Entropy Minimizing Matrix Factorization

Mulin Chen and Xuelong Li, Fellow, IEEE

Abstract—Nonnegative matrix factorization (NMF) is a widely used data analysis technique and has yielded impressive results in many real-world tasks. Generally, existing NMF methods represent each sample with several centroids and find the optimal centroids by minimizing the sum of the residual errors. However, outliers deviating from the normal data distribution may have large residuals and then dominate the objective value. In this study, an entropy minimizing matrix factorization (EMMF) framework is developed to tackle the above problem. Considering that outliers are usually much less than the normal samples, a new entropy loss function is established for matrix factorization, which minimizes the entropy of the residue distribution and allows a few samples to have large errors. In this way, the outliers do not affect the approximation of normal samples. Multiplicative updating rules for EMMF are derived, and the convergence is proven theoretically. In addition, a Graph regularized version of EMMF (G-EMMF) is also presented, which uses a data graph to capture the data relationship. Clustering results on various synthetic and real-world datasets demonstrate the advantages of the proposed models, and the effectiveness is also verified through the comparison with state-of-the-art methods.

Index Terms—Artificial intelligence, data representation, entropy loss, pattern recognition, robustness.

I. INTRODUCTION

NONNEGATIVE matrix factorization (NMF) is a popular unsupervised machine learning technique for handling matrix data. Based on the matrix factorization theory [1], Lee and Seung [2] imposed the nonnegative constraint to learn the part-of-whole interpretations. After that, NMF has attracted increasing attention due to its simplicity and interpretability, and shown encouraging performance in many real-world tasks, such as image processing [3], [4], document analysis [5], hyperspectral imagery [6], and recommendation systems [7].

Specifically, given the nonnegative data matrix, where each column is a sample, NMF approximates it as the product of two nonnegative factor matrices. One consists of a set of basis vectors, and another one is regarded as a coefficient matrix. By minimizing the residual error, each sample is represented by a linear combination of the basis vectors, and the coefficient matrix reflects the relationship between the basis vectors and samples. Therefore, the basis vectors act as the cluster centroids, and the coefficient matrix can be considered as the cluster indicator. In addition, benefiting from the nonnegative constraint, NMF allows only additive operation. Consequently, a part-based representation is obtained, which is able to provide an intuitive understanding of the input data.

Over the past decades, NMF has been studied from a wide variety of perspectives. For example, researchers have proved the connection between NMF and some popular machine learning techniques [8]–[10], such as k-means, spectral clustering, and linear discriminant analysis. Inspired by subspace clustering, a number of techniques [11]–[13] have been proposed to perform NMF in the low-dimensional subspace. Some graph-based NMF methods are proposed to exploit the data structure [4], [14]–[17], while some others deal with the missing data [18]–[21]. There are also many cost functions used for estimating the NMF model, such as β divergence [22] and Itakura–Saito (IS) divergence [23]. Recently, deep NMF [24]–[26] has become an attractive research area. Despite its salient properties and wide usage, traditional NMF has some major drawbacks, such as weak robustness, sensitivity to initialization, and the infeasibility in kernel space. In this article, we devote to tackle the robustness problem.

Traditional NMF models proposed by Lee and Seung [2] employ the Euclidean distance and the Kullback–Leibler divergence to calculate the residual error, which relies on an assumption that the noise is Gaussian or Poisson. Therefore, they are sensitive to outliers with large noise. To improve the robustness, some variants of NMF have been presented. Instead of using the Frobenius-norm, Ke and Kanade [27] proposed an α-stable MF, which is a generalized framework that includes Cauchy NMF [29] and Lévy NMF [30]. Student’s t distribution has also been used to formulate NMF models [31], [32], which shows robustness to heavy-tailed noise and outliers. The above methods weaken the effects of outliers by utilizing different loss functions and distributions. However, the effects of outliers still exist. If the error is extremely large, the outliers will affect the results as well. Gao et al. [33] designed a capped norm matrix factorization model. Similarly, Guan et al. [34] proposed the truncated Cauchy NMF. They found the outliers directly by thresholding the residual error and setting them as a constant. This strategy removes the outliers thoroughly, but it is unrealistic to find a suitable threshold for real-world applications.

In this article, an entropy minimizing matrix factorization (EMMF) framework is presented to improve the robust-
ness. Different from the previous works, we do not minimize the residual errors of all the samples. A new entropy loss function is designed, which models the whole distribution of the residues and avoids the effects of outliers naturally. The proposed loss function could also be applied in other tasks involving matrix computation, such as sparse representation [35] and matrix completion [36]. Since the data relationship is crucial for clustering [37], [38], a Graph regularized version of EMMF (G-EMMF) is also developed. The main contributions made in this study are summarized as follows.

1) We design a general entropy loss function for matrix factorization. By minimizing the entropy of the residue distribution, the proposed EMMF allows a few samples to be with relatively large errors and focuses on approximating most of the rest. Therefore, outliers with large errors do not affect the updating of centroids.

2) We provide efficient optimization algorithms for estimating the proposed models. The optimal solution can be obtained by multiplicative updating rules with proved convergence. The computation costs of the optimization algorithms are almost the same as NMF and its graph variants, which guarantees the practicability for real-world tasks.

3) We conduct extensive experiments to validate the effectiveness of the proposed framework. As demonstrated by the results, the objective function of EMMF is insensitive to outliers with extremely large errors, and it works well for data without outliers. The proposed G-EMMF also shows good performance.

This article is organized as follows. Section II reviews some existing NMF algorithms. Section III introduces the EMMF formulation and provides the corresponding optimization algorithm. Section IV presents the G-EMMF. Section V gives the experimental results of EMMF and discusses its advantages. Section VI shows the clustering performance of G-EMMF. Section VII concludes this article.

Notations: In this article, we write the matrices as uppercase and the vectors as lowercase. For a matrix \( A \), its \((i, k)\)th element is defined as \( a_{ik} \). Its ith row and column are denoted as \( a_i \) and \( a_i^T \), respectively. The trace of \( A \) is defined as \( \text{Tr}(A) \). The transposes of \( A \) and \( a_i \) are indicated by \( A^T \) and \( a_i^T \). \( I \) is the identity matrix. The \( \ell_p \) norm of \( a_i \) is calculated as

\[
\| a_i \|_p = \left( \sum_{k=1}^{d} |a_{ik}|^p \right)^{1/p} \quad (p > 0).
\]

II. PRELIMINARY

In this section, we revisit the formulations of the classical NMF and some existing methods. Numerous algorithms have been proposed to improve NMF from different aspects, and we mainly focus on the robust variants.

A. Nonnegative Matrix Factorization

Supposing that the data matrix is \( X = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^{d \times n} \) and the desired centroid number is \( c \), NMF aims to find the nonnegative matrices \( U \in \mathbb{R}^{d \times c} \) and \( V \in \mathbb{R}^{n \times c} \) which satisfy \( X \approx UV^T \). The least-squares error objective function is formulated as

\[
\min_{U \geq 0, V \geq 0} \|X - UV^T\|_F^2 = \sum_{i=1}^{n} \| x_i - UV^T x_i \|_2^2
\]

where \( \| \cdot \|_F \) is the Frobenius norm. Lee and Seung [2] derived the multiplicative updating rules for the above problem

\[
\begin{align*}
U_{ik} &\leftarrow U_{ik} \frac{(XV)_{ik}}{(UV^T)_{ik}} \\
V_{jk} &\leftarrow V_{jk} \frac{(X^T U)_{jk}}{(VU^T)_{jk}}.
\end{align*}
\]

Instead of using the Euclidean distance, another commonly used NMF model (KLNMF) takes the Kullback–Leibler divergence as the loss function

\[
\min_{U \succeq 0, V \succeq 0} \text{KL}(X||UV^T)
\]

where \( \text{KL}(A||B) = \sum_{i=1}^{d} \sum_{k=1}^{c} (A_{ik} \log(A_{ik}/B_{ik}) - A_{ik} + B_{ik}) \).

