Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(7): 12486–12509.
DOI: 10.3934/mbe.2023556
Received: 13 February 2023
Revised: 24 April 2023
Accepted: 15 May 2023
Published: 24 May 2023

Research article

Robust capped norm dual hyper-graph regularized non-negative matrix tri-factorization

Jiyang Yu[1], Baicheng Pan[2], Shanshan Yu[3,*] and Man-Fai Leung[4]

[1] College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China

[2] Chongqing Key Laboratory of Nonlinear Circuits and Intelligent Information Processing, College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China

[3] Training and Basic Education Management Office,Southwest University, Chongqing 400715, China

[4] School of Computing and Information Science, Faculty of Science and Engineering, Anglia Ruskin University, Cambridge, United Kingdom

* **Correspondence:** Email: yu33@swu.edu.cn.

Abstract: Non-negative matrix factorization (NMF) has been widely used in machine learning and data mining fields. As an extension of NMF, non-negative matrix tri-factorization (NMTF) provides more degrees of freedom than NMF. However, standard NMTF algorithm utilizes Frobenius norm to calculate residual error, which can be dramatically affected by noise and outliers. Moreover, the hidden geometric information in feature manifold and sample manifold is rarely learned. Hence, a novel robust capped norm dual hyper-graph regularized non-negative matrix tri-factorization (RCHNMTF) is proposed. First, a robust capped norm is adopted to handle extreme outliers. Second, dual hyper-graph regularization is considered to exploit intrinsic geometric information in feature manifold and sample manifold. Third, orthogonality constraints are added to learn unique data presentation and improve clustering performance. The experiments on seven datasets testify the robustness and superiority of RCHNMTF.

Keywords: non-negative matrix tri-factorization; capped norm; dual hyper-graph regularization; robust clustering

1. Introduction

As the dimension of data in machine learning and data mining is too high to learn, matrix factorization algorithms are adopted to deconstruct the high-dimensional data into several low-dimensional data and explore the hidden structure of low dimensional data. The widely used

matrix factorization approaches, include Principal Component Analysis (PCA) [1], Singular Value Decomposition (SVD) [2], Vector Quantization (VQ) [3] and Non-negative matrix factorization (NMF).

Different from PCA, SVD and VQ, the main motivations of NMF, are to decompose a high-dimensional data matrix into two low-rank non-negative matrices, whose product is approximate to the original data matrix. NMF is able to obtain a parts-based representation of data as the non-negative constraints allow only additive, not subtractive, combinations [4]. NMF algorithm and its extensions have been applied in several areas, e.g., medical research [5], community discovery [6], gene expression [7–10], text mining [11] and neural network [12–15].

As an extension of NMF, Non-negative matrix tri-factorization (NMTF) aims to decompose a high-dimensional data matrix into three low-rank non-negative matrices [16, 17]. With an extra decomposition matrix, NMTF can obtain higher degrees of freedom than NMF [18]. NMTF is helpful for co-clustering task as it can categorize feature space and sample space simultaneously. Nevertheless, Ding et al. [18] point out that unconstrained NMTF is equivalent to NMF while constrained NMTF brings new features to NMF and hereby propose orthogonal non-negative matrix tri-factorization (ONMTF). ONMTF can obtain a rigorous clustering presentation as non-negative and orthogonality constraints lead to a sparse solution. Abundant research show the superiority of spareness research [14, 19–27].

The standard NMF and NMTF algorithms rarely consider the hidden geometrical information in feature space and sample space. However, it is pointed out that the observed data lie on a nonlinear low dimensional manifold embedded in a high dimensional ambient space [28, 29]. Several manifold learning algorithms have been proposed to exploit intrinsic geometrical information [30–32], such as ISOMAP [33] and Laplacian Eigenmap (LE) [34]. Inspired by recent progress in manifold learning, Cai et al. propose graph regularized NMF (GNMF) [35]. GNMF firstly constructs a nearest neighbor graph to encode the geometrical information of the data space and incorporates the graph regularization into its objective function. To explore the geometrical information in feature space, which is neglected in GNMF, Shang et al. further propose graph dual regularized NMF (DNMF) [17]. DNMF constructs dual graph to discover the manifold embedded in sample space and feature space simultaneously. Unfortunately, the high-order relations among samples are seldom considered in graph-based learning methods as the constructed graph only considers the pairwise relationship between two samples or two features. This problem has been solved by hyper-graph learning [36, 37]. Contrast to graph, hyper-graph is constructed by edges connected with multiple samples. Therefore, hyper-graph can explore high-order relationship of data and features. Hyper-graph is popular in machine learning fields [38,39]. Feng et al. propose Hyper-graph Neural Networks (HNN) [40]. Jiang et al. propose dynamic hyper-graph neural networks [41]. HNN utilizes hyper-edge convolution operations to learn the hidden layer representation considering the high-order data structure [40, 42]. Zeng et al. first combine hyper-graph learning and NMF algorithm and propose hyper-graph regularized NMF (HNMF) [43]. HNMF utilizes single hyper-graph regularization to obtain a better geometrical information in sample space.

Although standard NMF and NMTF algorithms utilize Forbenius norm to calculate residual error and perform well on untainted data, the performance can be dramatically decreased by noise and outliers as Forbenius norm calculates the squared residual error. To alleviate the impact of noise and outliers, several robust M-estimator based NMF algorithms are proposed [44]. Kong et al. propose a robust NMF using $l_{2,1}$-norm to calculate the residual error of each sample point without squaring

it [45]. Gao et al. design a robust capped NMF [46]. The main contribution of robust capped NMF is that it caps the residual of extreme outlier to further enhance the robustness. Li et al. propose a robust NMF using $l_{2,p}$-norm to further reduce the impact of noise and outliers [47]. Guan et al. develop Truncated Cauchy non-negative matrix factorization which utilizes Truncated Cauchy loss to truncate large errors and handle outliers [48]. Guan et al. also propose Manhattan non-negative matrix factorization (MahNMF) which uses Manhattan distance instead of Euclidean distance to model the heavy tailed Laplacian noise [49].

Motivated by the good robustness of $l_{2,p}$-norm and capped $l_{2,1}$-norm and excellent performance of manifold learning based NMF algorithms, a robust capped norm dual hyper-graph regularized NMTF (RCHNMTF) is proposed. Specifically, RCHNMTF utilizes capped $l_{2,p}$-norm to enhance robustness. Second, RCHNMTF constructs dual hyper-graph to encode intrinsic geometrical information in sample manifold and feature manifold. Third, ONMTF framework is incorporated to RCHNMTF to improve clustering performance. The main contribution of the article is summarized as follows:

1) A novel method called RCHNMTF is proposed. RCHNMTF first utilizes capped $l_{2,p}$-norm to improve robustness. Dual hyper-graph regularization and ONMTF framework are also added to RCHNMTF to further improve clustering performance. 2) The optimization problem of RCHNMTF is reformulated and an alternative iteration algorithm is designed thereafter to simplify iteration steps. The computational complexity of RCHNMTF and its comparison methods are calculated thoroughly with three arithmetic operations and big O notation. 3) Experiments on seven real-world datasets and four noised datasets verify the superior clustering performance and robustness of RCHNMTF, compared with eight state-of-the-art algorithms.

The rest of the paper is arranged as follows: In Section 2, NMF algorithm, $l_{q,p}$-norm, capped $l_{2,1}$-norm NMF and hyper-graph regularization are introduced. In Section 3, the formulation of RCHNMTF is first described. The optimization problem of RCHNMTF is reformulated thereafter to simplify iteration steps. The computational complexity of RCHNMTF and its comparison algorithms are thoroughly analyzed. Section 4 demonstrates the clustering performance and robustness of RCHNMTF on seven real-world datasets and four contaminated datasets, compared with eight state-of-the-art algorithms. The paper is summarized in Section 5.

2. Related works

2.1. Non-negative matrix factorization

Given a data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n] \in R^{m\times n}$, each column of $\boldsymbol{X}$ denotes a sample vector $\boldsymbol{x}_n$. NMF aims to decompose the data matrix $\boldsymbol{X}$ into two low-dimensional non-negative matrices $\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_k] \in R^{m\times k}$ and $\boldsymbol{V^T} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_k] \in R^{k\times n}$, whose product $\boldsymbol{UV}^T$ is approximate to the original data matrix $\boldsymbol{X}$. The residual error between $\boldsymbol{X}$ and $\boldsymbol{UV}^T$ is calculated by Frobenius norm. The optimization problem of NMF is defined as [4]:

$$\min_{\boldsymbol{U},\boldsymbol{V}} \left\|\boldsymbol{X} - \boldsymbol{UV}^T\right\|_F^2 \quad s.t. \quad \boldsymbol{U} \geq 0, \boldsymbol{V} \geq 0 \tag{2.1}$$

where $\|\cdot\|_F$ represents the Frobenius norm of the matrix. Optimization problem (2.1) can be solved by the following multiplicative update rules [4]:

$$\boldsymbol{U} \leftarrow \boldsymbol{U}\frac{\boldsymbol{XV}}{\boldsymbol{UV}^T\boldsymbol{V}} \tag{2.2}$$

$$\boldsymbol{V} \leftarrow \boldsymbol{V}\frac{\boldsymbol{X}^T\boldsymbol{U}}{\boldsymbol{V}\boldsymbol{U}^T\boldsymbol{U}} \tag{2.3}$$

2.2. $L_{q,p}$-Norm

The $l_{q,p}$-norm of $\boldsymbol{X}$ is defined as follows [47]:

$$\|\boldsymbol{X}\|_{q,p} = \left(\sum_{i=1}^{N} \|\boldsymbol{x}_i\|_q^p\right)^{1/p}, \quad p \in (0, 1]. \tag{2.4}$$

where $\|\cdot\|_{q,p}$ represents $l_{q,p}$-norm. Let $q = 2$, the $l_{2,p}$-norm of $\boldsymbol{X}$ is defined as follows:

$$\|\boldsymbol{X}\|_{2,p} = \left(\sum_{i=1}^{N} \|\boldsymbol{x}_i\|_2^p\right)^{1/p}, \quad p \in (0, 1]. \tag{2.5}$$

where $\|\cdot\|_{2,p}$ represents $l_{2,p}$-norm. It should be noticed that $l_{2,p}$-norm ($0 < p < 1$) is not a valid matrix norm as it does not admit the triangular inequality. However, for simplicity, we term it a matrix norm. Moreover, the $l_{2,p}$ matrix pseudo norm is not convex or Lipschitz continuous as the l_p-norm ($0 < p < 1$) is neither convex nor Lipschitz continuous.

