Local Learning Regularized Nonnegative Matrix Factorization

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

Abstract

Nonnegative Matrix Factorization (NMF) has been widely used in machine learning and data mining. It aims to find two nonnegative matrices whose product can well approximate the nonnegative data matrix, which naturally lead to parts-based representation. In this paper, we present a local learning regularized nonnegative matrix factorization (LLNMF) for clustering. It imposes an additional constraint on NMF that the cluster label of each point can be predicted by the points in its neighborhood. This constraint encodes both the discriminative information and the geometric structure, and is good at clustering data on manifold. An iterative multiplicative updating algorithm is proposed to optimize the objective, and its convergence is guaranteed theoretically. Experiments on many benchmark data sets demonstrate that the proposed method outperforms NMF as well as many state of the art clustering methods.

1 Introduction

Nonnegative Matrix Factorization (NMF) [Lee and Seung, 2000] has been widely used in machine learning and data mining. It aims to find two nonnegative matrices whose product can well approximate the nonnegative data matrix, which naturally lead to parts-based representation. Recent years, many variants of NMF have been proposed. [Li *et al.*, 2001] proposed a local NMF (LNMF) which imposes a spatially localized constraint on the bases. [Hoyer, 2004] proposed a NMF with sparseness constraint. [Ding *et al.*, 2008] proposed a semi-NMF approach which relaxes the nonnegative constraint on the base matrix. [Ding *et al.*, 2006] proposed a nonnegative matrix tri-factorization for co-clustering. All the methods mentioned above are unsupervised, while [Wang *et al.*, 2004] and [Zafeiriou *et al.*, 2006] proposed independently 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 show that many real world data are actually sampled from a nonlinear low dimensional manifold which is embedded in the high dimensional ambient space [Roweis and Saul, 2000] [Niyogi, 2003]. 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, [Cai *et al.*, 2008] proposed a graph regularized NMF (GNMF), which assumes that the nearby data points are likely to be in the same cluster, i.e. *cluster assumption* [Chapelle *et al.*, 2006].

In this paper, we present a novel nonnegative matrix factorization method. It is based on the assumption that the cluster label of each point can be predicted by the data points in its neighborhood, i.e. *local learning assumption*, which is the philosophy of local learning algorithm [Bottou and Vapnik, 1992]. This assumption is embodied by a local learning regularization, which exploits both the discriminative information and the geometric structure. We constrain the NMF with local learning regularization, resulting in a local learning regularized NMF (LLNMF). LLNMF not only inherits the advantages of NMF, e.g. nonnegativity, but also overcomes the shortcomings of NMF, i.e. Euclidean assumption based and does not take into account discriminative information. We will show that it can be optimized via an iterative multiplicative updating algorithm and its convergence is theoretically guaranteed. Experiments on many benchmark data sets demonstrate that the proposed method outperforms NMF and its variants as well as many other state of the art clustering algorithms.

The remainder of this paper is organized as follows. In Section 2 we will briefly review NMF. In Section 3, we present LLNMF, followed with its optimization algorithm along with the proof of the convergence of the proposed algorithm. Experiments on many benchmark 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 [Lee and Seung, 2000]. 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

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

where $||\cdot||_F$ is Frobenius norm. To optimize the objective, [Lee and Seung, 2000] 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 clustering setting of NMF [Xu *et al.*, 2003], $\mathbf{V} \in \mathbb{R}_+^{m\times n}$ is the cluster assignment matrix where m is the prescribed number of clusters. And the cluster label y_i of data point $\mathbf{x}_i$ can be calculated as

$$y_i = \arg_{1\le j\le m} \max \mathbf{V}_{ij} \tag{3}$$

In the rest of this paper, we denote $\mathbf{V} = [\mathbf{v}^1, \ldots, \mathbf{v}^m]^T$ where $\mathbf{v}^l = [v_1^l, \ldots, v_n^l]^T \in \mathbb{R}_+^n, 1 \le l \le m$ is the row vector of $\mathbf{V}$, and v_i^l is the l-th cluster assignment of $\mathbf{x}_i$.

