One-Sided Non-Negative Matrix Factorization and Non-Negative Centroid Dimension Reduction for Text Classification

Haesun Park and Hyunsoo Kim

College of Computing
Georgia Institute of Technology
Atlanta, GA 30332, USA
{hpark, hskim}@cc.gatech.edu

Abstract

Non-negative matrix factorization (NMF) is an unsupervised subspace method that finds an approximate factorization $A \approx WH$ into non-negative factors W and H. In this paper, we introduce an one-sided NMF that imposes a non-negativity constraint on W without the non-negativity of H so as to obtain only non-negative basis vectors. In addition, we present an one-side NMF that enforces a non-negativity constraint on not W but H for the learning of non-subtractive combination of basis vectors that have arbitrary signs. Finally, we propose a non-negative centroid dimension reduction algorithm.

1 Introduction

In many data mining problems, dimension reduction is imperative for efficiently manipulating the massive quantity of high dimensional data, i.e. efficient classification [8], clustering [6], and faster searches [2]. To be useful, the lower dimensional representation must be a good approximation of the original dataset given in its full space. There are several types of unsupervised low-rank dimension reduction methods of the form $A \approx WH$, such as principal components analysis (PCA) and non-negative matrix factorization (NMF). NMF constrains the elements of W and H to be non-negative. Given a non-negative matrix A of size $m \times n$ and a desired rank k, NMF solves the following optimization problem:

$$\min_{W,H} \|A - WH\|_F^2, \;\; s.t. \;\; W, H \geq 0, \tag{1.1}$$

where $W \in \mathbb{R}^{m \times k}$ is a basis matrix, $H \in \mathbb{R}^{k \times n}$ is a mixing matrix, and $W, H \geq 0$ means that all elements of W and H are non-negative. NMF does not provide us with a unique solution if we can find a full rank square matrix X such that $A = WXX^{-1}H, WX \geq 0, X^{-1}H \geq 0$. A possible X is a rotation matrix, which is an orthogonal matrix with $|X| = 1$. NMF can sometimes learn a parts-based basis vectors [11]. NMF gives us more direct interpretation than PCA due to non-subtractive combinations of non-negative basis vectors. Also, some practical problems require non-negative basis vectors. For example, pixels in digital images, term frequencies in text mining, and chemical concentrations in bioinformatics should be non-negative [5]. It has been successfully applied to many problems including text data mining [11, 17], gene expression data analysis [9, 4]. Non-negative dimension reduction is desirable for handling the massive quantity of high-dimensional data that require non-negativity constraints.

In this paper, we introduce one-sided NMFs and a non-negative centroid dimension reduction algorithm. The rest of this paper is organized as follows. We review an algorithm for computing NMF based on multiplicative update rules (NMF/MUR) in Section 2, and NMF using alternating non-negativity constrained least squares (NMF/ANNLS) in Section 3. In Section 4, we propose an one-sided NMF that imposes a non-negativity constraint on W without the non-negativity of H so as to obtain only non-negative basis vectors, and an one-sided NMF that enforces a non-negativity constraint on not W but H for the learning of non-subtractive combination of basis vectors that have arbitrary signs. Section 5 presents experimental results illustrating properties of NMF/ANNLS and one-sided NMFs. In addition, the non-negative centroid dimension reduction is applied for text classification. Summary is given in Section 6.

Table 1: Comparison among NMF algorithms for $k = 9$ on the CBCL face recognition database [14] (A_{CBCL} of size 361×2429) and the ORL database of faces [1] (A_{ORL} of size 2576×400). We presented percentages of the number of zero elements and percentages of the number of very small non-negative elements that are smaller than 10^{-8} in W and H. We also presented the Frobenius norm of the error, i.e. $\|A - WH\|_F$, and convergence time for each method. The convergence criterion is described in the Results section.

