Neighborhood Preserving Nonnegative Matrix Factorization

Quanquan Gu
gqq03@mails.tsinghua.edu.cn
Jie Zhou
jzhou@tsinghua.edu.cn

State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology (TNList)
Department of Automation, Tsinghua University, Beijing 100084, China

Abstract

Nonnegative Matrix Factorization (NMF) has been widely used in computer vision and pattern recognition. It aims to find two nonnegative matrices whose product can well approximate the nonnegative data matrix, which naturally leads to parts-based and non-subtractive representation. In this paper, we present a neighborhood preserving nonnegative matrix factorization (NPNMF) for dimensionality reduction. It imposes an additional constraint on NMF that each data point can be represented as a linear combination of its neighbors. This constraint preserves the local geometric structure, and is good at dimensionality reduction on manifold. An iterative multiplicative updating algorithm is proposed to optimize the objective, and its convergence is guaranteed theoretically. Experiments on benchmark face recognition data sets demonstrate that the proposed method outperforms NMF as well as many state of the art dimensionality reduction methods.

1 Introduction

Nonnegative Matrix Factorization (NMF) [9] has been widely used in computer vision and pattern recognition. It aims to find two nonnegative matrices whose product can well approximate the nonnegative data matrix, which naturally leads to parts-based and non-subtractive representation. Recent years, many variants of NMF have been proposed. [10] proposed a local NMF (LNMF) which imposes a spatially localized constraint on the bases. [8] proposed a NMF with sparseness constraint. [5] proposed a semi-NMF approach which relaxes the nonnegative constraint on the base matrix. All the methods mentioned above are unsupervised, while [14] and [15] independently proposed a discriminative NMF (DNMF), which adds an additional constraint seeking to maximize the between-class scatter and minimize the within-class scatter in the subspace spanned by the bases.

Recent studies have shown that many real world data are actually sampled from a nonlinear low dimensional manifold which is embedded in the high dimensional ambient space [12] [11]. Yet NMF does not exploit the geometric structure of the data. In other word, it assumes that the data points are sampled from a Euclidean space. This greatly limits the application of NMF for the data lying on manifold. To address this problem, [3] proposed a

BMVC 2009 doi:10.5244/C.23.9

graph regularized NMF (GNMF), which assumes that the nearby data points are likely to be in the same cluster, i.e. *cluster assumption* [4] [6].

In this paper, we present a novel nonnegative matrix factorization method. It is based on the assumption that if a data point can be reconstructed from its neighbors in the input space, then it can be reconstructed from its neighbors by the same reconstruction coefficients in the low dimensional subspace, i.e. *local linear embedding assumption* [12] [7]. This assumption is embodied by a neighborhood preserving regularization, which preserves the local geometric structure. We constrain NMF with neighborhood preserving regularization, resulting in a neighborhood preserving NMF (NPNMF). NPNMF not only inherits the advantages of NMF, e.g. nonnegativity, but also overcomes the shortcomings of NMF, i.e. Euclidean assumption based. We will show that it can be optimized via an iterative multiplicative updating algorithm and its convergence is theoretically guaranteed. Experiments on benchmark face recognition data sets demonstrate that the proposed method outperforms NMF and its variants as well as many other state of the art dimensionality reduction methods.

The remainder of this paper is organized as follows. In Section 2 we briefly review NMF. In Section 3, we present NPNMF, followed with its optimization algorithm along with the proof of the convergence of the proposed algorithm. Experiments on many benchmark face recognition data sets are demonstrated in Section 4. Finally, we draw a conclusion in Section 5.

