Non-negative Matrix Factorization on Manifold*

Deng Cai[†‡] Xiaofei He[‡] Xiaoyun Wu[♯] Jiawei Han[†]

[†]Department of Computer Science, University of Illinois at Urbana-Champaign
[‡]The State Key Lab of CAD&CG, Zhejiang University
[♯]Google Inc.
{dengcai2, hanj}@cs.uiuc.edu, xiaofeihe@cad.zju.edu.cn, xiaoyunwu@google.com

Abstract

Recently Non-negative Matrix Factorization (NMF) has received a lot of attentions in information retrieval, computer vision and pattern recognition. NMF aims to find two non-negative matrices whose product can well approximate the original matrix. The sizes of these two matrices are usually smaller than the original matrix. This results in a compressed version of the original data matrix. The solution of NMF yields a natural parts-based representation for the data. When NMF is applied for data representation, a major disadvantage is that it fails to consider the geometric structure in the data. In this paper, we develop a graph based approach for parts-based data representation in order to overcome this limitation. We construct an affinity graph to encode the geometrical information and seek a matrix factorization which respects the graph structure. We demonstrate the success of this novel algorithm by applying it on real world problems.

1. Introduction

The techniques of matrix factorization have become popular in recent years for data representation. In many problems in information retrieval, computer vision and pattern recognition, the input data matrix is of very high dimension. This makes *learning from example* infeasible. One hopes then to find two or more lower dimensional matrices whose product provides a good approximation to the original matrix. The canonical matrix factorization techniques include LU-decomposition, QR-decomposition, Cholesky decomposition, and Singular Value Decomposition (SVD).

*The work was supported in part by the U.S. National Science Foundation grants IIS-08-42769 and BDI-05-15813, MIAS (a DHS Institute of Discrete Science Center for Multimodal Information Access and Synthesis). Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

SVD is one of the most frequently used matrix factorization tool. A singular value decomposition of an $m \times n$ matrix $\mathbf{X}$ is any factorization of the form

$$\mathbf{X} = \mathbf{USV}^T$$

where $\mathbf{U}$ is an $m \times m$ orthogonal matrix, $\mathbf{V}$ is an $n \times n$ orthogonal matrix, and $\mathbf{S}$ is an $m \times n$ diagonal matrix with $\mathbf{S}_{ij} = 0$ if $i \neq j$ and $\mathbf{S}_{ii} \geq 0$. The quantities $\mathbf{S}_{ii}$ are called the *singular values* of $\mathbf{X}$, and the columns of $\mathbf{U}$ and $\mathbf{V}$ are called left and right *singular vectors*, respectively. By removing those singular vectors corresponding to sufficiently small singular value, we get a natural low-rank approximation to the original matrix. This approximation is optimal in the sense of reconstruction error and thus optimal for data representation when Euclidean structure is concerned. For this reason, SVD has been applied to various real world applications, such as face recognition (*Eigenface*, [26]) and document representation (*Latent Semantic Indexing*, [8]).

Previous studies have shown there is psychological and physiological evidence for parts-based representation in human brain [23], [27], [20]. The Non-negative Matrix Factorization (NMF) algorithm is proposed to learn the parts of objects like human faces and text documents [22], [14]. NMF aims to find two non-negative matrices whose product provides a good approximation to the original matrix. The non-negative constraints lead to a parts-based representation because they allow only additive, not subtractive, combinations. NMF has been shown to be superior to SVD in face recognition [16] and document clustering [29]. NMF is optimal for learning the parts of objects. However, it fails to consider the geometrical structure of the data space which is essential for data clustering and classification problems.

In this paper, we propose a novel algorithm, called Graph regularized Non-negative Matrix Factorization (GNMF), to overcome the limitation of NMF. We encode the geometrical information of the data space by constructing a nearest neighbor graph. One hopes then to find a new representation space in which two data points are sufficiently close to each

other if they are connected in the graph. To achieve this, we design a new matrix factorization objective function and incorporates the graph structure into it. We also develop an optimization scheme to solve the objective function based on iterative updates of the two factor matrices. This leads to a new parts-based data representation which respects the geometrical structure of the data space. The convergence proof of our optimization scheme is provided.

The rest of the paper is organized as follows: in Section 2, we give a brief review of NMF. Section 3 introduces our algorithm and give a convergence proof of our optimization scheme. Extensive experimental results on clustering are presented in Section 4. Finally, we provide some concluding remarks and suggestions for future work in Section 5.

2. A Brief Review of NMF

Non-negative Matrix Factorization (NMF) [14] is a matrix factorization algorithm that focuses on the analysis of data matrices whose elements are nonnegative.

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$, each column of $\mathbf{X}$ is a sample vector. NMF aims to find two non-negative matrices $\mathbf{U} = [u_{ij}] \in \mathbb{R}^{m \times k}$ and $\mathbf{V} = [v_{ij}] \in \mathbb{R}^{n \times k}$ which minimize the following objective function:

$$O = \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_{\mathrm{F}}^2 \tag{1}$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the matrix *Frobenius norm*[1].

Although the objective function O in Eqn. (1) is convex in $\mathbf{U}$ only or $\mathbf{V}$ only, it is not convex in both variables together. Therefore it is unrealistic to expect an algorithm to find the global minimum of O. Lee & Seung [15] presented an iterative update algorithm as follows:

$$u_{ij}^{t+1} = u_{ij}^t \frac{\left(\mathbf{X}\mathbf{V}\right)_{ij}}{\left(\mathbf{U}\mathbf{V}^T\mathbf{V}\right)_{ij}} \tag{2}$$

$$v_{ij}^{t+1} = v_{ij}^t \frac{\left(\mathbf{X}^T\mathbf{U}\right)_{ij}}{\left(\mathbf{V}\mathbf{U}^T\mathbf{U}\right)_{ij}} \tag{3}$$

It is proved that the above update steps will find a local mimimum of the objective function O [15].