2.3. Capped $L_{2,1}$-norm NMF

To decrease the influence of extreme outliers, Gao et al. propose capped $l_{2,1}$-norm based NMF, whose optimization problem is defined as follows [46]:

$$\min_{\boldsymbol{U},\boldsymbol{V}} \sum_{i=1}^{n} \min\{\|\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{v}_i\|_2, \theta\}, \quad s.t. \quad \boldsymbol{U} \geq 0, \boldsymbol{V} \geq 0. \tag{2.6}$$

where $\theta > 0$ is a thresholding parameter to choose the extreme data outliers. In capped $l_{2,1}$-norm based NMF, if the residual error of a data point $\|\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{v}_i\| > \theta$, then data point x_i is determined as an extreme outlier and its residual is capped as θ to fix its effect on the whole model. For other data points $\|\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{v}_i\| < \theta$, the algorithm calculates the residual error using l_1-norm, which is also robust to regular data outliers. Moreover, θ is set according to the ratio of outliers. With an extra thresholding parameter θ to select extreme data outliers and cap their influence, capped $l_{2,1}$-norm based NMF is more robust than $l_{2,1}$-norm based NMF .

2.4. Hyper-graph regularization

With the development of graph theory, hyper-graph regularization has been widely used. Unlike graph regularization only considers the pairwise relationships between two samples or two features, hyper graph regularization considers the relationships between multiple samples or multiple features. Specifically, an edge of graph connects two nodes while an edge of hyper graph connects multiple nodes. Therefore, using hyper-graph prevents the forced conversion of multivariate relationships into binary relationships and explores the high-order information in sample space and feature space [43].

Hyper-graph $G = (V, E, \boldsymbol{W})$ consists of vertex set V, hyper-edge set E and diagonal hyper-edge weight matrix $\boldsymbol{W}$. The incidence matrix $\boldsymbol{H} \in R^{|V|\times|E|}$ is defined as follows:

$$\boldsymbol{H}(v,e) = \begin{cases} 1, & if \quad v \in e; \\ 0, & if \quad v \notin e. \end{cases} \tag{2.7}$$

$\boldsymbol{W}_i$ represents the weight of hyper-edge e_i. The calculation method of $\boldsymbol{W}_i$ depends on the specific situation. Denote V_x and V_y represent the sample points $\boldsymbol{X}_x$ and $\boldsymbol{X}_y$, respectively. In this paper, $\boldsymbol{W}_i$ is defined as:

$$\boldsymbol{W}_i = \sum_{V_x, V_y \in e_i} exp\left(-\frac{\|V_x - V_y\|_2^2}{\sigma^2}\right) \tag{2.8}$$

where σ^2 is defined as:

$$\sigma^2 = \sum_{V_x, V_y \in e_i} \frac{\|V_x - V_y\|_2^2}{k} \tag{2.9}$$

where parameter k represents the value of k-nearest neighbors for each vertex.

The degree of vertex v is expressed as:

$$d(v) = \sum_{e \in E} w(e)\boldsymbol{H}(w,e) \tag{2.10}$$

The degree of edge e is defined as:

$$f(e) = \sum_{v \in V} \boldsymbol{H}(w,e) \tag{2.11}$$

Given diagonal matrix $\boldsymbol{D}_v$ composed of $d(v)$ and diagonal matrix $\boldsymbol{D}_e$ composed of $f(e)$, the unnormalized hyper-graph Laplacian matrix $\boldsymbol{L}_{hyper}$ is defined as:

$$\begin{aligned} \boldsymbol{S} &= \boldsymbol{H}\boldsymbol{W}\boldsymbol{D}_e^{-1}\boldsymbol{H}^{-1} \\ \boldsymbol{L}_{hyper} &= \boldsymbol{D}_v - \boldsymbol{S} \end{aligned} \tag{2.12}$$

3. The proposed model

In this section, a novel algorithm called RCHNMTF is proposed. After that, an efficient optimization algorithm is designed. The computational complexity analysis of RCHNMTF and comparison algorithms are discussed subsequently. Figure 1 shows the framework of RCHNMTF.

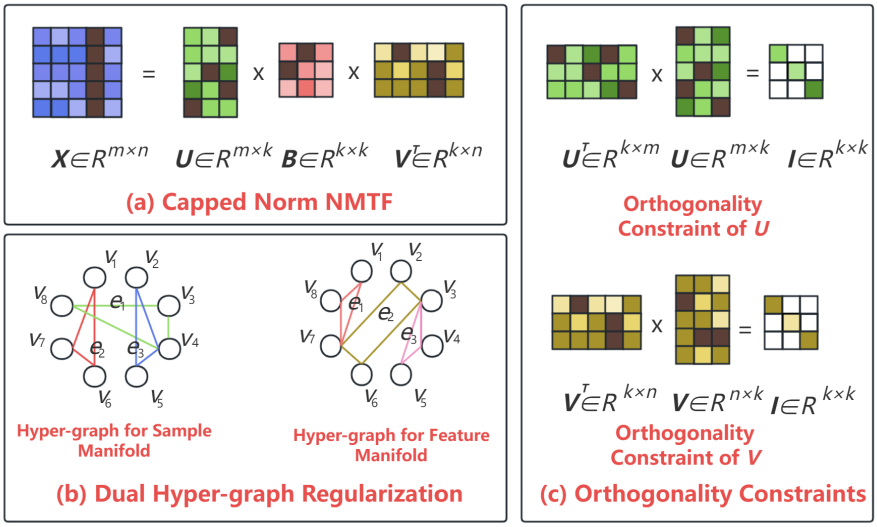


Figure 1. The framework of RCHNMTF: (a) $\boldsymbol{X} \in R^{m\times n}$ is a data matrix and its outlier sample point x_i is colored dark brown. The outlier sample point can contaminate the decomposed matrices $\boldsymbol{U}, \boldsymbol{B}, \boldsymbol{V}$ thoroughly and accordingly reduce the clustering performance. However, as RCHNMTF utilizing capped norm to limit the influence of outlier sample points, only a small part of decomposed matrices $\boldsymbol{U}, \boldsymbol{B}, \boldsymbol{V}$ is influenced, which is also colored dark brown; (b) let V_n denote vertex n and e_n denote hyper-edge n, the dual hyper-graph regularization constructs dual hyper-graph for sample manifold and feature manifold to obtain geometric information of data $\boldsymbol{X}$; (c) the orthogonality constraints of matrices $\boldsymbol{U}, \boldsymbol{V}$ guarantee a unique and sparse solution.

3.1. Problem formulation

The optimization problem of NMTF with capped $l_{2,p}$-norm is formulated as follows:

$$\begin{aligned} &\min_{\boldsymbol{U},\boldsymbol{B},\boldsymbol{V}} \sum_{i=1}^{n} \min\{\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p, \theta\} \\ &s.t. \boldsymbol{U} \geq 0, \boldsymbol{B} \geq 0, \boldsymbol{V} \geq 0, p \in (0,1]. \end{aligned} \tag{3.1}$$

where θ denotes a thresholding parameter.

In optimization problem (3.1), $\sum_{i=1}^{n} \min\{\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p, \theta\}$ indicates using capped $l_{2,p}$-norm to measure the residual error of each point. If the residual error $\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p$ of data point $\boldsymbol{x}_i$ is bigger than θ, data point $\boldsymbol{x}_i$ is determined as extreme outliers and its residual error is capped. The influence of outlier sample point $\boldsymbol{x}_i$ to the whole model is decreased by capping its residual error. For other data points whose residual error $\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p < \theta$, the optimization problem will minimize $\sum_{i=1}^{n} \left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p$, which is also robust as it calculates the residual error of each sample point to the p-th power, compared with Forbenius norm in original NMF measures the residual error to the power of 2. Therefore, utilizing capped $l_{2,p}$ -norm, the proposed algorithm can decrease the influence of extreme outliers and have good robustness.

To adopt geometric information of data space and sample space, dual hyper-graph regularization is incorporated to optimization problem (3.1). Let $\boldsymbol{L}_{hyper}^{\boldsymbol{U}}$ and $\boldsymbol{L}_{hyper}^{\boldsymbol{V}}$ represent the unnormalized hyper-

graph Laplacian matrix of matrix $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. With hyper-graph Laplacian matrix $\boldsymbol{L}^{\boldsymbol{U}}_{hyper}$ and $\boldsymbol{L}^{\boldsymbol{V}}_{hyper}$, we extend the optimization problem as:

$$\begin{aligned}\min_{\boldsymbol{U},\boldsymbol{B},\boldsymbol{V}} \sum_{i=1}^{n} \min\{\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p, \theta\} + \alpha Tr(\boldsymbol{V}^T\boldsymbol{L}^{\boldsymbol{V}}_{hyper}\boldsymbol{V}) \\ +\alpha Tr(\boldsymbol{U}^T\boldsymbol{L}^{\boldsymbol{U}}_{hyper}\boldsymbol{U}) \quad s.t. \quad \boldsymbol{U}\geq 0, \boldsymbol{B}\geq 0, \boldsymbol{V}\geq 0, p\in(0,1].\end{aligned} \tag{3.2}$$

where $Tr(\cdot)$ denotes the trace of a matrix and α denotes dual hyper-graph regularization parameter. The dual hyper-graph regularization is formulated as $\alpha Tr(\boldsymbol{V}^T\boldsymbol{L}^{\boldsymbol{V}}_{hyper}\boldsymbol{V}) + \alpha Tr(\boldsymbol{U}^T\boldsymbol{L}^{\boldsymbol{U}}_{hyper}\boldsymbol{U})$. Dual hyper-graph regularization utilizes two nearest neighbor hyper-graphs to exploit the geometric structure of the sample manifold and feature manifold.

