FLEXIBLE NON-NEGATIVE MATRIX FACTORIZATION WITH ADAPTIVELY LEARNED GRAPH REGULARIZATION

Yong Peng[1,2,*], *Yanfang Long*[1], *Feiwei Qin*[1], *Wanzeng Kong*[1], *Feiping Nie*[3] *and Andrzej Cichocki*[4,1]

[1] School of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Guangxi Key Laboratory of Multi-source Information Mining & Security, Guilin 541004, China
[3]Center for OPTIMAL, Northwestern Polytechnical University, Xi'an 710072, China
[4]Skolkovo Institute of Science and Technology (SKOLTECH), Moscow 143026, Russia
yongpeng@hdu.edu.cn

ABSTRACT

Non-negative matrix factorization (NMF) is an efficient model in learning parts-based data representation. Since the local geometrical structure can be effectively modeled by a nearest neighbor graph, the graph regularized NMF (GNMF) was proposed to make the learned representation more faithfully and better characterize the intrinsic structure of data. However, GNMF shares a similar paradigm with most of existing graph-based learning models which perform learning tasks on a fixed input graph. In this paper, we propose a new Flexible NMF model with adaptively learned Graph regularization (FNMFG) in which the graph is jointly learned with simultaneous performing the matrix factorization. An efficient iterative method with guaranteed convergence and relative low complexity is developed to optimize the FNMFG objective. Experiments compare FNMFG method with state-of-the-art algorithms and demonstrate its improved performance.

Index Terms— Non-negative matrix factorization, adaptive graph learning, clustering

1. INTRODUCTION

Matrix factorization techniques are popular to learning efficient representation of high dimensional data in many fields such as computer vision and data mining [1, 2, 3, 4]. Inspired by the psychological and physiological research finding that there exists parts-based representation in human neural systems, non-negative matrix factorization (NMF) was proposed to learn the parts-based data representation for pattern recognition [1]. Mathematically, NMF tries to minimize the approximation error between the target matrix and the product of two non-negative factor matrices. The non-negative constraint on factor matrices allows only the additive combinations and thus leads to the parts-based representation. A lot of studies were conducted recently on both new NMF models and applications [5, 3, 4, 6, 7, 8, 9].

Recent studies have shown that the learning performance can be significantly enhanced if the geometrical structure of data can be exploited and suitable regularization implemented [10, 11, 12, 13]. To incorporate the local invariance idea into NMF model, Cai *et al.* proposed the graph regularized NMF (GNMF) model in which the data geometrical information was characterized by a nearest neighbor graph and preserved by the graph regularization [14]. The underlying justification of the graph regularization is that the corresponding coefficient vectors of two data points should be similar if they are close to each other in data space (e.g., connected in the graph).

Although GNMF provided improved performance in many applications, it suffers from a drawback that the graph is constructed by fixed rules such as 'Heat Kernel' and '0-1' weighting. It cannot adapt to various data sets, especially when data is noisy. Therefore, it is meaningful to jointly learn an adaptive graph from data and use it to regularize the NMF. In this work, we propose a flexible NMF with adaptively learned graph regularization (FNMFG) model in which we simultaneously perform the graph learning and matrix factorization tasks. An efficient iterative method with guaranteed convergence and relative low complexity is designed to optimize the FNMFG objective. Extensive experiments are conducted to demonstrate the excellent performance of FNMFG. In conclusion we summarize results and indicate future research directions.

2. THE PROPOSED FNMFG MODEL

2.1. Model Formulation

As shown in [14], the objective of the standard GNMF is

$$\min_{\mathbf{U}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0}} \|\mathbf{X}-\mathbf{U}\mathbf{V}^T\|^2+\lambda\mathrm{Tr}(\mathbf{V}^T\mathbf{L}\mathbf{V}), \qquad (1)$$

This work was supported by NSFC (61602140,61671193,61633010), Zhejiang Science & Technology Program (2017C33049,2018C04012), China Postdoctoral Science Foundation (2017M620470), Ministry of Education and Science of the Russian Federation (14.756.31.0001), Jiangsu Key Lab. of Big Data Security & Intelligent Processing (BDSIP201804), Co-Innovation Center for Information Supply & Assurance Technology, Anhui University (ADXXBZ201704), and Guangxi Key Laboratory of Multi-source Information Mining & Security (MIMS18-06).

