Semi-Supervised Nonnegative Matrix Factorization

Hyekyoung Lee, Jiho Yoo, and Seungjin Choi

***Abstract*—Nonnegative matrix factorization (NMF) is a popular method for low-rank approximation of nonnegative matrix, providing a useful tool for representation learning that is valuable for clustering and classification. When a portion of data are labeled, the performance of clustering or classification is improved if the information on class labels is incorporated into NMF. To this end, we present semi-supervised NMF (SSNMF), where we jointly incorporate the data matrix and the (partial) class label matrix into NMF. We develop multiplicative updates for SSNMF to minimize a sum of weighted residuals, each of which involves the nonnegative 2-factor decomposition of the data matrix or the label matrix, sharing a common factor matrix. Experiments on document datasets and EEG datasets in BCI competition confirm that our method improves clustering as well as classification performance, compared to the standard NMF, stressing that semi-supervised NMF yields semi-supervised feature extraction.**

***Index Terms*—Collective factorization, nonnegative matrix factorization, semi-supervised learning.**

I. Introduction

NONNEGATIVE matrix factorization (NMF) is a multivariate analysis method which was proven to be useful in learning a fruitful representation of nonnegative data such as images, spectrograms, and documents [6]. A variety of successful applications of NMF have been reported in machine learning and signal processing, especially including feature extraction for electroencephalogram (EEG) classification [2], [3] and data clustering [13].

When a portion of data are labeled, it is desirable to incorporate the information on class labels into NMF, in order to improve the classification performance. To this end, supervised NMF was studied and most of which incorporate Fisher discriminant constraints into NMF [12], [14], [16] to extract more discriminative features than the standard NMF. As in Fisher linear discriminant (FLD) which extract more discriminative features than principal component analysis (PCA), supervised NMF [12], [14], [16] were shown to perform better in classification, compared to the standard NMF.

Manuscript received April 28, 2009; revised June 16, 2009. First published July 10, 2009; current version published September 25, 2009. This work was supported by the Korea Research Foundation Grant (KRF-2008-313-D00939), the Korea Ministry of Knowledge Economy under the ITRC support program supervised by NIPA (NIPA-2009-C1090-0902-0045), the CRC for Artificial Neurosensory Device and Cognitive System, the KOSEF WCU Program (Project R31-2008-000-10100-0), and by Microsoft Research Asia.

H. Lee is with Seoul National University College of Medicine, Seoul, Korea (e-mail: leehk@postech.ac.kr).

J. Yoo and S. Choi are with the Department of Computer Science, Pohang University of Science and Technology, Pohang 790-784, Korea (e-mail: zentasis@postech.ac.kr; seungjin@postech.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2009.2027163

However, in cases where only a small number of labeled examples are available, semi-supervised learning is preferred, exploiting both (plenty of) unlabeled data and labeled data. In this paper we present a modification of NMF which jointly incorporates the data matrix and the (partial) class label matrix into NMF, referred to as *semi-supervised NMF* (SSNMF). We develop multiplicative updates for SSNMF to minimize a sum of weighted residuals, each of which involves the nonnegative 2-factor decomposition of the data matrix or the label matrix, sharing a common factor matrix. Depending on weights, our SSNMF also behaves as the standard (unsupervised) NMF or a fully-supervised NMF. Such joint factorization (a.k.a. collective factorization) of two or more matrices with a common factor matrix shared for consistency becomes important and useful in various applications of machine learning, including multilabel informed latent semantic indexing [15], relational learning [10], and group analysis [7]. Matrix factorization methods for semi-supervised clustering are also available [1], [11]. However, our method is distinguished from existing work in two aspects.

- We develop SSNMF in the framework of NMF, while existing methods are based on SVD [15] or its exponential family generalization [10].
- We employ the joint weighted factorization, introducing weights in residuals in the decomposition. Especially 0/1 weights in residuals involving the label matrix decomposition enables us to cope with partially labeled data (i.e., semi-supervised learning).
- SSNMF can use the (partial) class label explicitly, serving as *semi-supervised feature extraction*, where most of factorization methods [1], [11] exploit cannot-link or must-link constraints for semi-supervised clustering.