3 The Proposed Method

In this section, we first introduce local learning regularization. Then we will present Local learning Regularized Nonnegative Matrix Factorization, followed with its optimization algorithm. The convergence of the proposed algorithm is also proved.

3.1 Local Learning Regularization

According to [Bottou and Vapnik, 1992], selecting a good predictor f in a global way might not be a good strategy because the function set $f(\mathbf{x})$ may not contain a good predictor for the entire input space. However, it is much easier for the set to contain some functions that are capable of producing good predictions on some specified regions of the input space. Therefore, if we split the whole input space into many local regions, then it is usually more effective to minimize predicting cost for each region. This inspires the work [Wu and Schölkopf, 2006], which adopted supervised learning idea for unsupervised learning problem. In the following, we will introduce how to construct the local predictors, and derive a local learning regularization.

For each data point $\mathbf{x}_i$, we use $\mathcal{N}(\mathbf{x}_i)$ to denote its neighborhood. We further construct predicting function $f_i^l(\mathbf{x}), 1 \le l \le m$ in $\mathcal{N}(\mathbf{x}_i)$, to predict the cluster label of $\{\mathbf{x}_j\}_{\mathbf{x}_j\in\mathcal{N}(\mathbf{x}_i)}$. Note that for $f_i^l(\mathbf{x})$, the superscript l indicates that it is for the l-th cluster, while the subscript i means that it is trained within the neighborhood of $\mathbf{x}_i$.

To fit the predictor $f_i^l(\mathbf{x})$, we use regularized ridge regression [Hastie *et al.*, 2001], which minimize the following loss function

$$J_i^l = \frac{1}{n_i}\sum_{\mathbf{x}_j\in\mathcal{N}(\mathbf{x}_i)} (f_i^l(\mathbf{x}_j) - v_j^l)^2 + \lambda_i ||f_i^l||_I^2. \tag{4}$$

where n_i is the cardinality of $\mathcal{N}(\mathbf{x}_i)$, v_j^l is the l-th cluster assignment of $\mathbf{x}_j$, $||f_i^l||_I$ measures the smoothness of f_i^l with respect to the intrinsic data manifold, $\lambda_i > 0$ is the regularization parameter. In this paper, we assume $\lambda_1 = \ldots = \lambda_n = \lambda$ and $n_1 = n_2 = \ldots = n_n = k$, for simplicity.

According to *Representor Theorem* [Schölkopf and Smola, 2002], we have

$$f_i^l(\mathbf{x}_i) = \sum_{j=1}^{n_i} \beta_{ij}^l K(\mathbf{x}_i, \mathbf{x}_j), \tag{5}$$

where $K : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$ is a positive definite kernel function, and β_{ij}^l are the expansion coefficients. Therefore we have

$$\mathbf{f}_i^l = \mathbf{K}_i\boldsymbol{\beta}_i^l, \tag{6}$$

where $\mathbf{f}_i^l \in \mathbb{R}^{n_i}$ denotes the vector $[f_i^l(\mathbf{x}_j)]^T, \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i)$, $\mathbf{K}_i \in \mathbb{R}^{n_i\times n_i}$ is the kernel matrix defined on the neighborhood of $\mathbf{x}_i$, i.e. $\mathcal{N}(\mathbf{x}_i)$, and $\boldsymbol{\beta}_i^l = [\beta_{i1}, \ldots, \beta_{in_i}]^T \in \mathbb{R}^{n_i\times 1}$ is the expansion coefficient vector. Bringing Eq.(5) back into Eq.(4), we can derive the following loss function

$$J_i^l = \frac{1}{n_i}||\mathbf{K}_i\boldsymbol{\beta}_i^l - \mathbf{v}_i^l||^2 + \lambda(\boldsymbol{\beta}_i^l)^T\mathbf{K}_i\boldsymbol{\beta}_i^l, \tag{7}$$

where $\mathbf{v}_i^l \in \mathbb{R}^{n_i}$ denotes the vector $[v_j^l]^T, \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i)$. By setting $\frac{\partial J_i^l}{\partial \boldsymbol{\beta}_i^l} = 0$, we can get that