Orthogonality constraints in NMTF is important as orthogonality constraints lead to a sparse and unique solution. To guarantee orthogonality of matrix $\boldsymbol{U}$ and $\boldsymbol{V}$, orthogonality penalty terms $\beta\left\|\boldsymbol{U}^T\boldsymbol{U}-\boldsymbol{I}_K\right\|_F^2$ and $\beta\left\|\boldsymbol{V}^T\boldsymbol{V}-\boldsymbol{I}_K\right\|_F^2$ are added to the optimization problem:

$$\begin{aligned}\min_{\boldsymbol{U},\boldsymbol{B},\boldsymbol{V}} \sum_{i=1}^{n} \min\{\left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})_i\right\|_2^p, \theta\} + \alpha Tr(\boldsymbol{V}^T\boldsymbol{L}^{\boldsymbol{V}}_{hyper}\boldsymbol{V}) \\ +\alpha Tr(\boldsymbol{U}^T\boldsymbol{L}^{\boldsymbol{U}}_{hyper}\boldsymbol{U}) + \beta\left\|\boldsymbol{U}^T\boldsymbol{U}-\boldsymbol{I}_K\right\|_F^2 + \beta\left\|\boldsymbol{V}^T\boldsymbol{V}-\boldsymbol{I}_K\right\|_F^2 \\ s.t. \quad \boldsymbol{U}\geq 0, \boldsymbol{B}\geq 0, \boldsymbol{V}\geq 0, p\in(0,1].\end{aligned} \tag{3.3}$$

where $\beta \geq 0$ denotes orthogonality parameter.

3.2. Optimization of RCHNMTF

First, the following equation can be easily proved:

$$\left\|\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}}\right\|_{2,p}^p = Tr((\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})\boldsymbol{H}(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^{\boldsymbol{T}})^T), p\in(0,1]. \tag{3.4}$$

where $\boldsymbol{H}$ is the diagonal matrix whose i -th diagonal element is calculated as:

$$h_{ii} = \frac{1}{\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T)_i\|_2^{2-p}} \tag{3.5}$$

It is apparent that optimization problem (3.3) is a non-convex optimization problem, so it is not easy to solve the optimization problem (3.3) directly. The optimization problem (3.3) is reformulated as follows:

$$\begin{aligned}\min_{\boldsymbol{U},\boldsymbol{B},\boldsymbol{V}} Tr\left(\left(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\right)\boldsymbol{D}\left(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\right)^T\right) + \alpha Tr(\boldsymbol{V}^T\boldsymbol{L}^{\boldsymbol{V}}_{hyper}\boldsymbol{V}) \\ + \alpha Tr(\boldsymbol{U}^T\boldsymbol{L}^{\boldsymbol{U}}_{hyper}\boldsymbol{U}) + \beta\left\|\boldsymbol{U}^T\boldsymbol{U}-\boldsymbol{I}_k\right\|_F^2 + \beta\left\|\boldsymbol{V}^T\boldsymbol{V}-\boldsymbol{I}_k\right\|_F^2 \\ s.t. \quad \boldsymbol{U}\geq 0, \boldsymbol{B}\geq 0, \boldsymbol{V}\geq 0, p\in(0,1].\end{aligned} \tag{3.6}$$

where $\boldsymbol{I}_k$ is the $a \times a$ identity matrix and $\boldsymbol{D}$ is the diagonal matrix whose i-th diagonal element is calculated as:

$$d_{ii} = \begin{cases}\dfrac{1}{\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T)_i\|_2^{2-p}}, & if \left\|(\boldsymbol{X}-\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T)_i\right\|_2^{2-p} < \theta; \\ 0, \quad otherwise.\end{cases} \tag{3.7}$$

3.3. Update rules of RCHNMF

Considering $\|\boldsymbol{X}\|_F^2 = Tr(\boldsymbol{X}\boldsymbol{X}^T)$, the optimization problem (3.6) is rewritten as follows:

$$\begin{aligned}\min_{\boldsymbol{U},\boldsymbol{B},\boldsymbol{V}} & Tr\left(\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\right) - 2Tr\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{X}^T\right) + Tr\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T\boldsymbol{U}^T\right) \\ & + \alpha Tr\left(\boldsymbol{V}^T\boldsymbol{L}_{hyper}^{\boldsymbol{V}}\boldsymbol{V}\right) + \alpha Tr\left(\boldsymbol{U}^T\boldsymbol{L}_{hyper}^{\boldsymbol{U}}\boldsymbol{U}\right) \\ & + \beta Tr\left(\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{U} - 2\boldsymbol{U}^T\boldsymbol{U} + \boldsymbol{I}_k\right) \\ & + \beta Tr\left(\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{V}^T\boldsymbol{V} - 2\boldsymbol{V}^T\boldsymbol{V} + \boldsymbol{I}_k\right)\end{aligned} \tag{3.8}$$

Optimization problem (3.8) can be solved by multiplicative iteration method. Let $\boldsymbol{\phi} = [\phi_{jk}] \in R_{\geq 0}^{m\times k}$, $\boldsymbol{\omega} = [\omega_{kk'}] \in R_{\geq 0}^{k\times k}$, $\boldsymbol{\psi} = [\psi_{ik}] \in R_{\geq 0}^{n\times k}$ be the Lagrange multipliers for $\boldsymbol{U}$, $\boldsymbol{B}$, $\boldsymbol{V}$ respectively, Lagrange function $\mathcal{L}$ is obtained as follows:

$$\begin{aligned}\mathcal{L} = & Tr\left(\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\right) - 2Tr\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{X}^T\right) + Tr\left(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T\boldsymbol{U}^T\right) \\ & + \alpha Tr\left(\boldsymbol{V}^T\boldsymbol{L}_{hyper}^{\boldsymbol{V}}\boldsymbol{V}\right) + \alpha Tr\left(\boldsymbol{U}^T\boldsymbol{L}_{hyper}^{\boldsymbol{U}}\boldsymbol{U}\right) \\ & + \beta Tr\left(\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{U} - 2\boldsymbol{U}^T\boldsymbol{U} + \boldsymbol{I}_k\right) \\ & + \beta Tr\left(\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{V}^T V - 2\boldsymbol{V}^T\boldsymbol{V} + \boldsymbol{I}_k\right) \\ & + Tr\left(\boldsymbol{\phi}\boldsymbol{U}^T\right) + Tr\left(\boldsymbol{\omega}\boldsymbol{B}^T\right) + Tr\left(\boldsymbol{\psi}\boldsymbol{V}^T\right)\end{aligned} \tag{3.9}$$

The partial derivatives of $\mathcal{L}$ with respect to $\boldsymbol{U}$, $\boldsymbol{B}$ and $\boldsymbol{V}$ are as follows:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{U}} = & -2\boldsymbol{X}\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T + 2\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T \\ & + 2\alpha\boldsymbol{L}_{hyper}^{\boldsymbol{U}} + 4\beta(\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{U} - \boldsymbol{U}) + \boldsymbol{\phi}\end{aligned} \tag{3.10}$$

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{B}} = -2\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{V} + 2\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V} + \boldsymbol{\omega} \tag{3.11}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{V}} = & -2\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{U}\boldsymbol{B} + 2\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T U^T\boldsymbol{U}\boldsymbol{B} \\ & + 2\alpha\boldsymbol{L}_{hyper}^{\boldsymbol{V}} + 4\beta(\boldsymbol{V}\boldsymbol{V}^T\boldsymbol{V} - \boldsymbol{V}) + \boldsymbol{\psi}\end{aligned} \tag{3.12}$$

Considering that $\boldsymbol{L}_{hyper}^{\boldsymbol{U}}$ and $\boldsymbol{L}_{hyper}^{\boldsymbol{V}}$ are not non-negative matrices, we refer to (2.12) and define non-negative matrices $\boldsymbol{D}_{hyper}^{\boldsymbol{U}}$, $\boldsymbol{S}_{hyper}^{\boldsymbol{U}}$, $\boldsymbol{D}_{hyper}^{\boldsymbol{V}}$, $\boldsymbol{S}_{hyper}^{\boldsymbol{V}}$ as follows:

$$\begin{aligned}\boldsymbol{L}_{hyper}^{\boldsymbol{U}} &= \boldsymbol{D}_{hyper}^{\boldsymbol{U}} - \boldsymbol{S}_{hyper}^{\boldsymbol{U}} \\ \boldsymbol{L}_{hyper}^{\boldsymbol{V}} &= \boldsymbol{D}_{hyper}^{\boldsymbol{V}} - \boldsymbol{S}_{hyper}^{\boldsymbol{V}}\end{aligned} \tag{3.13}$$

Using the KKT condition $\boldsymbol{\phi}\boldsymbol{U} = 0$, $\boldsymbol{\omega}\boldsymbol{B} = 0$, and $\boldsymbol{\beta}\boldsymbol{V} = 0$, we have:

$$\begin{aligned}-(\boldsymbol{X}\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T + \alpha\boldsymbol{S}_{hyper}^{\boldsymbol{U}}U + 2\beta\boldsymbol{U})_{jk}u_{jk} \\ +(\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V}\boldsymbol{B}^T + \alpha\boldsymbol{D}_{hyper}^{\boldsymbol{U}}\boldsymbol{U} + 2\beta\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{U})_{jk}u_{jk} = 0\end{aligned} \tag{3.14}$$

$$-(\boldsymbol{U}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{V})_{kk'}b_{kk'} + (\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{B}\boldsymbol{V}^T\boldsymbol{D}\boldsymbol{V})_{kk'}b_{kk'} = 0 \tag{3.15}$$

$$\begin{aligned} &-(\boldsymbol{DX}^T\boldsymbol{UB} + \alpha\boldsymbol{S}^{\boldsymbol{V}}_{hyper}\boldsymbol{V} + 2\beta\boldsymbol{V})_{ik}v_{ik} \\ &+(\boldsymbol{DVB}^T\boldsymbol{U}^T\boldsymbol{UB} + \alpha\boldsymbol{D}^{\boldsymbol{V}}_{hyper}\boldsymbol{V} + 2\beta\boldsymbol{VV}^T\boldsymbol{V})_{ik}v_{ik} = 0 \end{aligned} \tag{3.16}$$

According to the above equations, the multiplicative update rules of RCHNMTF are as follows:

$$u_{jk} \leftarrow u_{jk}\frac{(\boldsymbol{XDVB}^T + \alpha\boldsymbol{S}^{U}_{hyper}U + 2\beta\boldsymbol{U})_{jk}}{(\boldsymbol{UBV}^T\boldsymbol{DVB}^T + \alpha\boldsymbol{D}^{\boldsymbol{U}}_{hyper}\boldsymbol{U} + 2\beta\boldsymbol{UU}^T\boldsymbol{U})_{jk}} \tag{3.17}$$

$$b_{kk'} \leftarrow b_{kk'}\frac{(\boldsymbol{U}^T\boldsymbol{XDV})_{kk'}}{(\boldsymbol{U}^T\boldsymbol{UBV}^T\boldsymbol{DV})_{kk'}} \tag{3.18}$$

$$v_{ik} \leftarrow v_{ik}\frac{(\boldsymbol{DX}^T\boldsymbol{UB} + \alpha\boldsymbol{S}^{\boldsymbol{V}}_{hyper}\boldsymbol{V} + 2\beta\boldsymbol{V})_{ik}}{(\boldsymbol{DVB}^T\boldsymbol{U}^T\boldsymbol{UB} + \alpha\boldsymbol{D}^{\boldsymbol{V}}_{hyper}\boldsymbol{V} + 2\beta\boldsymbol{VV}^T\boldsymbol{V})_{ik}} \tag{3.19}$$

Algorithm 1 Alternative iteration algorithm for RCHNMTF

Input: Data matrix $\boldsymbol{X} \in R^{m\times n}$, parameter α, β, k.
Output: $\boldsymbol{U} \in R^{m\times k}$, $\boldsymbol{V} \in R^{n\times k}$

1: Initialize $\boldsymbol{U} \geq 0, \boldsymbol{B} \geq 0, \boldsymbol{V} \geq 0$
2: Initialize matrix $\boldsymbol{D}$ based on Eq (3.7)
3: Repeat:
4: Calculate matrix $\boldsymbol{D}$ based on Eq (3.7)
5: Update $\boldsymbol{U}$ based on Eq (3.17)
6: Update $\boldsymbol{B}$ based on Eq (3.18)
7: Update $\boldsymbol{V}$ based on Eq (3.19)
8: Until Converges
9: Return $\boldsymbol{U}, \boldsymbol{V}$
10: Apply K-means clustering method to $\boldsymbol{U}, \boldsymbol{V}$ to get learned cluster label information.