II. Weighted NMF

Given a nonnegative data matrix $\boldsymbol{X} \in \mathbb{R}_+^{m\times n}$, NMF seeks a 2-factor decomposition of $\boldsymbol{X}$ such that a rank-r approximation is determined by minimizing $||\boldsymbol{X}-\boldsymbol{AS}||^2$ where factor matrices $\boldsymbol{A} \in \mathbb{R}_+^{m\times r}$ and $\boldsymbol{S} \in \mathbb{R}_+^{r\times n}$ are required to be nonnegative matrices as well. When columns in $\boldsymbol{X}$ are treated as data points in m-dimensional space, columns in $\boldsymbol{A}$ are considered as *basis vectors* and each column in $\boldsymbol{S}$ is *encoding* that represents the extent to which each basis vector is used to reconstruct data vectors. In such a case, $\boldsymbol{A}$ is referred to as *basis matrix* and $\boldsymbol{S}$ is called as *encoding matrix* or *feature matrix* since column vectors correspond to features in classification.

In practice, the data matrix is often incomplete, i.e., some of entries are missing or unobserved. In order to handle missing entries in the decomposition, we consider an objective function that is a sum of weighted residuals, as in [5], [9],

$$\mathcal{J} = \sum_{i,j} W_{ij}\left(X_{ij} - [\boldsymbol{AS}]_{ij}\right)^2 = ||\boldsymbol{W} \odot (\boldsymbol{X} - \boldsymbol{AS})||^2 \quad (1)$$

where W_{ij} are binary weights, i.e.,

$$W_{ij} = \begin{cases} 1, & \text{if } X_{ij} \text{ is observed} \\ 0, & \text{if } X_{ij} \text{ is unobserved.} \end{cases}$$

Multiplicative updates for $\boldsymbol{A}$ and $\boldsymbol{S}$ which minimize (1) are derived as follows. Suppose that the gradient of an error function has a decomposition that is of the form

$$\nabla \mathcal{J} = [\nabla \mathcal{J}]^+ - [\nabla \mathcal{J}]^- \tag{2}$$

where $[\nabla \mathcal{J}]^+ > 0$ and $[\nabla \mathcal{J}]^- > 0$. Then multiplicative update for parameters Θ has the form

$$\Theta \leftarrow \Theta \odot \left(\frac{[\nabla \mathcal{J}]^-}{[\nabla \mathcal{J}]^+} \right). \tag{3}$$

It can be easily seen that the multiplicative update (3) preserves the nonnegativity of the parameter Θ, while $\nabla \mathcal{J} = 0$ when the convergence is achieved.

Derivatives of the error function (1) with respect to $\boldsymbol{A}$ and $\boldsymbol{S}$, are given by

$$\begin{aligned} \nabla_A \mathcal{J} &= [\nabla_A \mathcal{J}]^+ - [\nabla_A \mathcal{J}]^- \\ &= [\boldsymbol{W} \odot \boldsymbol{AS}]\boldsymbol{S}^\top - [\boldsymbol{W} \odot \boldsymbol{X}]\boldsymbol{S}^\top \qquad (4) \\ \nabla_S \mathcal{J} &= [\nabla_S \mathcal{J}]^+ - [\nabla_S \mathcal{J}]^- \\ &= \boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{AS}] - \boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{X}]. \qquad (5) \end{aligned}$$

With these gradient calculations, invoking (3) yields multiplicative updates for WNMF

$$\boldsymbol{A} \leftarrow \boldsymbol{A} \odot \frac{[\boldsymbol{W} \odot \boldsymbol{X}]\boldsymbol{S}^\top}{[\boldsymbol{W} \odot \boldsymbol{AS}]\boldsymbol{S}^\top} \tag{6}$$

$$\boldsymbol{S} \leftarrow \boldsymbol{S} \odot \frac{\boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{X}]}{\boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{AS}]}. \tag{7}$$

III. Semi-Supervised NMF

Suppose that the data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}_+^{m \times n}$ consists of m-dimensional nonnegative data vectors, each of which belongs to one of k classes. Associated labels are encoded in the label matrix $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n] \in \mathbb{R}_+^{k \times n}$, where each $\boldsymbol{y}_i$ is a binary vector such that only jth entry is 1 and remaining elements are zero if $\boldsymbol{x}_i$ belongs to class j.