Accordingly, the updating rules are given as

\[
\begin{align*}
U_{ik} &\leftarrow U_{ik} \frac{\sum_{j=1}^{d} X_{ij} V_{jk} / (UV^T)_{ij}}{\sum_{j=1}^{d} V_{jk}} \\
V_{jk} &\leftarrow V_{jk} \frac{\sum_{i=1}^{n} X_{ij} U_{ik} / (UV^T)_{ij}}{\sum_{i=1}^{n} U_{ik}}.
\end{align*}
\]

By solving \( U \) and \( V \) iteratively, the local optimal solution of both problems (1) and (3) will be found. Besides, the problems can be decomposed into the approximation of each sample, i.e., \( x_i \approx UV^T x_i \). The largest element in \( v_j \) indicates the closet centroid to \( x_i \). Therefore, \( V \) indicates the clustering results directly.

Equations (1) and (3) correspond to the maximum likelihood estimation of Gaussian and Poisson distributions, respectively, so they fail to handle the heavy-tailed noise. Taking the least-squares error loss for instance, since it minimizes the errors of all the samples, outliers with large noise will affect the loss function inevitably. Furthermore, the square of the errors compounds the problem.

B. Robust NMF

Based on whether a generative model is established, existing robust NMF methods can be roughly classified into two categories: 1) distribution-based methods and 2) distance-based methods.\(^1\)

The first class introduces a generative model based on a robust noise distribution. Lam [40] developed the \( l_1 \)-norm loss function based on the Laplacian distribution. Simekli et al. [28] proposed the matrix factorization model with a stable distribution, which is suitable to handle the impulsive data. Liutkus et al. [29] presented the Cauchy NMF for sound source separation such that the robustness to impulsive noise is improved. Magron et al. [30] used the Lévy distribution to model the heavy-tailed signals. Yoshii et al. [32] improved the NMF model based on complex Student’s \( t \) distributions, which shows more robustness to the heavy-tailed distributed data. Leglaive et al. [31] considered the signals as Student’s \( t \) latent variables and developed a Bayesian framework for multichannel source separation. Jia and Darrell [41] proposed the gamma-compound-Laplace distribution to model the noise in gradient-based image descriptors.

\(^1\)Different strategies may lead to the same model. In fact, each distance category may lead to different optimality of the loss function.
The second class formulates the loss function with different kinds of distances directly. Guan et al. [42] handled the Laplace noise with the Manhattan distance. Févotte and Doubigeon [6] utilized the $\ell_{2,1}$-norm regularization term to fit the data noise. Ke and Kanade [27] replaced the Frobenius norm with the $\ell_1$ distance, which yields the following formulation:

$$\min_{U \geq 0, V \geq 0} \|X - UV^T\|_1 = \sum_{i=1}^{n} |x_i - Uv_i^T|.$$  \tag{5}

Instead of squaring the errors, problem (5) takes the square operation. Besides, because $\log \frac{1}{x}$ is the concept of entropy. Defining $p_i$ as the probability distribution of a random variable, the Shannon entropy is given by

$$H = -\sum_i p_i \log p_i.$$  \tag{7}

According to the information theory [47], the entropy is maximized when the distribution is uniform, i.e., all the probabilities are with the same value. Conversely, the less value of the entropy indicates the imbalanced distribution.

Defining $M = X - UV^T$, we define $p_i$ as

$$p_i = \frac{\|m_i\|_2}{\|M\|_{2,1}}, \quad i \in [1,n]$$  \tag{8}

where $m_i$ is the $i$th column of $M$. According to the aforementioned definitions, it is manifest that $\sum_{i=1}^{n} p_i = 1$. Therefore, $\{p_i\}$ is exact the samples’ residue distribution, and the entropy is computed as

$$H(M) = -\sum_{i=1}^{n} \frac{\|m_i\|_2}{\|M\|_{2,1}} \log \frac{\|m_i\|_2}{\|M\|_{2,1}}.$$  \tag{9}

The value of $H(M)$ is minimized when the residue distribution is extremely imbalanced. However, the distribution cannot reflect the exact value of the residues, i.e., $H(M)$ equals to $H(\rho M)$ for any $\rho > 0$. To keep the uniqueness, the matrix residue $\|M\|_{2,1}$ should be also minimized. Since both the entropy and matrix residue are with positive values, we propose to minimize their product

$$\min_{U,V} H(M) \times \|M\|_{2,1}$$  \tag{10}

s.t. $M = X - UV^T$, $U \geq 0$, $V \geq 0$ which yields the objective function of EMMF

$$\min_{U,V} -\sum_{i=1}^{n} \|m_i\|_2 \log \frac{\|m_i\|_2}{\|M\|_{2,1}}$$  \tag{11}

s.t. $M = X - UV^T$, $U \geq 0$, $V \geq 0$.

If several samples are with large errors, EMMF ignores them and moves the centroids toward the remaining ones. By searching an imbalanced residue distribution, the effects of the outliers are avoided. One may doubt the correctness of the model for the data without outliers. In fact, since the samples from the same class usually obey the uniform distribution, the centroid will not change too much if a few samples are considered outliers. Therefore, the mistaken outliers can still be connected with the correct centroid. This statement will be verified in Section V-A.

### III. Entropy Minimizing Matrix Factorization

In this section, the EMMF framework is introduced. The optimization strategy and the convergence analysis are also given.

#### A. Methodology

Before describing the formulation of EMMF, we first introduce the concept of entropy. Defining $\{p_i\}$ as the probability distribution of a random variable, the Shannon entropy is given by

$$\mathcal{L}_1(m_i) = -\sum_{i=1}^{n} \frac{|m_i|_2}{\|M\|_{2,1}} \log \frac{|m_i|_2}{\|M\|_{2,1}} + \mathcal{G}(m_i, \Lambda_1)$$  \tag{12}

where $\mathcal{G}(m_i, \Lambda_1)$ represents the constraint on $m_i$ and $\Lambda_1$ is the Lagrange multiplier. Computing the derivative of $\mathcal{L}_1(m_i)$ and setting it as zero, we get the optimal solution as

$$Q_{ii} m_i + \frac{\partial \mathcal{G}(m_i, \Lambda_1)}{\partial m_i} = 0$$  \tag{13}

where $Q \in \mathbb{R}^{n \times n}$ is the diagonal matrix with

$$Q_{ii} = -\frac{1}{\|m_i\|_2} \log \frac{|m_i|_2}{\|M\|_{2,1}}.$$  \tag{14}

In the implementation, we add a small enough factor $\epsilon$ on $\|m_i\|_2$ to prevent it from being zero. When $Q$ is set as
stationary, (13) is also the optimal solution to the following problem:

\[
\min_{M} \text{Tr}(M^TQ^TM) = \sum_{i=1}^{n} Q_{ik}\|m_i\|_2^2
\]

s.t. \(M = X - UV^T\), \(U \geq 0\), \(V \geq 0\). (15)

Then, the optimal \(M\) of objective (11) can be obtained by solving problem (15). Accordingly, we search the optimal \(U\) and \(V\) by solving

\[
\min_{U \geq 0, V \geq 0} \text{Tr}[(X - UV^T)Q(X - UV^T)^T].
\]

In each iteration, \(Q\) is updated with the current \(U, V\) according to (14). The updating rules of \(U\) and \(V\) are given as follows. Updated \(U\), problem (16) becomes

\[
\min_{U \geq 0} \text{Tr}(UV^TQV^TU) - 2\text{Tr}(XQVU^T).
\]