2 A Review of NMF

In this section, we will briefly review NMF [9]. Given a nonnegative data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}_+^{d \times n}$, each column of $\mathbf{X}$ is a data point. NMF aims to find two nonnegative matrices $\mathbf{U} \in \mathbb{R}_+^{d \times m}$ and $\mathbf{V} \in \mathbb{R}_+^{m \times n}$ which minimize the following objective

$$\begin{aligned} J_{NMF} &= ||\mathbf{X} - \mathbf{U}\mathbf{V}||_F^2, \\ \text{s.t.} \quad & \mathbf{U} \geq 0, \mathbf{V} \geq 0, \end{aligned} \tag{1}$$

where $||\cdot||_F$ is Frobenius norm. To optimize the objective, [9] presented an iterative multiplicative updating algorithm as follows

$$\begin{aligned} \mathbf{U}_{ij} &\leftarrow \mathbf{U}_{ij} \frac{(\mathbf{X}\mathbf{V}^T)_{ij}}{(\mathbf{U}\mathbf{V}\mathbf{V}^T)_{ij}} \\ \mathbf{V}_{ij} &\leftarrow \mathbf{V}_{ij} \frac{(\mathbf{U}^T\mathbf{X})_{ij}}{(\mathbf{U}^T\mathbf{U}\mathbf{V})_{ij}} \end{aligned} \tag{2}$$

In the rest of this paper, we denote $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ where $\mathbf{v}_i \in \mathbb{R}_+^m$ is the i-th column vector of $\mathbf{V}$.

3 The Proposed Method

In this section, we first introduce neighborhood preserving regularization. Then we will present neighborhood preserving nonnegative matrix factorization, followed with its optimization algorithm. The convergence of the proposed algorithm is also proved.

3.1 Neighborhood Preserving Regularization

Recent studies have shown that many real world data are actually sampled from a nonlinear low dimensional manifold which is embedded in the high dimensional ambient space [12] [11]. In order to consider the geometric structure in the data, we assume that if a data point can be reconstructed from its neighbors in the input space, then it can be reconstructed from its neighbors by the same reconstruction coefficients in the low dimensional subspace, i.e. *local linear embedding assumption* [12] [2].

For each data point $\mathbf{x}_i$, we use $\mathcal{N}_k(\mathbf{x}_i)$ to denote its k-nearest neighborhood. And we characterize the local geometric structure of its neighborhood by the linear coefficients that reconstruct $\mathbf{x}_i$ from its neighbors, i.e. $\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)$. The reconstruction coefficients are computed by the following objective function

$$\begin{aligned} \min \quad & ||\mathbf{x}_i - \sum_{\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)} M_{ij}\mathbf{x}_j||^2, \\ \text{s.t.} \quad & \sum_{\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)} M_{ij} = 1 \end{aligned} \tag{3}$$

And $M_{ij} = 0$ if $\mathbf{x}_j \notin \mathcal{N}_k(\mathbf{x}_i)$.

Then $\mathbf{v}_i, 1 \le i \le n$ in the low dimensional subspace can be reconstructed by minimizing

$$\begin{aligned} & \sum_i ||\mathbf{v}_i - \sum_{\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)} M_{ij}\mathbf{v}_j||^2 \\ = \quad & \mathrm{tr}(\mathbf{V}(\mathbf{I}-\mathbf{M})(\mathbf{I}-\mathbf{M})\mathbf{V}^T) \\ = \quad & \mathrm{tr}(\mathbf{V}\mathbf{L}\mathbf{V}^T) \end{aligned} \tag{4}$$

where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is identity matrix and $\mathbf{L} = (\mathbf{I}-\mathbf{M})(\mathbf{I}-\mathbf{M})$. Eq.(4) is called *Neighborhood Preserving Regularization*. The better each point is reconstructed from its neighborhood in the low dimensional subspace, the smaller the neighborhood preserving regularizer will be.

3.2 Neighborhood Preserving NMF

Our assumption is that each point can be reconstructed by the data points in its neighborhood. To apply this idea for NMF, we constrain NMF in Eq.(1) with neighborhood preserving regularization in Eq.(4) as follows

$$\begin{aligned} J_{NPNMF} \quad = \quad & ||\mathbf{X}-\mathbf{U}\mathbf{V}||_F^2 + \mu\mathrm{tr}(\mathbf{V}\mathbf{L}\mathbf{V}^T), \\ \text{s.t.} \quad & \mathbf{U} \ge 0, \mathbf{V} \ge 0, \end{aligned} \tag{5}$$

where μ is a positive regularization parameter controlling the contribution of the additional constraint. We call Eq.(5) *Neighborhood Preserving Nonnegative Matrix Factorization* (NPNMF). Let $\mu = 0$, Eq.(5) degenerates to the original NMF. To make the objective in Eq.(5) lower bounded, we use L_2 normalization on rows of $\mathbf{V}$ in the optimization, and compensate the norms of $\mathbf{V}$ to $\mathbf{U}$.