$$\boldsymbol{\beta}_i^l = (\mathbf{K}_i + n_i\lambda\mathbf{I})^{-1}\mathbf{v}_i^l. \tag{8}$$

where $\mathbf{I} \in \mathbb{R}^{n_i\times n_i}$ is the identity matrix. Substituting Eq.(8) into Eq.(5), we have

$$f_i^l(\mathbf{x}_i) = \mathbf{k}_i^T(\mathbf{K}_i + n_i\lambda\mathbf{I})^{-1}\mathbf{v}_i^l = \boldsymbol{\alpha}_i^T\mathbf{v}_i^l, \tag{9}$$

where $\mathbf{k}_i \in \mathbb{R}^{n_i}$ denotes the vector $[k(\mathbf{x}_i, \mathbf{x}_j)]^T, \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i)$, and $\boldsymbol{\alpha}_i = [\alpha_{i1}, \ldots, \alpha_{in_i}]^T \in \mathbb{R}^{n_i\times 1}$.

After the local predictors are constructed, we will combine them together by minimizing the sum of their prediction errors

$$J = \sum_{l=1}^{m}\sum_{i=1}^{n} ||f_i^l(\mathbf{x}_i) - v_i^l||^2 \tag{10}$$

Substitute Eq.(9) into Eq.(10), we obtain

$$\begin{aligned} J &= \sum_{l=1}^{m}\sum_{i=1}^{n} ||\mathbf{k}_i^T(\mathbf{K}_i + n_i\lambda\mathbf{I})^{-1}\mathbf{v}_i^l - v_i^l||^2 \\ &= \sum_{l=1}^{m} ||\mathbf{G}\mathbf{v}^l - \mathbf{v}^l||^2 \\ &= \mathrm{tr}(\mathbf{V}(\mathbf{G}-\mathbf{I})(\mathbf{G}-\mathbf{I})\mathbf{V}^T) \\ &= \mathrm{tr}(\mathbf{V}\mathbf{L}\mathbf{V}^T) \end{aligned} \tag{11}$$

where $\mathbf{v}^l = [v_1^l, \ldots, v_n^l]^T \in \mathbb{R}^n$ is the l-th row of $\mathbf{V}$, $\mathbf{I} \in \mathbb{R}^{n\times n}$ is identity matrix, $\mathbf{L} = (\mathbf{G}-\mathbf{I})(\mathbf{G}-\mathbf{I})$ and $\mathbf{G} \in \mathbb{R}^{n\times n}$ is defined as follows

$$G_{ij} = \begin{cases} \alpha_{ij}, & \text{if } \mathbf{x}_j \in \mathcal{N}(\mathbf{x}_i) \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Eq.(11) is called as *Local Learning Regularization*. The better the cluster label of each point is predicted by the data points in its neighborhood, the smaller the local learning regularizer will be.

3.2 Local Learning Regularized NMF

Our assumption is that the cluster label of each point can be predicted by the data points in its neighborhood. To apply this idea for NMF, we constrain NMF in Eq.(1) with local learning regularization in Eq.(11) as follows

$$\begin{aligned} J_{LLNMF} &= ||\mathbf{X}-\mathbf{UV}||_F^2 + \mu \text{tr}(\mathbf{VLV}^T), \\ &\text{s.t.} \quad \mathbf{U} \geq 0, \mathbf{V} \geq 0, \end{aligned} \tag{13}$$

where μ is a positive regularization parameter controlling the contribution of the additional constraint. We call Eq.(13) as *Local Learning Regularized Nonnegative Matrix Factorization* (LLNMF). Letting $\mu = 0$, Eq.(13) degenerates to the original NMF. To make the objective in Eq.(13) 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.(13).