The above subproblem is convex, and the Lagrangian function is

\[
L_2(U) = \text{Tr}(UV^TQV^TU) - 2\text{Tr}(XQVU^T) + \Lambda_2 U^T
\]

where \(\Lambda_2 \in \mathbb{R}^{d \times c}\) is the Lagrangian multiplier. Let \(((\partial L_2(U))/\partial U) = 0; we have

\[
2UV^TQV - 2XQV + \Lambda_2 = 0.
\]

(19)

According to the KKT conditions \((\Lambda_2)_{ik} U_{ik} = 0\), we get

\[
(UV^TQV)_{ik} U_{ik}^2 - (XQV)_{ik} U_{ik}^2 = 0.
\]

(20)

Then, the updating rule of \(U\) is

\[
U_{ik} \leftarrow U_{ik} \sqrt{(XQV)_{ik}/(UV^TQV)_{ik}}.
\]

(21)

At convergence, the equality holds for (21), so the condition in (19) is satisfied. Updated \(V\), the subproblem is

\[
\min_{V \geq 0} \text{Tr}(V^TQVU^TU) - 2\text{Tr}(V^TXQVU^TU).
\]

The Lagrangian function is

\[
L_3(V) = \text{Tr}(V^TQVU^TU) - 2\text{Tr}(V^TXQVU^TU) + \Lambda_3 V^T
\]

where \(\Lambda_3 \in \mathbb{R}^{n \times c}\) is the Lagrangian multiplier. Similar to (20), we have

\[
(QVU^TU)_{ik} V_{ik}^2 - (XQVU^TU)_{ik} V_{ik}^2 = 0
\]

(24)

and the updating rule of \(V\) is

\[
V_{ik} \leftarrow V_{ik} \sqrt{(XQVU^TU)_{ik}/(QVU^TU)_{ik}}.
\]

(25)

At convergence, \(V\) satisfies the condition \(((\partial L(V, \Lambda_3))/\partial V) = 0\).

The details of the optimization for problem (16) are described in Algorithm 1. In each iteration, the computation of the diagonal matrix \(Q\) requires \(O(ndc)\) operations. The costs for updating \(U\) and \(V\) are also \(O(ndc)\). After \(t\) iterations, the overall cost of EMMF is \(O(tndc)\). Note that the suggested optimization algorithm is not the only choice. The proposed model can be also optimized by other algorithms, such as gradient descent [49], expectation–maximization [23], and alternating direction method of multipliers [15].

### C. Convergence

The convergence analysis consists of two parts. First, we demonstrate that objective (11) can be optimized by solving problem (15). After that, since the updating rules of \(U\) and \(V\) are in similar forms, we only prove the convergence of problem (17).

1) **Convergence of Problem (15):** The convergence is proven by introducing the following theorem.

**Theorem 1:** The optimization of problem (15) decreases the objective value of problem (11) monotonically.

**Proof:** Denote the value of \(M\) and \(Q\) at the \(t\)th iteration as \(M(t)\) and \(Q(t)\), and suppose that the value of problem (15) decreases through the optimization, i.e.,

\[
\sum_{i=1}^{n} Q_{ik}(t) \|m_i(t)\|_2^2 \geq \sum_{i=1}^{n} Q_{ik}^{(t+1)} \|m_i^{(t+1)}\|_2^2.
\]

(26)

With the definition of \(Q\) in (14), the above inequality is transformed into

\[
\sum_{i=1}^{n} -\|m_i^{(t)}\|_2^2 \log \left(\frac{\|m_i^{(t)}\|_2}{\|M(t)\|_2}ight) \geq \sum_{i=1}^{n} -\|m_i^{(t+1)}\|_2^2 \log \left(\frac{\|m_i^{(t+1)}\|_2}{\|M(t+1)\|_2}ight)
\]

(27)

which leads to

\[
\sum_{i=1}^{n} \left(\|m_i^{(t+1)}\|_2 - \|m_i^{(t)}\|_2\right) \log \left(\frac{\|m_i^{(t+1)}\|_2}{\|M(t+1)\|_2}\right) \geq 0.
\]

(28)

According to the log sum inequality [50], we know that

\[
\sum_{i=1}^{n} \|m_i^{(t+1)}\|_2 \left(\log \frac{\|m_i^{(t+1)}\|_2}{\|M(t+1)\|_2} - \log \frac{\|M(t+1)\|_2}{\|M(t+1)\|_2}\right) \geq 0.
\]

(29)

Summing up (28) and (29), the following inequality holds:

\[
\sum_{i=1}^{n} \|m_i^{(t)}\|_2 \log \left(\frac{\|m_i^{(t)}\|_2}{\|M(t)\|_2}\right) \geq \sum_{i=1}^{n} \|m_i^{(t+1)}\|_2 \log \left(\frac{\|m_i^{(t+1)}\|_2}{\|M(t+1)\|_2}\right)
\]

(30)

which completes the proof.

---

**Algorithm 1 Optimization Algorithm of EMMF**

**Input:** Data matrix \(X\), centroid number \(c\).

1. Initialize \(U\) and \(V\).
2. **repeat**
   3. Compute \(Q\) with Eq. (14).
   4. Update \(U\) with Eq. (21).
   5. Update \(V\) with Eq. (25).
3. **until** Converge

**Output:** Optimal \(U, V\).
2) Convergence of Problem (17): To demonstrate that the updating rule (21) decreases the value of problem (17), the following definition [2] is introduced.

Definition 1: \(g(U, \tilde{U})\) is the auxiliary function for \(f(U)\) if, for any \(U\) and \(\tilde{U}\), it satisfies

\[
g(U, \tilde{U}) \geq f(U), \quad g(U, U) = f(U).
\]

As proved by Lee and Seung [2], we have the following lemma.

Lemma 1: Given the auxiliary function \(g(U, \tilde{U})\), \(f(U^{t+1}) \leq f(U^{t})\) holds if \(U^{t+1}\) is the solution to

\[
\min_{U} g(U, U^{t}).
\]

We propose the following theorem to demonstrate the convergence of problem (22).

Theorem 2: Updating rule (21) decreases the Lagrangian function \(\mathcal{L}_{2}(U)\) in (18) monotonically.

Proof: As Ding et al. [51] pointed out, for any nonnegative matrices \(A \in \mathbb{R}^{d \times d}, B \in \mathbb{R}^{c \times c}, C \in \mathbb{R}^{d \times c}\), and \(G \in \mathbb{R}^{d \times c}\), if \(A\) and \(B\) are symmetric, the following inequality holds:

\[
\text{Tr}(C^T A B) \leq \sum_{i=1}^{d} \sum_{k=1}^{c} \frac{(AGB)_{ik} C_{ik}^2}{G_{ik}}.
\]

Based on the above equation, the upper bound of the first term of \(f(U)\) is written as

\[
\text{Tr}(UV^T Q V U^T) = \text{Tr}(U^T UV^T Q V) \\
\leq \sum_{i=1}^{d} \sum_{k=1}^{c} (U^{(t)}_{ik} V^{(t)}_{kj})^2 U^{(t)}_{ik} - 2 \sum_{i=1}^{d} \sum_{k=1}^{c} (XQV)_{ik} U^{(t)}_{ik} \left(1 + \log \frac{U^{(t)}_{ik}}{U^{(t)}_{ij}}\right).
\]

The last term of \(\mathcal{L}_{2}(U)\) equals to zero, so we do not consider it. Based on the bounds of the first two terms, the auxiliary function of \(\mathcal{L}_{2}(U)\) is

\[
g(U, U^{(t)}) = \sum_{i=1}^{d} \sum_{k=1}^{c} (U^{(t)}_{ik} V^{(t)}_{kj})^2 U^{(t)}_{ik} - 2 \sum_{i=1}^{d} \sum_{k=1}^{c} (XQV)_{ik} U^{(t)}_{ik} \left(1 + \log \frac{U^{(t)}_{ik}}{U^{(t)}_{ij}}\right)
\]

which satisfies the conditions in (31).

