53</b>	77.41±03.62	76.14±03.33	75.34±01.88	74.35±02.19	83.47±02.62
50	82.00±02.15	<b>82.21±01.57</b>	76.87±02.41	74.51±02.63	73.32±02.28	73.76±01.95	82.13±03.12
60	79.53±02.44	79.96±02.94	75.62±02.12	72.68±02.02	74.72±02.33	71.86±01.46	<b>81.72±03.25</b>
70	79.94±01.98	79.50±01.80	76.40±03.02	73.87±02.48	74.99±02.21	73.43±01.33	<b>80.14±03.33</b>
80	80.31±01.73	76.79±01.89	73.82±02.42	72.26±01.76	74.62±02.70	70.20±02.04	<b>78.45±03.26</b>
90	78.94±02.60	75.50±02.10	72.32±01.84	70.96±01.68	73.99±03.03	71.10±02.47	<b>79.49±02.73</b>
100	78.55±01.94	73.24±01.75	71.20±02.18	72.03±01.54	75.07±02.48	70.88±01.40	<b>79.72±01.83</b>
110	77.50±01.75	71.80±02.10	71.19±02.10	70.78±01.97	75.94±01.73	70.46±02.45	<b>83.15±01.97</b>
113	79.60	72.30	71.42	72.48	73.05	70.27	<b>82.74</b>
Average	82.01	80.49	76.35	75.60	76.11	75.08	<b>83.66</b>
N	Correntropy						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
10	0.1787±0.1101	0.0840±0.0954	0.3204±0.1183	0.1994±0.1247	0.3162±0.1439	0.1512±0.1072	<b>0.0000±0.0000</b>
20	0.3270±0.0636	0.2713±0.1083	0.4544±0.0889	0.4392±0.0956	0.5036±0.1071	0.3871±0.1030	<b>0.1849±0.0514</b>
30	0.3937±0.1208	0.3867±0.0995	0.5662±0.0923	0.6162±0.0883	0.5789±0.0823	0.6521±0.1650	<b>0.2683±0.0635</b>
40	0.4733±0.0911	0.4375±0.0950	0.7194±0.1241	0.7534±0.1872	0.6857±0.0711	0.8528±0.1603	<b>0.2735±0.0392</b>
50	0.4843±0.0691	0.5024±0.0475	0.7458±0.0941	0.8172±0.1093	0.7998±0.0812	0.8413±0.1491	<b>0.3272±0.0498</b>
60	0.5707±0.0985	0.5687±0.1066	0.7858±0.0744	0.9123±0.1017	0.7463±0.1141	1.0110±0.1022	<b>0.3366±0.0630</b>
70	0.5544±0.0601	0.6596±0.0874	0.7516±0.1077	0.8848±0.1546	0.6933±0.0812	0.8803±0.0537	<b>0.3609±0.0663</b>
80	0.5424±0.0772	0.7135±0.0707	0.8688±0.0911	0.9858±0.1250	0.7062±0.0934	1.1352±0.1575	<b>0.3846±0.0730</b>
90	0.6329±0.0909	0.8180±0.1182	0.9332±0.0589	1.1122±0.1031	0.7664±0.1151	1.1002±0.2046	<b>0.3807±0.0644</b>
100	0.6188±0.0753	0.8805±0.1351	0.9704±0.0664	1.0406±0.1290	0.7376±0.0701	1.0906±0.1520	<b>0.3855±0.0340</b>
110	0.6873±0.0870	0.9955±0.1141	0.9919±0.0790	1.1137±0.1233	0.7187±0.0595	1.1320±0.1398	<b>0.4113±0.0586</b>
113	0.6349	1.0132	0.9937	1.1091	0.8288	0.9821	<b>0.4083</b>
Average	0.5082	0.6108	0.7585	0.8320	0.6734	0.8513	<b>0.3101</b>

all possible combinations of parameters. This strategy applies to all data sets and algorithms. We will provide detailed comparison of clustering performance of all methods in next subsection.