	CBCL		ORL	
Methods	NMF/MUR	NMF/ANNLS	NMF/MUR	NMF/ANNLS
$W = 0$ (%)	0%	45%	0%	21%
$H = 0$ (%)	2%	32%	0%	10%
$0 \leq W < 10^{-8}$ (%)	31%	45%	8%	21%
$0 \leq H < 10^{-8}$ (%)	31%	32%	5%	10%
$\|A - WH\|_F$	69.07	68.35	3.17	3.16
time (sec.)	198.2	37.4	219.5	26.1

Table 2: Comparison among NMF algorithms for $k = 25$ on the CBCL data matrix A_{CBCL} of size $361 \times 2,429$ and the ORL data matrix A_{ORL} of size $2,576 \times 400$. We presented percentages of the number of very small non-negative elements that are smaller than 10^{-8} in W and H.

	CBCL		ORL	
Methods	NMF/MUR	NMF/ANNLS	NMF/MUR	NMF/ANNLS
$0 \leq W < 10^{-8}$ (%)	51%	64%	20%	31%
$0 \leq H < 10^{-8}$ (%)	42%	40%	13%	24%
$\|A - WH\|_F$	52.64	52.02	2.53	2.51
time (sec.)	673.6	601.6	769.8	597.5

2 NMF based on Multiplicative Update Rules (NMF/MUR)

Lee and Seung [12] suggested an algorithm for computing NMF based on multiplicative update rules (NMF/MUR) of W and H, and proved that the Euclidean distance $\|A - WH\|_F$ is monotonically non-increasing under the update rules:

$$H_{aj} \leftarrow H_{aj} \frac{(W^T A)_{aj}}{(W^T W H)_{aj} + \epsilon},$$

for $1 \leq a \leq k$ and $1 \leq j \leq n$,

$$W_{ia} \leftarrow W_{ia} \frac{(AH^T)_{ia}}{(WHH^T)_{ia} + \epsilon},$$

for $1 \leq i \leq m$ and $1 \leq a \leq k$, where ϵ is a small positive value to avoid zero in the denominators of the approximations W and H. One can normalize the columns of the basis matrix W to unit norm. Then, the column vectors of W are mapped to the surface of a hypersphere. This normalization procedure can be explained by the scaling procedure $(\mathbf{w}_i/d_i)(d_i\mathbf{h}_i)$, where $\mathbf{w}_i$ is the ith column of W, $\mathbf{h}_i$ is the ith row of H, and d_i is $\|\mathbf{w}_i\|_2$.

3 NMF based on Alternating Non-Negativity-Constrained Least Squares (NMF/ANNLS)

Given a non-negative matrix $A \in \mathbb{R}^{m \times n}$, NMF based on alternating non-negativity-constrained least squares (NMF/ANNLS) starts with the initialization of $W \in \mathbb{R}^{m \times k}$ and $H \in \mathbb{R}^{k \times n}$ with non-negative values, and scales the columns of W to unit L_2-norm. Then, it iterates the following ANNLS until convergence:

$$\min_H \|WH - A\|_F^2, \ \ s.t. \ H \geq 0, \tag{3.2}$$

which fixes W and solves the optimization with respect to H, and

$$\min_W \|H^T W^T - A^T\|_F^2, \ \ s.t. \ W \geq 0, \tag{3.3}$$

which fixes H and solves the optimization with respect to W. Paatero and Tapper [15] originally proposed