We consider a joint factorization of the data matrix $\boldsymbol{X}$ and the label matrix $\boldsymbol{Y}$, sharing a common factor matrix $\boldsymbol{S}$

$$\widetilde{\mathcal{J}} = \|\boldsymbol{W} \odot (\boldsymbol{X} - \boldsymbol{AS})\|^2 + \lambda \|\boldsymbol{L} \odot (\boldsymbol{Y} - \boldsymbol{BS})\|^2 \; eqno(8)$$

where λ is a tradeoff parameter determining the importance of the supervised term, $\boldsymbol{A} \in \mathbb{R}^{m \times r}$ and $\boldsymbol{B} \in \mathbb{R}^{k \times r}$ are basis matrices for $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively, and $\boldsymbol{L} \in \mathbb{R}^{k \times n}$ is a weight matrix to handle missing labels. Columns of $\boldsymbol{L}$ are filled with either 1 or 0, depending on the availability of labels of corresponding data points, i.e.,

$$[\boldsymbol{L}]_{:,j} = \begin{cases} \mathbf{1}_k, & \text{if the label of } \boldsymbol{x}_j \text{ is known} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $\mathbf{1}_k = [1, \ldots, 1]^\top \in \mathbb{R}^k$.

As in the standard NMF, multiplicative updates for $\boldsymbol{A}$, $\boldsymbol{S}$ and $\boldsymbol{B}$ for minimizing (8) are easily derived, which are summarized below.

Algorithm Outline: **SSNMF**

$$\boldsymbol{A} \leftarrow \boldsymbol{A} \odot \frac{[\boldsymbol{W} \odot \boldsymbol{X}]\boldsymbol{S}^\top}{[\boldsymbol{W} \odot \boldsymbol{AS}]\boldsymbol{S}^\top} \tag{10}$$

$$\boldsymbol{B} \leftarrow \boldsymbol{B} \odot \frac{[\boldsymbol{L} \odot \boldsymbol{Y}]\boldsymbol{S}^\top}{[\boldsymbol{L} \odot \boldsymbol{BS}]\boldsymbol{S}^\top} \tag{11}$$

$$\boldsymbol{S} \leftarrow \boldsymbol{S} \odot \frac{\boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{X}] + \lambda \boldsymbol{B}^\top [\boldsymbol{L} \odot \boldsymbol{Y}]}{\boldsymbol{A}^\top [\boldsymbol{W} \odot \boldsymbol{AS}] + \lambda \boldsymbol{B}^\top [\boldsymbol{L} \odot \boldsymbol{BS}]}. \tag{12}$$

The computational complexity of SSNMF is $\mathcal{O}((m+k)nr)$ while NMF requires $\mathcal{O}(mnr)$. Since, in general, $m \gg k$, the computational complexity of SSNMF reduces down to $\mathcal{O}(mnr)$.