The first-order derivative of \(g(U, U^{(t)})\) is

\[
\frac{\partial g(U, U^{(t)})}{\partial U_{ik}} = 2 (U^{(t)}_{ik} V^{(t)}_{kj} - U^{(t)}_{ik}) U^{(t)}_{ik} - 2 (XQV)_{ik} U^{(t)}_{ik} \left(1 + \log \frac{U^{(t)}_{ik}}{U^{(t)}_{ij}}\right)
\]

The Hessian matrix is

\[
\frac{\partial^2 g(U, U^{(t)})}{\partial U_{ik} \partial U_{ij}} = 2 \delta_{ij} \delta_{kl} \left(\frac{(U^{(t)}_{ik} V^{(t)}_{kj})^2 - 2 (XQV)_{ik} U^{(t)}_{ik}}{U^{(t)}_{ik}} + \frac{(XQV)_{ik} U^{(t)}_{ik}^2}{U^{(t)}_{ik}^2}\right)
\]

where \(\delta_{ij}\) is the defined as

\[
\delta_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise}. \end{cases}
\]

The Hessian matrix is a positive definite diagonal matrix, so \(g(U, U^{(t)})\) is convex on \(U\). The global optimal solution \(U^{(t+1)}\) to \(\min_{U} g(U, U^{(t)})\) is computed by setting the first-order derivative to zero:

\[
U_{ik}^{(t+1)} = \left(U_{ik}^{(t)} \sqrt{\frac{(XQV)_{ik}}{(U^{(t)}_{ik} V^{(t)} QV)_{ik}}} \right).
\]

According to Lemma 1, \(\mathcal{L}_{2}(U)\) is nonincreasing with the above updating rule.

Therefore, by searching the global optimal solutions to the subproblems iteratively, the local optimal solution to objective (11) can be obtained.

IV. GRAPH REGULARIZED EMMF

EMMF uses the global centroids to represent the samples, so it cannot handle the data with complex manifold structures. To explore the local data relationship, the G-EMMF is introduced. Since this research mainly focuses on robustness, we simply incorporate a graph regularization term into EMMF to improve the performance.

A. Methodology

Supposing that \(S \in \mathbb{R}^{n \times n}\) is the similarity graph of the data matrix \(X\), a large value of \(S_{ij}\) indicates the high similarity between \(x_i\) and \(x_j\). Intuitively, if \(x_i\) is similar to \(x_j\), their coefficient vectors should also be similar. Using the inner product to measure the distance between the vectors, the graph regularization term is given as

\[
\min_{V \geq 0} ||S - V V^T||_2^2.
\]

Ideally, we can obtain a block diagonal \(VV^T\). Kuang et al. [52] proved that the above term is equivalent to spectral clustering if \(S\) is doubly stochastic and \(V\) is orthogonal. Normalizing the graph \(S = D^{-1/2} S D^{-1/2}\), where \(D\) is the degree matrix of \(S\), problem (41) becomes

\[
\min_{V \geq 0, V V^T = I} ||S - V V^T||_2^2.
\]

As mentioned in Section II, the orthogonal constraint also facilitates the clustering interpretation. Combining (42) with the objective (11), the model of G-EMMF is

\[
\min_{U, V} - \sum_{i=1}^{n} \|m_i\|_2 \log \frac{\|m_i\|_2}{\|M\|_{1,1}} + \lambda ||S - V V^T||_2^2
\]

s.t. \(M = X - U V^T\), \(U \geq 0\), \(V \geq 0\), \(V^T V = I\)

where \(\lambda\) is the regularization parameter. With the above formulation, the coefficient matrix \(V\) preserves the local correlations between the samples.
B. Optimization

The objective of G-EMMF is equivalent to
\[
\min_{U,V} \text{Tr}((X - UV^T)(X - UV)) + \lambda \|S - VV^T\|_2^2
\]
\[
s.t. \quad U \geq 0, \quad V \geq 0, \quad V^TV = I.
\] (44)

Removing the irrelevant terms, the above problem is simplified into
\[
\min_{U,V} \text{Tr}([X - UV^T]Q[X - UV]) - 2\lambda \text{Tr}(V^T SV)
\]
\[
s.t. \quad U \geq 0, \quad V \geq 0, \quad V^TV = I.
\] (45)

The updating rule of \(U\) is the same as EMMF, so we only give the updating rule of \(V\).

Updating \(V\), problem (45) is transformed into
\[
\min_{V} \text{Tr}(V^T QVU^T U) - 2\lambda \text{Tr}(V^T Vq^T) + \text{Tr}(V^T SV)
\]
\[
s.t. \quad V \geq 0, \quad V^TV = I.
\] (46)

The Lagrangian function is
\[
\mathcal{L}_4(V) = \text{Tr}(V^T QVU^T U) - 2\lambda \text{Tr}(V^T Vq^T) + \text{Tr}(V^T SV) + \text{Tr}[\Lambda_5(V^T V - I)^T V - \Lambda_5]
\] (47)

where \(\Lambda_4 \in \mathbb{R}^{n \times c}\) and \(\Lambda_5 \in \mathbb{R}^{n \times c}\) are Lagrangian multipliers. Given \((\Lambda_4)_{ik} = 0\), setting \((\partial \mathcal{L}_4(V)) / \partial V) = 0\) gives rise to
\[
(QVU^T U - QX^T U - 2\lambda SV + V\Lambda_5)_{ik} V_{ik} = 0
\] (48)
so the updating rule is
\[
V_{ik} \leftarrow V_{ik} \sqrt{(QX^T U + 2\lambda SV + V\Lambda_5^-)_{ik}}
\] (49)

where \(\Lambda_5^-\) and \(\Lambda_5^+\) are the negative and positive parts of \(\Lambda_5\), i.e., \((\Lambda_5^-)_{ik} = (\{(\Lambda_5)_{ik} - (\Lambda_5)_{ik})/2\) and \((\Lambda_5^+)_{ik} = (\{(\Lambda_5)_{ik} + (\Lambda_5)_{ik})/2\). Since we also have
\[
(QVU^T U - QX^T U - 2\lambda SV + V\Lambda_5)_{ik} V_{ik} = 0
\] (50)
\(\Lambda_5\) is computed as
\[
\Lambda_5 = V^T QX^T U + 2\lambda SV - V^T QVU^T U
\] (51)\) and \(\Lambda_5^-\) and \(\Lambda_5^+\) are computed as
\[
\Lambda_5^- = V^T QVU^T U
\]
\[
\Lambda_5^+ = V^T QX^T U + 2\lambda V^T SV.
\] (52)

Therefore, the updating rule in (49) is rewritten as
\[
V_{ik} \leftarrow V_{ik} \sqrt{(QX^T U + 2\lambda SV + VV^T QVU^T U)_{ik}}
\] (53)

The optimization strategy for G-EMMF is described in Algorithm 2. G-EMMF takes \(O(n^2d)\) additional complexity to construct and normalize the similarity graph, and the remaining costs are the same with EMMF, i.e., \(O(tndc)\). After \(t\) iterations, the overall computational cost of G-EMMF is \(O(n^2d + tndc)\).