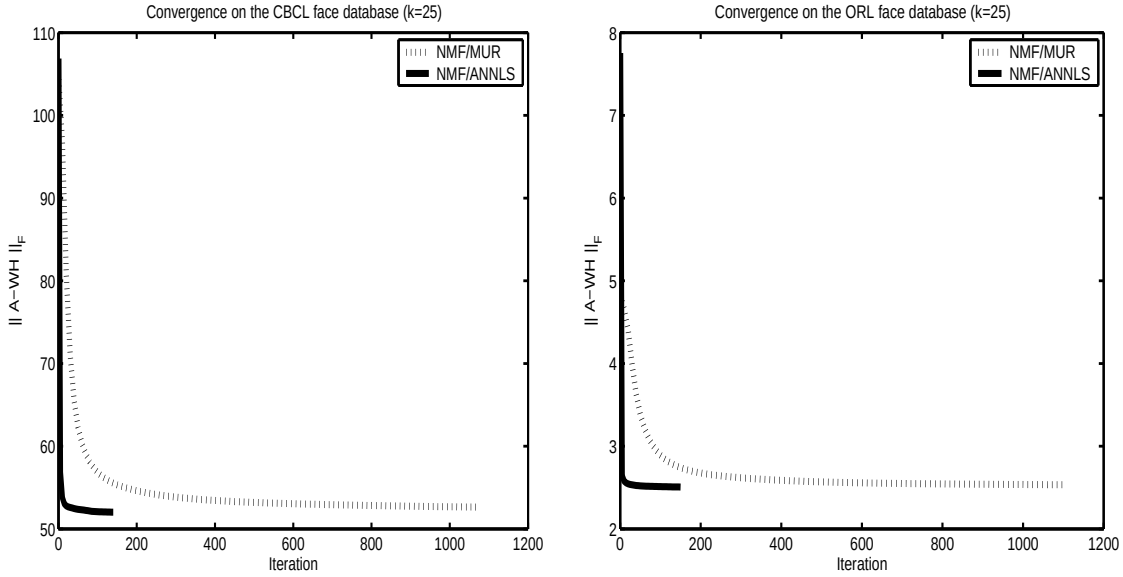

Figure 1: The number of iterations for the convergence of NMF/MUR and NMF/ANNLS with $k = 25$ on the CBCL data matrix A_{CBCL} of size 361×2429 and the ORL data matrix A_{ORL} of size 2576×400. The convergence criterion is described in the Results section.

using a constrained alternating least squares algorithm to solve Eqn. (1.1). The columns of the basis matrix W are normalized to unit L_2-norm at each iteration. When $k < m$, the non-negative low-rank representation of A is given by H. The low-rank representation $\mathbf{h} \in \mathbb{R}^{k\times 1}$ of a new data point $\mathbf{x} \in \mathbb{R}^{m\times 1}$ is computed by solving following non-negativity-constrained least squares problem:

$$\min_{\mathbf{h}} \|W\mathbf{h} - \mathbf{x}\|_2^2, \ \ s.t. \ \mathbf{h} \geq 0. \tag{3.4}$$

Here, we adopt a fast algorithm for large scale non-negativity-constrained least squares (NNLS) problems [18] to solve Eqns. (3.2)-(3.4). Bro and de Jong [3] made a substantial speed improvement to Lawson and Hanson's algorithm [10] for large scale NNLS problems. This algorithm precomputes parts of the pseudoinverse and the cross-product matrices that appear in the normal equations for solving least squares problems. Van Benthem and Keenan [18] devised an algorithm that further improves the performance of NNLS for multivariate data. This algorithm deals with the following NNLS optimization problem given $B \in \mathbb{R}^{m\times k}$ and $A \in \mathbb{R}^{m\times n}$:

$$\min_{G} \|BG - A\|_F^2, \ \ s.t. \ G \geq 0, \tag{3.5}$$

where $G \in \mathbb{R}^{k\times n}$ is a solution. It is based on the active/passive set method. It uses the unconstrained solution G_u obtained from the unconstrained least squares problem, i.e. $\min_{G_u} \|BG_u - A\|_F^2$, so as to determine the initial passive sets $\mathcal{P}$. A passive set $\mathcal{P}_j$ $(1 \leq j \leq n)$ contains locations (rows) of the positive entries in the jth column of G. The active set $\mathcal{A}_j$ is merely the complement of $\mathcal{P}_j$, which contains the row indices of elements that are constrained to equal zero, in the jth column of G. Thus, there are a set of passive sets $\mathcal{P} = \{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$ and a set of active sets is $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$. This algorithm also makes a set $\mathcal{F}$ of column indices of G for solutions that are not optimal (See [10, 3] for the description of the optimality testing) and an infeasible solution set $\mathcal{H}$ that is a subset of $\mathcal{F}$ containing the indices for solutions that are currently infeasible (i.e. column indices of G containing negative values). This algorithm consists of a main loop and an inner loop. The main loop obtains unconstraint least squares solutions for the passive variables for columns of G in $\mathcal{F}$, i.e. $\min_{G_\mathcal{F}} \|BG_\mathcal{F} - A_\mathcal{F}\|_F^2$ using $\mathcal{P}_\mathcal{F}$, where $G_\mathcal{F}$ and $A_\mathcal{F}$ are respectively the submatrices of G and A obtained from the set of column indices $\mathcal{F}$. The algorithm terminates when $\mathcal{F}$ has been emptied. If there are infeasible solutions ($\mathcal{H} \neq \emptyset$), it makes them feasible in the inner loop by computing $\min_{G_\mathcal{H}} \|BG_\mathcal{H} - A_\mathcal{H}\|_F^2$ using updated $\mathcal{P}_\mathcal{H}$. The inner loop terminates when $\mathcal{H}$ becomes empty. Then, the set $\mathcal{F}$ is updated by removing indices of columns whose solutions are optimal. The passive sets $\mathcal{P}_\mathcal{F}$ are updated accordingly. Each unconstrained least squares solution is computed

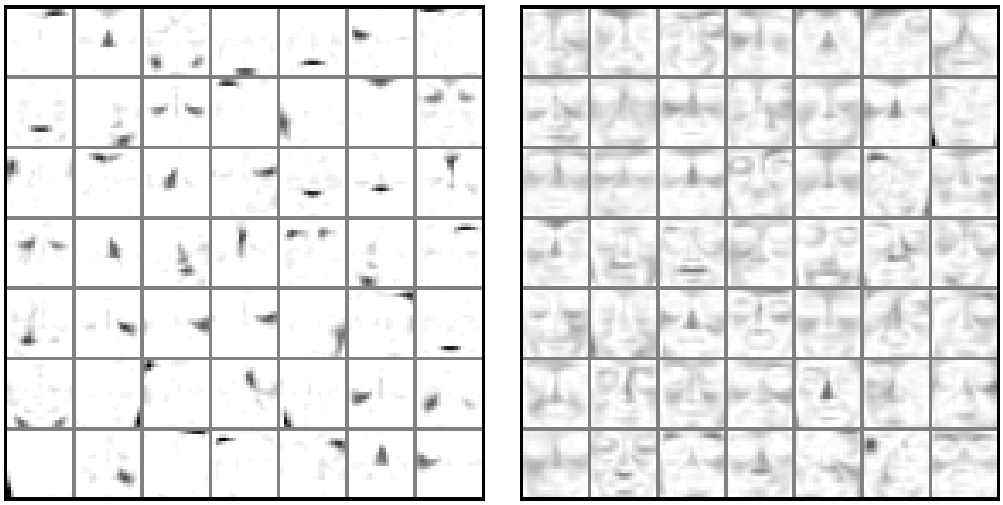

Figure 2: (Left panel) Basis images given by NMF/ANNLS for $k = 49$ on the CBCL data matrix A_{CBCL} of size $361 \times 2,429$ ($\|A_{CBCL} - WH\|_F = 39.87$ s.t. $A_{CBCL} \geq 0$, $W \geq 0$ and $H \geq 0$, elapsed time: 2116.4 sec.). (Right panel) Basis images given by ONMF/L for $k = 49$ on the CBCL data matrix ($\|A_{CBCL} - WH\|_F = 32.98$ s.t. $A_{CBCL} \geq 0$ and $W \geq 0$, elapsed time: 52.6 sec.). Each column of $W \in \mathbb{R}^{361\times 49}$ is a basis image of size 19×19. Zeros were drawn in white and larger positive values were drawn in darker gray.