### 6.5. Clustering Performance

In this subsection, we evaluate the algorithms on PIX, Jaffe, Semeion, and Alphadigit data sets. For purpose of illustration, we visually show some examples of these data sets in Fig. 1. With the above discussed experimental settings, we present the experimental results in Tables 1 to 4, with  $N$  values being used



in our experiments. In each table, three subtables are given corresponding to three evaluating measures, respectively. The best performance is bold-faced.

From Tables 1 to 4, we can see that the KLS-NMF achieves the best performance among all methods in comparison. In clustering accuracy, KLS-NMF achieves the best performance in almost all cases and the improvements over other methods are significant. For example, on JAFFE data with large  $N$  value, KLS-NMF can improve the performance by about 8%. Generally, the proposed method can improve the average performance by around 3-6% compared with the most competing method. Among the baseline methods, it should be noted that none of them consistently achieves the best performance on all data sets, whereas the proposed method always shows stability on all data sets. For example, RMNMF has some of the best results on PIX, but its performance on other data sets are less competitive. For other evaluation metrics, similar observations can be found. When  $N$  increases, it is seen that all methods show degraded clustering performance, which is reasonable since it is increasingly difficult to performance clustering on a data set with more clusters. Compared with other methods, the performance of KLS-NMF degrades far less significantly, implying better clustering capability. For example, on Alphadigit data set, KLS-NMF and MKNMF have similar performance with  $N = 2$ . However, as  $N$  increases, KLS-NMF has superior performance to MKNMF with significant performance. Similar observations can be found for other methods and data sets. Moreover, the improvements of KLS-NMF over competing methods suggests that learning local similarity indeed provides advantages in clustering.

#### 6.6. Clustering Performance on Larger Data

In the above subsection, we have evaluated the proposed method on some widely used benchmark data sets. Among them, 3 out of 4 data sets have up to 10 classes while the rest one has up to 36 classes included in the experiment. In this subsection, we aim at testifying the capability of our method in handling larger data with more classes. To further testify how the proposed method performs on larger data, we conduct experiments on Faces94 data set. In this



Figure 2: Examples selected images from Faces94 data.

test, we use a subset of Faces94, which includes images of the males. The corresponding subset has up to 113 individuals, which is fairly large for this test. For purpose of clearer illustration, we show some examples of this data in Fig. 2. We follow the same settings as in above subsection and report the results in Table 5. It is observed that the proposed method outperforms the baseline methods with significant improvements. It is seen that the KLS-NMF obtains the best performance in all cases of accuracy, NMI, and correntropy. In purity, the KLS-NMF obtains the best performance in 8 out 12 cases. In the other cases, the KLS-NMF obtains the top second performance, which is very competitive to RMNMF and superior to other methods. These observations, again, ensure the effectiveness of the proposed method.

### 6.7. Convergence

In Section 5, we have provided theoretical analysis on the convergence of the proposed optimization algorithm. In this subsection, we will show some examples to further empirically testify this. In particular, we use Yale, PIX, Alphadigit, and Semeion data sets in this test. Without loss of generality, for these data sets, we use their smallest subsets as used in Section 6.5. For all these subsets, we fix the parameter  $\lambda = 0.001$  and set 1 for the radial of rbf kernel.