3.4. Computational complexity analysis

In this part, the computational complexity of the proposed RCHNMTF algorithm and comparison algorithms are calculated and presented. The computational complexity of NMF is also added for reference. To accurately present the computational complexity of RCHNMTF and comparison algorithms, big O notation and three arithmetic operations, namely addition, multiplication and division, are adopted. Table 1 shows the detailed computational complexity information of each algorithm in each iteration. It can be seen that the extra factorization matrix, graph regularization, hyper-graph regularization and orthogonality constraints will all increase the computational complexity. However, these extra term will not dramatically affect the runtime of algorithms. It can be explained by two reasons. First, if measured by big O notation, the computational complexity of all algorithms are $O(mnk)$. Second, considering $k << \min(m, n)$, mnk term counts much in computational complexity analysis and the coefficient of mnk term in RCHNMTF is not high, which indicates the computational complexity of RCHNMTF is not high. Moreover, compared with other M-estimator based NMF algorithms, the computational complexity of capped $l_{q,p}$-norm are small (e.g., correntropy based NMF algorithms require exponential operation). If calculate the

computational complexity of RCHNMTF algorithm from the beggining to the n-th iterations, then the computational complexity of RCHNMTF is $O(tmnk + mn^2 + nm^2)$, due to the extra $O(mn^2 + nm^2)$ from constructing dual hyper-graph in feature space and sample space.

Table 1. Computational complexity comparison.

	floating point addition	floating point multiplication	floating point division	overall
CNMTF	$5mnk + 9mk^2 + 6nk^2 + 2k^3 + 11mk + 5nk + (m+n)pk$	$5mnk + 9mk^2 + 6nk^2 + 2k^3 + 8mk + 2nk + k^2 + (m+n)pK$	$mk + nk + k^2$	$O(mnk)$
CHNMF	$2mnk + 2mk^2 + 2nk^2 + 4mk + 3nk + npk$	$2mnk + 2mk^2 + 2nk^2 + 5mk + 2nk + npk$	$mk + nk$	$O(mnk)$
DNMF	$2mnk + 2mk^2 + 2nk^2 + 3mk + 3nk + (m+n)pk$	$2mnk + 2mk^2 + 2nk^2 + 2mk + 2nk + (m+n)pk$	$mk + nk$	$O(mnk)$
GNMF	$2mnk + 2mk^2 + 2nk^2 + mk + 3nk + npk$	$2mnk + 2mk^2 + 2nk^2 + mk + 2nk + npk$	$mk + nk$	$O(mnk)$
HNMF	$2mnk + 2mk^2 + 2nk^2 + mk + 3nk + mpk$	$2mnk + 2mk^2 + 2nk^2 + mk + 2nk + npk$	$mk + nk$	$O(mnk)$
RHNMF	$2mnk+2mk^2+2nk^2+7nk+npk$	$2mnk + 2mk^2 + 2nk^2 + mk + 6nk + npp$	$mk + nk$	$O(mnk)$
CaNMF	$3mnk + mk^2 + 3nk^2 + 4nk$	$3mnk+mk^2+3nk^2+mk+5nk$	$mk + nk$	$O(mnk)$
DHSNMF	$2mnk + 2mk^2 + 2nk^2 + 3mk + 3nk + (m+n)pk$	$2mnk + 2mk^2 + 2nk^2 + 2mk + 2nk + (m+n)pk$	$mk + nk$	$O(mnk)$
RCHNMTF	$3mnk + 8mk^2 + 5nk^2 + 6k^3 + 5mk + 11nk + (m+n)pk$	$3mnk + 8mk^2 + 5nk^2 + 6k^3 + 2mk + 8nk + k^2 + (m+n)pk$	$mk + nk + k^2$	$O(mnk)$

4. Experiments

In this section, RCHNMTF algorithm is compared with eight state-of-the-art algorithms on seven datasets (i.e., AR100*, ORL†, YALE‡, PIE§, MSRA25¶, PENDIGITS‖ and COIL100**) to verify the effectiveness and robustness of the proposed algorithm.

4.1. Experiments setting

4.1.1. Datasets

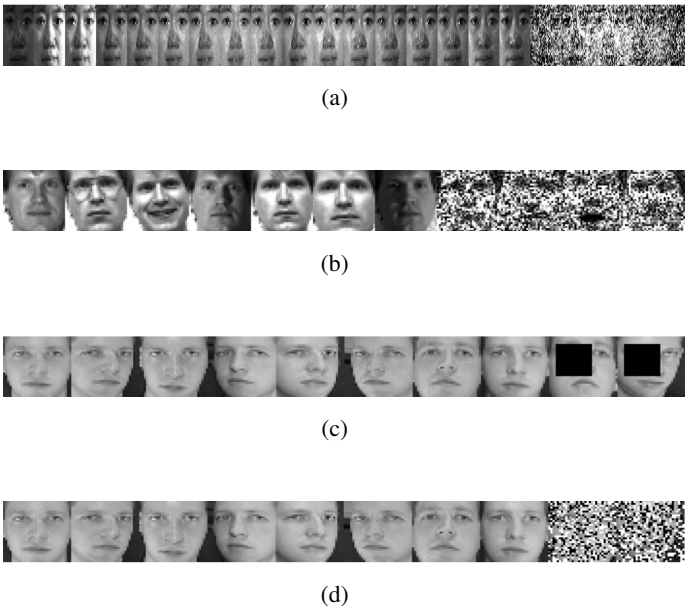
Seven real-world datasets are used in the experiment. The description of these datasets can be found in Table 2. Besides, to simulate the situation in reality where parts of sample points are affected by noise

*http://www2.ece.ohio-state.edu/ aleix/ARdatabase.html
†http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.tar.Z
‡http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html
§https://www.ri.cmu.edu/publications/the-cmu-pose-illumination-and-expression-pie-database-of-human-faces
¶http://www.escience.cn/people/fpnie/index.html
‖http://archive.ics.uci.edu/ml/datasets/pen-based+recognition+of+handwritten+digits
**https://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php

and outliers, four contaminated datasets are produced by adding noise to parts of the sample points of the original dataset while leaving the other data points unchanged. Specifically, the formulations are: 1) The last 5 images of each class in PIE dataset are noised by Gaussian noise with mean 0 and variance $V = 0.1, 0.2, 0.3, 0.4, 0.5$; 2) The last 4 images of each class in YALE dataset are contaminated with Speckle noise with mean 0 and variance $V = 0.08, 0.16, 0.24, 0.32, 0.4$; 3) The last 2 images of each class in ORL dataset are polluted by $a \times a$-blocks noise with $a = 6, 7, 8, 9, 10$; 4) The last 2 images of each class in ORL dataset are contaminated by Salt & Pepper noise with noise density $D = 10, 20, 30, 40, 50\%$. Figure 2 shows the example class of each polluted datasets.

Table 2. Datasets description.

Datasets	Sample	Feature	Class	Data types
AR100	1300	1024	100	Face Image
ORL	400	1024	40	Face Image
YALE	165	1024	15	Face Image
PIE	1166	1024	53	Face Image
MSRA25	1799	256	12	Face Image
PENDIGITS	10992	16	10	Handwritten digits
COIL100	7200	1024	100	Object image



(a)

(b)

(c)

(d)

Figure 2. Sample class of polluted datasets: (a) an example class from PIE dataset and its last 5 images are contaminated by Gaussian noise with mean 0 and variance 0.5; (b) an example class from YALE dataset and its last 4 images are polluted by Speckle noise with mean 0 and variance 0.4; (c) an example class from ORL dataset with its last 2 images polluted by 10×10-blocks noise. (d) an example class from ORL dataset and its last 2 images are contaminated by Salt & Pepper noise with noise density $D = 50\%$;

4.1.2. Comparison algorithms

The proposed RCHNMTF algorithm are compared with eight state-of-the-art algorithms to verify its clustering performance and robustness, which are listed as follows:

1). CNMTF [50]. A correntropy based NMTF combines dual graph regularization and orthogonal constraints.

2). CHNMF [7]. It incorporates correntropy and hyper-graph regularization into NMF.

3). DNMF [17]. It utilizes dual graph regularization to learn the geometrical information of the sample manifold and the feature manifold.

4). GNMF [35]. It constructs a single graph to consider the geometrical information of the sample manifold.

5). HNMF [43]. A hyper-graph regularized NMF constructs a hyper-graph to explore the geometrical information of the sample manifold.

6). RHNMF [8]. A robust NMF utilizes $l_{2,1}$-norm and hyper-graph regularization.

7). CaNMF [46]. A robust NMF adopts capped norm to cap the residual error of extreme outliers.

8). DHSNMF [51]. A robust dual hyper-graph regularized supervised NMF. To compare DHSNMF with other unsupervised algorithms in a fair way, the label information parameter of DHSNMF is set to 0 specifically.

4.1.3. Evaluation metrics

To evaluate the performance and robustness of RCHNMTF and comparison algorithms in a sound manner, three evaluation metrics, namely purity (PUR) , normalized mutual information (NMI) and accuracy (ACC) are introduced.