by solving the normal equations $B^T BG_{\mathcal{S}} = B^T A_{\mathcal{S}}$ under certain set of columns $\mathcal{S}$ given passive sets $\mathcal{P}_{\mathcal{S}}$. In the subroutine for computing the unconstrained least squares problems, a set of unique passive sets $\mathcal{U} = \{\mathcal{U}_1, \ldots, \mathcal{U}_u\}$ are found from $\mathcal{P}_{\mathcal{S}}$. For each unique passive set $\mathcal{U}_j$, ($1 \leq j \leq u$), the system of normal equations $\Gamma(\mathcal{U}_j, \mathcal{U}_j)G(\mathcal{U}_j, \mathcal{E}_j) = \Delta(\mathcal{U}_j, \mathcal{E}_j)$ is solved inside the subroutine, where $\Gamma = B^T B$, $\Delta = B^T A$, and $\mathcal{E}_j$ is a set of column indices sharing the same passive set of $\mathcal{U}_j$. This grouping strategy is an essential part that contributes to the computational efficiency of this algorithm. Although there are several numerical methods to solve normal equations, this algorithm uses the LU factorization with pivoting and forward- and back-substitution. More detailed explanations of this algorithm can be found in [18].

4 One-Sided NMF

We suggest a new decomposition that imposes a constraint of non-negativity on W without non-negativity of H so as to obtain only non-negative basis vectors. This decomposition is referred to as ONMF/L, where 'L' denotes that the left side factor has the nonnegativity imposed. Given a non-negative matrix $A \in \mathbb{R}^{m\times n}$ and k, ONMF/L starts with an initial guess of $W \in \mathbb{R}^{m\times k} \geq 0$ and $H \in \mathbb{R}^{k\times n}$ and scales the columns of W to unit L_2-norm. Then it iteratively solves the following LS and NNLS problems until convergence:

$$\min_H \|WH - A\|_F^2,$$

$$\min_W \|H^T W^T - A^T\|_F^2, \ \ s.t. \ W \geq 0.$$

The columns of the basis matrix W are normalized to unit L_2-norm at each iteration. This dimension reduction algorithm can produce non-negative basis vectors. When $k \ll m$, the low-rank representation of A is given by H. The low-rank representation $\mathbf{h} \in \mathbb{R}^{k\times 1}$ of a new data point $\mathbf{x} \in \mathbb{R}^{m\times 1}$ is computed by solving following LS problem:

$$\min_{\mathbf{h}} \|W\mathbf{h} - \mathbf{x}\|_2^2. \tag{4.6}$$

On the other hand, we can think of the following one-sided NMF only for the non-negativity of H. This decomposition is referred to as ONMF/R, where 'R' denotes that the right side factor has the nonnegativity imposed. ONMF/R also begins with an initial guess of W and $H \geq 0$ and scales the columns of W to unit L_2-norm. Then, it iteratively solves the following NNLS and LS problems:

$$\min_H \|WH - A\|_F^2, \ \ s.t. \ H \geq 0,$$

$$\min_W \|H^T W^T - A^T\|_F^2.$$

The columns of the basis matrix W are normalized to unit L_2-norm at each iteration. This dimension reduction algorithm allow the entries of a basis matrix W to be of arbitrary sign. But, the coefficients of the linear combination should be non-negative. When $k \ll m$, the non-negative low-rank representation of A is given by H. Hence, this decomposition can be used when we want to obtain only a non-negative low-rank representation. The non-negative low-rank representation $\mathbf{h} \in \mathbb{R}^{k\times 1}$ of a new data point $\mathbf{x} \in \mathbb{R}^{m\times 1}$ is computed by solving following NNLS problem:

$$\min_{\mathbf{h}} \|W\mathbf{h} - \mathbf{x}\|_2^2, \;\; s.t. \;\; \mathbf{h} \geq 0. \tag{4.7}$$