Algorithm 2 Optimization Algorithm of G-EMMF
\[
\text{Input}: \text{Data matrix } X, \text{ centroid number } c, \text{ graph } S, \text{ parameter } \lambda.
\]
1: Initialize \(U\) and \(V\).
2: Normalize \(S\) as \(D^{-1/2}SD^{-1/2}\).
3: repeat
4: \quad Compute \(Q\) with Eq. (14).
5: \quad Update \(U\) with Eq. (21).
6: \quad Update \(V\) with Eq. (53).
7: until Converge
\[
\text{Output}: \text{Optimal } U, V.
\]

C. Convergence

Here, we demonstrate the convergence of problem (46). We first introduce the following theorem.

Theorem 3: Updating rule (49) decreases the Lagrangian function \(L_4(V)\) in (47) monotonically.

Proof: According to the proof of Theorem 2, denoting the value of \(V\) at the \(r\)th iteration as \(V^{(r)}\), the bounds of the nonzero terms in \(L_4(V)\) are given as follows:
\[
\text{Tr}(V^T QVU^T U) \leq \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(QVU^T U)_{ik} V_{ik}^2}{V_{ik}^{(r)} V_{ik}^{(r)}}
\]
\[
\text{Tr}(V^T QX^T U) \geq \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(QX^T U)_{ik} V_{ik}^{(r)} (1 + \log \frac{V_{ik}}{V_{ik}^{(r)}})}{V_{ik}^{(r)} V_{ik}^{(r)}}
\]
\[
V^T SV \geq \sum_{i=1}^{n} \sum_{k=1}^{c} \sum_{l=1}^{c} S_{il} V_{ik}^{(r)} V_{il}^{(r)} (1 + \log \frac{V_{ik} V_{il}}{V_{ik}^{(r)} V_{il}^{(r)}})
\]
\[
\text{Tr}(\Lambda_5 V^T V) \leq \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(V^T \Lambda_5^+)_{ik} V_{ik}^2}{V_{ik}^{(r)} V_{ik}^{(r)}}.
\] (54)

Combining the bounds, the auxiliary function for \(L_4(V)\) is
\[
g(V, V^{(r)})
= \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(QVU^T U)_{ik} V_{ik}^2}{V_{ik}^{(r)} V_{ik}^{(r)}}
\]
\[
-2 \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(QX^T U)_{ik} V_{ik}^{(r)} (1 + \log \frac{V_{ik}}{V_{ik}^{(r)}})}{V_{ik}^{(r)} V_{ik}^{(r)}}
\]
\[
-2\lambda \sum_{i=1}^{n} \sum_{k=1}^{c} \sum_{l=1}^{c} S_{il} V_{ik}^{(r)} V_{il}^{(r)} (1 + \log \frac{V_{ik} V_{il}}{V_{ik}^{(r)} V_{il}^{(r)}})
\]
\[
- \sum_{i=1}^{n} \sum_{k=1}^{c} \sum_{l=1}^{c} (\Lambda_5^-)_{kl} V_{ik}^{(r)} V_{il}^{(r)} (1 + \log \frac{V_{ik} V_{il}}{V_{ik}^{(r)} V_{il}^{(r)}})
\]
\[
+ \sum_{i=1}^{n} \sum_{k=1}^{c} \frac{(V^T \Lambda_5^+)_{ik} V_{ik}^2}{V_{ik}^{(r)} V_{ik}^{(r)}}.
\] (55)

The Hessian matrix is positive definite, so the optimal solution \(V^{(r+1)}\) that minimizes \(g(V, V^{(r)})\) is calculated by setting the
first-order derivative to zero

\[ V_{jk}^{(t+1)} = V_{jk}^{(t)} \sqrt{\frac{(QXU + 2SV + V^T A_s)_{jk}}{(QV^TU + V^TA_s)_{jk}}} \]  \hspace{1cm} (56)

According to Lemma 1, the above updating rule decreases \( \mathcal{L}_4(V) \).

V. EVALUATION OF EMMF

In this section, the proposed EMMF is evaluated on several synthetic and real-world datasets.

A. Experiments on Synthetic Datasets

Synthetic datasets are constructed to validate the robustness of EMMF. Some static properties of EMMF are also discussed.

As shown in Fig. 1(a), the first dataset consists of 13 2-D samples. The first ten samples are with normal distribution, and the last three samples are outliers. Fig. 1(a) also visualizes the approximation results of NMF [2], \( \ell_{2,1}-\text{NMF} \) [43], and EMMF. The approximated samples of NMF deviate from the normal distribution largely, which indicates the approximation is dominated by the outliers. As shown in Fig. 1(b), the errors of some normal samples are larger than the outliers. \( \ell_{2,1}-\text{NMF} \) shows better performance because the outliers are depressed. However, the outliers still affect the approximation. EMMF just let the outliers be with large errors, as shown in Fig. 1(d), such that the normal samples are approximated correctly. As shown in Fig. 1(f), it is obvious that the weights, i.e., \( Q_{ii} \), of outliers are small. Therefore, EMMF shows more robustness.

Fig. 2 shows the approximation results for the data without outliers. It can be seen that all the methods approximate the samples successfully. As visualized in Fig. 2(c) and (d), both \( \ell_{2,1}-\text{NMF} \) and EMMF cannot achieve the zero error due to the factor \( \varepsilon \) added on \( Q \). In Fig. 2(d), some samples are with relatively large errors because EMMF treats them as outliers. However, the errors of most samples are close to zero, so the overall error is still less than \( \ell_{2,1}-\text{NMF} \). That is to say, a few mistaken outliers cannot affect the approximation largely, and their weights are close to those of normal ones. Therefore, EMMF works well even when there is no outlier.

We also demonstrate that the objective of EMMF is unlikely to be dominant by outliers. Supposing that \( x_i \) is the outlier, its effects on the objectives of NMF, \( \ell_{2,1}-\text{NMF} \), and EMMF are computed as

\[ \phi_{\text{NMF}}(x_i) = \frac{\| m_i \|_2^2}{\sum_{i=1}^{n} \| m_i \|_2^2} \]
Taking the step length of $\mu$ bound of curve is plotted in Fig. 4. We can see that the upper maximum value when varying $p$.

Given a certain randomly generated data $X$ exceeding a certain value.

As shown in Fig. 3, the outlier effect on NMF increases dramatically. Compared with NMF, $\mu$ increases slower. However, both of them reach to 1 when $\sigma$ is very large. Meanwhile, $\phi_{EMMF}(x_i)$ decreases when $\sigma$ exceeds a certain value. Therefore, EMMF is insensitive to the outliers with extremely large errors.

Suppose that there is only one outlier $x_i$, and its residue ratio is $p$, i.e., $(\|m_i\|_2/\|M\|_2) = p$. The entropy of the distribution is minimized when all the remaining samples are with the same residue ratio, i.e., $((1 - p)/(n - 1))$. In such situation, $\phi_{EMMF}(x_i)$ is

$$\phi_{EMMF}(x_i) = \frac{p \log(p)}{p \log(p) + (1 - p)\log(1 - p) - \log(n - 1)}.$$  \hspace{0.5cm} (58)

Given a certain $n$, the upper bound is calculated as the maximum value when varying $p$ within the range $(0, 1]$. Taking the step length of $p$ as 0.01 and increasing $n$, the upper bound curve is plotted in Fig. 4. We can see that the upper bound of $\phi_{EMMF}(x_i)$ decreases monotonically with the value of $n$, which complies with the human perception that the effect of an outlier should be small when there are many normal samples.

**B. Experiments on Real-World Datasets**

The clustering performance of EMMF is evaluated on real-world datasets. Clustering accuracy (ACC) and normalized mutual information (NMI) are used as measurements.

1) **Datasets:** Eight benchmarks for clustering are employed, including two object image datasets, i.e., COIL20 and COIL100 [14], three face image datasets, i.e., YALE [53], JAFFE [54], and UMIST [55], a spoken letter recognition dataset, i.e., ISOLET [56], a biological dataset, i.e., GLIOMA [57], and a hand movement dataset, i.e., Movement [58]. The samples are normalized as the unit vectors in the experiments. The details of the datasets are exhibited in Table I.