To show the convergence of  $\{W_t\}$  and  $\{G_t\}$ , we show how the differences of two consecutive  $W_t$ 's and  $G_t$ 's, i.e.,  $\|W_{t+1} - W_t\|_F$  and  $G_{t+1} - G_t$ , change with

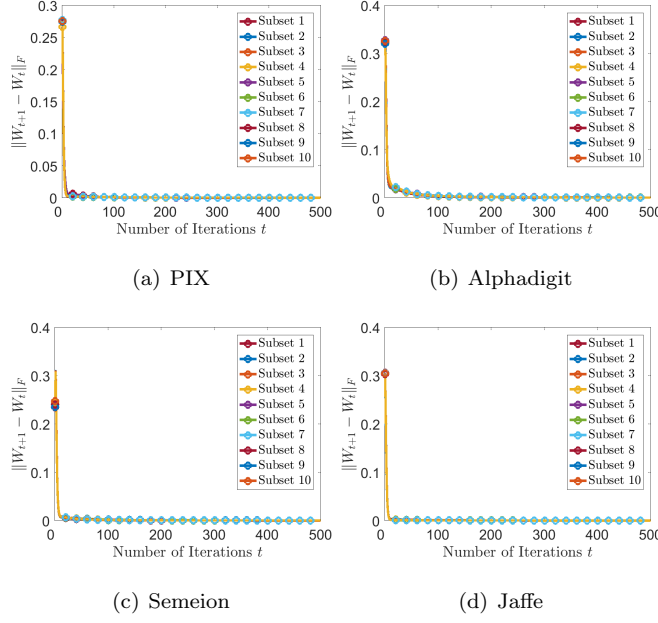


Figure 3: Example of the difference between consecutive  $W_t$ 's by KLS-NMF on PIX, Alphadigit, Semeion, and Jaffe.

respect to iteration number  $t$  in Figs. 3 and 4, respectively. It is seen that both  $\{W_t\}$  and  $\{G_t\}$  sequences can converge within a small number of iterations, which verifies the effectiveness and correctness of the optimization scheme.

Moreover, to further experimentally verify the convergence of objective value sequence, we show some results in Fig. 5. It is seen that the objective function indeed decreases its value and the objective value sequences tend to converge within about 100 iterations, which verifies the fast convergence and effectiveness of the proposed method. In addition to the theoretical guarantees, these empirical observations indeed further strengthen the applicability of our method in real world problems.

To further testify the efficiency of the KLS-NMF, we compare the convergence behavior of all methods in comparison. In particular, for each method we record the number of iterations needed for convergence. Here, we terminate

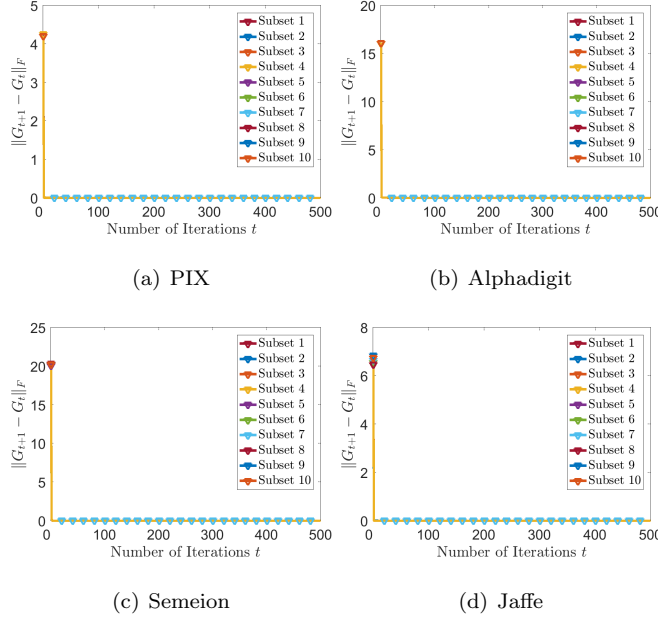


Figure 4: Example of the difference between consecutive  $G_t$ 's by KLS-NMF on PIX, Alphadigit, Semeion, and Jaffe.