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

$$\begin{aligned} L(\mathbf{U},\mathbf{V}) &= ||\mathbf{X}-\mathbf{UV}||_F^2 + \mu \text{tr}(\mathbf{VLV}^T) \\ &- \text{tr}(\boldsymbol{\gamma}\mathbf{U}^T) - \text{tr}(\boldsymbol{\eta}\mathbf{V}^T) \end{aligned} \tag{14}$$

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

$$\begin{aligned} \boldsymbol{\gamma} &= -2\mathbf{XV}^T + 2\mathbf{UVV}^T \\ \boldsymbol{\eta} &= -2\mathbf{U}^T\mathbf{X} + 2\mathbf{U}^T\mathbf{UV} + 2\mu\mathbf{VL} \end{aligned} \tag{15}$$

Using the Karush-Kuhn-Tucker condition [Boyd and Vandenberghe, 2004] $\boldsymbol{\gamma}_{ij}\mathbf{U}_{ij} = 0$ and $\boldsymbol{\eta}_{ij}\mathbf{V}_{ij} = 0$, we get

$$\begin{aligned} (-\mathbf{XV}^T + \mathbf{UVV}^T)_{ij}\mathbf{U}_{ij} &= 0 \\ (-\mathbf{U}^T\mathbf{X} + \mathbf{U}^T\mathbf{UV} + \mu\mathbf{VL})_{ij}\mathbf{V}_{ij} &= 0 \end{aligned} \tag{16}$$

Introduce

$$\mathbf{L} = \mathbf{L}^+ - \mathbf{L}^- \tag{17}$$

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

Substitute Eq.(17) into Eq.(16), we obtain

$$\begin{aligned} (-\mathbf{XV}^T + \mathbf{UVV}^T)_{ij}\mathbf{U}_{ij} &= 0 \\ (-\mathbf{U}^T\mathbf{X} + \mathbf{U}^T\mathbf{UV} + \mu\mathbf{VL}^+ - \mu\mathbf{VL}^-)_{ij}\mathbf{V}_{ij} &= 0 \end{aligned} \tag{18}$$

Eq.(18) leads to the following updating formula

$$\begin{aligned} \mathbf{U}_{ij} &\leftarrow \mathbf{U}_{ij}\sqrt{\frac{(\mathbf{XV}^T)_{ij}}{(\mathbf{UVV}^T)_{ij}}} \\ \mathbf{V}_{ij} &\leftarrow \mathbf{V}_{ij}\sqrt{\frac{(\mathbf{U}^T\mathbf{X} + \mu\mathbf{VL}^-)_{ij}}{(\mathbf{U}^T\mathbf{UV} + \mu\mathbf{VL}^+)_{ij}}} \end{aligned} \tag{19}$$

3.3 Convergence Analysis

In this section, we will investigate the convergence of the updating formula in Eq.(19). We use the auxiliary function approach [Lee and Seung, 2000] to prove the convergence. Here we first introduce the definition of auxiliary function [Lee and Seung, 2000].

Definition 3.1. *[Lee and Seung, 2000] $Z(h,h')$ 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. *[Lee and Seung, 2000] 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. *[Ding* et al.*, 2008] 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*

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

Theorem 3.4. *Let*

$$J(\mathbf{U}) = tr(-2\mathbf{X}^T\mathbf{UV} + \mathbf{V}^T\mathbf{U}^T\mathbf{UV}) \tag{20}$$

Then the following function

$$\begin{aligned} Z(\mathbf{U},\mathbf{U}') &= -2\sum_{ij}(\mathbf{XV}^T)_{ij}\mathbf{U}'_{ij}(1+\log\frac{\mathbf{U}_{ij}}{\mathbf{U}'_{ij}}) \\ &+ \sum_{ij}\frac{(\mathbf{U}'\mathbf{VV}^T)_{ij}\mathbf{U}_{ij}^2}{\mathbf{U}'_{ij}} \end{aligned}$$

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

$$\mathbf{U}_{ij} = \mathbf{U}_{ij}\sqrt{\frac{(\mathbf{XV}^T)_{ij}}{(\mathbf{UVV}^T)_{ij}}} \tag{21}$$