In reality, we have $k \ll m$ and $k \ll n$. Thus, NMF essentially try to find a compressed approximation of the original data matrix, $\mathbf{X} \approx \mathbf{U}\mathbf{V}^T$. We can view this approximation column by column as

$$\mathbf{x}_i \approx \sum_{j=1}^{k} \mathbf{u}_j v_{ij} \tag{4}$$

where $\mathbf{u}_j$ is the j-th column vector of $\mathbf{U}$. Thus, each data vector $\mathbf{x}_i$ is approximated by a linear combination of the columns of $\mathbf{U}$, weighted by the components of $\mathbf{V}$. Therefore $\mathbf{U}$ can be regarded as containing a basis that is optimized for the linear approximation of the data in $\mathbf{X}$. Since relatively few basis vectors are used to represent many data vectors, good approximation can only be achieved if the basis vectors discover structure that is latent in the data [15].

The non-negative constraints on $\mathbf{U}$ and $\mathbf{V}$ only allow addictive combinations among different basis. This is the most significant difference between NMF and other other matrix factorization methods, *e.g.*, SVD. Unlike SVD, no subtractions can occur in NMF. For this reason, it is believed that NMF can learn a *parts-based* representation [14]. The advantages of this parts-based representation has been observed in many real world problems such as face analysis [16], document clustering [29] and DNA gene expression analysis [4].

3. Graph Regularized Non-negative Matrix Factorization

By using the non-negative constraints, NMF can learn a parts-based representation. However, NMF performs this learning in the Euclidean space. It fails to to discover the intrinsic geometrical and discriminating structure of the data space, which is essential to the real applications. In this Section, we introduce our *Graph regularized Non-negative Matrix Factorization* (GNMF) algorithm which avoids this limitation by incorporating a geometrically based regularizer.

3.1. The Objective Function

Recall that NMF tries to find a basis that is optimized for the linear approximation of the data which are drawn according to the distribution P_X. One might hope that knowledge of the distribution P_X can be exploited for better discovery of this basis. A natural assumption here could be that if two data points $\mathbf{x}_i, \mathbf{x}_j$ are *close* in the *intrinsic* geometry of the data distribution, then the representations of this two points in the new basis are also close to each other. This assumption is usually referred to as *manifold assumption* [2], which plays an essential rule in developing various kinds of algorithms including dimensionality reduction algorithms [2] and semi-supervised learning algorithms [3, 32, 31].

Let $f_k(\mathbf{x}_i) = v_{ik}$ be function that produce the mapping of the original data point $\mathbf{x}_i$ onto the axis $\mathbf{u}_k$, we use $\|f_k\|_M^2$ to measure the smoothness of f_k along the geodesics in the intrinsic geometry of the data. When we consider the case that the data is a compact submanifold $\mathcal{M} \subset \mathbb{R}^m$, a natural choice for $\|f_k\|_M^2$ is

$$\|f_k\|_M^2 = \int_{\mathbf{x} \in \mathcal{M}} \|\nabla_{\mathcal{M}} f_k\|^2 dP_X(\mathbf{x}) \tag{5}$$

where $\nabla_{\mathcal{M}}$ is the gradient of f_k along the manifold $\mathcal{M}$ and

[1] One can use other cost functions to measure how good $\mathbf{U}\mathbf{V}^T$ approximates $\mathbf{X}$[15]. In this paper, we will only focus on the Frobenius norm because of the space limitation.

the integral is taken over the distribution P_X.

In reality, the data manifold is usually unknown. Thus, $\|f_k\|_M^2$ in Eqn. (5) can not be computed. Recent studies on spectral graph theory [7] and manifold learning theory [1] have demonstrated that $\|f_k\|_M^2$ can be discretely approximated through a nearest neighbor graph on a scatter of data points.

Consider a graph with n vertices where each vertex corresponds to a data point. Define the edge weight matrix W as follows:

$$\mathbf{W}_{ij} = \begin{cases} 1, & \text{if } \mathbf{x}_i \in N_p(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_p(\mathbf{x}_i) \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where $N_p(\mathbf{x}_i)$ denotes the set of p nearest neighbors of $\mathbf{x}_i$. Define $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is a diagonal matrix whose entries are column (or row, since $\mathbf{W}$ is symmetric) sums of $\mathbf{W}$, $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$. $\mathbf{L}$ is called graph Laplacian [7], which is a discrete approximation to the Laplace-Beltrami operator $\triangle_{\mathcal{M}}$ on the manifold [1]. Thus, the discrete approximation of $\|f_k\|_M^2$ can be computed as follows:

$$\begin{aligned} \mathcal{R}_k &= \frac{1}{2}\sum_{i,j=1}^{N} \left(f_k(\mathbf{x}_i) - f_k(\mathbf{x}_j)\right)^2 \mathbf{W}_{ij} \\ &= \sum_{i=1}^{N} f_k(\mathbf{x}_i)^2 \mathbf{D}_{ii} - \sum_{i,j=1}^{N} f_k(\mathbf{x}_i) f_k(\mathbf{x}_j) \mathbf{W}_{ij} \\ &= \sum_{i=1}^{N} v_{ik}^2 \mathbf{D}_{ii} - \sum_{i,j=1}^{N} v_{ik} v_{jk} \mathbf{W}_{ij} \\ &= \mathbf{v}_k^T \mathbf{D} \mathbf{v}_k - \mathbf{v}_k^T \mathbf{W} \mathbf{v}_k \\ &= \mathbf{v}_k^T \mathbf{L} \mathbf{v}_k \end{aligned} \tag{7}$$

$\mathcal{R}_k$ can be used to measure the smoothness of mapping function f_k along the geodesics in the intrinsic geometry of the data set. By minimizing $\mathcal{R}_k$, we get a mapping function f_k which is sufficiently smooth on the data manifold. A intuitive explanation of minimizing $\mathcal{R}_k$ is that if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close (*i.e.* W_{ij} is big), $f_k(\mathbf{x}_i)$ and $f_k(\mathbf{x}_j)$ are similar to each other.