In the following, we will give the solution to Eq.(5).

Since $\mathbf{U} \ge 0$, $\mathbf{V} \ge 0$, we introduce the Lagrangian multiplier $\gamma \in \mathbb{R}^{d\times m}$ and $\eta \in \mathbb{R}^{m\times n}$, thus, the Lagrangian function is

$$L(\mathbf{U},\mathbf{V}) = ||\mathbf{X}-\mathbf{U}\mathbf{V}||_F^2 + \mu\mathrm{tr}(\mathbf{V}\mathbf{L}\mathbf{V}^T) - \mathrm{tr}(\gamma\mathbf{U}^T) - \mathrm{tr}(\eta\mathbf{V}^T) \tag{6}$$

Setting $\frac{\partial L(\mathbf{U},\mathbf{V})}{\partial \mathbf{U}} = 0$ and $\frac{\partial L(\mathbf{U},\mathbf{V})}{\partial \mathbf{V}} = 0$, we obtain

$$\begin{aligned} \gamma &= -2\mathbf{X}\mathbf{V}^T + 2\mathbf{U}\mathbf{V}\mathbf{V}^T \\ \eta &= -2\mathbf{U}^T\mathbf{X} + 2\mathbf{U}^T\mathbf{U}\mathbf{V} + 2\mu\mathbf{V}\mathbf{L} \end{aligned} \tag{7}$$

Using the Karush-Kuhn-Tucker condition [2] $\gamma_{ij}\mathbf{U}_{ij} = 0$ and $\eta_{ij}\mathbf{V}_{ij} = 0$, we get

$$\begin{aligned} (-\mathbf{X}\mathbf{V}^T + \mathbf{U}\mathbf{V}\mathbf{V}^T)_{ij}\mathbf{U}_{ij} &= 0 \\ (-\mathbf{U}^T\mathbf{X} + \mathbf{U}^T\mathbf{U}\mathbf{V} + \mu\mathbf{V}\mathbf{L})_{ij}\mathbf{V}_{ij} &= 0 \end{aligned} \tag{8}$$

Introduce

$$\mathbf{L} = \mathbf{L}^+ - \mathbf{L}^- \tag{9}$$

where $\mathbf{L}^+_{ij} = (|\mathbf{L}_{ij}| + \mathbf{L}_{ij})/2$ and $\mathbf{L}^-_{ij} = (|\mathbf{L}_{ij}| - \mathbf{L}_{ij})/2$.

Substitute Eq.(9) into Eq.(8), we obtain

$$\begin{aligned} (-\mathbf{X}\mathbf{V}^T + \mathbf{U}\mathbf{V}\mathbf{V}^T)_{ij}\mathbf{U}_{ij} &= 0 \\ (-\mathbf{U}^T\mathbf{X} + \mathbf{U}^T\mathbf{U}\mathbf{V} + \mu\mathbf{V}\mathbf{L}^+ - \mu\mathbf{V}\mathbf{L}^-)_{ij}\mathbf{V}_{ij} &= 0 \end{aligned} \tag{10}$$

Eq.(10) leads to the following updating formula

$$\begin{aligned} \mathbf{U}_{ij} &\leftarrow \mathbf{U}_{ij}\sqrt{\frac{(\mathbf{X}\mathbf{V}^T)_{ij}}{(\mathbf{U}\mathbf{V}\mathbf{V}^T)_{ij}}} \\ \mathbf{V}_{ij} &\leftarrow \mathbf{V}_{ij}\sqrt{\frac{(\mathbf{U}^T\mathbf{X} + \mu\mathbf{V}\mathbf{L}^-)_{ij}}{(\mathbf{U}^T\mathbf{U}\mathbf{V} + \mu\mathbf{V}\mathbf{L}^+)_{ij}}} \end{aligned} \tag{11}$$

3.3 Convergence Analysis

In this section, we will investigate the convergence of the updating formula in Eq.(11). We use the auxiliary function approach [9] to prove the convergence. Here we first introduce the definition of auxiliary function [9].