2) **Competitors:** Seven state-of-the-arts are taken for comparison, including the following.

- 1) NMF [2]: NMF with the Frobenius norm formulation.
- 2) KLNMF [2]: NMF with KL divergence formulation.
- 3) $\ell_{2,1}$-NMF [43]: NMF with the $\ell_{2,1}$-norm loss.
- 4) HxNMF [59]: NMF with the logarithmic loss.
- 5) ISNMF [60]: Complex NMF model with IS divergence.
- 6) $\beta$NMF [61]: Complex NMF model with $\beta$ divergence.
- 7) tNMF [31]: NMF model based on student’s $t$ distribution.
- 8) CNMF [29]: NMF model based on Cauchy distribution.
- 9) TCNMF [34]: NMF with a truncated Cauchy loss.

Complex ISNMF and complex $\beta$NMF are designed to handle complex-valued data; we employ them instead of the real-valued versions because of their good implementations.

**TABLE I**

| Description on the Real-World Datasets |
|----------------------------------------|
| **COIL20** | **COIL100** | **YALE** | **JAFFE** | **UMIST** | **ISOLET** | **GLIOMA** | **Movement** |
| Samples | 1440 | 7200 | 165 | 213 | 575 | 1560 | 50 | 360 |
| Dimension | 1024 | 1024 | 256 | 676 | 644 | 617 | 4434 | 90 |
| Class | 20 | 100 | 15 | 10 | 20 | 26 | 4 | 15 |
TABLE II

| Method | COIL20 | COIL100 | YALE | JAFFE | UMIST | ISOLET | GIOMA | Movement |
|--------|--------|---------|------|-------|-------|--------|-------|----------|
| NMF    | 44.86 ± 1.73 | 25.32 ± 1.22 | 34.12 ± 3.93 | 63.90 ± 5.80 | 34.03 ± 2.41 | 38.91 ± 2.90 | 44.80 ± 2.86 | 32.97 ± 2.53 |
| KL-NMF | 50.97 ± 2.56 | 39.54 ± 1.07 | 41.88 ± 2.22 | 91.36 ± 2.66 | 38.31 ± 1.86 | 58.08 ± 2.61 | 43.40 ± 0.92 | 43.72 ± 0.99 |
| ℓ₂,₁-NMF | 58.26 ± 1.91 | 42.56 ± 1.34 | 41.82 ± 2.57 | 93.29 ± 2.50 | 40.26 ± 1.90 | 64.42 ± 2.55 | 46.00 ± 0.00 | 44.53 ± 0.96 |
| HxNMF  | 58.08 ± 1.66 | 43.74 ± 1.31 | 40.97 ± 2.80 | 93.57 ± 2.50 | 40.90 ± 1.91 | 64.49 ± 2.55 | 46.00 ± 0.00 | 44.75 ± 0.98 |
| ISNMF  | 54.93 ± 3.24 | 38.79 ± 1.17 | 39.64 ± 2.20 | 85.92 ± 4.12 | 37.39 ± 1.82 | 54.10 ± 2.72 | 44.80 ± 2.40 | 43.61 ± 1.69 |
| βNMF   | 56.18 ± 1.98 | 36.35 ± 0.92 | 37.82 ± 1.83 | 91.08 ± 3.13 | 36.87 ± 1.64 | 47.37 ± 2.52 | 46.60 ± 0.92 | 41.11 ± 1.35 |
| rNMF   | 47.50 ± 3.02 | 41.90 ± 0.95 | 40.61 ± 2.34 | 69.01 ± 3.22 | 39.13 ± 1.69 | 38.47 ± 2.95 | 44.40 ± 3.67 | 44.72 ± 1.84 |
| CNMF   | 50.98 ± 1.68 | 40.05 ± 1.21 | 31.21 ± 2.69 | 90.09 ± 1.10 | 36.89 ± 2.43 | 39.79 ± 4.11 | 45.80 ± 4.85 | 42.69 ± 1.75 |
| TCNMF  | 58.68 ± 2.39 | 44.46 ± 1.31 | 42.42 ± 1.80 | 91.24 ± 5.87 | 40.52 ± 1.82 | 59.92 ± 1.74 | 44.25 ± 5.52 | 43.22 ± 2.24 |
| EMMF   | 59.72 ± 2.21 | 45.42 ± 1.36 | 43.27 ± 2.44 | 94.23 ± 2.43 | 41.32 ± 1.78 | 65.06 ± 2.52 | 46.80 ± 0.98 | 45.42 ± 1.66 |

Fig. 5. Convergence curves of EMMF. (a) COIL20. (b) COIL100. (c) YALE. (d) JAFFE. (e) UMIST. (f) ISOLET. (g) GIOMA. (h) Movement.

For ISNMF, βNMF, rNMF, CNMF, and TCNMF, the hyperparameters are tuned within the ranges suggested by the authors. For all the methods, we initialize U and V with k-means [62] and set the stop criteria as 500 iterations. Given a data matrix, where the centroid number c is given, matrix factorization is performed to get the coefficient matrix V. After that, clustering results are obtained by finding the largest element in each row of V. To alleviate the influence of the initiation condition, all the methods are performed for 20 repetitions. Averaged results and standard deviations are reported.

3) Performance: The clustering results on different datasets are exhibited in Table II. The proposed EMMF shows the best performance in terms of ACC and NMI. On these datasets, most samples obey the normal distribution. ℓ₂,₁-NMF, HxNMF, rNMF, and CNMF show better results than NMF, which demonstrates the advantage of improving robustness. By filtering the large outliers, TCNMF performs better than CNMF on some occasions, but it is difficult to determine the threshold. EMMF performs well on all the datasets because it moves the centroids toward the samples with small errors.
In this way, most of the samples are well represented by the centroids, and the clustering performance is improved. EMMF learns an imbalanced residue distribution while minimizing the overall error. In this way, it does not enlarge the errors of the outliers but reduces those of the remaining ones to a larger extent. Therefore, it works well on the datasets without outliers, as can be seen from the results on GLIOMA, where the samples are high dimensional and scarce. The convergence curves of EMMF are also given in Fig. 5. The optimization algorithm converges within 40 iterations on all the datasets, which ensures efficiency.

To further prove the robustness of EMMF, we introduce large outliers into real-world datasets. For each dataset, randomly generated noise is added into elements of the data matrix. The noise values are within the range of $[0, 0.5 \times \sigma]$, where $\sigma$ is the largest value in the original data matrix. We perform all the methods on the outlier datasets with different numbers of noise values ([$0, (n/3)$]) and show the clustering results of the original samples in Fig. 6. As the outlier number increases, the curves of NMF, KLNMF, ISNMF, and $\beta$NMF drop dramatically since they do not consider the robustness problem. The performance of TCNMF decreases because the threshold becomes unreliable. $\ell_2$NMF and Hx-NMF show similar robustness by depressing the outliers. The ACC curves of EMMF are more stable than competitors on all the datasets because the outliers have fewer effects on the updating of centroids. Therefore, EMMF is able to handle the data with large outliers.
The effectiveness of G-EMMF is validated through experiments. The datasets used in this part are the same as in Section V-B.

A. Competitors

Seven graph-regularized NMF methods are used for comparison, including the following.

1) RMNMF [9]: Robust manifold NMF, which integrates the spectral clustering term into the objective of \( \ell_{2,1} \)-NMF.

2) NLCF [63]: Nonnegative local coordinate factorization, which uses local coordinate learning to encode the data structure.

3) LCF [64]: Local coordinate concept factorization, which combines local coordinate learning and convex matrix factorization.