Our GNMF incorporates the $\mathcal{R}_k$ term and minimize the objective function

$$\begin{aligned} \mathcal{O} &= \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_{\mathrm{F}}^2 + \lambda \sum_{i=1}^{k} \mathcal{R}_k \\ &= \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_{\mathrm{F}}^2 + \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{L} \mathbf{V}) \end{aligned} \tag{8}$$

with the constraint that u_{ij} and v_{ij} are non-negative. $\operatorname{Tr}(\cdot)$ denotes the trace of a matrix. The $\lambda \geq 0$ is the regularization parameter.

3.2. An Algorithm

The objective function $\mathcal{O}$ of GNMF in Eqn. (8) is not convex in both $\mathbf{U}$ and $\mathbf{V}$ together. Therefore it is unrealistic to expect an algorithm to find the global minimum of $\mathcal{O}$. In the following, we introduce an iterative algorithm which can achieve a local minimum.

The objective function $\mathcal{O}$ can be rewritten as:

$$\begin{aligned} \mathcal{O} &= \operatorname{Tr}\left((\mathbf{X} - \mathbf{U}\mathbf{V}^T)(\mathbf{X} - \mathbf{U}\mathbf{V}^T)^T\right) + \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{L} \mathbf{V}) \\ &= \operatorname{Tr}\left(\mathbf{X}\mathbf{X}^T\right) - 2\operatorname{Tr}\left(\mathbf{X}\mathbf{V}\mathbf{U}^T\right) + \operatorname{Tr}\left(\mathbf{U}\mathbf{V}^T\mathbf{V}\mathbf{U}^T\right) \\ &\quad + \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{L} \mathbf{V}) \end{aligned} \tag{9}$$

where the second step of derivation uses the matrix property $\operatorname{Tr}(\mathbf{AB}) = \operatorname{Tr}(\mathbf{BA})$ and $\operatorname{Tr}(\mathbf{A}) = \operatorname{Tr}(\mathbf{A}^T)$. Let ψ_{ij} and ϕ_{ij} be the Lagrange multiplier for constraint $u_{ij} \geq 0$ and $v_{ij} \geq 0$ respectively, and $\Psi = [\psi_{ij}]$, $\Phi = [\phi_{ij}]$, the Lagrange $\mathcal{L}$ is

$$\begin{aligned} \mathcal{L} &= \operatorname{Tr}\left(\mathbf{X}\mathbf{X}^T\right) - 2\operatorname{Tr}\left(\mathbf{X}\mathbf{V}\mathbf{U}^T\right) + \operatorname{Tr}\left(\mathbf{U}\mathbf{V}^T\mathbf{V}\mathbf{U}^T\right) \\ &\quad + \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{L} \mathbf{V}) + \operatorname{Tr}(\Psi \mathbf{U}^T) + \operatorname{Tr}(\Phi \mathbf{V}^T) \end{aligned} \tag{10}$$

The partial derivatives of $\mathcal{L}$ with respect to $\mathbf{U}$ and $\mathbf{V}$ are:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{U}} = -2\mathbf{X}\mathbf{V} + 2\mathbf{U}\mathbf{V}^T\mathbf{V} + \Psi \tag{11}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{V}} = -2\mathbf{X}^T\mathbf{U} + 2\mathbf{V}\mathbf{U}^T\mathbf{U} + 2\lambda\mathbf{L}\mathbf{V} + \Phi \tag{12}$$

Using the KKT conditions $\psi_{ij}u_{ij} = 0$ and $\phi_{ij}v_{ij} = 0$, we get the following equations for u_{ij} and v_{ij}:

$$-\left(\mathbf{X}\mathbf{V}\right)_{ij} u_{ij} + \left(\mathbf{U}\mathbf{V}^T\mathbf{V}\right)_{ij} u_{ij} = 0 \tag{13}$$

$$-\left(\mathbf{X}^T\mathbf{U}\right)_{ij} v_{ij} + \left(\mathbf{V}\mathbf{U}^T\mathbf{U}\right)_{ij} v_{ij} + \lambda\left(\mathbf{L}\mathbf{V}\right)_{ij} v_{ij} = 0 \tag{14}$$

These equations lead to the following update rules:

$$u_{ij} \leftarrow u_{ij} \frac{\left(\mathbf{X}\mathbf{V}\right)_{ij}}{\left(\mathbf{U}\mathbf{V}^T\mathbf{V}\right)_{ij}} \tag{15}$$

$$v_{ij} \leftarrow v_{ij} \frac{\left(\mathbf{X}^T\mathbf{U} + \lambda\mathbf{W}\mathbf{V}\right)_{ij}}{\left(\mathbf{V}\mathbf{U}^T\mathbf{U} + \lambda\mathbf{D}\mathbf{V}\right)_{ij}} \tag{16}$$

Regarding these two update rules, we have the following theorem:

Theorem 1 *The objective function $\mathcal{O}$ in Eqn. (8) is nonincreasing under the update rules in Eqn. (15) and (16). The objective function is invariant under these updates if and only if $\mathbf{U}$ and $\mathbf{V}$ are at a stationary point.*

Theorem 1 grantees that the update rules of $\mathbf{U}$ and $\mathbf{V}$ in Eqn. (15) and (16) converge and the final solution will be a local optimum. Please see the Appendix for a detailed proof.

4. Related Works

Several authors have noted the shortcomings of standard NMF, and suggested extensions and modifications of the original model.