Definition 3.1. *[9] $Z(h,h')$ is an auxiliary function for $F(h)$ if the conditions*

$$Z(h,h') \geq F(h), Z(h,h) = F(h),$$

are satisfied.

Lemma 3.2. *[9] If Z is an auxiliary function for F, then F is non-increasing under the update*

$$h^{(t+1)} = \arg\min_h Z(h,h^{(t)})$$

Proof. $F(h^{(t+1)}) \leq Z(h^{(t+1)},h^{(t)}) \leq Z(h^{(t)},h^{(t)}) = F(h^{(t)})$ □

Lemma 3.3. *[5] For any nonnegative matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{k\times k}$, $\mathbf{S} \in \mathbb{R}^{n\times k}$,$\mathbf{S}' \in \mathbb{R}^{n\times k}$, and $\mathbf{A}$, $\mathbf{B}$ are symmetric, then the following inequality holds*

$$\sum_{i=1}^{n}\sum_{p=1}^{k}\frac{(\mathbf{A}\mathbf{S}'\mathbf{B})_{ip}\mathbf{S}^2_{ip}}{\mathbf{S}'_{ip}} \geq tr(\mathbf{S}^T\mathbf{A}\mathbf{S}\mathbf{B})$$

Theorem 3.4. *Let*

$$J(\mathbf{U}) = tr(-2\mathbf{X}^T\mathbf{U}\mathbf{V} + \mathbf{V}^T\mathbf{U}^T\mathbf{U}\mathbf{V}) \tag{12}$$

Then the following function

$$Z(\mathbf{U},\mathbf{U}') = -2\sum_{ij}(\mathbf{X}\mathbf{V}^T)_{ij}\mathbf{U}'_{ij}(1+\log\frac{\mathbf{U}_{ij}}{\mathbf{U}'_{ij}}) + \sum_{ij}\frac{(\mathbf{U}'\mathbf{V}\mathbf{V}^T)_{ij}\mathbf{U}^2_{ij}}{\mathbf{U}'_{ij}}$$

is an auxiliary function for $J(\mathbf{U})$. Furthermore, it is a convex function in $\mathbf{U}$ and its global minimum is

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\sqrt{\frac{(\mathbf{X}\mathbf{V}^T)_{ij}}{(\mathbf{U}\mathbf{V}\mathbf{V}^T)_{ij}}} \tag{13}$$

Proof. See Appendix A □

Theorem 3.5. *Updating $\mathbf{U}$ using Eq.(11) will monotonically decrease the value of the objective in Eq.(5), hence it converges.*

Proof. By Lemma 3.2 and Theorem 3.4, we can get that $J(\mathbf{U}^0) = Z(\mathbf{U}^0,\mathbf{U}^0) \geq Z(\mathbf{U}^1,\mathbf{U}^0) \geq J(\mathbf{U}^1) \geq \ldots$ So $J(\mathbf{U})$ is monotonically decreasing. Since $J(\mathbf{U})$ is obviously bounded below, we prove this theorem. □

Theorem 3.6. *Let*

$$J(\mathbf{V}) = tr(-2\mathbf{X}^T\mathbf{U}\mathbf{V} + \mathbf{V}^T\mathbf{U}^T\mathbf{U}\mathbf{V} - \mu\mathbf{V}\mathbf{L}\mathbf{V}^T) \tag{14}$$