Table 6: Average Number of Iteration Used for Convergence

Data	Average Number of Iteration Used for Convergence						
	WNMF	RMNMF	CNMF	KNMF	ONMF	MKNMF	KLS-NMF
Jaffe	383	500	500	500	500	11	12
PIX	462	500	500	350	500	11	18
Semeion	191	500	500	500	500	13	47
Alphadigit	500	500	500	500	500	15	31
Faces94	500	500	500	500	500	153	35

the algorithm when the difference of objective value of two consecutive updates is smaller than 0.001 or when a maximum number of 500 iterations is reached. Without loss of generality, we conduct these experiments on the subsets with smallest  $N$  value as used in Tables 1 to 5 and the average iteration number is reported in Table 6. It is seen that KLS-NMF needs the least number of iterations to converge on Faces94 data set while the second least on the other 4 data sets. Generally, the KLS-NMF can converge within 50 iterations on all

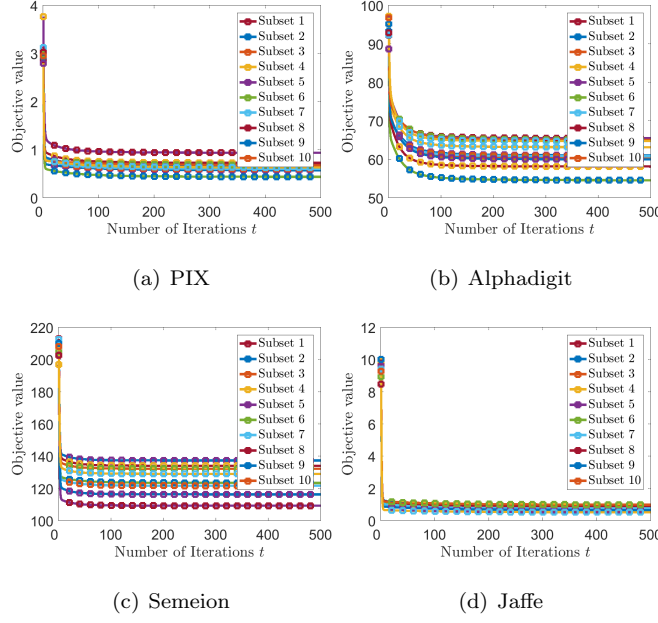


Figure 5: Example of objective value sequences by KLS-NMF on PIX, Alphadigit, Semeion, and Jaffe.

data sets whereas the other methods cannot. Moreover, except MKNMF and WNMF, often times the other methods cannot converge within 500 iterations. Concerning the superior performance in clustering, these observations confirm the fast convergence and efficiency of the KLS-NMF.

### 6.8. Ablation Study

In this test, we empirically show the importance of local similarity learning in the KLS-NMF model. In particular, we treat the KLS-NMF with  $\lambda = 0$  as baseline and compare it with the KLS-NMF with  $\lambda$  being general nonzero values. We follow the experimental settings in previous tests and use the RBF kernel with radial ranging within  $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ . Without loss of generality, we use the overall Alphadigit and Jaffe data sets. For each data set, we report the performance of the KLS-NMF with respect to the values of radial. Particularly, for KLS-NMF with  $\lambda$  being general values, for each radial value