where $\mathbf{X} \in \mathbb{R}^{d\times n}$, $\mathbf{U} \in \mathbb{R}^{d\times c}$, $\mathbf{V} \in \mathbb{R}^{n\times c}$, $\mathbf{L} = \mathbf{D}_{\mathbf{W}} - \mathbf{W}$ is the graph Laplacian matrix corresponding to the affinity matrix $\mathbf{W}$ (d is the dimensionality, n is the number of samples, c is the number of clusters).

For graph-based learning methods, the affinity matrix $\mathbf{W}$ has a significant influence on their performance. Under some constraints, $\mathbf{W}$ can be viewed as a probability matrix. If data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar to each other, a large value will be assigned to w_{ij} meaning that $\mathbf{x}_i$ has a higher probability to be neighbor of $\mathbf{x}_j$. However, the $\mathbf{W}$ in GNMF is not an optimal graph for characterizing the complex intrinsic structure of data and it may even mislead the subsequent coefficient matrix learning. In this paper, we propose to learn an optimal graph $\mathbf{S}$ based on the affinity matrix $\mathbf{W}$. That is, we assume that $\mathbf{S}$ will approximate $\mathbf{W}$ but with suitable properties and constraints including **non-negativity**, **row-sum-to-one** and **constrained rank** [15]. The normalization constraint needs the sum of entries in each row of $\mathbf{S}$ to be one. The third constraint means that the graph Laplacian $\mathbf{L_S} = \mathbf{D_S} - \frac{\mathbf{S}+\mathbf{S}^T}{2}$ should satisfy rank$(\mathbf{L_S}) = n - c$, when $\mathbf{S}$ is expected to have exactly c block diagonals. In other words, we formulate the following optimization problem

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geq\mathbf{0}} \|\mathbf{S}-\mathbf{W}\|_F^2,\ s.t.\ \text{rank}(\mathbf{L_S}) = n - c. \quad (2)$$

Since it is difficult to deal with the rank constraint directly on the graph, we need to reformulate (2) as a tractable one. Let $\sigma_i(\mathbf{L_S}) \geq 0$ be the i-th smallest eigenvalue of $\mathbf{L_S}$. Then the constraint rank$(\mathbf{L_S}) = n - c$ could be satisfied if the smallest c eigenvalues of $\mathbf{L_S}$ are zero. Given a large enough value γ, problem (2) is equivalent to

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geq\mathbf{0}} \|\mathbf{S}-\mathbf{W}\|_F^2 + \gamma \sum\nolimits_{i=1}^{c} \sigma_i(\mathbf{L_S}). \quad (3)$$

According to its definition, NMF can be roughly viewed as a feature mapping model in which coefficient vector as the feature of each data point. Ideally, each column of the basis matrix $\mathbf{U}$ can represent the main characteristics of each cluster and then the coefficient matrix $\mathbf{V}$ should have sparse discriminative structure to be utilized for clustering [16]. According to Ky Fan's Theorem [17], we can rewrite the rank constraint in (3) and formulate our proposed flexible GNMF with adaptive graph learning (FNMFG) model as follows

$$\min_{\mathbf{S},\mathbf{U},\mathbf{V}} \|\mathbf{S}-\mathbf{W}\|_F^2 + \lambda\|\mathbf{X}-\mathbf{U}\mathbf{V}^T\|_F^2 + \gamma\text{Tr}(\mathbf{V}^T\mathbf{L_S}\mathbf{V})$$
$$s.t.\ \mathbf{S}\geq\mathbf{0}, \mathbf{S1}=\mathbf{1}, \mathbf{U}\geq\mathbf{0}, \mathbf{V}\geq\mathbf{0}, \mathbf{V}^T\mathbf{V}=\mathbf{I}_c,$$

where parameter λ controls the approximation error of NMF, and γ controls how well the coefficient matrix fits the geometrical structure of the data.