We denote by $L^{(1)}$ the weighted matrix introduced in (9). Alternatively we make use of a different weight matrix given by

$$L_{ij}^{(2)} = \begin{cases} 0.001, & \text{if } Y_{ij} = 1 \\ 1 & \text{if, } Y_{ij} = 0 \\ 0, & \text{if } Y_{ij} \text{ is unknown.} \end{cases} \tag{13}$$

The basic idea behind is as follows. For minimization of (8), $[\boldsymbol{BS}]_{ij}$ is forced to be $Y_{ij} = 1$ or 0. When $Y_{ij} = 1$, however, $[\boldsymbol{BS}]_{ij}$ does not need to be exactly equal to 1 (nonzero constant is fine), while $[\boldsymbol{BS}]_{ij}$ is desirable to be close to 0 for $Y_{ij} = 0$. Thus the weights in (13) de-emphasize the case for $Y_{ij} = 1$. Empirically this further improves the performance of SSNMF, as will be shown in Section IV. When $\boldsymbol{W} = \mathbf{1}_m \mathbf{1}_n^\top$ and $\boldsymbol{L} = 0$, SSNMF reduces to the standard NMF. When $\boldsymbol{L} = \mathbf{1}_k \mathbf{1}_n^\top$, SSNMF fully makes use of labeled data.

IV. Numerical Experiments

Our method, SSNMF, is compared to NMF with Fisher discriminant constraints (such as Fisher NMF [12]), in a task of supervised feature extraction for classification and is compared to the standard NMF and the harmonic function [17] in a task of semi-supervised clustering in order to show how much partially-labeled data improves the performance. Two tasks considered here are listed.

- Supervised feature extraction for classification
  - — Partition the dataset $\boldsymbol{X}$ into a training data $\boldsymbol{X}_{\text{train}}$ and a test data $\boldsymbol{X}_{\text{test}}$. Labels of all the data points in $\boldsymbol{X}_{\text{train}}$ are assumed to be known in advance. That is, we set $\boldsymbol{L} = \mathbf{1}_{k \times n_{\text{train}}}$.
  - — Feature matrices $\boldsymbol{S}_{\text{train}}$ and $\boldsymbol{S}_{\text{test}}$ are computed by $\boldsymbol{S}_{\text{train}} = \boldsymbol{A}^\dagger \boldsymbol{X}_{\text{train}}$ and $\boldsymbol{S}_{\text{test}} = \boldsymbol{A}^\dagger \boldsymbol{X}_{\text{test}}$, where $\dagger$ denotes the pseudoinversion.
  - — $\boldsymbol{S}_{\text{train}}$ are used to train a classifier (support vector machine (SVM) or Viterbi algorithm) and classification of $\boldsymbol{X}_{\text{test}}$ is performed by feeding $\boldsymbol{S}_{\text{test}}$ into a trained classifier.

TABLE I
CLUSTERING ACCURACY (AVERAGED OVER 20 INDEPENDENT TRIALS) IS SUMMARIZED AS THE PORTION OF LABELED DATA INCREASES FROM 0% TO 50% (SUPERVISED)

Datasets	# docs	# classes	**0 %**	**10 %**	**20 %**	**30 %**	**40 %**	**50 %**
20-news	2000	20	40.75	51.29	64.33	74.70	83.38	89.13
CSTR	600	4	77.33	86.48	90.99	93.19	94.43	95.43
k1a	2340	20	50.17	65.05	75.59	87.49	94.32	95.92
k1b	2340	6	82.44	89.48	92.54	93.85	94.94	95.28
re0	1504	13	47.52	57.39	67.27	79.53	85.74	89.81
Reut10	2184	10	65.80	79.51	87.57	90.90	92.74	94.29
wap	1560	20	49.68	59.35	64.92	70.63	91.08	92.47
WebKB4	1200	4	56.33	70.38	79.50	83.89	86.71	89.03