Purity demonstrates how well each cluster contains sample from primarily one, which is defined as follows:

$$Purity = \frac{1}{N}\sum_{j=1}^{N} max(n_i^j) \tag{4.1}$$

where n_i^j denotes the sample in cluster i that also belongs to original class j.

NMI calculates the shared information of two clusters and it expresses the degree of agreement between the two clusters. Given the ground truth cluster C and the cluster $\overline{C}$ from clustering algorithm result, NMI of C and $\overline{C}$ is calculated as follows:

$$NMI = \frac{MI(C,\overline{C})}{max(H(C), H(\overline{C}))} \tag{4.2}$$

where $MI(C,\overline{C})$ denotes the mutual information of the cluster C and $\overline{C}$, $H(C)$ and $H(\overline{C})$ are the entropies of C and $\overline{C}$, respectively.

Given ground truth label l_i and the learned cluster label r_i, accuracy is defined as follows:

$$ACC = \frac{\sum_{n=1}^{N}\delta(l_n, map(r_n))}{n} \tag{4.3}$$

where $\delta(x,y) = 1$ if $x = y$ and $\delta(x,y) = 0$ otherwise. Mapping function $Map(\cdot)$ is solved by Hungarian algorithm [52]. For the mentioned evaluation metrics PUR, NMI and ACC, the higher they are, the better clustering performance the algorithm is.

4.1.4. Experimental setup

In this part, experiments setup are reported in detail. The dimension k of the $\boldsymbol{B} \in R^{k\times k}$ is adjusted to be the same as the number of real classes for all datasets. K-nearest neighbor method is applied to construct graph and hyper-graph in graph regularized and hyper-graph regularized algorithms. The nearest neighbor parameter p in k-nearest neighbor method is set to 5 empirically. HeatKernel method is adopted to assign weights for each edge of a graph. For hyper-graph, the weight of a hyper-edge is calculated by (2.8). Parameter θ is not fixed but set according to the outliers radio s. Outliers ratio s denotes the ratio of outlier samples to the whole sample. Specifically, in the first five iterations, the outlier data with the largest $l_{2,p}$ loss at a ratio of s are selected to determine θ. To better reduce the influence of extreme sample points, s is set to 0.05 and 0.1 for unpolluted real-world datasets and polluted real-world datasets, respectively. Additionally, dual hyper-graph regularization parameter α in RCHNMTF is tuned as 100 and orthogonality parameter β is set to 0.01. Moreover, the parameter p in the capped $l_{2,p}$-norm is set to 0.5 to achieve good clustering performance and robustness. The parameters of the comparison algorithms are set according to the default given values. Matrices $\boldsymbol{U}, \boldsymbol{B}, \boldsymbol{V}$ are initialized randomly and we run each method on different datasets 20 times. After decomposition to obtain low-dimensional representation matrix $\boldsymbol{V}$, we apply k-means method to $\boldsymbol{V}$ to get clustering performance measured by PUR, NMI and ACC.

4.2. Experiment results

To testify the clustering performance of RCHNMTF, RCHNMTF and eight comparison algorithms are first experimented on seven real-world datasets. Tables 3–5 demonstrate the clustering performance on seven real-world datasets measured by PUR, NMI and ACC. Figures 3–6 present the clustering performance on polluted datasets. Figure 7 shows the convergent results on seven datasets. To validate the robustness of RCHNMTF, RCHNMTF and the comparison algorithms are further experimented on four contaminated datasets. From the experiment results on seven real-world datasets and four polluted datasets, the following conclusions are obtained:

1) RCHNMTF outperforms other algorithms on most original datasets (e.g., AR100, PIE, ORL, MSRA25, PENDIGITS, COIL100). The reasons are as follows: i.) Capped $l_{2,p}$-norm can alleviate the impact caused by the noise and outliers inherent in the original datasets; ii.) Dual hyper-graph regularization helps RCHNMTF to obtain intrinsic geometrical information of feature manifold and data manifold; iii.) Orthogonal NMTF framework provides a decomposed results with more degree of freedom and enables RCHNMTF to learn a unique decomposition result.

2) The algorithms which utilize $l_{q,p}$-norm or capped $l_{q,p}$-norm (e.g., RHNMF, CaNMF, RCHNMTF) are more robust than other algorithms as the former calculates the residual error of each sample point. When parts of sample points are contaminated, $l_{q,p}$-norm and capped $l_{q,p}$-norm are able to distinguish the polluted sample points and reduces the influence of outlier sample points. Moreover, RCHNMTF is more robust to CaNMF and RHNMF because capped $l_{2,p}$-norm caps the residual error of extreme outlier points and uses $l_{2,0.5}$-norm to calculate the residual error of other data, which can alleviate the impact of outlier points as much as possible. Therefore, RCHNMTF outperforms other algorithms when the dataset has extreme outliers.

3) The convergence curve in Figure 7 shows the convergence results of RCHNMTF.

Table 3. Clustering accuracy and standard deviation (accuracy ± standard deviation) on different datasets.

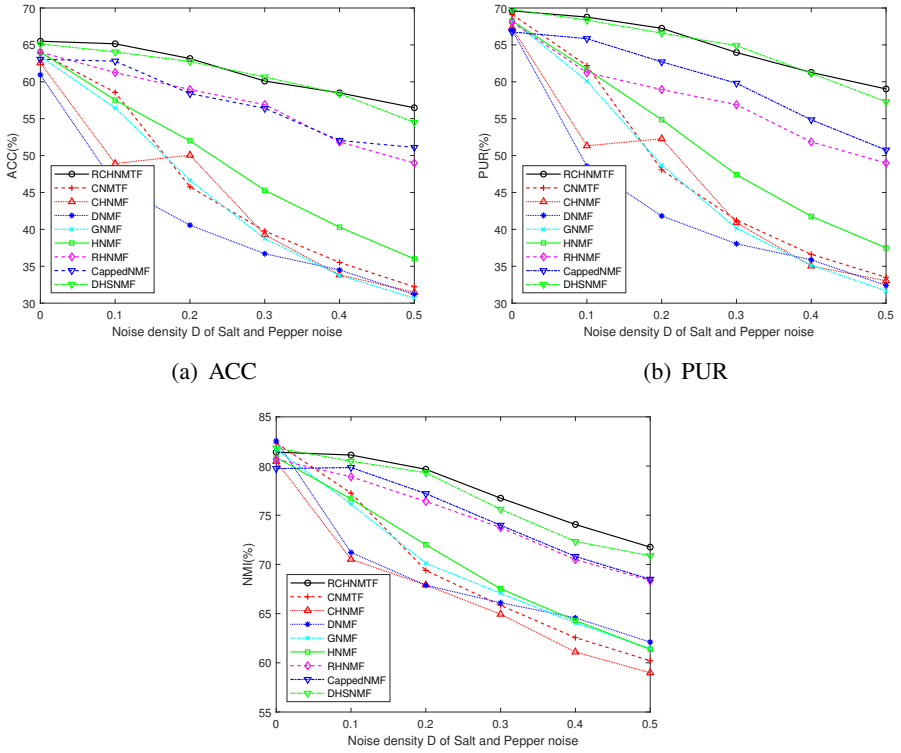
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
CNMTF [50]	48.78±1.24	70.00±2.18	63.92±2.81	57.31±4.16	47.03±2.42	73.44±3.17	47.72±1.17
CHNMF [7]	52.28±1.37	74.83±1.60	62.56±1.54	57.42±2.16	46.42±2.10	67.12±4.06	47.86±1.55
DNMF [17]	59.70±1.47	77.87±2.71	60.94±1.88	54.78±2.84	42.42±3.18	73.64±4.45	54.81±2.14
GNMF [35]	58.14±1.63	78.73±1.95	63.44±2.99	53.00±2.66	42.24±3.12	71.00±5.19	55.33±1.10
HNMF [43]	55.43±1.57	76.44±2.81	64.05±1.67	55.04±1.87	42.60±2.42	73.24±5.46	58.00±1.29
RHNMF [8]	59.05±1.22	78.49±2.21	63.05±1.50	54.37±1.78	**47.93±1.39**	71.46±6.22	50.62±1.58
DHSNMF [51]	55.96±1.77	75.28±2.58	65.12±1.58	52.57±1.998	43.15±2.58	70.27±4.32	51.91±1.34
RCHNMTF	**60.08±1.91**	**80.04±2.44**	**65.50±2.31**	**59.13±2.72**	45.33±2.39	**74.24±2.91**	**58.26±1.64**

Table 4. Purity (purity± standard deviation) on different datasets.

Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
CNMTF [50]	51.98±1.37	74.10±1.63	69.12±1.96	58.84±3.63	**48.00±2.70**	75.13±2.71	51.91±1.06
CHNMF [7]	55.79±1.42	78.30±1.41	67.36±1.57	59.57±1.49	47.21±2.25	70.22±3.02	52.73±1.34
DNMF [17]	**62.99±1.23**	81.50±2.02	67.02±1.50	56.85±2.31	43.87±2.53	75.56±3.37	58.83±1.59
GNMF [35]	61.20±1.45	82.50±1.85	68.34±1.73	55.05±2.03	43.75±2.90	73.61±3.31	59.12±0.84
HNMF [43]	58.37±1.54	80.64±2.07	68.25±1.55	57.67±1.25	43.87±2.06	74.86±3.58	62.35±0.93
RHNMF [8]	61.52±0.78	79.53±2.39	68.10±1.56	61.11±3.73	44.97±1.81	74.38±2.78	55.78±1.16
CaNMF [46]	61.96±0.99	81.97±1.81	66.75±1.31	56.81±1.63	48.66±1.64	73.42±4.53	53.36±1.18
DHSNMF [51]	58.89±1.46	79.19±1.95	69.20±1.33	55.53±1.88	44.06±2.08	72.48±2.84	56.30±1.21
RCHNMTF	62.50±1.64	**83.35±1.68**	**69.61±2.01**	**61.22±2.82**	46.48±2.40	**75.87±2.67**	**62.75±1.44**

Table 5. Normalized mutual information (NMI± standard deviation) on different datasets.

Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
CNMTF [50]	75.85±1.24	83.37±1.22	82.35±1.28	61.77±3.72	51.68±2.37	**71.10±2.55**	74.99±0.55
CHNMF [7]	78.41±0.72	86.29±1.11	80.48±0.79	63.61±2.46	**51.70±2.24**	65.52±2.22	75.43±0.31
DNMF [17]	82.54±0.54	88.48±1.11	**82.56±0.37**	60.87±2.19	48.26±2.11	69.61±2.98	**81.77±0.32**
GNMF [35]	82.15±0.78	89.52±1.47	81.70±1.29	59.03±1.71	48.02±1.72	69.63±1.23	81.65±0.24
HNMF [43]	80.62±0.96	88.54±1.27	80.87±0.88	60.85±1.84	47.83±1.38	70.33±1.31	78.29±0.81
RHNMF [8]	82.31±0.20	87.21±1.42	80.69±0.93	63.92±3.28	49.38±2.10	69.81±1.43	78.27±0.18
CaNMF [46]	82.08±0.50	88.53±1.20	79.74±0.70	59.68±1.95	51.51±1.41	68.85±3.42	76.18±0.46
DHSNMF [51]	79.8±0.64	87.54±1.03	81.84±1.13	60.45±2.79	48.33±1.43	68.10±1.71	77.00±0.45
RCHNMTF	**82.56±0.77**	**89.64±0.99**	81.84±0.56	**64.92±2.28**	50.46±1.08	67.46±2.36	80.78±0.67



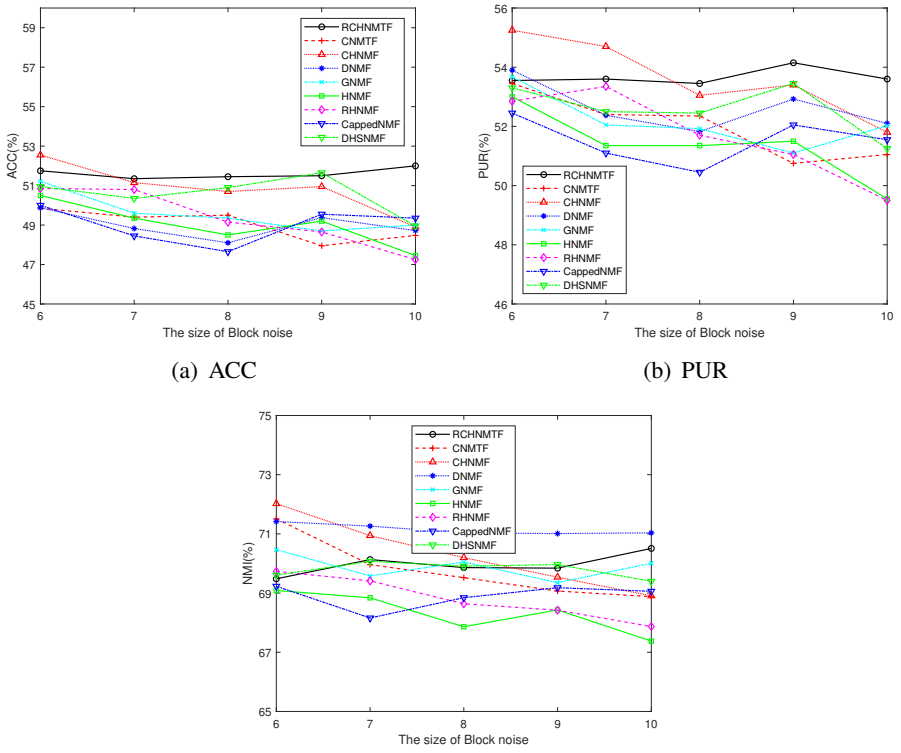
(a) ACC (b) PUR

(c) NMI

Figure 3. Clustering results on ORL dataset contaminated by Salt & Pepper noise.

Table 6. Tunning p on different datasets.

Accuracy under different values of p							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
$p = 0.75$	58.67	77.83	63.25	52.15	44.72	66.88	54.975
$p = 0.5$	**60.08**	**80.04**	**65.50**	**59.13**	**45.33**	**74.24**	58.26
$p = 0.25$	59.09	77.39	63.0	54.43	43.515	70.63	**58.50**
Purity under different values of p							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
$p = 0.75$	61.40	81.32	67.7	55.23	46.42	69.22	59.48
$p = 0.5$	**62.50**	**83.35**	**69.61**	**61.22**	**46.48**	**75.87**	62.75
$p = 0.25$	61.75	81.02	67.95	56.77	44.24	71.78	**62.91**
NMI under different values of p							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
$p = 0.75$	82.68	87.90	80.665	58.24	49.94	65.61	78.83
$p = 0.5$	82.56	**89.64**	**81.84**	**64.92**	**50.46**	**67.46**	**81.78**
$p = 0.25$	**83.24**	88.50	80.67	60.42	48.18	66.29	80.61



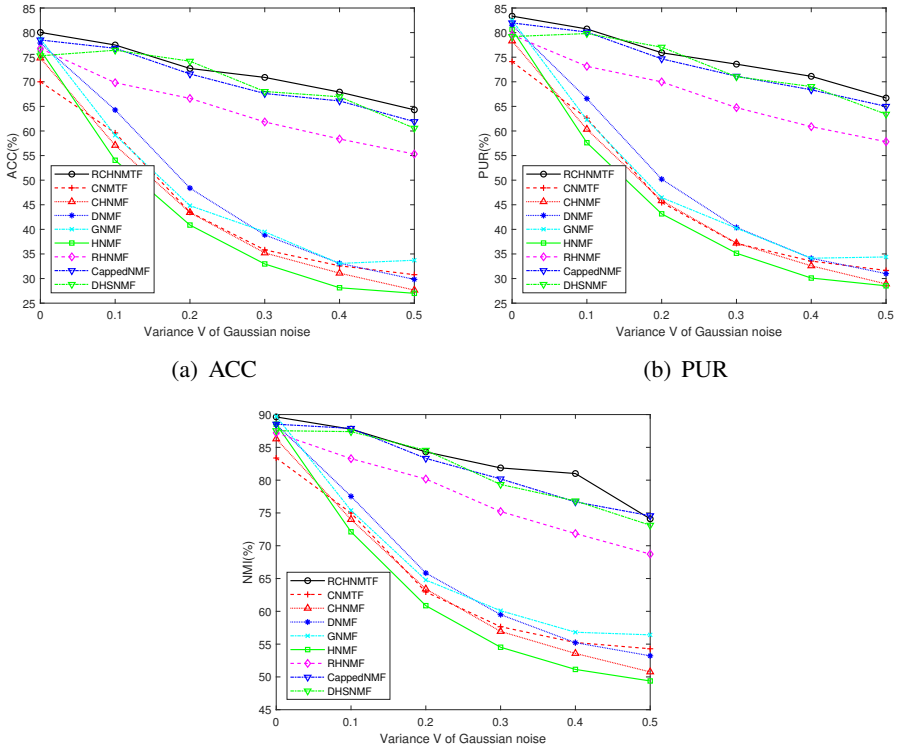
(a) ACC (b) PUR

(c) NMI

Figure 4. Clustering results on ORL dataset contaminated by block noise.

Table 7. Ablation experiments.

RCHNMTF without dual hyper-graph regularization							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
PUR	53.93	79.14	58.85	55.22	44.84	70.34	26.05
NMI	76.76	86.69	73.89	58.14	49.43	64.54	58.28
ACC	51.33	75.48	55.40	53.61	43.75	69.62	24.02
RCHNMTF without orthogonality constraints							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
PUR	33.82	78.21	67.10	46.51	39.39	61.50	29.03
NMI	61.54	86.31	80.15	51.28	45.32	56.48	48.43
ACC	30.32	74.97	62.75	43.46	37.57	60.78	24.33
RCHNMTF with dual hyper-graph regularization and orthogonality constraints							
Datasets	AR100	PIE	ORL	MSRA	YALE	PENDIGITS	COIL100
PUR	62.50	83.35	69.61	61.22	46.48	75.87	62.75
NMI	82.56	89.64	81.84	64.92	50.46	67.46	81.78
ACC	60.08	80.04	65.50	59.13	45.33	74.24	58.26



(a) ACC (b) PUR

(c) NMI

Figure 5. Clustering results on PIE dataset contaminated by Gaussian noise.

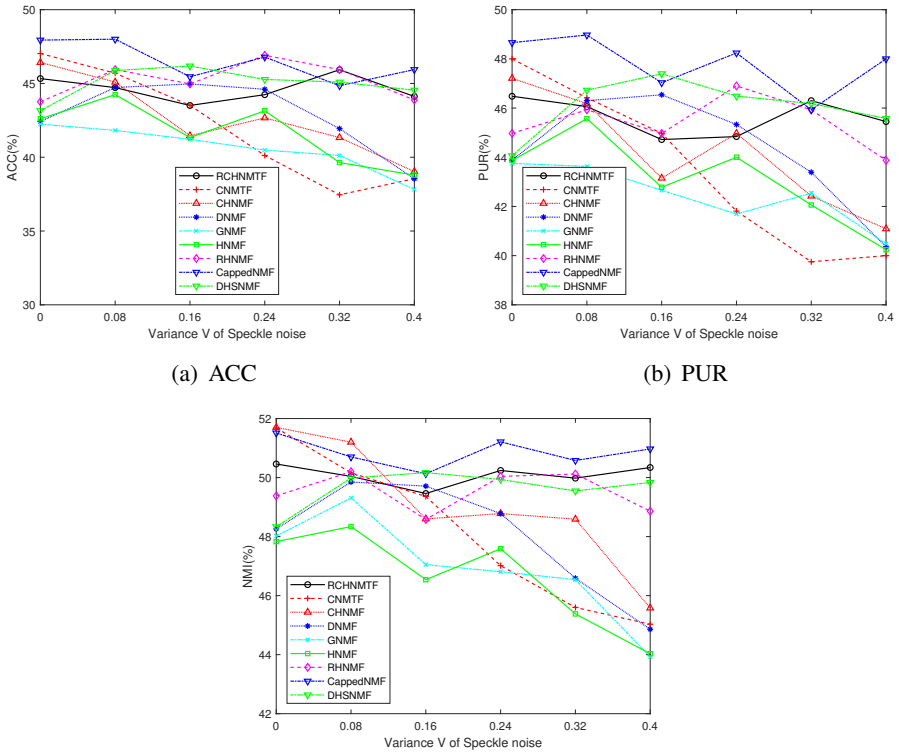
4.3. Parameter selection

Parameter selection are crucial to RCHNMTF algorithm. The main parameters in RCHNMTF are p in capped $l_{2,p}$-norm, outlier ratio s, thresholding parameter θ, hyper-graph graph regularization parameter α and orthogonality constrains parameter β. During the experiments, α and β are fixed as 100 and 0.01 [8, 50]. The value of θ is set according to the ratio of the outliers s. Specifically, in the first five iterations, we select outlier data with the largest $l_{2,p}$ loss at a ratio of s and the value of $l_{2,p}$ loss is assigned to θ. By this means, outlier data with the largest $l_{2,p}$ loss is capped and its impact to the whole model is reduced. In parameter selection, s and p are adjusted to find their optimal values. Figure 8 shows the clustering accuracy on different datasets with s ranging in $\{0, 0.05, 0.1, 0.15, 0.2\}$. Table 6 demonstrates the clustering performance with p ranging in $\{0.25, 0.5, 0.75\}$. The conclusions are as follows:

1) Compared with uncapped $l_{2,p}$-norm, a proper value of s in capped $l_{2,p}$-norm can improve clustering performance on both original and noised datasets (e.g., $s = 0.05$ for original datasets and $s = 0.1$ for noised datasets), because the residual error of intrinsic noised points in original real-world datasets and the formulated outlier sample points in noised datasets are capped and thus the influence of noise and outliers are decreased. However, relatively large s will reduce the clustering performance, as the unpolluted sample points are misidentified as outlier sample points.