2.2. Optimization Procedure

In out approach, we use the alternating direction method (ADM) to optimize (4), that is, we update one variable with the others fixed. Detailed descriptions are given below.

1) Update $\mathbf{S}$. Since $\mathbf{S}$ is represented by $\mathbf{L_S}$, we need to decouple the third term in (4), so the objective associated with $\mathbf{S}$ can be written as

$$\min_{s_{ij}\geq 0, \sum_j s_{ij}=1} \sum_{i,j=1}^{n} (s_{ij}-w_{ij})^2 + \frac{\gamma}{2} \sum_{i,j=1}^{n} \|\mathbf{v}_i - \mathbf{v}_j\|^2 s_{ij}. \quad (4)$$

For each $\mathbf{s}_i$ and $\mathbf{w}_i$, we have

$$\min_{s_{ij}\geq 0, \mathbf{s}_i\mathbf{1}=1} \|\mathbf{s}_i - \mathbf{w}_i\|_2^2 + \frac{\gamma}{2}\mathbf{s}_i\mathbf{d}_i^T, \quad (5)$$

where $\mathbf{s}_i = [s_{i1}, \cdots, s_{in}]$ and $\mathbf{w}_i = [w_{i1}, \cdots, w_{in}]$ are both n-dimensional row vectors of $\mathbf{S}$ and $\mathbf{W}$, respectively. The j-th entry of $\mathbf{d}_i$ is $\|\mathbf{v}_i - \mathbf{v}_j\|_2^2$. Transforming the square form w.r.t. $\mathbf{s}_i$, (5) can be rewritten in equivalent form as

$$\min_{\mathbf{s}_i\geq\mathbf{0}, \mathbf{s}_i\mathbf{1}=1} \|\mathbf{s}_i - (\mathbf{w}_i - \frac{\gamma}{4}\mathbf{d}_i)\|_2^2. \quad (6)$$

This optimization problem can be solved with a closed from solution by an efficient iterative algorithm [18].

2) Update $\mathbf{U}$. The objective associated with matrix $\mathbf{U}$ is the same as NMF and GNMF. Therefore, we directly have the standard form of updating rule below.

$$u_{ik} \leftarrow (u_{ik}(\mathbf{X}\mathbf{V})_{ik})/(\mathbf{U}\mathbf{V}^T\mathbf{V})_{ik}, \quad (7)$$

3) Update $\mathbf{V}$. In order to eliminate the orthogonal constraint, we add a penalty term $\frac{\delta}{2}\|\mathbf{V}^T\mathbf{V}-\mathbf{I}\|^2$ in which $\delta \gg 0$ is a large value (we set it to 10^7 in our experiments). Therefore, we have the objective function associated with $\mathbf{V}$ as

$$\min_{\mathbf{V}\geq\mathbf{0}} \lambda\|\mathbf{X}-\mathbf{U}\mathbf{V}^T\|^2 + \gamma\text{Tr}(\mathbf{V}^T\mathbf{L_S}\mathbf{V}) + \frac{\delta}{2}\|\mathbf{V}^T\mathbf{V}-\mathbf{I}\|^2. \quad (8)$$

The corresponding Lagrangian function $\mathcal{L}$ is

$$\min_{\mathbf{V}} \lambda\|\mathbf{X}-\mathbf{U}\mathbf{V}^T\|^2 + \gamma\text{Tr}(\mathbf{V}^T\mathbf{L_S}\mathbf{V}) + \frac{\delta}{2}\|\mathbf{V}^T\mathbf{V}-\mathbf{I}\|^2 + \text{Tr}(\mathbf{\Phi}^T\mathbf{V}).$$

Taking its derivative w.r.t. $\mathbf{V}$ and setting it to zero, we have

$$2\lambda(\mathbf{X}^T - \mathbf{V}\mathbf{U}^T)\mathbf{U} + 2\gamma\mathbf{L_S}\mathbf{V} + 2\delta\mathbf{V}(\mathbf{V}^T\mathbf{V} - \mathbf{I}) + \mathbf{\Phi} = \mathbf{0}.$$