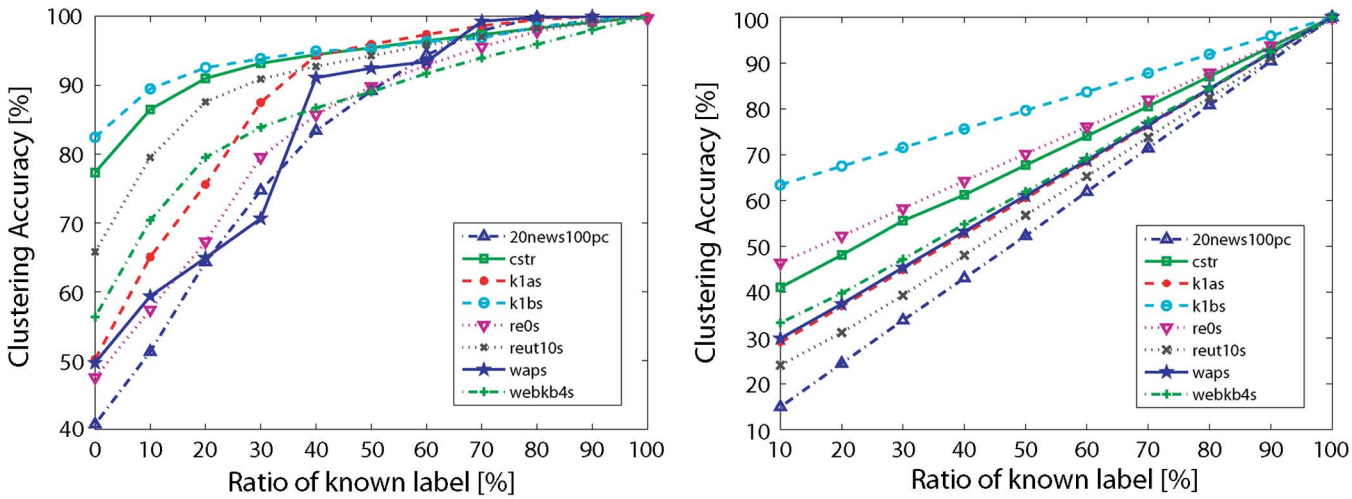

Fig. 1. Clustering accuracy (averaged over 50 independent trials) is shown as the portion of labeled data increases from 0% (unsupervised) to 100% (supervised): (a) our method, SSNMF and (b) label propagation [17].

- Semi-supervised clustering
  - — Construct the data matrix $\boldsymbol{X} \in \mathbb{R}_+^{m\times n}$ and the associated weight matrix $\boldsymbol{W}$, depending on the availability of data points.
  - — The weight matrix $\boldsymbol{L}$ is set by $\boldsymbol{L}^{(1)}$ in (9) or $\boldsymbol{L}^{(2)}$ in (13), in order to reflect only available labels.
  - — Apply k-means clustering to the feature matrix $\boldsymbol{S}$ learned by SSNMF or NMF.

Experiments were carried out on several datasets, including document datasets and EEG datasets. The trade-off parameter, λ is chosen by cross validation. If $\lambda = 0$, SSNMF is the same with the original NMF. We can obtain slightly better performance than original NMF if λ is selected between 0.5 and 3. We set $\lambda = 1$ for all experiments.

A. Document Clustering

1) Document Data Matrix: In the vector-space model of text data, each document is represented by an m-dimensional vector $\boldsymbol{x}_t \in \mathbb{R}^m$, where m is the number of terms in the dictionary. Given n documents, we construct a term-document matrix $\boldsymbol{X} \in \mathbb{R}^{m\times n}$ where X_{ij} corresponds to the significance of term t_i in document d_j [13] that is calculated by

$$X_{ij} = \mathrm{TF}_{ij} \log\left(\frac{n}{\mathrm{DF}_i}\right)$$

where TF_{ij} denotes the frequency of term t_i in document d_j and DF_i represents the number of documents containing term t_i. Elements X_{ij} are always nonnegative and equal zero only when corresponding terms do not appear in the document.

2) Datasets: We used eight document datasets, the statistics of which is summarized in Table I and detailed description for the datasets are listed below.

- *CSTR* is a dataset from a list of computer science technical reports on the homepage of the Computer Science department of University of Rochester (http://www.cs.rochester.edu/trs/).
- *WebKB4* is a dataset consisting of the webpages from computer science department of various universities (http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/).
- *wap*, *k1a* and *k1b* datasets consist of web pages in various subject directories of Yahoo! (http://www.yahoo.com/).
- *re0* and *Reut10* datasets are from the news articles of the Reuters newswire (http://www.daviddlewis.com/resources/testcollections/reuters 21578/).
- *20-News* dataset is from 20 Newsgroup dataset (http://people.csail.mit.edu/jrennie/20Newsgroups/).