Proof. See Appendix A □

Theorem 3.5. *Updating $\mathbf{U}$ using Eq.(19) will monotonically decrease the value of the objective in Eq.(13), 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*

$$J(\mathbf{V}) = tr(-2\mathbf{X}^T\mathbf{UV} + \mathbf{V}^T\mathbf{U}^T\mathbf{UV} - \mu\mathbf{VLV}^T) \tag{22}$$

Then the following function

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

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

$$\mathbf{V}_{ij} = \mathbf{V}_{ij}\sqrt{\frac{(\mathbf{U}^T\mathbf{X} + \mu\mathbf{V}\mathbf{L}^+)_{ij}}{(\mathbf{U}^T\mathbf{U}\mathbf{V} + \mu\mathbf{V}\mathbf{L}^-)_{ij}}} \tag{23}$$

Proof. See Appendix B □

Theorem 3.7. *Updating $\mathbf{V}$ using Eq.(19) will monotonically decrease the value of the objective in Eq.(13), hence it converges.*

Proof. By Lemma 3.2 and Theorem 3.6, we can get that $J(\mathbf{V}^0) = Z(\mathbf{V}^0, \mathbf{V}^0) \geq Z(\mathbf{V}^1, \mathbf{V}^0) \geq J(\mathbf{V}^1) \geq \dots$ 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 Kmeans, Normalized Cut (NC) [Shi and Malik, 2000], Local Learning Clustering (LLC) [Wu and Schölkopf, 2006], NMF [Lee and Seung, 2000], Semi-NMF [Ding *et al.*, 2008], Orthogonal Nonnegative Matrix Tri-Factorization (ONMTF) [Ding *et al.*, 2006] and GNMF [Cai *et al.*, 2008].

4.1 Evaluation Metrics

To evaluate the clustering results, we adopt the performance measures used in [Cai *et al.*, 2008]. These performance measures are the standard measures widely used for clustering.

Clustering Accuracy Clustering Accuracy discovers the one-to-one relationship between clusters and classes and measures the extent to which each cluster contained data points from the corresponding class. Clustering Accuracy is defined as follows:

$$Acc = \frac{\sum_{i=1}^{n} \delta(map(r_i), l_i)}{n}, \tag{24}$$

where r_i denotes the cluster label of $\mathbf{x}_i$, and l_i denotes the true class label, n is the total number of documents, $\delta(x, y)$ is the delta function that equals one if $x = y$ and equals zero otherwise, and $map(r_i)$ is the permutation mapping function that maps each cluster label r_i to the equivalent label from the data set.

Normalized Mutual Information The second measure is the Normalized Mutual Information (NMI), which is used for determining the quality of clusters. Given a clustering result, the NMI is estimated by

$$NMI = \frac{\sum_{k=1}^{C}\sum_{m=1}^{C} n_{k,m} \log \frac{n n_{k,m}}{n_k \hat{n}_m}}{\sqrt{(\sum_{k=1}^{C} n_k \log \frac{n_k}{n})(\sum_{m=1}^{C} \hat{n}_m \log \frac{\hat{n}_m}{n})}}, \tag{25}$$

where n_k denotes the number of data contained in the cluster $\mathcal{D}_k (1 \leq k \leq C)$, $\hat{n}_m$ is the number of data belonging to the $\mathcal{L}_m (1 \leq m \leq C)$, and $n_{k,m}$ denotes the number of data that are in the intersection between the cluster $\mathcal{D}_k$ and the class $\mathcal{L}_m$. The larger the NMI is, the better the clustering result will be.

4.2 Data Sets

In our experiment, we use 6 data sets which are widely used as benchmark data sets in clustering literature [Cai *et al.*, 2008] [Ding *et al.*, 2006].

Coil20[1] This data set contains 32×32 gray scale images of 20 3D objects viewed from varying angles. For each object there are 72 images.

PIE The CMU PIE face database [Sim *et al.*, 2003] contains 68 individuals with 41368 face images as a whole. The face images were captured by 13 synchronized cameras and 21 flashes, under varying pose, illumination and expression. All the images were also resized to 32×32.