One of the shortcomings of NMF is that it can only be applied to data containing non-negative values. Ding *et al*. [10] proposed a semi-NMF approach which relaxes the non-negative constraint on **U**. Thus, semi-NMF can be used to model data containing negative values. Xu & Gong [28] proposed a Concept Factorization approach in which the input data matrix is factorized into three matrix $\mathbf{X} \approx \mathbf{XWV}^T$. Both **W** and **V** are non-negative. Such modification makes it possible to kernelize concept factorization. This concept factorization approach is also referred as convex-NMF [10].

Another shortcoming of NMF is that it does not always result in parts-based representations. Several researchers addressed this problem by incorporating the sparseness constraints on **U** and/or **V** [11], [19], [12]. These approaches extended the NMF framework to include an adjustable sparseness parameter. With a suitable sparseness parameter, these approaches are guaranteed to result in parts-based representations.

Besides the most well known multiplicative update algorithm [15], there are many other optimization methods that can solve the NMF problem in Eqn. (1). One of the most promising approaches is projected gradient method. Lin [18] shows that projected gradient method converges faster than the popular multiplicative update algorithm. Moreover, it is easy to use projected gradient method to solve the NMF problem with sparse constraints [12].

The above extensions and modifications focus on the different aspects of the original NMF. However, they all fail to consider the geometrical structure in the data. Our approach discussed in this paper presents a new direction for extending NMF. For more discussions on the relationship between various NMF extensions, please refer [17], [12], [6].

5. Experimental Results

Previous studies show that NMF is very powerful on clustering [29, 24]. It can achieve similar or better performance than most of the state-of-the-art clustering algorithms, including the popular spectral clustering methods [29]. In this section, we also evaluate our GNMF algorithm on clustering problems.

Two data sets are used in the experiment. The first one is COIL20 image library[2], which contains 32×32 gray scale images of 20 objects viewed from varying angles. The second one is the CMU PIE face database[3], which contains 32×32 gray scale face images of 68 persons. Each person has 21 facial images under different light conditions.

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

[3]http://www.ri.cmu.edu/projects/project_418.html

There are two parameters in our GNMF approach: the number of nearest neighbors p and the regularization parameter λ. Throughout our experiments, we empirically set the number of nearest neighbors p to 5, the value of the regularization parameter λ to 100.

5.1. Evaluation Metric

The clustering result is evaluated by comparing the obtained label of each sample with that provided by the data set. Two metrics, the accuracy (AC) and the normalized mutual information metric ($\overline{MI}$) are used to measure the clustering performance [29][5]. Given a data point $\mathbf{x}_i$, let r_i and s_i be the obtained cluster label and the label provided by the corpus, respectively. The AC is defined as follows:

$$AC = \frac{\sum_{i=1}^{n} \delta(s_i, map(r_i))}{n}$$

where n is the total number of samples and $\delta(x, y)$ 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 corpus. The best mapping can be found by using the Kuhn-Munkres algorithm [21].

Let C denote the set of clusters obtained from the ground truth and C' obtained from our algorithm. Their mutual information metric $MI(C, C')$ is defined as follows:

$$MI(C, C') = \sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \cdot log_2 \frac{p(c_i, c'_j)}{p(c_i) \cdot p(c'_j)}$$

where $p(c_i)$ and $p(c'_j)$ are the probabilities that a sample arbitrarily selected from the data set belongs to the clusters c_i and c'_j, respectively, and $p(c_i, c'_j)$ is the joint probability that the arbitrarily selected sample belongs to the clusters c_i as well as c'_j at the same time. In our experiments, we use the normalized mutual information $\overline{MI}$ as follows:

$$\overline{MI}(C, C') = \frac{MI(C, C')}{\max(H(C), H(C'))}$$

where $H(C)$ and $H(C')$ are the entropies of C and C', respectively. It is easy to check that $\overline{MI}(C, C')$ ranges from 0 to 1. $\overline{MI} = 1$ if the two sets of clusters are identical, and $\overline{MI} = 0$ if the two sets are independent.

5.2. Performance Evaluations and Comparisons

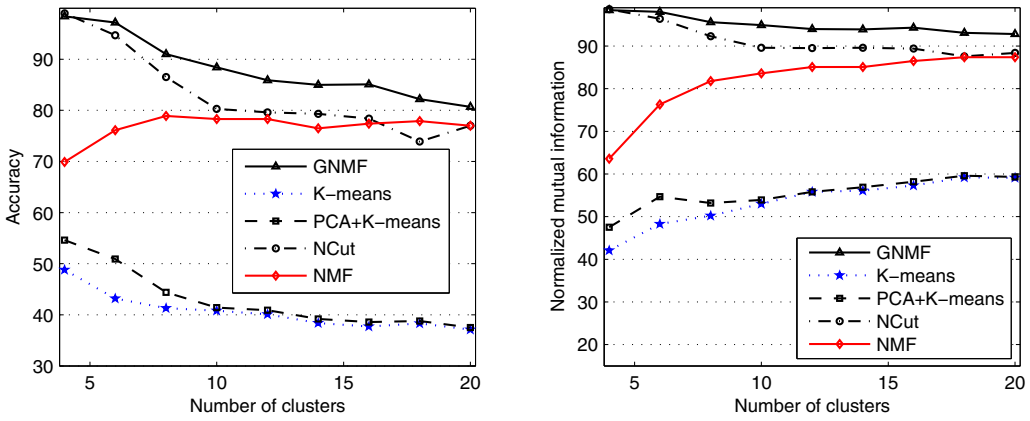
To demonstrate how the clustering performance can be improved by our method, we compared GNMF with other four popular clustering algorithms as follows:

- Canonical K-means clustering method (K-means in short).