5 Experimental Results and Discussion

5.1 Datasets Description We used the CBCL face recognition database [14], the ORL database of faces [1], and the MEDLINE information retrieval dataset. The CBCL face database contains 2,429 faces with 19×19 grayscale PGM format images in the training set. We built a big matrix A_{CBCL} of size $(19 \cdot 19) \times 2429$. The ORL database of faces [1] contains ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background. The faces are in an upright, frontal position with tolerance for some side movement. The files are also in PGM format. The size of each image is 112×92 pixels, with 256 grey levels per pixel. We reduced the size of the images to 56×46 pixels and built a big matrix A_{ORL} of size $(56 \cdot 46) \times 400$. MEDLINE consists of $1,033$ abstracts from medical journals (available from ftp://ftp.cs.cornell.edu/pub/smart). This dataset has been extensively used as a benchmark collection in information retrieval that aims at extracting documents that are relevant to user's query, effectively and efficiently. We obtained a MEDLINE term-document matrix of size $5,735 \times 1,033$ in the form of Matlab sparse arrays generated by Text to Matrix Generator (TMG) (available from http:// scgroup.hpclab.ceid.upatras.gr/scgroup/Projects/TMG/). TMG applies common filtering techniques (e.g. removal of common words, removal of words that are too infrequent or frequent, removal of words that are too short or too long, etc) to reduce the size of the term dictionary. The matrix was provided by using the simplest term-weighting scheme (i.e. term frequency local function, with no global weighting and normalization). Stemming was not applied. In the MEDLINE dataset, there are 30 natural language queries and relations giving relevance judgements between query and document. We prepared a term-document matrix of size $5,735 \times 696$ since only 696 documents among 1,033 documents have matched with 30 queries.

5.2 Characteristics of NMFs We tested convergence every 10 iterations for NMF and 5 iterations for the other NMF algorithms using NNLS. We store the Frobenius norm of the error, i.e. $f = \|A - WH\|_F$, when testing convergence. The convergence criterion was

$$\frac{f_{prev} - f_{curr}}{f_{prev}} < 10^{-4}, \tag{5.8}$$

where f_{prev} and f_{curr} are the Frobenius norms in the previous and current convergence tests respectively. The initial Frobenius norm was computed by the random initial guess of W and H.

We implemented algorithms in Matlab 6.5 [13]. All our experiments were performed on a PentiumIII 600MHz machine with 512MB memory running Windows2000. In Table 1, we presented percentages of the number of zero elements and percentages of the number of very small non-negative elements that are smaller than 10^{-8} in W and H. We also presented the Frobenius norm of the error, i.e. $\|A - WH\|_F$, and convergence time for each method. NMF/MUR could only yield a few percentage of the number of the exact zero elements in H on the CBCL dataset. We also observed a number of elements within the range of $[0,\ 10^{-8})$. On the other hand, NMF/ANNLS introduced many the exact zeros. NMF/ANNLS produced sparser basis images and more accurate decompositions (i.e., smaller $\|A - WH\|_F$) within shorter time than NMF on the both datasets. Figure 1 shows the number of iterations for the convergence of NMF/MUR and NMF/ANNLS with $k = 25$ on the CBCL dataset and the ORL dataset. NMF/ANNLS consistently converged within fewer numbers of iterations than NMF/MUR. However, the fewer number of iterations generally does not mean quicker convergence. NMF/ANNLS converged in much less seconds than NMF/MUR when $k = 9$ (see Table 1), but it showed convergence speed similar to that of NMF/MUR when $k = 25$ (see Table 2).

We observed that ONMF/L generated holistic basis images instead of parts-based basis images (See Figure 2). However, we would like to emphasize that ONMF/L generates non-negative basis vectors. In ONMF/R, it

Table 3: Five-fold cross-validation (CV) errors (%) on the MEDLINE information retrieval dataset (30 categories, $A_{MEDLINE}$ of size $5,735 \times 696$). We used NMF/ANNLS, ONMF/L, and ONMF/R in order to obtain a k-dimensional representation. For classification, we used the 1-nearest neighbor classifier (1-NN) based on L_2-norm or cosine similarity in the reduced k-dimensional space on the MEDLINE dataset.