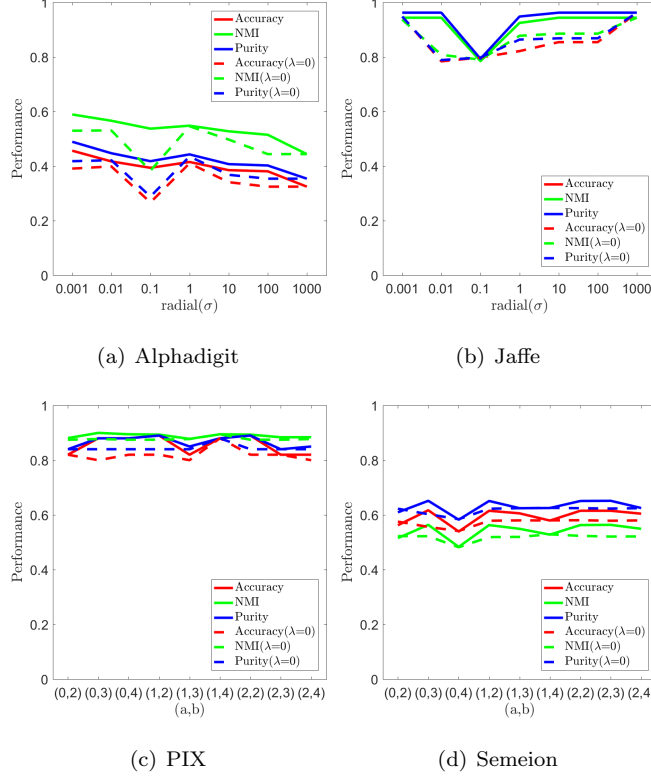


Figure 6: Illustration of the importance of local similarity learning by the performance comparison of KLS-NMF with  $\lambda$  equal to general values and 0, respectively. (a)-(b) are the results on Alphadigit and Jaffe data sets with RBF kernel. (c)-(d) are the results on PIX and Semeion data sets with polynomial kernel.

we tune  $\lambda \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$  such that the best performance is obtained. We show the results in Figs. 6(a) and 6(b). It is seen that the KLS-NMF always has better performance with  $\lambda$  being general values than 0. This observation implies the effectiveness and importance of the local similarity learning in the KLS-NMF.

Moreover, to better illustrate our observation and conclusion, we further test the performance of the KLS-NMF with other kernel functions to show that

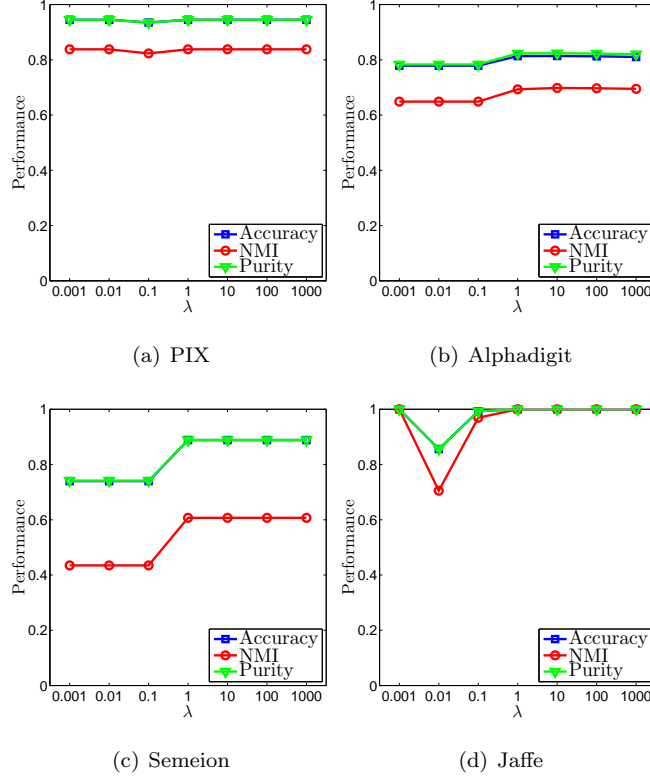


Figure 7: Performance variations of KLS-NMF in accuracy, NMI, and purity with respect to different  $\lambda$  values on PIX, Alphadigit, Semeion, and Jaffe.

the above conclusion holds for general kernels. Without loss of generality, we adopt the polynomial kernel with different parameters in this test. For the polynomial kernel, we vary the two parameters with different combinations. For the other settings, we remain the same as above. Without loss of generality, we conduct experiments on the overall PIX and Semeion data sets and show the results in Figs. 6(c) and 6(d). It is seen that the performance of KLS-NMF with polynomial kernel has similar pattern to RBF kernel, which suggests the effectiveness of the local similarity learning in KLS-NMF.