Table 1. Clustering performance on PIE

k	Accuracy (%)					Normalized Mutual Information (%)				
	K-means	PCA+K-means	NCut	NMF	GNMF	K-means	PCA+K-means	NCut	NMF	GNMF
4	48.8	54.6	99.0	69.9	98.4	42.1	47.5	98.6	63.6	98.4
6	43.2	50.9	94.7	76.1	97.2	48.3	54.7	96.4	76.3	98.0
8	41.3	44.4	86.5	78.9	91.0	50.2	53.2	92.3	81.8	95.6
10	40.8	41.4	80.3	78.3	88.4	53.0	53.9	89.6	83.6	94.9
12	40.1	40.9	79.6	78.3	85.9	55.8	55.8	89.5	85.1	94.0
14	38.4	39.2	79.3	76.5	85.0	56.1	56.9	89.6	85.1	93.9
16	37.7	38.6	78.4	77.4	85.1	57.3	58.2	89.4	86.5	94.3
18	38.3	38.8	73.9	77.9	82.2	59.2	59.6	87.6	87.4	93.1
20	37.1	37.5	77.0	77.0	80.7	59.1	59.3	88.4	87.4	92.8
Avg	40.6	42.9	83.2	76.7	88.2	53.5	55.5	91.3	81.9	95.0

k is the number of clusters



(a) (b)

Figure 1. (a) Accuracy (b) Normalized mutual information vs. the number of clusters on PIE database

- K-means clustering in the Principle Component subspace (PCA+K-means in short). Principle Component Analysis (PCA) [13] is one of the most well known unsupervised dimensionality reduction algorithms. It is expected that the cluster structure will be more explicit in the principle component subspace. Interestingly, Zha *et al*. [30] has shown that K-means clustering in the PCA subspace has close connection with Average Association [25], which is a popular spectral clustering algorithm. They showed that if the inner product is used to measure the similarity and construct the graph, K-means after PCA is equivalent to average association.

- Normalized Cut [25], one of the typical spectral clustering algorithms (NCut).

- Nonnegative Matrix Factorization based clustering (NMF in short). We implemented a normalized cut weighted version of NMF as suggested in [29].

Table 1 and 2 show the evaluation results on the PIE data set and the COIL20 data set, respectively. The evaluations were conducted with the cluster numbers ranging from two to ten. For each given cluster number k, 20 test runs were conducted on different randomly chosen clusters. The average performance is reported in the tables.

These experiments reveal a number of interesting points:

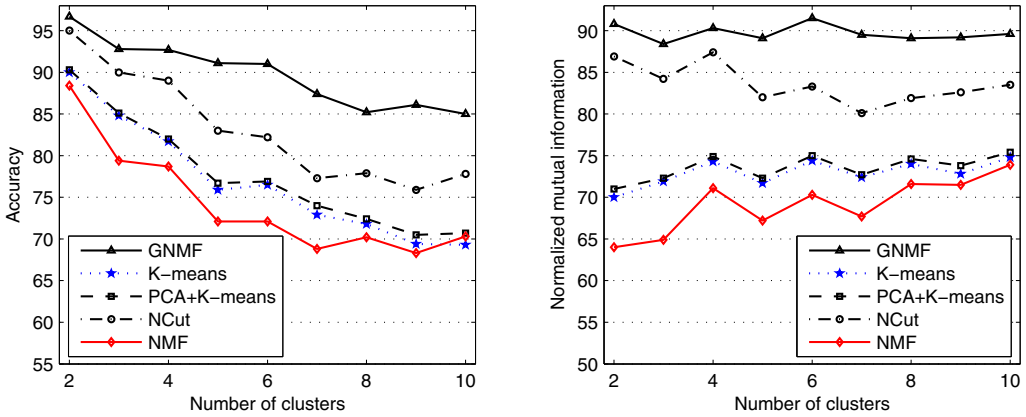
- The ordinary NMF approach outperforms K-means and PCA + K-means on PIE database while fails to get good performance on COIL20 database. Our GNMF approach gets significantly better performance than the ordinary NMF. This shows that by considering the intrinsic geometrical structure of the data, GNMF can learn a better compact representation in the sense of semantic structure.

- Both NCut and GNMF consider the geometrical structure of the data and achieve better performance than the other three algorithms. This suggests the impor-

Table 2. Clustering performance on COIL20

k	Accuracy (%)					Normalized Mutual Information (%)				
	K-means	PCA+K-means	NCut	NMF	GNMF	K-means	PCA+K-means	NCut	NMF	GNMF
2	90.0	90.3	95.0	88.4	96.7	70.0	71.0	86.9	64.0	90.8
3	84.8	85.1	90.0	79.4	92.8	71.9	72.3	84.2	64.9	88.4
4	81.7	82.0	89.0	78.7	92.7	74.3	74.9	87.4	71.1	90.3
5	75.9	76.7	83.0	72.1	91.1	71.7	72.3	82.0	67.2	89.1
6	76.5	76.9	82.2	72.1	91.0	74.4	75.0	83.3	70.3	91.5
7	72.9	74.0	77.3	68.8	87.4	72.4	72.7	80.1	67.7	89.5
8	71.8	72.4	77.9	70.2	85.2	74.0	74.6	81.9	71.6	89.1
9	69.4	70.5	75.9	68.3	86.1	72.8	73.8	82.6	71.5	89.2
10	69.3	70.7	77.8	70.3	85.0	74.8	75.4	83.5	73.9	89.6
Avg	76.9	77.6	83.1	74.3	89.8	72.9	73.6	83.5	69.1	89.7

k is the number of clusters



(a) (b)

Figure 2. (a) Accuracy (b) Normalized mutual information vs. the number of clusters on COIL20 database

tance of the geometrical structure in learning the hidden topic structure.

5.3. Parameters Selection

Our GNMF model has two essential parameters: the number of nearest neighbors p and the regularization parameter λ. Figure 3 and Figure 4 show how the performance of GNMF varies with the parameters λ and p, respectively. As we can see, the GNMF is very stable with respect to both the parameter λ and p. It achieves consistent good performance with the λ varying from 50 to 1000 and p varying from 3 to 6.