	$k = 30$		$k = 100$	
Methods	1-NN (L_2)	1-NN (cosine)	1-NN (L_2)	1-NN (cosine)
NMF/ANNLS	43.97%	37.93%	37.36%	30.17%
ONMF/L	38.22%	34.20%	38.21%	25.29%
ONMF/R	42.81%	37.50%	44.10%	33.34%
	$k = 150$		$k = 200$	
Methods	1-NN (L_2)	1-NN (cosine)	1-NN (L_2)	1-NN (cosine)
NMF/ANNLS	39.50%	32.90%	40.80%	30.17%
ONMF/L	41.23%	26.01%	42.24%	25.43%
ONMF/R	54.03%	36.78%	58.77%	38.36%

is hard to interpret the negative elements in the basis components of the matrix W due to the lack of intuitive meaning. On the other hand, positive coefficients in H make non-subtractive combination of basis vectors. ONMF/L and ONMF/R have a computational merit that they usually converge faster than NMF/MUR or NMF/ANNLS. In addition, they produce a better approximation of the original dataset given in its full space due to their only one-sided non-negativity constraint.

5.3 Text Classification Table 3 shows that five-fold cross-validation (CV) errors (%) on the MEDLINE information retrieval dataset. We obtained reduced k-dimensional representations of training and test data by NMF/ANNLS, ONMF/L, and ONMF/R. Then, we classified test data points by the 1-nearest neighbor classifier (1-NN) based on L_2-norm or cosine similarity in the reduced k-dimensional space. Overall, the cross-validation errors were not small. This is an expectable result since NMF does not have any discriminative power since it is an unsupervised dimension reduction algorithm. Also, one important issue is the selection of the parameter k. When the optimal value of k is significantly smaller than $\min(m, n)$, we can reduce computational costs and storage requirements by dimension reduction. LDA/EVD-QRD [16] produced lower five-fold CV errors of 20.40% and 14.79% for 1-NN based on L_2-norm and cosine similarity, respectively. This is an expectable result since it is a supervised dimension reduction that maximizes between-class scatter and minimizes within-class scatter.

In order to incorporate class information in training data, we used the following non-negative centroid dimension reduction algorithm. Given a non-negative matrix $A \in \mathbb{R}^{m \times n}$ with p classes, the non-negative centroid dimension reduction algorithm solves the following NNLS problem:

$$\min_{H_C} \|CH_C - A\|_F^2, \ \ s.t. \ H_C \geq 0, \tag{5.9}$$

where $C \in \mathbb{R}^{m \times p}$ is a non-negative centroid matrix of the given input matrix A. The non-negative lower-dimensional representation of A is given by $H_C \in \mathbb{R}^{p \times n}$. The non-negative low-rank representation $\mathbf{h} \in \mathbb{R}^{p \times 1}$ of a new test data point $\mathbf{x} \in \mathbb{R}^{m \times 1}$ is computed by solving following NNLS problem:

$$\min_{\mathbf{h}} \|C\mathbf{h} - \mathbf{x}\|_2^2, \ \ s.t. \ \mathbf{h} \geq 0. \tag{5.10}$$

Then, we can assign the new data point by using κ-nearest neighbors based on L_2-norm or cosine similarity. This algorithm is the same as the centroid dimension reduction algorithm [7] with non-negativity constraints.

Table 4 shows that five-fold CV errors (%) on the MEDLINE information retrieval dataset by using the non-negative centroid dimension reduction. We obtained better results than those of Table 3 since it takes advantage of class information. Interestingly, the best result in Table 4 yielded more favorable results than LDA/EVD-QRD on the MEDLINE dataset despite its non-negativity constraints.