### 6.9. Parameter Sensitivity

For unsupervised learning methods, it is still an open problem to determine the optimal parameter. To study how the parameter affects the final clustering performance of the KLS-NMF, we conduct experiments to empirically show the sensitivity of the proposed method to different parameter values. In particular, we test KLS-NMF and show its performance with respect to different  $\lambda$  values. In this test, we use Alphadigit, Jaffe, PIX, and Semeion data sets, where we use the 10 subsets with the smallest  $N$  values as used in Tables 1 to 4. We plot the performance versus  $\lambda$  in terms of accuracy, NMI, and purity, where for each  $\lambda$  value we tune the kernel parameter such that the best performance is obtained. It is observed that KLS-NMF is quite insensitive to variation of parameter values and promising performance can be obtained with a wide range of parameter variation. Similar observation can also be found for Correntropy. This insensitivity to parameter variation may reduce parameter tuning efforts, affording ease of use of our models in real world applications.

## 7. Conclusion

This paper proposes a novel NMF method, which simultaneously exploits global and local structures of the data to construct basis vectors and coefficient matrix. The learned basis and coefficients well preserve intrinsic geometrical structures of the data and thus are more representative. An orthogonality constraint enforced on the coefficient and the embedding of local similarity learning mutually ensure the uniqueness of the factorization and provide an immediate and improved clustering interpretation. Nonlinear variant is developed and efficient multiplicative updating rules are derived with theoretical convergence guarantee. Extensive experimental results have verified the effectiveness of the proposed method.



## Acknowledgment

C. Chen is the corresponding author. This work is supported by National Natural Foundation of China (NSFC) under Grants 61806106, 61802215, and 61806045, and Natural Science Foundation of Shandong Province under Grants ZR2019QF009 and ZR2019BF011; Q.C. is partially supported by NIH UH3 NS100606-03 and a grant from the University of Kentucky.

## References

- [1] C. Peng, Q. Cheng, Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data, *IEEE Transactions on Neural Networks & Learning Systems* PP (99) (2020) 1–15.
- [2] X. Zhu, S. Zhang, Y. Li, J. Zhang, L. Yang, Y. Fang, Low-rank sparse subspace for spectral clustering, *IEEE Transactions on Knowledge and Data Engineering* 31 (8) (2019) 1532–1543. doi:[10.1109/TKDE.2018.2858782](https://doi.org/10.1109/TKDE.2018.2858782).
- [3] W. Wang, H. Zhang, Z. Zhang, L. Liu, L. Shao, Sparse graph based self-supervised hashing for scalable image retrieval, *Information Sciences* 547 (2021) 622 – 640. doi:<https://doi.org/10.1016/j.ins.2020.08.092>.
- [4] L. Chen, D. Chen, F. Yang, J. Sun, A deep multi-task representation learning method for time series classification and retrieval, *Information Sciences* 555 (2021) 17 – 32. doi:<https://doi.org/10.1016/j.ins.2020.12.062>.
- [5] C. Peng, Z. Kang, Q. Cheng, Subspace clustering via variance regularized ridge regression, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 682–691.
- [6] C. Peng, C. Chen, Z. Kang, J. Li, Q. Cheng, Res-pca: A scalable approach to recovering low-rank matrices, in: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7309–7317.