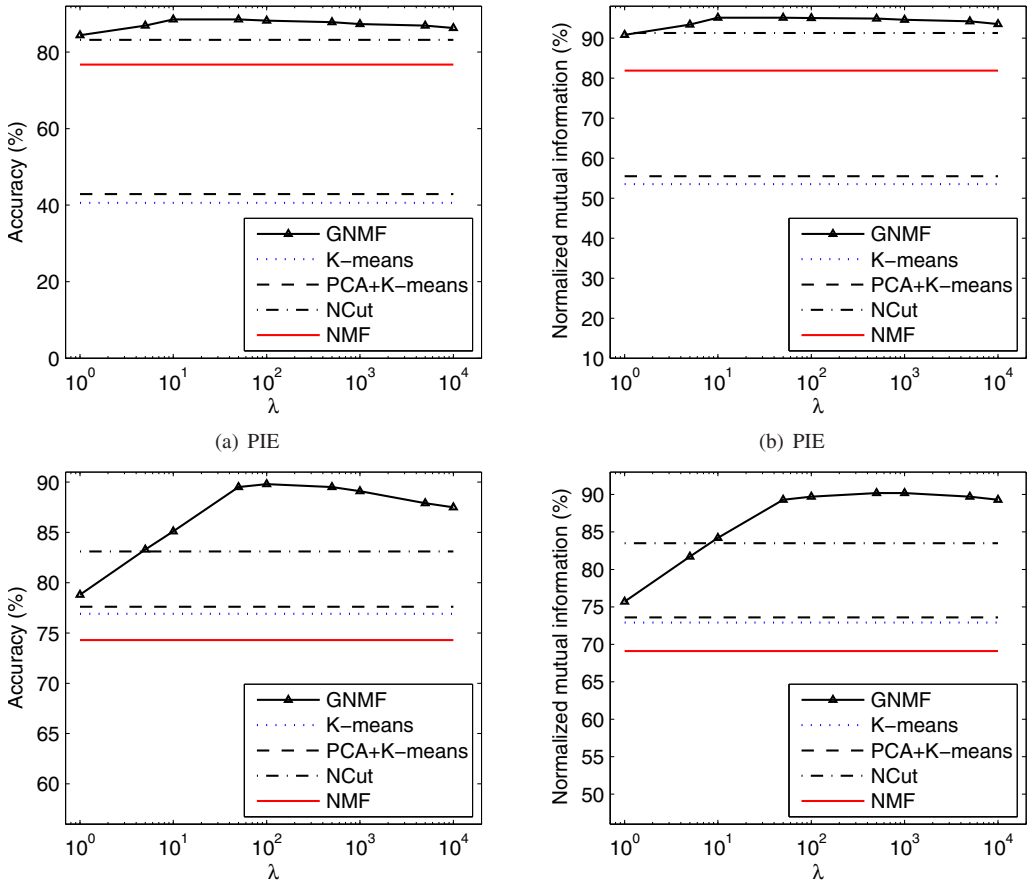
6. Conclusions and Future Work

We have presented a novel method for matrix factorization, called Graph regularized Non-negative Matrix Factorization (GNMF). GNMF models the data space as a submanifold embedded in the ambient space and performs the non-negative matrix factorization on this manifold in question. As a result, GNMF can have more discriminating power than the ordinary NMF approach which only considers the Euclidean structure of the data. Experimental results on visual objects clustering show that GNMF provides better representation in the sense of semantic structure.

Several questions remain to be investigated in our future work:

1. There is a parameter λ which controls the smoothness of our GNMF model. GNMF boils down to original NMF when $\lambda = 0$. Thus, a suitable value of λ is critical to our algorithm. It remains unclear how to do model selection theoretically and efficiently.

2. It would be very interesting to explore different ways of constructing the graphes to model the semantic structure in the data. There is no reason to believe that the nearest neighbor graph is the only or the most natural choice. For example, for web page data it may be more natural to use the hyperlink information to construct the graph.



(a) PIE (b) PIE

(c) COIL20 (d) COIL20

Figure 3. The performance of GNMF vs. parameter λ. The GNMF is very stable with respect to the parameter λ. It achieves consistent good performance with the λ varying from 50 to 1000.

References

[1] M. Belkin. *Problems of Learning on Manifolds*. PhD thesis, University of Chicago, 2003.

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, Cambridge, MA, 2001.

[3] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[4] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences*, 101(12):4164–4169, 2004.

[5] D. Cai, X. He, and J. Han. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637, December 2005.