4) LSNMF [49]: Local-centroids structured NMF, which employs multiple local centroids to represent a cluster.

5) SRMCF [65]: Self-representative manifold concept factorization, which optimizes the data graph during performing convex matrix factorization.

6) NMFAN [66]: NMF with adaptive neighbors, which learns the local data relationship adaptively.

7) KMM [67]: k multiple means, which finds multiple local centroids within each class.

The best parameters of the competitors are found by searching the grid \( \{10^{-5}, 10^{-4}, \ldots, 10^5\} \). The graph regularization parameter \( \lambda \) in G-EMMF is found within the grid \( \{10^1, 10^3, \ldots, 10^9\} \). The methods are initialized with k-means, and the 0-1 weighting five-nearest-neighbor graph is constructed as the similarity graph. The stop criteria for all the methods are set as 500 maximum iterations. After repeating the methods 20 times, the average ACC, NMI, and standard deviations are reported.

B. Performance

The results of the graph regularized NMF methods are given in Table III. RMNMF fixes the data graph during optimization. The outliers may affect the graph quality and further influence the clustering results. Compared with RMNMF, the other competitors show better performance because they learn the data relationship iteratively such that the graph quality is improved. Similar to RMNMF, the proposed G-EMMF also relies on the input graph. It still achieves the best performance because the graph noise is depressed. In addition, compared with the results of EMMF in Table II, G-EMMF improves the clustering performance stably, which demonstrates the necessity of exploiting data structure.

The convergence curves are shown in Fig. 7, which verifies the effectiveness of the optimization algorithm. Besides, to investigate the effect of parameter \( \lambda \) on the clustering results, we plot the curves of ACC and NMI with varying \( \lambda \), as shown in Fig. 8. The clustering performance is not very
TABLE III
PERFORMANCE (MEAN ± STD %) OF GRAPH REGULARIZED NMF METHODS ON REAL-WORLD DATASETS. BEST RESULTS ARE IN BOLDFACE

| Method | COIL20 | COIL100 | YALE | JAFPE | UMIST | ISOLET | GLOIMA | Movement |
|--------|--------|---------|------|-------|-------|--------|--------|----------|
| RMNMF  | 57.92 ± 1.90 | 41.19 ± 2.69 | 44.48 ± 3.14 | 91.08 ± 4.16 | 41.91 ± 2.23 | 55.90 ± 2.83 | 43.60 ± 3.35 | 41.67 ± 1.83 |
| NLCF   | 64.31 ± 1.12 | 41.53 ± 1.10 | 43.52 ± 2.80 | 93.15 ± 2.68 | 39.10 ± 1.34 | 61.67 ± 1.52 | 45.80 ± 1.26 | 44.44 ± 0.81 |
| LCF    | 65.83 ± 2.22 | 45.96 ± 1.27 | 41.38 ± 3.16 | 92.68 ± 3.42 | 39.20 ± 2.15 | 61.99 ± 1.57 | 45.80 ± 1.26 | 46.67 ± 1.46 |
| LSNMF  | 65.28 ± 2.02 | 46.51 ± 2.13 | 24.85 ± 6.21 | 75.02 ± 6.99 | 39.86 ± 5.42 | 43.14 ± 2.29 | 37.40 ± 1.20 | 32.78 ± 3.31 |
| SRMCF  | 66.60 ± 1.57 | 39.90 ± 1.25 | 41.21 ± 2.17 | 96.15 ± 3.52 | 42.54 ± 1.47 | 63.72 ± 1.01 | 44.00 ± 0.00 | 45.83 ± 1.44 |
| NMFAN  | 65.00 ± 1.37 | 45.58 ± 0.79 | 42.30 ± 3.20 | 93.05 ± 3.22 | 39.69 ± 1.91 | 62.44 ± 1.68 | 45.40 ± 0.92 | 44.17 ± 1.07 |
| KMM    | 59.38 ± 2.46 | 40.50 ± 1.19 | 38.06 ± 3.01 | 87.89 ± 5.53 | 42.26 ± 1.92 | 57.14 ± 1.48 | 42.20 ± 2.57 | 43.33 ± 1.88 |
| G-EMMF | 68.82 ± 1.27 | 48.03 ± 1.45 | 45.58 ± 2.06 | 98.12 ± 2.14 | 45.88 ± 2.42 | 64.31 ± 1.59 | 46.00 ± 0.00 | 47.78 ± 1.67 |

TABLE IV
PERFORMANCE OF GRAPH REGULARIZED NMF METHODS ON YALE DATASETS WITH OUTLIERS. BEST RESULTS ARE IN BOLDFACE

| Method | ACC | NMI |
|--------|-----|-----|
| RMNMF  | 0.69 ± 0.68 | 66.05 ± 0.99 | 49.09 ± 2.21 | 88.94 ± 1.84 | 59.38 ± 1.01 | 68.63 ± 1.24 | 20.10 ± 4.09 | 52.17 ± 1.41 |
| NLCF   | 0.73 ± 0.66 | 66.16 ± 0.54 | 48.67 ± 2.40 | 91.97 ± 1.57 | 58.83 ± 0.76 | 76.21 ± 0.72 | 21.49 ± 2.96 | 61.58 ± 0.49 |
| LCF    | 0.74 ± 0.98 | 72.13 ± 0.30 | 48.09 ± 1.51 | 90.37 ± 1.94 | 59.01 ± 1.05 | 75.98 ± 0.96 | 20.08 ± 2.19 | 62.37 ± 0.51 |
| LSNMF  | 0.76 ± 0.65 | 63.33 ± 1.32 | 25.10 ± 6.91 | 84.04 ± 3.78 | 53.10 ± 5.47 | 62.24 ± 1.52 | 9.92 ± 2.27 | 43.93 ± 4.28 |
| SRMCF  | 0.76 ± 0.91 | 68.86 ± 0.38 | 47.45 ± 1.12 | 95.56 ± 2.51 | 62.92 ± 0.73 | 77.63 ± 0.44 | 17.63 ± 0.00 | 59.90 ± 0.28 |
| NMFAN  | 0.74 ± 0.76 | 73.30 ± 0.30 | 46.82 ± 2.31 | 92.08 ± 2.06 | 59.63 ± 1.09 | 78.11 ± 0.81 | 19.23 ± 1.05 | 61.67 ± 0.65 |
| KMM    | 0.75 ± 0.91 | 67.04 ± 1.53 | 43.70 ± 2.90 | 90.83 ± 4.08 | 61.11 ± 2.30 | 75.36 ± 0.99 | 19.05 ± 3.27 | 55.48 ± 2.01 |
| G-EMMF | 0.80 ± 0.55 | 75.47 ± 0.57 | 52.03 ± 1.12 | 97.31 ± 1.37 | 66.96 ± 0.57 | 78.92 ± 0.74 | 21.79 ± 0.00 | 63.62 ± 0.76 |

sensitive to $\lambda$ within a wide range. When $\lambda$ becomes very large, the performance tends to decrease because the matrix factorization residue increases.

Here, we evaluate the robustness. Given the faces images from YALE, we randomly select three images from each class and add block noise on them, so these images become outliers, as shown in Fig. 9. The clustering results are given in Table IV. We can see RMNMF outperforms several competitors due to the robust formulation. LSNMF and KMM show unsatisfying results since they aim to learn a graph with exact connected components, which is unrealistic on the data with outliers. G-EMMF performs well on different scales of block noise.

VII. CONCLUSION AND FUTURE WORK

This article proposes an EMMF framework. A novel matrix factorization formulation with entropy loss is designed. Instead of approximating all the samples, the proposed model pursues an imbalanced residue distribution, and the outliers with relatively large errors are not taken into consideration. In this way, outliers have fewer effects on the learned centroids. In addition, a graph regularized EMMF is introduced to deal with complex data structures. The models can be solved by the suggested optimization algorithms efficiently. Experiments on various datasets demonstrate the robustness of the proposed methods and show their applicability on data without outliers. The comparison with the state-of-the-arts validates the superiorities of our methods.