- [7] C. Peng, Y. Chen, Z. Kang, C. Chen, Q. Cheng, Robust principal component analysis: A factorization-based approach with linear complexity, *Information Sciences* 513 (2020) 581–599.
- [8] F. Nie, H. Huang, C. Ding, Low-rank matrix recovery via efficient Schatten  $p$ -norm minimization, in: *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
- [9] X. Ma, B. Zhang, C. Ma, Z. Ma, Co-regularized nonnegative matrix factorization for evolving community detection in dynamic networks, *Information Sciences*.
- [10] X. Ma, P. Sun, G. Qin, Nonnegative matrix factorization algorithms for link prediction in temporal networks using graph communicability, *Pattern Recognition* 71 (2017) 361 – 374. doi:<https://doi.org/10.1016/j.patcog.2017.06.025>.
- [11] C. Peng, Z. Kang, H. Li, Q. Cheng, Subspace clustering using log-determinant rank approximation, in: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2015, pp. 925–934.
- [12] J. Ma, Y. Zhang, L. Zhang, Discriminative subspace matrix factorization for multiview data clustering, *Pattern Recognition* 111 (2021) 107676. doi:<https://doi.org/10.1016/j.patcog.2020.107676>.
- [13] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, *Nature* 401 (6755) (1999) 788–791.
- [14] S. E. Palmer, Hierarchical structure in perceptual representation, *Cognitive psychology* 9 (4) (1977) 441–474.
- [15] E. Wachsmuth, M. Oram, D. Perrett, Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque, *Cerebral Cortex* 4 (5) (1994) 509–522.

- [16] N. K. Logothetis, D. L. Sheinberg, Visual object recognition, *Annual review of neuroscience* 19 (1) (1996) 577–621.
- [17] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, R. A. Harshman, Indexing by latent semantic analysis, *JAsIs* 41 (6) (1990) 391–407.
- [18] S. A. Vavasis, On the complexity of nonnegative matrix factorization, *SIAM Journal on Optimization* 20 (3) (2009) 1364–1377.
- [19] S. Arora, R. Ge, R. Kannan, A. Moitra, Computing a nonnegative matrix factorization—provably, *SIAM Journal on Computing* 45 (4) (2016) 1582–1611. [doi:10.1137/130913869](https://doi.org/10.1137/130913869).
- [20] C. Ding, X. He, H. D. Simon, On the equivalence of nonnegative matrix factorization and spectral clustering, in: *Proceedings of the 2005 SIAM International Conference on Data Mining*, SIAM, 2005, pp. 606–610.
- [21] I. S. Dhillon, Y. Guan, B. Kulis, Weighted graph cuts without eigenvectors a multilevel approach, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29 (11) (2007) 1944–1957.
- [22] R. Zass, A. Shashua, A unifying approach to hard and probabilistic clustering, in: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, Vol. 1, IEEE, 2005, pp. 294–301.
- [23] C. Ding, T. Li, W. Peng, H. Park, Orthogonal nonnegative matrix t-factorizations for clustering, in: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2006, pp. 126–135.
- [24] S. Peng, W. Ser, B. Chen, Z. Lin, Robust orthogonal nonnegative matrix tri-factorization for data representation, *Knowledge-Based Systems* 201-202 (2020) 106054.

- [25] I. Buciu, N. Nikolaidis, I. Pitas, Nonnegative matrix factorization in polynomial feature space, *IEEE Transactions on Neural Networks* 19 (6) (2008) 1090–1100.
- [26] Y.-X. Wang, Y.-J. Zhang, Nonnegative matrix factorization: A comprehensive review, *IEEE Transactions on Knowledge and Data Engineering* 25 (6) (2013) 1336–1353.
- [27] C. H. Ding, T. Li, M. I. Jordan, Convex and semi-nonnegative matrix factorizations, *IEEE transactions on pattern analysis and machine intelligence* 32 (1) (2010) 45–55.
- [28] P. He, X. Xu, J. Ding, B. Fan, Low-rank nonnegative matrix factorization on stiefel manifold, *Information Sciences* 514 (2020) 131–148.
- [29] D. Cai, X. He, J. Han, T. S. Huang, Graph regularized nonnegative matrix factorization for data representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (8) (2011) 1548–1560.
- [30] J. Huang, F. Nie, H. Huang, C. Ding, Robust manifold nonnegative matrix factorization, *ACM Transactions on Knowledge Discovery from Data (TKDD)* 8 (3) (2014) 11.
- [31] H. Cai, B. Liu, Y. Xiao, L. Lin, Semi-supervised multi-view clustering based on orthonormality-constrained nonnegative matrix factorization, *Information Sciences* 536 (2020) 171 – 184. doi:<https://doi.org/10.1016/j.ins.2020.05.073>.
- [32] Y. Meng, R. Shang, F. Shang, L. Jiao, S. Yang, R. Stolkin, Semi-supervised graph regularized deep nmf with bi-orthogonal constraints for data representation, *IEEE Transactions on Neural Networks and Learning Systems* 31 (9) (2020) 3245–3258. doi:[10.1109/TNNLS.2019.2939637](https://doi.org/10.1109/TNNLS.2019.2939637).
- [33] X. Liu, L. Wang, J. Zhang, J. Yin, H. Liu, Global and local structure preservation for feature selection, *IEEE Transactions on Neural Networks and Learning Systems* 25 (6) (2014) 1083–1095.