2) RCHNMTF with $p = 0.5$ is better than the ones with $p = 0.75$ and $p = 0.25$. Given that $l_{2,p}$-norm calculates the residual error of each sample points to the p-th power, a relatively large p will decrease the robustness of capped $l_{2,p}$-norm while a relatively small p will blur the difference between normal

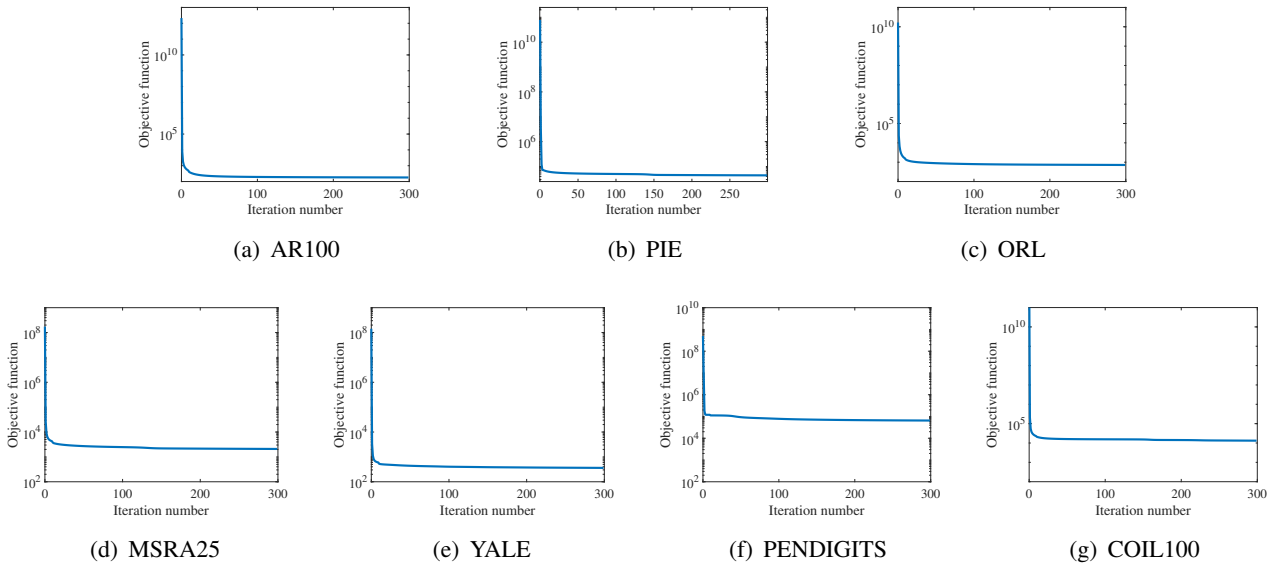
sample points and outlier sample points.



(a) ACC (b) PUR

(c) NMI

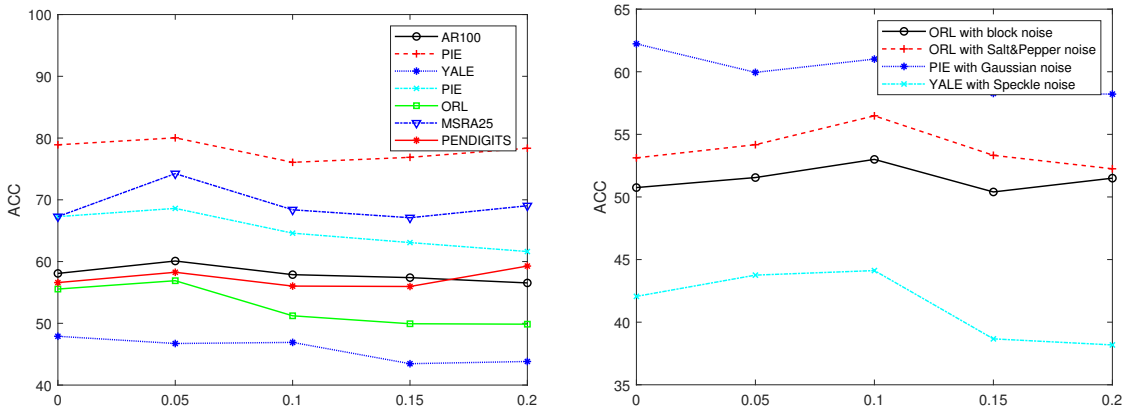
Figure 6. Clustering results on YALE dataset contaminated by Speckle noise.



(a) AR100 (b) PIE (c) ORL

(d) MSRA25 (e) YALE (f) PENDIGITS (g) COIL100

Figure 7. Convergence results on seven datasets.



(a) Tuning outlier ratio s on unpolluted datasets. (b) Tuning outlier ratio s on noised world datasets.

Figure 8. Tuning outlier ratio s.

4.4. Ablation study

In this subsection, ablation study is made to demonstrate the superiority of dual hyper-graph regularization and orthogonality constraints. Table 7 records the ablation study of RCHNMTF for three cases: 1) without dual hyper-graph regularization; 2)without orthogonality constraints; 3) with dual hyper-graph regularization and orthogonality constraints. α and β are accordingly set to: 1) 0 and 0.01; 2) 100 and 0; 3) 100 and 0.01. From Table 7, we can learn that both hyper-graph regularization and orthogonality constraints will improve the clustering performance of RCHNMTF.

5. Conclusions and feature work

In this paper, the robustness of NMTF is improved by introducing capped $l_{2,p}$-norm to cap the extreme outlier points. Dual hyper-graph is constructed to encode the high-dimensional geometrical information of the sample space and feature space. ONMTF framework is incorporated to RCHNMTF to get unique clustering solution and improve clustering performance. To solve the formulated problem, the optimization problem is rewritten and an alternative algorithm is designed. The computational complexity of RCHNMTF and comparison algorithms are thoroughly analyzed. Abundant experiments verify that RCHNMTF performs well on real-world datasets and performs better on datasets with extreme outliers.

Although RCHNMTF is robust and has good clustering performance, it still has some limitations. Firstly, RCHNMTF has many parameters, which makes adjusting parameters troublesome. Secondly, though the first term in (3.3) can distinguish and cap outlier sample points, the orthogonal penalty term and hyper-graph regularization term regard outlier sample points as normal sample points and can not cap the influence of outliers. In the future, several investigations will be conducted to improve RCHNMTF:

1) Systematic mechanisms for simplifying parameter selections are needed.

2) The orthogonality penalty term and hyper-graph regularization term also need to be optimized for finding and capping outlier sample points.

3) Multi-view clustering is popular in machine learning field as multi-view clustering can analyze

multi-view data [53–62]. It is worth the time to utilize RCHNMTF in multi-view clustering field.

Acknowledgements

The authors are grateful for the reviewers and editor for their beneficial suggestions.

conflict of interest

The authors declare that there is no conflict of interest.

References

1. I. T. Jolliffe, J. Cadima, in Principal component analysis: a review and recent developments, *Philosophical transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **374** (2016), 20150202. https://doi.org/10.1098/rsta.2015.0202
2. R. O. Duda, P. E. Hart, *Pattern Classification*, John Wiley & Sons, 2006.
3. A. Gersho, R. M. Gray, *Vector Quantization and Signal Compression*, Springer Science & Business Media, 2012.
4. D. Seung, L. Lee, Algorithms for non-negative matrix factorization, *Adv. Neural Inf. Process. Syst.*, **13** (2001), 556–562.
5. D. Li, S. Zhang, X. Ma, Dynamic module detection in temporal attributed networks of cancers, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **19** (2021), 2219–2230. https://doi.org/10.1109/TCBB.2021.3069441
6. Z. Zhao, Z. Ke, Z. Gou, H. Guo, K. Jiang, R. Zhang, The trade-off between topology and content in community detection: An adaptive encoder–decoder-based nmf approach, *Expert Syst. Appl.*, **209** (2022), 118230. https://doi.org/10.1016/j.eswa.2022.118230
7. N. Yu, M. J. Wu, J. X. Liu, C. H. Zheng, Y. Xu, Correntropy-based hypergraph regularized nmf for clustering and feature selection on multi-cancer integrated data, *IEEE Trans. Cybern.*, **51** (2020), 3952–3963. https://doi.org/10.1109/TCYB.2020.3000799
8. N. Yu, Y. L. Gao, J. X. Liu, J. Wang, J. Shang, Hypergraph regularized nmf by l 2, 1-norm for clustering and com-abnormal expression genes selection, in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, (2018), 578–582.
9. M. Venkatasubramanian, K. Chetal, D. J. Schnell, G. Atluri, N. Salomonis, Resolving single-cell heterogeneity from hundreds of thousands of cells through sequential hybrid clustering and nmf, *Bioinformatics*, **36** (2020), 3773–3780. https://doi.org/10.1093/bioinformatics/btaa201
10. W. Wu, X. Ma, Network-based structural learning nonnegative matrix factorization algorithm for clustering of scrna-seq data, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **20** (2022), 566–575. https://doi.org/10.1038/s41579-022-00790-1
11. R. Egger, J. Yu, A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts, *Front. Soc.*, **7** (2022).