Then the following function

$$\begin{aligned} Z(\mathbf{V},\mathbf{V}') &= \sum_{ij}\frac{(\mathbf{U}^T\mathbf{U}\mathbf{V}')_{ij}\mathbf{V}^2_{ij}}{\mathbf{V}'_{ij}} + \mu\sum_{ij}\frac{(\mathbf{V}'\mathbf{L}^-)_{ij}\mathbf{V}^2_{ij}}{\mathbf{V}'_{ij}} \\ &- \sum_{ij}(\mathbf{U}^T\mathbf{X})_{ij}\mathbf{V}'_{ij}(1+\log\frac{\mathbf{V}_{ij}}{\mathbf{V}'_{ij}}) - \mu\sum_{ijk}\mathbf{L}^+_{jk}\mathbf{V}'_{ij}\mathbf{V}'_{ik}(1+\log\frac{\mathbf{V}_{ij}\mathbf{V}_{ik}}{\mathbf{V}'_{ij}\mathbf{V}'_{ik}}) \end{aligned}$$

is an auxiliary function for $J(\mathbf{V})$. Furthermore, it is a convex function in $\mathbf{V}$ and its global minimum is

$$\mathbf{V}_{ij} = \mathbf{V}_{ij}\sqrt{\frac{(\mathbf{U}^T\mathbf{X}+\mu\mathbf{V}\mathbf{L}^+)_{ij}}{(\mathbf{U}^T\mathbf{U}\mathbf{V}+\mu\mathbf{V}\mathbf{L}^-)_{ij}}} \tag{15}$$

Proof. See Appendix B □

Theorem 3.7. *Updating $\mathbf{V}$ using Eq.(11) will monotonically decrease the value of the objective in Eq.(5), hence it converges.*

Proof. By Lemma 3.2 and Theorem 3.6, we can get that $J(\mathbf{V}^0) = Z(\mathbf{V}^0,\mathbf{V}^0) \geq Z(\mathbf{V}^1,\mathbf{V}^0) \geq J(\mathbf{V}^1) \geq \ldots$ So $J(\mathbf{V})$ is monotonically decreasing. Since $J(\mathbf{V})$ is obviously bounded below, we prove this theorem. □

4 Experiments

In this section, we evaluate the performance of the proposed method. We compare our method with Principal Component Analysis (PCA) [1], Linear Discriminant Analysis (LDA) [1], NMF [9], LNMF [10], DNMF [14] [15] and Neighborhood Preserving Embedding (NPE) [7]. We use nearest neighbor (NN) classifier as baseline.

4.1 Data Sets

In our experiments, we use three standard face recognition data sets which are widely used as benchmark data sets in dimensionality reduction literature.

The ORL face database[1]. There are ten images for each of the 40 human subjects, which were taken at different times, varying the lightings, facial expressions and facial details. The original images (with 256 gray levels) have size 92×112, which are resized to 32×32 for efficiency;

The Yale face database[2]. It contains 11 gray scale images for each of the 15 individuals. The images demonstrate variations in lighting condition, facial expression and with/without glasses. In our experiments, the images were also resized to 32×32;

The CMU PIE face database [13]. It contains 68 individuals with 41368 face images as a whole. The face images were captured by 13 synchronized cameras and 21 flashes, under varying poses, illuminations and expressions. In our experiments, one near frontal pose ($C27$) is selected under different illuminations, lightings and expressions which leaves us about 49 near frontal face images for each individual, and all the images were also resized to 32×32.

Figure 1 shows some sample images from ORL, Yale and PIE data sets.

Figure 1: Some sample images. The top row is from ORL data set, the middle row is from Yale data set, and the bottom row is from PIE data set.