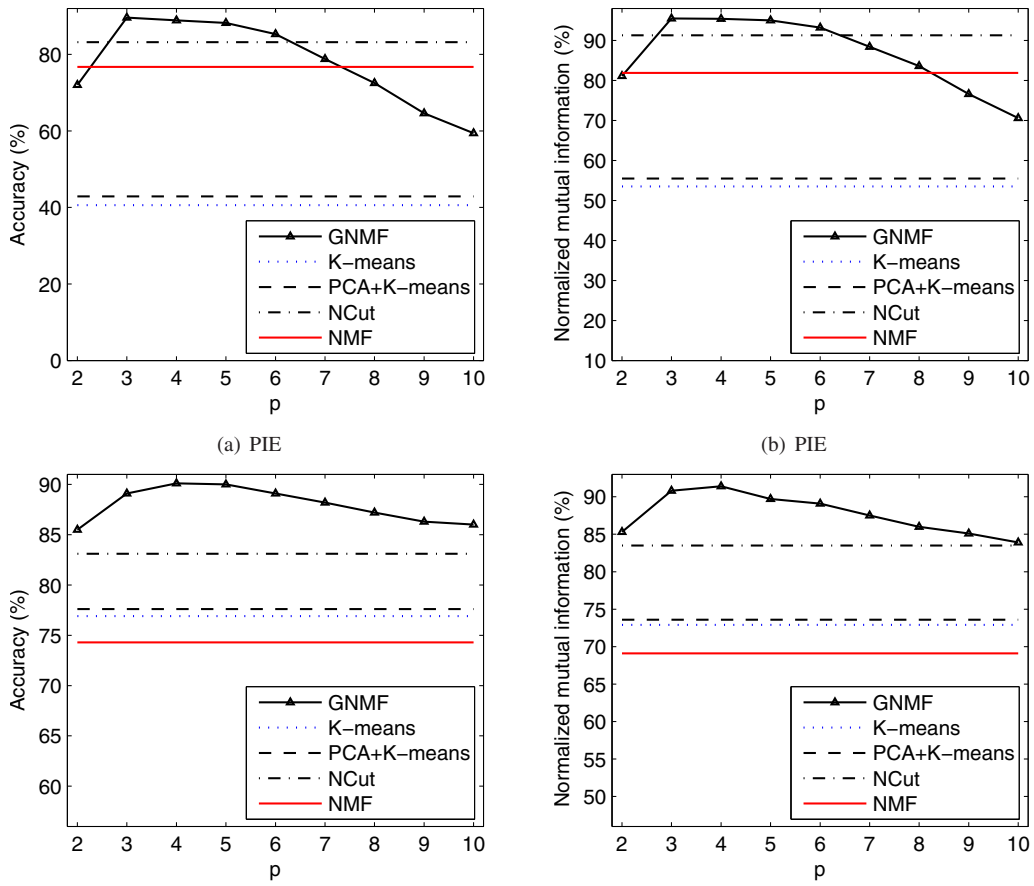
[6] M. Chu, F. Diele, R. Plemmons, and S. Ragni. *Optimality, Computation, and Interpretation of Nonnegative Matrix Factoriaztions*, October 2004.

[7] F. R. K. Chung. *Spectral Graph Theory*, volume 92 of *Regional Conference Series in Mathematics*. AMS, 1997.

[8] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[10] C. Ding, T. Li, and M. Jordan. Convex and semi-nonnegative matrix factorizations for clustering and low-dimension representation. Technical report,



(a) PIE (b) PIE

(c) COIL20 (d) COIL20

Figure 4. The performance of GNMF vs. parameter p. GNMF achieves consistent good performance with the parameter p varying from 3 to 6.

LBNL-60428, Lawrence Berkeley National Laboratory, 2006.

[11] P. O. Hoyer. Non-negative sparse coding. In *Proc. IEEE Workshop on Neural Networks for Signal Processing*, pages 557–565, 2002.

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

[13] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, 1989.

[14] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[15] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*. 2001.

[16] S. Z. Li, X. Hou, H. Zhang, and Q. Cheng. Learning spatially localized, parts-based representation. In *2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, pages 207–212, 2001.

[17] T. Li and C. Ding. The relationships among various nonnegative matrix factorization methods for clustering. In *Proc. Int. Conf. on Data Mining (ICDM'06)*, 2006.

[18] C.-J. Lin. Projected gradient methods for non-negative matrix factorization. *Neural Computation*, 19(10):2756–2779, 2007.

[19] W. Liu, N. Zheng, and X. Lu. Non-negative matrix factorization for visual coding. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'2003)*, 2003.

[20] N. K. Logothetis and D. L. Sheinberg. Visual object recognition. *Annual Review of Neuroscience*, 19:577–621, 1996.

[21] L. Lovasz and M. Plummer. *Matching Theory*. Akadémiai Kiadó, North Holland, Budapest, 1986.

[22] P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

[23] S. E. Palmer. Hierarchical structure in perceptual representation. *Cognitive Psychology*, 9:441–474, 1977.

[24] F. Shahnaza, M. W. Berrya, V. Paucab, and R. J. Plemmonsb. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

[25] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[26] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[27] E. Wachsmuth, M. W. Oram, and D. I. Perrett. Recognition of objects and their component parts: Responses of single units in the temporal cortex of the macaque. *Cerebral Cortex*, 4:509–522, 1994.

[28] W. Xu and Y. Gong. Document clustering by concept factorization. In *Proc. 2004 Int. Conf. on Research and Development in Information Retrieval (SIGIR'04)*, pages 202–209, Sheffield, UK, July 2004.

[29] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *Proc. 2003 Int. Conf. on Research and Development in Information Retrieval (SIGIR'03)*, pages 267–273, Toronto, Canada, Aug. 2003.

[30] H. Zha, C. Ding, M. Gu, X. He, , and H. Simon. Spectral relaxation for k-means clustering. In *Advances in Neural Information Processing Systems 14*, pages 1057–1064. MIT Press, Cambridge, MA, 2001.

[31] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*, 2003.

[32] X. Zhu and J. Lafferty. Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 1052–1059, 2005.

Appendix (Proofs of Theorem 1):

The objective function $\mathcal{O}$ of GNMF in Eqn. (8) is certainly bounded from below by zero. To prove Theorem 1, we need to show that $\mathcal{O}$ is nonincreasing under the update steps in Eqn. (15) and (16). Since the second term of $\mathcal{O}$ is only related to $\mathbf{V}$, we have exactly the same update formula for $\mathbf{U}$ in GNMF as the original NMF. Thus, we can use the convergence proof of NMF to show that $\mathcal{O}$ is nonincreasing under the update step in Eqn. (15). Please see [15] for details.

Now we only need to prove that $\mathcal{O}$ is nonincreasing under the update step in Eqn. (16). we will follow the similar procedure described in [15]. Our proof will make use of an auxiliary function similar to that used in the Expectation-Maximization algorithm [9]. We begin with the definition of the *auxiliary function*.