12. H. Che, J. Wang, Nonnegative matrix factorization algorithm based on a discrete-time projection neural network, *Neural Networks*, **103** (2018), 63–71. https://doi.org/10.1016/j.neunet.2018.03.003

13. H. Che, J. Wang, A. Cichocki, Bicriteria sparse nonnegative matrix factorization via two-timescale duplex neurodynamic optimization, *IEEE Trans. Neural Networks Learn. Syst.*, **2021** (2021).

14. H. Che, J. Wang, A two-timescale duplex neurodynamic approach to mixed-integer optimization, *IEEE Trans. Neural Networks Learn. Syst.*, **32** (2020), 36–48. https://doi.org/10.1109/TNNLS.2020.2973760

15. X. Ma, W. Zhao, W. Wu, Layer-specific modules detection in cancer multi-layer networks, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **2022** (2022).

16. S. Wang, A. Huang, Penalized nonnegative matrix tri-factorization for co-clustering, *Expert Syst. Appl.*, **78** (2017), 64–73.

17. F. Shang, L. Jiao, F. Wang, Graph dual regularization non-negative matrix factorization for co-clustering, *Pattern Recognit.*, **45** (2012), 2237–2250. https://doi.org/10.1016/j.patcog.2011.12.015

18. 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*, (2006), 126–135.

19. J. Li, H. Che, X. Liu, Circuit design and analysis of smoothed l_0 norm approximation for sparse signal reconstruction, *Circuits Syst. Signal Process.*, (2022), 1–25.

20. X. Ju, H. Che, C. Li, X. He, Solving mixed variational inequalities via a proximal neurodynamic network with applications, *Neural Process. Lett.*, **54** (2022), 207–226. https://doi.org/10.1007/s11063-021-10628-1

21. H. Che, J. Wang, A collaborative neurodynamic approach to global and combinatorial optimization, *Neural Networks*, **114** (2019), 15–27. https://doi.org/10.1016/j.neunet.2019.02.002

22. X. Ju, H. Che, C. Li, X. He, G. Feng, Exponential convergence of a proximal projection neural network for mixed variational inequalities and applications, *Neurocomputing*, **454** (2021), 54–64. https://doi.org/10.1016/j.neucom.2021.04.059

23. C. Dai, H. Che, M.-F. Leung, A neurodynamic optimization approach for l 1 minimization with application to compressed image reconstruction, *Int. J. Artif. Intell. Tools*, **30** (2021), 2140007. https://doi.org/10.1142/S0218213021400078

24. H. Che, J. Wang, A. Cichocki, Sparse signal reconstruction via collaborative neurodynamic optimization, *Neural Networks*, **154** (2022), 255–269. https://doi.org/10.1016/j.neunet.2022.07.018

25. H. Che, J. Wang, A. Cichocki, Neurodynamics-based iteratively reweighted convex optimization for sparse signal reconstruction, in *2022 12th International Conference on Information Science and Technology (ICIST)*, *IEEE*, (2022), 45–51.

26. Y. Wang, J. Wang, H. Che, Two-timescale neurodynamic approaches to supervised feature selection based on alternative problem formulations, *Neural Networks*, **142** (2021), 180–191. https://doi.org/10.1016/j.neunet.2021.04.038

27. X. Ju, C. Li, H. Che, X. He, G. Feng, A proximal neurodynamic network with fixed-time convergence for equilibrium problems and its applications, *IEEE Trans. Neural Networks Learn. Syst.*, **2022** (2022).
28. F. Shang, L. Jiao, J. Shi, J. Chai, Robust positive semidefinite l-isomap ensemble, *Pattern Recognit. Lett.*, **32** (2011), 640–649. https://doi.org/10.1016/j.patrec.2010.12.005
29. M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples., *J. Mach. Learn. Res.*, **7** (2006).
30. K. Chen, H. Che, X. Li, M. F. Leung, Graph non-negative matrix factorization with alternative smoothed l_0 regularizations, *Neural Comput. Appl.*, **2022** (2022), 1–15.
31. X. Yang, H. Che, M. F. Leung, C. Liu, Adaptive graph nonnegative matrix factorization with the self-paced regularization, *Appl. Intell.*, **2022** (2022), 1–18.
32. Z. Huang, Y. Wang, X. Ma, Clustering of cancer attributed networks by dynamically and jointly factorizing multi-layer graphs, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **19** (2021), 2737–2748. https://doi.org/10.1137/19M1301746
33. J. B. Tenenbaum, V. d. Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, *Science*, **290** (2000), 2319–2323. https://doi.org/10.1126/science.290.5500.2319
34. M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, *Adv. Neural Inf. Process. Syst.*, **14** (2001).
35. D. Cai, X. He, J. Han, T. S. Huang, Graph regularized nonnegative matrix factorization for data representation, *IEEE Trans. Pattern Anal. Mach. Intell.*, **33** (2010), 1548–1560.
36. D. Zhou, J. Huang, B. Schölkopf, Learning with hypergraphs: Clustering, classification, and embedding, *Adv. Neural Inf. Process. Syst.*, **19** (2006).
37. J. Yu, D. Tao, M. Wang, Adaptive hypergraph learning and its application in image classification, *IEEE Trans. Image Process.*, **21** (2012), 3262–3272.
38. P. Zhou, X. Wang, L. Du, X. Li, Clustering ensemble via structured hypergraph learning, *Inf. Fusion*, **78** (2022), 171–179. https://doi.org/10.1016/j.inffus.2021.09.003
39. L. Xia, C. Huang, Y. Xu, J. Zhao, D. Yin, J. Huang, Hypergraph contrastive collaborative filtering, in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, (2022), 70–79.
40. Y. Feng, H. You, Z. Zhang, R. Ji, Y. Gao, Hypergraph neural networks, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **33** (2019), 3558–3565. https://doi.org/10.1609/aaai.v33i01.33013558
41. J. Jiang, Y. Wei, Y. Feng, J. Cao, Y. Gao, Dynamic hypergraph neural networks, in *International Joint Conference on Artificial Intelligence*, (2019), 2635–2641.
42. X. Liao, Y. Xu, H. Ling, Hypergraph neural networks for hypergraph matching, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 1266–1275.
43. K. Zeng, J. Yu, C. Li, J. You, T. Jin, Image clustering by hyper-graph regularized non-negative matrix factorization, *Neurocomputing*, **138** (2014), 209–217. https://doi.org/10.1016/j.neucom.2014.01.043

44. L. Du, X. Li, Y. D. Shen, Robust nonnegative matrix factorization via half-quadratic minimization, in *2012 IEEE 12th International Conference on Data Mining*, (2012), 201–210.

45. D. Kong, C. Ding, H. Huang, Robust nonnegative matrix factorization using l21-norm, in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, (2011), 673–682. https://doi.org/10.3917/ag.682.0673

46. H. Gao, F. Nie, W. Cai, H. Huang, Robust capped norm nonnegative matrix factorization: Capped norm nmf, in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, **2015** 2015, 871–880.

47. Z. Li, J. Tang, X. He, Robust structured nonnegative matrix factorization for image representation, *IEEE Trans. Neural Networks Learn. Syst.*, **29** (2017), 1947–1960. https://doi.org/10.1109/TNNLS.2017.2691725

48. N. Guan, T. Liu, Y. Zhang, D. Tao, L. S. Davis, Truncated cauchy non-negative matrix factorization, *IEEE Trans. Pattern Anal. Mach. Intell.*, **41** (2017), 246–259.

49. N. Guan, D. Tao, Z. Luo, J. Shawe-Taylor, Mahnmf: Manhattan non-negative matrix factorization, *Statics*, **1050** (2012), 14.

50. S. Peng, W. Ser, B. Chen, Z. Lin, Robust orthogonal nonnegative matrix tri-factorization for data representation, *Knowl. Based Syst.*, **201** (2020), 106054. https://doi.org/10.1016/j.knosys.2020.106054

51. C. Y. Wang, N. Yu, M. J. Wu, Y. L. Gao, J. X. Liu, J. Wang, Dual hyper-graph regularized supervised nmf for selecting differentially expressed genes and tumor classification, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **18** (2020), 2375–2383. https://doi.org/10.1109/TCBB.2020.2975173

52. L. Lovász, M. D. Plummer, *Matching theory*, American Mathematical Society, 2009. https://doi.org/10.1090/chel/367

53. X. Gao, X. Ma, W. Zhang, J. Huang, H. Li, Y. Li, J. Cui, Multi-view clustering with self-representation and structural constraint, *IEEE Trans. Big Data*, **8** (2021), 882–893. https://doi.org/10.1109/TBDATA.2021.3128906

54. C. Liu, W. Cao, S. Wu, W. Shen, D. Jiang, Z. Yu, H.-S. Wong, Supervised graph clustering for cancer subtyping based on survival analysis and integration of multi-omic tumor data, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **19** (2020), 1193–1202.

55. C. Liu, S. Wu, R. Li, D. Jiang, H. S. Wong, Self-supervised graph completion for incomplete multi-view clustering, *IEEE Trans. Knowl. Data Eng.*, **2023** (2023), forthcoming.

56. C. Liu, R. Li, S. Wu, H. Che, D. Jiang, Z. Yu, H.-S. Wong, Self-guided partial graph propagation for incomplete multiview clustering, *IEEE Trans. Neural Networks Learn. Syst.*, **2023** (2023).

57. C. Li, H. Che, M. F. Leung, C. Liu, Z. Yan, Robust multi-view non-negative matrix factorization with adaptive graph and diversity constraints, *Inf. Sci.*, **2023** (2023).

58. B. Pan, C. Li, H. Che, Nonconvex low-rank tensor approximation with graph and consistent regularizations for multi-view subspace learning, *Neural Networks*, **161** (2023), 638–658.

59. S. Wang, Z. Chen, S. Du, Z. Lin, Learning deep sparse regularizers with applications to multi-view clustering and semi-supervised classification, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2021), 5042–5055.

60. S. Du, Z. Liu, Z. Chen, W. Yang, S. Wang, Differentiable bi-sparse multi-view co-clustering, *IEEE Trans. Signal Process.*, **69** (2021), 4623–4636. https://doi.org/10.1109/TSP.2021.3101979

61. S. Wang, X. Lin, Z. Fang, S. Du, G. Xiao, Contrastive consensus graph learning for multi-view clustering, *IEEE/CAA J. Autom. Sin.*, **9** (2022), 2027–2030. https://doi.org/10.1109/JAS.2022.105959

62. Z. Fang, S. Du, X. Lin, J. Yang, S. Wang, Y. Shi, Dbo-net: Differentiable bi-level optimization network for multi-view clustering, *Inf. Sci.*, **2023** (2023).

AIMS Press