Using KKT condition $\phi_{ij}v_{ij} = 0$, we obtain updating of $\mathbf{V}$ as

$$v_{ij} \leftarrow v_{ij}\frac{(\lambda\mathbf{X}^T\mathbf{U} + \gamma\mathbf{S}\mathbf{V} + \delta\mathbf{V})_{ij}}{(\lambda\mathbf{V}\mathbf{U}^T\mathbf{U} + \gamma\mathbf{D_S}\mathbf{V} + \delta\mathbf{V}\mathbf{V}^T\mathbf{V})_{ij}}. \quad (9)$$

After updating $\mathbf{V}$, we need to normalize it to satisfy the orthogonal constraint $\mathbf{V}^T\mathbf{V} = \mathbf{I}$.

Based on the above analysis, we summarize the optimization procedure for FNMFG in Algorithm 1.

2.3. Complexity and Convergence Analysis

The complexity of Algorithm 1 is determined by the loop which consists of three blocks. We need $O(nt_1)$ operations to

Table 1. Computation operations for each iteration in standard NMF, GNMF and new FNMFG for updating $\mathbf{U}$ and $\mathbf{V}$.

	fladd	flmlt	fldiv	overall
NMF	$2dnc + 2(d+n)c^2$	$2dnc + 2(d+n)c^2 + (d+n)c$	$(d+n)c$	$\mathcal{O}(dnc)$
GNMF	$2dnc + 2(d+n)c^2 + n(p+3)c$	$2dnc + 2(d+n)c^2 + (d+n)c + n(p+1)c$	$(d+n)c$	$O(dnc)$
FNMFG	$2dnc + 2(d+n)c^2 + n(\frac{n}{c}+5)c + 2nc^2$	$2dnc + 2(d+n)c^2 + (d+n)c + n(\frac{n}{c}+1)c + 2nc^2$	$(d+n)c$	$O(dnc)$

Algorithm 1 Optimization to FNMFG objective in (4).

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$, given affinity matrix $\mathbf{W} \in \mathbb{R}^{n\times n}$, parameters λ and γ, and the number of clusters c;

Output: basis matrix $\mathbf{U}$, and coefficient matrix $\mathbf{V}$.

1: Initialize $\mathbf{W}$ by 'Heatkernel' function in which the bandwidth parameter is set as the average of the squared pairwise distances, and $\mathbf{U}$, $\mathbf{V}$ are initialized randomly;
2: **while** not converged **do**
3: Update $\mathbf{S}$ by (6);
4: Update $\mathbf{U}$ by (7);
5: Update $\mathbf{V}$ by (9);
6: **end while**

obtain the desirable affinity matrix $\mathbf{S}$ by an efficient iterative method in which t_1 is the number of iterations of the Newton method. For the updating rules to $\mathbf{U}$ and $\mathbf{V}$, we count the basic arithmetic operations for NMF, GNMF and FNMFG in Table 1 where p is the number of nearest neighbors in GNMF, *fladd*, *flmlt* and *fldiv* respectively mean the *floating-point addition*, *floating-point multiplication* and *floating-point division*. In GNMF, $\mathbf{W}$ is a p-sparse matrix, while $\mathbf{S}$ in FNMFG is an $\frac{n}{c}$-sparse matrix on average (Each cluster has $\frac{n}{c}$ samples on average). Considering that typically $c \ll d$ and $d \ll n$, we have the overall complexity of updating $\mathbf{U}$ and $\mathbf{V}$ in each iteration as $O(dnc)$. As a whole, the complexity for FNMFG is $O(t(nt_1 + dnc))$ where t is the number of iterations.

Regarding the convergence of Algorithm 1, we have the following theorem.