- [34] J. Chen, Z. Ma, Y. Liu, Local coordinates alignment with global preservation for dimensionality reduction, *IEEE transactions on neural networks and learning systems* 24 (1) (2013) 106–117.
- [35] K. Luong, R. Nayak, Learning inter- and intra-manifolds for matrix factorization-based multi-aspect data clustering, *IEEE Transactions on Knowledge & Data Engineering* PP (99) (2020) 1–1.
- [36] H. Wang, S. Chen, Z. Hu, W. Zheng, Locality-preserved maximum information projection, *IEEE Transactions on Neural Networks* 19 (4) (2008) 571–585.
- [37] J. Chen, J. Ye, Q. Li, Integrating global and local structures: A least squares framework for dimensionality reduction, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [38] L. Zhen, D. Peng, W. Wang, X. Yao, Kernel truncated regression representation for robust subspace clustering, *Information Sciences* 524 (2020) 59 – 76. doi:<https://doi.org/10.1016/j.ins.2020.03.033>.
- [39] F. R. Chung, *Spectral graph theory*, Vol. 92, American Mathematical Soc., 1997.
- [40] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, *IEEE Transactions on Knowledge & Data Engineering* 23 (6) (2011) 902–913.
- [41] C. Peng, Q. Zhang, Z. Kang, C. Chen, Q. Cheng, Kernel two-dimensional ridge regression for subspace clustering, *Pattern Recognition* (2020) 107749doi:<https://doi.org/10.1016/j.patcog.2020.107749>.
- [42] F. Nie, X. Wang, H. Huang, Clustering and projected clustering with adaptive neighbors, in: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2014, pp. 977–986.

- [43] C. Peng, Z. Kang, Y. Hu, J. Cheng, Q. Cheng, Robust graph regularized nonnegative matrix factorization for clustering, *Acm Transactions on Knowledge Discovery from Data* 11 (3) (2017) 33.
- [44] D. D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, in: *Advances in neural information processing systems*, 2001, pp. 556–562.
- [45] L. Xu, M. I. Jordan, On convergence properties of the em algorithm for gaussian mixtures, *Neural computation* 8 (1) (1996) 129–151.
- [46] T. Ensari, J. Chorowski, J. M. Zurada, Correntropy-based document clustering via nonnegative matrix factorization, in: *International Conference on Artificial Neural Networks & Machine Learning-volume Part II*, 2012.
- [47] C. Peng, Z. Kang, S. Cai, Q. Cheng, Integrate and conquer: Double-sided two-dimensional k-means via integrating of projection and manifold construction, *ACM Trans. Intell. Syst. Technol.* 9 (5).
- [48] Y.-D. Kim, S. Choi, Weighted nonnegative matrix factorization, in: *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, IEEE, 2009, pp. 1541–1544.
- [49] Y. Yang, Y. Zhang, Y. Zhu, Application of the multi-kernel non-negative matrix factorization on the mechanical fault diagnosis, *Advances in Mechanical Engineering* 7 (5) (2015) 1687814015584494.