CSTR This is the data set of the abstracts of technical reports published in the Department of Computer Science at a university. The data set contained 476 abstracts, which were divided into four research areas: Natural Language Processing (NLP), Robotics/Vision, Systems and Theory.

Newsgroup4 The Newsgroup4 data set used in our experiments is selected from the famous 20-newsgroups data set[2]. The topic *rec* containing *autos, motorcycles, baseball* and *hockey* was selected from the version 20news-18828. The Newsgroup4 data set contains 3970 documents.

WebKB4 The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into 7 categories: student, faculty, staff, course, project, department and other, among which student, faculty, course and project are four most populous entity-representing categories.

WebACE The data set contains 2340 documents consisting of news articles from Reuters new service via the Web in October 1997. These documents are divided into 20 classes.

Table.1 summarizes the characteristics of the real world data sets used in this experiment.

Table 1: Description of the real world datasets

Data Sets	#samples	#features	#classes
Coil20	1440	1024	20
PIE	1428	1024	68
CSTR	476	1000	4
Newsgroup4	3970	1000	4
WebKB4	4199	1000	4
WebACE	2340	1000	20

4.3 Parameter Settings

Since many clustering algorithms have one or more parameters to be tuned, under each parameter setting, we repeat clustering 20 times, and the mean result is computed. We report the best mean result for each method to compare with each other. We set the number of clusters equal to the true number of classes for all the data sets and clustering algorithms.

For NC [Shi and Malik, 2000], the scale parameter of Gaussian kernel for constructing adjacency matrix is set by the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$.

[1] http://www1.cs.columbia.edu/CAVE/software/softlib/coil-20.php

[2] http://people.csail.mit.edu/jrennie/20Newsgroups/

For LLC, the neighborhood size for computing the local learning regularization is determined by the grid $\{5, 10, 20, 30, 40, 50, 80\}$ according to [Wu and Schölkopf, 2006]. For the image datasets (e.g. Coil20, PIE), Gaussian kernel is used with the scale parameter tuned the same as in NC, while for the text datasets (e.g. CSTR, Newsgroup4, WebKB4, WebACE), the cosine kernel is adopted.

For ONMTF, the number of word clusters is set to be the same as the number of document clusters, i.e. the true number of classes in our experiment, according to [Ding *et al.*, 2006].

For GNMF, the neighborhood size to construct the graph is set by search the grid $\{1, 2, 3, \ldots, 10\}$ according to [Cai *et al.*, 2008], and the regularization parameter is set by the grid $\{0.1, 1, 10, 100, 500, 1000\}$.

For the proposed algorithm, the neighborhood size k for computing the local learning regularization is determined by the grid $\{5, 10, 20, 30, 40, 50, 80\}$, and the regularization parameter μ is set by search the grid $\{0.1, 1, 10, 100, 500, 1000\}$. Kernel selection is the same as that in LLC.

Note that no parameter selection is needed for Kmeans, NMF and Semi-NMF, given the number of clusters.

4.4 Clustering Results

The clustering results are shown in Table 2 and Table 3. Table 2 shows the clustering accuracy of all the algorithms on all the data sets, while Table 3 shows the normalized mutual information.

We can see that our method outperforms the other clustering methods on all the data sets. The superiority of our method may arise in the following two aspects: (1) the *local learning assumption*, which is usually more powerful than *cluster assumption* [Chapelle *et al.*, 2006] [Cai *et al.*, 2008] for clustering data on manifold. (2) the *nonnegativity*, inheriting from NMF, which is suitable for nonnegative data, e.g. image data and text data.

5 Conclusion

In this paper, we present a local learning regularized nonnegative matrix factorization (LLNMF) for clustering, which considers both the discriminative information and the geometric structure. The convergence of the proposed algorithm is proved theoretically. Experiments on many benchmark data sets demonstrate that the proposed method outperforms NMF as well as many state of the art clustering methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No.60721003, No.60673106 and No.60573062) and the Specialized Research Fund for the Doctoral Program of Higher Education. We thank the anonymous reviewers for their helpful comments.