Theorem 1 *The objective of FNMFG in (4) is nonincreasing under the updating rules in Algorithm 1.*

Due to the limited space, we only give some analysis instead of a strict proof to Theorem 1. Given a fixed point $(\mathbf{U}^t, \mathbf{V}^t)$, it is obvious that $\mathcal{O}(\mathbf{S}^{t+1}, \mathbf{U}^t, \mathbf{V}^t) \leq \mathcal{O}(\mathbf{S}^t, \mathbf{U}^t, \mathbf{V}^t)$ since the updating of $\mathbf{S}$ can get a closed form solution. The updating rule to $\mathbf{U}$ is exactly the same as that of standard NMF and GNMF and thus we can use the convergence proof of NMF to show that $\mathcal{O}(\mathbf{S}^{t+1}, \mathbf{U}^{t+1}, \mathbf{V}^t) \leq \mathcal{O}(\mathbf{S}^{t+1}, \mathbf{U}^t, \mathbf{V}^t)$. With $\mathbf{S}^{t+1}$ and $\mathbf{U}^{t+1}$ fixed, by introducing an auxiliary function for (8) as in [1], we can easily prove that $\mathcal{O}(\mathbf{V}^{t+1}) \leq \mathcal{O}(\mathbf{V}^t)$. So finally, we have $\mathcal{O}(\mathbf{S}^{t+1}, \mathbf{U}^{t+1}, \mathbf{V}^{t+1}) \leq \mathcal{O}(\mathbf{S}^t, \mathbf{U}^t, \mathbf{V}^t)$.

3. EXPERIMENTS

3.1. Experimental Settings

Six representative benchmark data sets were used in following experiments including one object data set (COIL20), three face image data sets (UMIST, PIE, AR), one text data set (TDT2), and one hand written digit data set (MNIST). Table 2 shows the detailed information of these data sets.

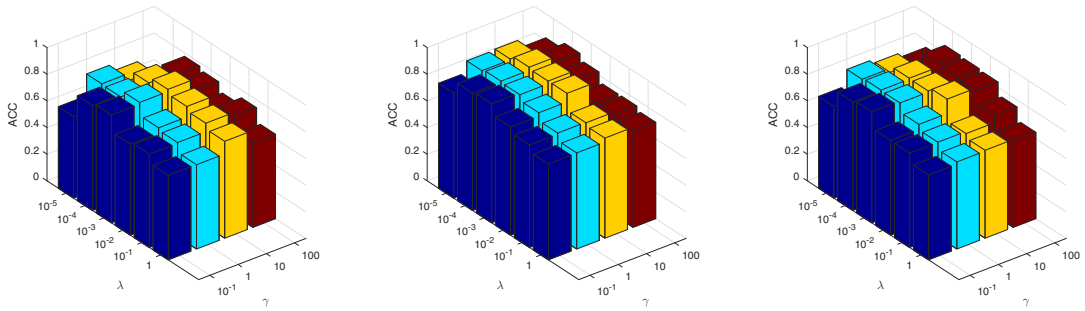
In our experiments, we set the number of columns in basis matrix $\mathbf{U}$ to be equal to the number of clusters c. After obtaining the coefficient matrix $\mathbf{V}$, we perform 20 times K-means clustering for different starting points and report the best results. Three metrics, *i.e.*, Accuracy (ACC), Normalized Mutual Information (NMI) and Purity are used to measure the clustering performance. We compare FNMFG with K-means, Normalized Cut (NCut), standard NMF [1], concept factorization (CF) [19], GNMF [14] and locally consistent concept factorization (LCCF) [20]. To keep fair comparison, we tune the parameters involved in respective methods in wide range (10^{-3} to 10^3). The best clustering results under the optimal parameter combination are only reported.

Table 2. Description of Data Sets.

Data set	# Samples	# Dimensions	# Clusters
COIL20	1440	1024	20
UMIST	575	644	20
PIE	2856	1024	68
TDT2	9394	36771	30
AR	700	2580	100
MNIST	4000	768	10