4.2 Parameter Settings

For each data set, we randomly divide it into training and testing sets, and evaluate the recognition accuracy on the testing set. In detail, for each individual in the ORL and Yale data sets, $p = 2,3,4$ images were randomly selected as training samples, and the rest were used for testing, while for each individual in the PIE data set, $p = 5,10,20$ images were randomly selected as training samples[3]. The training set was used to learn a subspace, and the recog-

[1] http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data

[2] http://cvc.yale.edu/projects/yalefaces/yalefaces.html

[3] As we know, face recognition with small training samples is more challenging, so we use only about $10\% - 40\%$ samples of each data set for training.

nition was performed in the subspace by NN classifier. Since the training set was randomly chosen, we repeated each experiment 20 times and calculated the average recognition accuracy. In general, the recognition rate varies with the dimensionality of the subspace. The best average performance obtained as well as the corresponding dimensionality is reported.

For LDA, as in [1], we first use PCA to reduce the dimensionality to $n-c$ and then perform LDA (Note that in PIE data set with $p = 20$, $n-c$ is larger than the original dimensionality d, hence we perform LDA directly without PCA in this case). The regularization parameter μ is set by searching the grid $\{0.01, 0.1, 1, 10, 100\}$. The neighborhood size k in NPE and NPNMF is set by searching the grid $\{1, 2, 3, 4, 5, \ldots, 10\}$.

For NMF, LNMF, DNMF and NPNMF, the projection is computed as $\mathbf{U}^\dagger = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T$, except for Yale data set, in which we use $\mathbf{U}^T$ instead of $\mathbf{U}^\dagger$ because the classification capability of $\mathbf{U}^T$ is much better than that of $\mathbf{U}^\dagger$ on this data set.

4.3 Classification Results

Table 1, 2 and 3 show the experimental results of all the methods on the three data sets respectively, where the value in each entry represents the average recognition accuracy of 20 independent trials, and the number in brackets is the corresponding projection dimensionality.

Table 1: Face Recognition accuracy on the ORL data set. The number in brackets is the corresponding projection dimensionality.

Method	2 Train	3 Train	4 Train
Baseline	70.67	78.88	84.12
PCA	70.67(79)	78.88(118)	84.21(152)
LDA	72.80(25)	83.79(39)	90.13(39)
NPE	73.19(36)	84.29(54)	91.06(73)
NMF	70.87(97)	78.98(81)	84.48(95)
LNMF	71.73(178)	81.09(168)	86.31(195)
DNMF	74.00(75)	83.32(84)	88.10(74)
NPNMF	**75.31(200)**	**84.73(94)**	**91.35(81)**

Table 2: Face Recognition accuracy on the Yale data set. The number in brackets is the corresponding projection dimensionality.

Method	2 Train	3 Train	4 Train
Baseline	46.04	49.96	55.62
PCA	46.04(29)	49.96(44)	55.67(58)
LDA	42.81(11)	60.33(14)	68.10(13)
NPE	48.19(13)	62.00(19)	69.00(68)
NMF	44.11(112)	49.00(195)	52.19(164)
LNMF	44.00(157)	48.83(198)	53.57(197)
DNMF	48.15(161)	60.50(169)	66.67 (102)
NPNMF	**50.36(119)**	**62.62(137)**	**70.33(151)**

We can see that our method outperforms other dimensionality reduction methods on all the three data sets. The superiority of our method may arise in the following two aspects: (1) *local linear embedding assumption* [12] [7], which preserves the local geometric structure

Table 3: Face Recognition accuracy on the PIE data set. The number in brackets is the corresponding projection dimensionality.

Method	5 Train	10 Train	20 Train
Baseline	43.02	62.90	83.19
PCA	42.87(199)	62.51(195)	82.84(200)
LDA	83.39(67)	90.47(67)	93.98(67)
NPE	84.71(166)	91.48(200)	94.33(200)
NMF	78.66(200)	88.98(200)	95.52(200)
LNMF	76.47(200)	87.91(200)	95.61(196)
DNMF	80.51(200)	90.85(200)	96.40(191)
NPNMF	**85.02(200)**	**91.97(198)**	**96.46(182)**

of the data. (2) the *nonnegativity*, inheriting from NMF, which is suitable for nonnegative data, e.g. image data.