A Proof of Theorem 3.4

Proof. We rewrite Eq.(20) as

$$L(\mathbf{U}) = \mathrm{tr}(-2\mathbf{V}\mathbf{X}^T\mathbf{U} + \mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T) \quad (26)$$

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

$$z \ge 1 + \log z, \forall z > 0 \quad (27)$$

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) $\mathbf{Z}(\mathbf{U}, \mathbf{U}') \ge J_{LLNMF}(\mathbf{U})$; (2)$\mathbf{Z}(\mathbf{U}, \mathbf{U}) = J_{LLNMF}(\mathbf{U})$

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

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

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.(21). □

B Proof of Theorem 3.6

Proof. We rewrite Eq.(22) as

$$\begin{aligned} L(\mathbf{V}) &= \mathrm{tr}(-2\mathbf{X}^T\mathbf{U}\mathbf{V} + \mathbf{V}^T\mathbf{U}^T\mathbf{U}\mathbf{V} - \mu\mathbf{V}\mathbf{L}^+\mathbf{V}^T \\ &+ \mu\mathbf{V}\mathbf{L}^-\mathbf{V}^T) \end{aligned} \quad (28)$$

By applying Lemma 3.3, we have

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

By the inequality in Eq.(27), we have

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

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

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

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

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

Table 2: Clustering Accuracy of the 8 algorithms on the 6 data sets.

Data Sets	Kmeans	NC	LLC	NMF	SNMF	ONMTF	GNMF	LLNMF
Coil20	0.5864	0.6056	0.7174	0.4517	0.3678	0.5527	0.6665	**0.7311**
PIE	0.3018	0.3880	0.7290	0.3952	0.2975	0.3351	0.7583	**0.7647**
CSTR	0.7634	0.6597	0.7483	0.7597	0.6976	0.7700	0.7437	**0.7768**
Newsgroup4	0.8158	0.6056	0.7275	0.8805	0.8214	0.8399	0.8877	**0.9000**
WebKB4	0.6973	0.6716	0.7008	0.6659	0.6214	0.6885	0.7264	**0.7567**
WebACE	0.5142	0.4679	0.4397	0.4936	0.4007	0.5415	0.5047	**0.5791**

Table 3: Normalized Mutual Information of the 8 algorithms on the 6 data sets.

Data Sets	Kmeans	NC	LLC	NMF	SNMF	ONMTF	GNMF	LLNMF
Coil20	0.7588	0.7407	0.8011	0.5954	0.4585	0.7110	0.8136	**0.8532**
PIE	0.6276	0.6843	0.9343	0.6743	0.5430	0.6787	0.9368	**0.9423**
CSTR	0.6531	0.5761	0.5787	0.6645	0.5941	0.6716	0.6302	**0.6887**
Newsgroup4	0.7129	0.7212	0.6100	0.7294	0.6432	0.7053	0.7106	**0.7334**
WebKB4	0.4665	0.4437	0.4476	0.4255	0.3643	0.4552	0.4571	**0.4755**
WebACE	0.6157	0.5959	0.4996	0.5850	0.4649	0.6012	0.6007	**0.6373**

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.(23). □

References

[Bottou and Vapnik, 1992] L. Bottou and V. Vapnik. Local learning algorithms. *Neural Computation*, 4(6):888–900, 1992.

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

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

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

[Ding *et al.*, 2006] Chris H. Q. Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *KDD*, pages 126–135, 2006.

[Ding *et al.*, 2008] 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.

[Hastie *et al.*, 2001] Trevor Hastie, Robert Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, July 2001.

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

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

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

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

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

[Schölkopf and Smola, 2002] B. Schölkopf and A.J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, USA, 2002.

[Shi and Malik, 2000] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[Sim *et al.*, 2003] 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.

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

[Wu and Schölkopf, 2006] Mingrui Wu and Bernhard Schölkopf. A local learning approach for clustering. In *NIPS*, pages 1529–1536, 2006.

[Xu *et al.*, 2003] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *SIGIR*, pages 267–273, 2003.

[Zafeiriou *et al.*, 2006] 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.