6 Summary

We designed an one-sided NMF $A \approx WH$ that imposes a non-negativity constraint on W without the

Table 4: Five-fold cross-validation (CV) errors (%) on the MEDLINE information retrieval dataset (30 categories, i.e $p = 30$). We used a non-negative centroid dimension reduction to obtain a p-dimensional representation. For classification, we used the κ-nearest neighbor classifier (κ-NN) based on L_2-norm or cosine similarity in the reduced p-dimensional space.

Methods	κ-NN (L_2)	κ-NN (cosine)
$\kappa = 1$	15.09%	13.36%
$\kappa = 5$	17.96%	13.65%
$\kappa = 10$	18.25%	12.78%
$\kappa = 15$	19.83%	13.07%
$\kappa = 20$	20.11%	13.93%

non-negativity of H so as to obtain only non-negative basis vectors. Also, we devised an one-side NMF that enforces a non-negativity constraint on not W but H for the learning of non-subtractive combination of basis vectors that have arbitrary signs. Finally, we proposed a non-negative centroid dimension reduction algorithm. It is a supervised dimension reduction algorithm under non-negativity constraints. Our experiments showed that a low-rank representation obtained from this algorithm could well discriminate text documents since it is using a centroid matrix to incorporate a priori knowledge of class labels of training data.

Acknowledgment

We thank Dr. Lars Eldén for his valuable discussion. This material is based upon work supported in part by the National Science Foundation Grants CCR-0204109 and ACI-0305543.

References

[1] AT&T Laboratories Cambridge. The ORL Database of Faces, 1994.

[2] M. W. Berry, Z. Drmac, and E. R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Review*, 41:335–362, 1999.

[3] R. Bro and S. de Jong. A fast non-negativity-constrained least squares algorithm. *J. Chemometrics*, 11:393–401, 1997.

[4] J. P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proc. Natl Acad. Sci. USA*, 101(12):4164–4169, 2004.

[5] M. Chu and R. J. Plemmons. Nonnegative matrix factorization and applications, 2005. preprint.

[6] C. Ding, X. He, H. Zha, and H. D. Simon. Adaptive dimension reduction for clustering high dimensional data. In *Proc. of the 2nd IEEE Int'l Conf. Data Mining*. Maebashi, Japan, 2002.

[7] M. Jeon, H. Park, and J. B. Rosen. Dimensional reduction based on centroids and least squares for efficient processing of text data. In *Proceedings of the First SIAM International Workshop on Text Mining*. Chiago, IL, 2001.

[8] H. Kim, P. Howland, and H. Park. Dimension reduction in text classification with support vector machines. *Journal of Machine Learning Research*, 6(January):37–53, 2005.

[9] P. M. Kim and B. Tidor. Subsystem identification through dimensionality reduction of large-scale gene expression data. *Genome Research*, 13:1706–1718, 2003.

[10] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*. Prentice-Hall, Englewood Cliffs, NJ, 1974.

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

[12] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Proceedings of Neural Information Processing Systems*, pages 556–562, 2000.

[13] MATLAB. *User's Guide*. The MathWorks, Inc., Natick, MA 01760, 1992.

[14] MIT Center for Biological and Computational Learning. CBCL face database #1, 1996.

[15] P. Paatero and U. Tapper. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. *Envirometrics*, 5:111–126, 1994.

[16] C. H. Park and H. Park. A fast dimension reduction algorithm with applications on face recognition and text classification. Technical Report 03-050, Department of Computer Science and Engineering, University of Minnesota, 2003.

[17] V. P. Pauca, F. Shahnaz, M. W. Berry, and R. J. Plemmons. Text mining using non-negative matrix factorizations. In *Proc. SIAM Int'l Conf. Data Mining (SDM'04)*, April 2004.

[18] M. H. van Benthem and M. R. Keenan. Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems. *J. Chemometrics*, 18:441–450, 2004.