Definition $G(v, v')$ is an *auxiliary function* for $F(v)$ if the conditions

$$G(v, v') \geq F(v), \quad G(v, v) = F(v)$$

are satisfied.

The auxiliary function is very useful because of the following lemma.

Lemma 2 *If G is an auxiliary function of F, then F is nonincreasing under the update*

$$v^{(t+1)} = \arg\min_{v} G(v, v^{(t)}) \tag{17}$$

Proof

$$F(v^{(t+1)}) \leq G(v^{(t+1)}, v^{(t)}) \leq G(v^{(t)}, v^{(t)}) = F(v^{(t)})$$

∎

Now we will show that the update step for $\mathbf{V}$ in Eqn. (16) is exactly the update in Eqn. (17) with a proper auxiliary function.

We rewrote the objective function $\mathcal{O}$ of GNMF in Eqn. (8) as follows

$$\begin{aligned}\mathcal{O} &= \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_{\mathrm{F}}^2 + \lambda \operatorname{Tr}(\mathbf{V}^T\mathbf{L}\mathbf{V}) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij} - \sum_{l=1}^{k} u_{il}v_{jl})^2 + \lambda \sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{n} v_{jl}L_{ji}v_{il}\end{aligned} \tag{18}$$

Considering any element v_{ab} in $\mathbf{V}$, we use F_{ab} to denote the part of $\mathcal{O}$ which is only relevant to v_{ab}. It is easy to check that

$$F'_{ab} = \left(\frac{\partial \mathcal{O}}{\partial \mathbf{V}}\right)_{ab} = \left(-2\mathbf{X}^T\mathbf{U} + 2\mathbf{V}\mathbf{U}^T\mathbf{U} + 2\lambda\mathbf{L}\mathbf{V}\right)_{ab} \tag{19}$$

$$F''_{ab} = 2\left(\mathbf{U}^T\mathbf{U}\right)_{bb} + 2\lambda\mathbf{L}_{aa} \tag{20}$$

Since our update is essentially element-wise, it is sufficient to show that each F_{ab} is nonincreasing under the update step of Eqn. (16).

Lemma 3 *Function*

$$G(v, v_{ab}^{(t)}) = F_{ab}(v_{ab}^{(t)}) + F'_{ab}(v_{ab}^{(t)})(v - v_{ab}^{(t)}) + \frac{\big(\boldsymbol{VU}^T\boldsymbol{U}\big)_{ab} + \lambda\big(\boldsymbol{DV}\big)_{ab}}{v_{ab}^{(t)}}(v - v_{ab}^{(t)})^2 \quad (21)$$

is an auxiliary function for F_{ab}, the part of $\mathcal{O}$ which is only relevant to v_{ab}.

Proof Since $G(v, v) = F_{ab}(v)$ is obvious, we need only show that $G(v, v_{ab}^{(t)}) \geq F_{ab}(v)$. To do this, we compare the Taylor series expansion of $F_{ab}(v)$

$$F_{ab}(v) = F_{ab}(v_{ab}^{(t)}) + F'_{ab}(v_{ab}^{(t)})(v - v_{ab}^{(t)}) + \big[\big(\mathbf{U}^T\mathbf{U}\big)_{bb} + \lambda\mathbf{L}_{aa}\big](v - v_{ab}^{(t)})^2 \quad (22)$$

with Eqn. (21) to find that $G(v, v_{ab}^{(t)}) \geq F_{ab}(v)$ is equivalent to

$$\frac{\big(\mathbf{VU}^T\mathbf{U}\big)_{ab} + \lambda\big(\mathbf{DV}\big)_{ab}}{v_{ab}^{(t)}} \geq \big(\mathbf{U}^T\mathbf{U}\big)_{bb} + \lambda\mathbf{L}_{aa}. \quad (23)$$

We have

$$\big(\mathbf{VU}^T\mathbf{U}\big)_{ab} = \sum_{l=1}^{k} v_{al}^{(t)}\big(\mathbf{U}^T\mathbf{U}\big)_{lb} \geq v_{ab}^{(t)}\big(\mathbf{U}^T\mathbf{U}\big)_{bb} \quad (24)$$

and

$$\begin{aligned}\lambda\big(\mathbf{DV}\big)_{ab} &= \lambda\sum_{j=1}^{m}\mathbf{D}_{aj}v_{jb}^{(t)} \geq \lambda\mathbf{D}_{aa}v_{ab}^{(t)} \\ &\geq \lambda\big(\mathbf{D} - \mathbf{W}\big)_{aa}v_{ab}^{(t)} = \lambda\mathbf{L}_{aa}v_{ab}^{(t)}\end{aligned} \quad (25)$$

.

Thus, Eqn. (23) holds and $G(v, v_{ab}^{(t)}) \geq F_{ab}(v)$. ∎

We can now demonstrate the convergence of Theorem 1:

Proof of Theorem 1 Replacing $G(v, v_{ab}^{(t)})$ in Eqn. (17) by Eqn. (21) results in the update rule:

$$\begin{aligned}v_{ab}^{(t+1)} &= v_{ab}^{(t)} - v_{ab}^{(t)}\frac{F'_{ab}(v_{ab}^{(t)})}{2\big(\mathbf{VU}^T\mathbf{U}\big)_{ab} + 2\lambda\big(\mathbf{DV}\big)_{ab}} \\ &= v_{ab}^{(t)}\frac{\big(\mathbf{X}^T\mathbf{U} + \lambda\mathbf{WV}\big)_{ab}}{\big(\mathbf{VU}^T\mathbf{U} + \lambda\mathbf{DV}\big)_{ab}}\end{aligned} \quad (26)$$

Since Eqn. (21) is an auxiliary function, F_{ab} is nonincreasing under this update rule. ∎