5 Conclusion

In this paper, we present a neighborhood preserving nonnegative matrix factorization (NPNMF) for dimensionality reduction, which preserves the local geometric structure. We show that it can be optimized by an iterative multiplicative updating algorithm. The convergence of the algorithm is proved theoretically. Experiments on many benchmark face recognition data sets demonstrate that the proposed method outperforms NMF as well as many state of the art dimensionality reduction methods.

A Proof of Theorem 3.4

Proof. We rewrite Eq.(12) as

$$L(\mathbf{U}) = \mathrm{tr}(-2\mathbf{V}\mathbf{X}^T\mathbf{U}+\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T) \tag{16}$$

By applying Lemma 3.3, we have

$$\mathrm{tr}(\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T) \le \sum_{ij}\frac{(\mathbf{U}'\mathbf{V}\mathbf{V}^T)_{ij}\mathbf{U}_{ij}^2}{\mathbf{U}'_{ij}}$$

To obtain the lower bound for the remaining terms, we use the inequality that

$$z \ge 1+\log z, \forall z>0 \tag{17}$$

Then

$$\mathrm{tr}(\mathbf{V}\mathbf{X}^T\mathbf{U}) \ge \sum_{ij}(\mathbf{X}\mathbf{V}^T)_{ij}\mathbf{U}'_{ij}(1+\log\frac{\mathbf{U}_{ij}}{\mathbf{U}'_{ij}})$$

By summing over all the bounds, we can get $\mathbf{Z}(\mathbf{U},\mathbf{U}')$, which obviously satisfies (1) $\mathbf{Z}(\mathbf{U},\mathbf{U}') \ge J_{NPNMF}(\mathbf{U})$; (2)$\mathbf{Z}(\mathbf{U},\mathbf{U}) = J_{NPNMF}(\mathbf{U})$

To find the minimum of $\mathbf{Z}(\mathbf{U},\mathbf{U}')$, we take the Hessian matrix of $Z(\mathbf{U},\mathbf{U}')$

$$\frac{\partial^2 Z(\mathbf{U},\mathbf{U}')}{\partial \mathbf{U}_{ij}\partial \mathbf{U}_{kl}} = \delta_{ik}\delta_{jl}\left(\frac{2(\mathbf{U}'\mathbf{V}\mathbf{V}^T)_{ij}}{\mathbf{U}'_{ij}} + 2(\mathbf{X}\mathbf{V}^T)_{ij}\frac{\mathbf{U}'_{ij}}{\mathbf{U}^2_{ij}}\right)$$

which is a diagonal matrix with positive diagonal elements. So $Z(\mathbf{U},\mathbf{U}')$ is a convex function of $\mathbf{U}$, and we can obtain the global minimum of $Z(\mathbf{U},\mathbf{U}')$ by setting $\frac{\partial Z(\mathbf{U},\mathbf{U}')}{\partial \mathbf{U}_{ij}} = 0$ and solving for $\mathbf{U}$, from which we can get Eq.(13). □

B Proof of Theorem 3.6

Proof. We rewrite Eq.(14) as

$$L(\mathbf{V}) = \mathrm{tr}(-2\mathbf{X}^T\mathbf{U}\mathbf{V} + \mathbf{V}^T\mathbf{U}^T\mathbf{U}\mathbf{V} - \mu\mathbf{V}\mathbf{L}^+\mathbf{V}^T + \mu\mathbf{V}\mathbf{L}^-\mathbf{V}^T) \tag{18}$$

By applying Lemma 3.3, we have

$$\begin{aligned}\mathrm{tr}(\mathbf{V}^T\mathbf{U}^T\mathbf{U}\mathbf{V}) &\le \sum_{ij}\frac{(\mathbf{U}^T\mathbf{U}\mathbf{V}')_{ij}\mathbf{V}^2_{ij}}{\mathbf{V}'_{ij}}\\ \mathrm{tr}(\mathbf{V}\mathbf{L}^-\mathbf{V}^T) &\le \sum_{ij}\frac{(\mathbf{V}'\mathbf{L}^-)_{ij}\mathbf{V}^2_{ij}}{\mathbf{V}'_{ij}}\end{aligned}$$