We apply SSNMF($\boldsymbol{L}^{(2)}$) to the task of semi-supervised document clustering, where we set r to be equal to the number of classes in the decomposition. If we denote the true label for the ith data to be c_i, and the estimated label $\hat{c}_i$, the accuracy can be computed by $\sum_{i=1}^{n}(\delta(c_i,\hat{c}_i)/n)$, where where $\delta(x,y) = 1$ for $x = y$ and $\delta(x,y) = 0$ for $x \neq y$. We evaluate the performance in terms of clustering accuracy (using known labels) by increasing the portion of labeled data (from 0% to 100%), which is summarized in Table I and Fig. 1(a). We select labeled data

TABLE II
CLASSIFICATION ACCURACY AVERAGED OVER 10 INDEPENDENT TRIALS IS SHOWN FOR VARIOUS NMF METHODS. (THE PORTION OF UNLABELED DATA IS 25%)

[%]	NMF	Local NMF	Fisher NMF	SSNMF ($\boldsymbol{L}^{(1)}$)	SSNMF ($\boldsymbol{L}^{(2)}$)	BCI comp. winner
Sub1	79.03 ($\pm$ 0.53)	78.11 ($\pm$ 0.84)	79.35 ($\pm$ 0.18)	78.49 ($\pm$ 3.49)	**85.62** ($\pm$ 3.02)	79.60
Sub2	69.13 ($\pm$ 2.23)	65.13 ($\pm$ 0.13)	67.45 ($\pm$ 2.24)	76.24 ($\pm$ 3.44)	**78.61** ($\pm$ 0.76)	70.31
Sub3	49.14 ($\pm$ 1.04)	50.54 ($\pm$ 1.24)	48.52 ($\pm$ 1.26)	64.51 ($\pm$ 2.17)	**65.10** ($\pm$ 1.63)	56.02
Avg.	65.77 ($\pm$ 2.29)	64.59 ($\pm$ 0.85)	65.11 ($\pm$ 2.25)	73.08 ($\pm$ 3.03)	**76.44** ($\pm$ 3.11)	68.65

uniformly. We compare our method to a representative semi-supervised learning method [17], which is shown in Fig. 1(b). Performance is evaluated in terms of clustering accuracy (which is nothing but classification accuracy since ground truth is known in these datasets) Clustering accuracy of SSNMF goes beyond 90% across document datasets, if 40% labeled data is used.

B. EEG Classification

NMF was shown to be useful in extracting discriminative features from EEG data [3]. For our empirical study, we use the dataset V in BCI competition III, which was provided by the IDIAP Research Institute [4]. *IDIAP dataset* contains EEG data recorded from 3 normal subjects and involves three tasks, including the imagination of repetitive self-paced left/right hand movements and the generation of words beginning with the same random letter.

We use the precomputed features which were obtained by the power spectral density (PSD) in the band 8–30 Hz every 62.5 ms, (i.e., 16 times per second) over the last second of data with a frequency resolution of 2 Hz for the eight centro-parietal channels C_3, C_z, C_4, CP_1, CP_2, P_3, P_z, and P_4.