In future work, we are desired to develop the deep model of EMMF and apply it in large-scale clustering tasks. We also want to incorporate the graph information into EMMF without introducing any additional parameters.

REFERENCES

[1] P. Paatero and U. Tapper, “Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values,” Environmetrics, vol. 5, no. 2, pp. 111–126, 1994.
[2] D. Lee and H. Seung, “Algorithms for nonnegative matrix factorization,” in Proc. Adv. Neural Inf. Process. Syst., 2000, vol. 13, no. 6, pp. 556–562.
[3] C. Li, H.-B. Xie, X. Fan, R. Y. D. Xu, S. Van Huffel, and K. Mengersen, “Kernelized sparse Bayesian matrix factorization,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 1, pp. 391–404, Jan. 2021.
[4] 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.
S. Wei, J. Wang, G. Yu, C. Domeniconi, and X. Zhang, “Multi-view L

M. Chen and X. Li, “Concept factorization with local centroids,” in Proc. SIAM Int. Conf. Data Mining, Apr. 2005, pp. 606–610.

J. Huang, F. Nie, H. Huang, and C. Ding, “Robust manifold nonnegative matrix factorization,” ACM Trans. Knowl. Discovery Data, vol. 8, no. 3, p. 11, 2014.

X. Li, M. Chen, and Q. Wang, “Discrimination-aware projected matrix factorization,” IEEE Trans. Knowl. Data Eng., vol. 32, no. 4, pp. 809–814, Apr. 2020.

D. Cai, X. He, and J. Han, “Locally consistent concept factorization for document clustering,” in IEEE Trans. Knowl. Data Eng., vol. 23, no. 6, pp. 902–913, Jun. 2011.

W. Xu and Y. Gong, “Document clustering by concept factorization,” in Proc. 27th Annu. Int. Conf. Res. Develop. Inf. Retr. (SIGIR), 2004, pp. 202–209.

M. Chen and X. Li, “Concept factorization with local centroids,” IEEE Trans. Neural. Netw. Learn. Syst., vol. 32, no. 11, pp. 5247–5253, Nov. 2021, doi: 10.1109/TNNLS.2020.3027068.

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.

L. Zhang, Q. Zhang, B. Du, J. You, and D. Tao, “Adaptive manifold regularized matrix factorization for data clustering,” in Proc. Int. Joint Conf. Artif. Intell., 2017, pp. 3399–3405.

M. Chen, Q. Wang, and X. Li, “Adaptive projected matrix factorization method for data clustering,” Neurocomputing, vol. 306, pp. 182–188, Sep. 2018.

M. Chen and X. Li, “Robust matrix factorization with spectral embedding,” IEEE Trans. Neural. Netw. Learn. Syst., vol. 32, no. 12, pp. 5698–5707, Dec. 2021, doi: 10.1109/TNNLS.2020.3027351.

X. He, J. Tang, X. Du, R. Hong, T. Ren, and T. Chua, “Fast matrix factorization with nonuniform weights on missing data,” IEEE Trans. Neural. Netw. Learn. Syst., vol. 31, no. 8, pp. 2791–2804, Aug. 2020.

N. Kallus, X. Mao, and M. Udell, “Causal inference with noisy and missing covariates via matrix factorization,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 6921–6932.

Z. Lin, C. Xu, and H. Zha, “Robust matrix factorization by majorization minimization,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 1, pp. 208–220, Jan. 2017.

S. Ohsawa, Y. Obara, and T. Osogami, “Gated probabilistic matrix factorization: Learning users’ attention from missing values,” in Proc. Int. Joint Conf. Artif. Intell., 2016, pp. 1888–1894.

C. Févotte and J. Idier, “Algorithms for nonnegative matrix factorization with the β-divergence,” Neurocomput., vol. 239, no. 9, pp. 2421–2456, Sep. 2017.

C. Févotte, N. Bertin, and J.-L. Durrieu, “Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis,” in Neurocomput., vol. 21, no. 3, pp. 793–830, 2009.

Z. Zhang, Q. Wang, and Y. Yuan, “Hyperspectral unmixing VIA L1/4 sparsity-constrained multilayer NMF,” in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS), Jul. 2019, pp. 2143–2146.

S. Xu, C. Zhang, and J. Zhang, “Bayesian deep matrix factorization network for multiple images denoising,” Neurocomput., vol. 123, pp. 420–428, Mar. 2020.

S. Wei, J. Wang, G. Yu, C. Domeniconi, and X. Zhang, “Multi-view multiple clusterings using deep matrix factorization,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 6348–6355.

Q. Ke and T. Kanade, “Robust L1 norm factorization in the presence of outliers and missing data by alternative convex programming,” in IEEE Trans. Comput. Soc. Comput. Conf. Vis. Pattern Recognit. (CVPR), Jun. 2005, pp. 739–746.

U. Simsekli, A. Liutkus, and A. T. Cemgil, “Alpha-stable matrix factorization,” IEEE Signal Process. Lett., vol. 22, no. 12, pp. 2289–2293, Dec. 2015.
[55] D. Graham and N. Allinson, “Characterising virtual eigensignatures for general purpose face recognition,” *Face Recognit. Theory Appl.*, vol. 163, no. 2, pp. 446–456, 1998.

[56] D. Cai, X. He, and J. Han, “Speed up kernel discriminant analysis,” *VLDB J.*, vol. 20, no. 1, pp. 21–33, 2011.

[57] J. Li et al., “Feature selection: A data perspective,” *ACM Comput. Surv.*, vol. 50, no. 6, p. 94, 2016.

[58] D. Dua and C. Graff. (2017). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[59] Q. Wang, X. He, X. Jiang, and X. Li, “Robust bi-stochastic graph regularized matrix factorization for data clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 390–403, Jan. 2022, doi: 10.1109/TPAMI.2020.3007673.

[60] P. Magron and T. Virtanen, “Complex ISNMF: A phase-aware model for monaural audio source separation,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 1, pp. 20–31, Jan. 2019.

[61] P. Magron and T. Virtanen, “Towards complex nonnegative matrix factorization with the beta-divergence,” in *Proc. 16th Int. Workshop Acoustic Signal Enhancement (IWAENC)*, Sep. 2018, pp. 156–160.

[62] J. Macqueen, “Some methods for classification and analysis of multivariate observations,” in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, 1967, pp. 281–297.

[63] Y. Chen, J. Zhang, D. Cai, W. Liu, and X. He, “Nonnegative local coordinate factorization for image representation,” *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 969–979, Mar. 2013.

[64] H. Liu, Z. Yang, J. Yang, Z. Wu, and X. Li, “Local coordinate concept factorization for image representation,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1071–1082, Jun. 2014.

[65] S. Ma, L. Zhang, W. Hu, Y. Zhang, J. Wu, and X. Li, “Self-representative manifold concept factorization with adaptive neighbors for clustering,” in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 2539–2545.

[66] S. Huang, Z. Xu, and F. Wang, “Nonnegative matrix factorization with adaptive neighbors,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 486–493.

[67] F. Nie, C. Wang, and X. Li, “K-multiple-means: A multiple-means clustering method with specified K clusters,” in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 959–967.

**Mulin Chen** received the B.E. degree in software engineering and the Ph.D. degree in computer application technology from Northwestern Polytechnical University, Xi’an, China, in 2014 and 2019, respectively.

He is currently a Researcher with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University.

**Xuelong Li** (Fellow, IEEE) is currently a Full Professor with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi’an, China.