By the inequality in Eq.(17), we have

$$\begin{aligned}\mathrm{tr}(\mathbf{X}^T\mathbf{U}\mathbf{V}) &\ge \sum_{ij}(\mathbf{U}^T\mathbf{X})_{ij}\mathbf{V}'_{ij}(1+\log\frac{\mathbf{V}_{ij}}{\mathbf{V}'_{ij}})\\ \mathrm{tr}(\mathbf{V}\mathbf{L}^+\mathbf{V}^T) &\ge \sum_{ijk}\mathbf{L}^+_{jk}\mathbf{V}'_{ij}\mathbf{V}'_{ik}(1+\log\frac{\mathbf{V}_{ij}\mathbf{V}_{ik}}{\mathbf{V}'_{ij}\mathbf{V}'_{ik}})\end{aligned}$$

By summing over all the bounds, we can get $\mathbf{Z}(\mathbf{V},\mathbf{V}')$, which obviously satisfies (1) $\mathbf{Z}(\mathbf{V},\mathbf{V}') \ge J_{NPNMF}(\mathbf{V})$; (2)$\mathbf{Z}(\mathbf{V},\mathbf{V}) = J_{NPNMF}(\mathbf{V})$

To find the minimum of $\mathbf{Z}(\mathbf{V},\mathbf{V}')$, we take the Hessian matrix of $Z(\mathbf{V},\mathbf{V}')$

$$\frac{\partial^2 Z(\mathbf{V},\mathbf{V}')}{\partial \mathbf{V}_{ij}\partial \mathbf{V}_{kl}} = \delta_{ik}\delta_{jl}\left(\frac{2(\mathbf{U}^T\mathbf{X}+\mu\mathbf{V}'\mathbf{L}^+)_{ij}\mathbf{V}'_{ij}}{\mathbf{V}^2_{ij}} + \frac{2(\mathbf{U}^T\mathbf{U}\mathbf{V}'+\mu\mathbf{V}'\mathbf{L}^-)_{ij}}{\mathbf{V}'_{ij}}\right)$$

which is a diagonal matrix with positive diagonal elements. So $Z(\mathbf{V},\mathbf{V}')$ is a convex function of $\mathbf{V}$, and we can obtain the global minimum of $Z(\mathbf{V},\mathbf{V}')$ by setting $\frac{\partial Z(\mathbf{V},\mathbf{V}')}{\partial \mathbf{V}_{ij}} = 0$ and solving for $\mathbf{V}$, from which we can get Eq.(15). □

References

[1] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):711–720, 1997.

[2] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, 2004.

[3] Deng Cai, Xiaofei He, Xiaoyun Wu, and Jiawei Han. Non-negative matrix factorization on manifold. In *ICDM*, 2008.

[4] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

[5] Chris H.Q. Ding, Tao Li, and Michael I. Jordan. Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99 (1), 2008. ISSN 0162-8828.

[6] Xiaofei He and Partha Niyogi. Locality preserving projections. In *NIPS*, 2003.

[7] Xiaofei He, Deng Cai, Shuicheng Yan, and HongJiang Zhang. Neighborhood preserving embedding. In *ICCV*, pages 1208–1213, 2005.

[8] Patrik O. Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

[9] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[10] Stan Z. Li, XinWen Hou, HongJiang Zhang, and QianSheng Cheng. Learning spatially localized, parts-based representation. In *CVPR (1)*, pages 207–212, 2001.

[11] Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2003.

[12] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

[13] Terence Sim, Simon Baker, and Maan Bsat. The cmu pose, illumination, and expression database. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(12):1615–1618, 2003.

[14] Yuan Wang, Yunde Jia, Changbo Hu, and Matthew Turk. Fisher non-negative matrix factorization for learning local features. In *ACCV*, January 2004.

[15] Stefanos Zafeiriou, Anastasios Tefas, Ioan Buciu, and Ioannis Pitas. Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification. *IEEE Transactions on Neural Networks*, 17(3):683–695, 2006.