Spectral components $\bar{P}_{f,t}^{(k)} \in \mathbb{R}^{12\times 10528}$ is the normalized precomputed features satisfying $\sum_f \bar{P}_{f,t}^{(k)} = 1$ for $f \in \{8, 10, \ldots, 28, 30\}$ Hz, $k = 1, \ldots, 8$ (eight different channels), and $t = 1, \ldots, 10528$ where 10528 is the number of data points in the training set. Then we construct the training data matrix $\boldsymbol{X}_{\text{train}} \in \mathbb{R}^{96\times 10528}$ by collecting 12×10528 spectral matrices computed at eight different channels

$$\boldsymbol{X} = \left[\bar{\boldsymbol{P}}^{(1)}; \bar{\boldsymbol{P}}^{(2)}; \ldots; \bar{\boldsymbol{P}}^{(8)}\right] \in \mathbb{R}^{m\times n} \tag{14}$$

where $m = 12 \times 8$. In the same way, we construct the test data matrix, $\boldsymbol{X}_{\text{test}} \in \mathbb{R}^{96\times 3504}$. Classification results are summarized in Table II, where the Viterbi classifier [8] is used.

V. CONCLUSIONS

We have presented a semi-supervised version of NMF which jointly exploited both (partial) labeled and unlabeled data to extract more discriminative features than the standard NMF. We formulated SSNMF as a joint factorization of the data matrix and the label matrix, sharing a common factor matrix $\boldsymbol{S}$ for consistency. Weighted residuals for the decomposition of the data matrix and the label matrix were introduced to handle missing data and to incorporate partially labeled data, respectively. Extensive numerical experiments confirmed that: 1) SSNMF performed better than existing supervised NMF methods based on Fisher discriminant constraints in the task of semi-supervised or supervised feature extraction and 2) SSNMF improved clustering performance by incorporating labeled data samples.

REFERENCES

[1] Y. Chen, M. Rege, M. Dong, and J. Hua, "Non-negative matrix factorization for semi-supervised data clustering," *Knowl. Inform. Syst.*, vol. 17, pp. 355–379.

[2] A. Cichocki, H. Lee, Y. D. Kim, and S. Choi, "Nonnegative matrix factorization with α-divergence," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1433–1440, 2008.

[3] A. Cichocki, Y. Washizawa, T. Rutkowski, H. Bakardjian, A. H. Phan, S. Choi, H. Lee, Q. Zhao, L. Zhang, and Y. Li, "Noninvasive BCIs: Multiway signal-processing array decompositions," *IEEE Computer*, vol. 41, no. 10, pp. 34–42, Oct. 2008.

[4] J. del R. Millán, "On the need for on-line learning in brain-computer interfaces," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, Budapest, Hungary, 2004.

[5] Y. D. Kim and S. Choi, "Weighted nonnegative matrix factorization," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, R.O.C., 2009.

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

[7] H. Lee and S. Choi, "Group nonnegative matrix factorization for EEG classification," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, Clearwater Beach, FL, 2009.

[8] H. Lee, Y. D. Kim, A. Cichocki, and S. Choi, "Nonnegative tensor factorization for continuous EEG classification," *Int. J. Neural Syst.*, vol. 17, no. 4, pp. 305–317, 2007.

[9] Y. Mao and L. K. Saul, "Modeling distances in large-scale networks by matrix factorization," in *Proc. ACM Internet Measurement Conf.*, Taormina, Sicily, Italy, 2004.

[10] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD)*, Las Vegas, NV, 2008.

[11] F. Wang, T. Li, and C. Zhang, "Semi-supervised clustering via matrix factorization," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Atlanta, GA, 2008.

[12] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Fisher non-negative matrix factorization for learning local features," in *Proc. Asian Conf. Computer Vision (ACCV)*, Jeju Island, Korea, 2004.

[13] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proc. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, Toronto, ON, Canada, 2003.

[14] Y. Xue, C. S. Tong, W. S. Chen, W. Zhang, and Z. He, "A modified non-negative matrix factorization algorithm for face recognition," in *Proc. Int. Conf. Pattern Recognition (ICPR)*, Hong Kong, 2006, pp. 495–498.

[15] K. Yu, S. Yu, and V. Tresp, "Multi-label informed latent semantic indexing," in *Proc. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, Salvador, Brazil, 2005.

[16] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 683–695, May 2006.

[17] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Proc. Int. Conf. Machine Learning (ICML)*, 2003, pp. 912–919.