3.2. Experimental Results

The results for all investigated methods on the benchmark data sets are summarized in Table 3 in which the best results are in boldface. From this table, we have several important findings. First, matrix factorization models are efficient in learning data representation, which delivers better results in most cases of our experiments than directly performing K-means clustering in the original data space. Second, the graph regularization in matrix factorization models is more efficient since the geometrical structure of data is exploited and the local invariance is imposed. In GNMF and LCCF, the data geometrical structure is characterized by graph and the learned coefficient matrix attempts to preserve the data structure. Therefore, they consistently achieve better results than the counterparts, NMF and CF. Third, adaptively learning of an optimal graph from data for matrix factorization models is better than regularizing the coefficient matrix with a fixed graph. Existing models such as GNMF and LCCF construct graph based on predefined rules (*i.e.*, 'HeatKernel' function and '0-1' weighting) which may not adapt to complex data structure of different data sets. Our FNMFG method can jointly perform graph learning and matrix factorization in which the two sub-objectives can co-evolve to the solution



(a) COIL20: ACC (b) COIL20: NMI (c) COIL20: Purity

Fig. 1. The performance of FNMFG versus variation of parameters λ and γ for COIL20 data set.

close to optimal one. This effectively avoids the limitations existed in the two-stage strategy employed in most of existing graph-based learning models, that is, constructing a graph first and then performing learning process on this fixed graph.

Table 3. Comparison of clustering performance of different methods on benchmark data sets.

(a) Accuracy (%)

	COIL20	UMIST	PIE	TDT2	AR	MNIST
Kmeans	60.67	44.09	24.54	62.88	30.57	51.75
NCut	71.46	50.26	65.30	68.74	39.71	58.63
NMF	61.46	45.57	56.62	69.60	40.14	55.05
CF	57.50	44.52	57.21	66.97	37.29	50.15
GNMF	81.46	62.43	72.37	85.75	43.29	61.18
LCCF	77.92	57.04	64.22	82.17	41.14	60.98
FNMFG	**84.31**	**69.22**	**79.17**	**87.30**	**45.57**	**62.08**

(b) Normalized Mutual Information (%)

	COIL20	UMIST	PIE	TDT2	AR	MNIST
Kmeans	71.63	63.77	53.77	71.46	62.85	46.44
NCut	83.55	71.97	83.33	76.61	69.68	61.23
NMF	72.59	64.43	81.71	77.01	68.65	46.83
CF	68.10	60.82	74.57	74.23	65.52	46.73
GNMF	88.63	77.13	88.22	83.99	70.19	59.77
LCCF	85.56	73.37	85.14	81.65	68.95	57.91
FNMFG	**91.34**	**81.54**	**89.92**	**85.72**	**72.31**	**60.79**

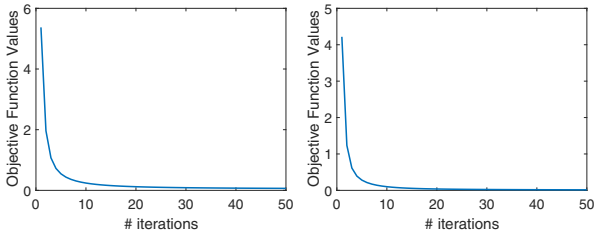
(c) Purity (%)

	COIL20	UMIST	PIE	TDT2	AR	MNIST
Kmeans	62.87	51.96	28.29	83.70	32.86	55.38
NCut	75.76	61.39	71.11	85.87	40.29	66.90
NMF	63.06	52.70	60.89	87.31	43.57	56.70
CF	60.90	51.65	61.66	85.54	38.97	54.81
GNMF	83.75	69.22	76.86	89.84	46.43	69.98
LCCF	80.42	64.87	70.90	88.43	44.65	68.02
FNMFG	**86.60**	**75.30**	**82.67**	**91.43**	**48.21**	**71.31**

3.3. Parameter Sensitivity and Convergence Study

There are two regularization parameters in our FNMFG algorithm. Concretely, λ controls the approximation error of matrix factorization and γ controls the adaptation of the coefficient matrix $\mathbf{V}$ to the geometrical manifold characterized by the learned graph. Figure 1 shows how the performance of FNMFG varies in terms of parameters λ and γ. As we can see, FNMFG is not sensitive to wide variation of them. FNMFG can consistently achieve excellent performance even for λ varies from 10^{-4} to 10^{-1} and γ varies from 1 to 100. In FNMFG, we assume $\mathbf{V}$ to be orthogonal to guarantee more discriminative information which accordingly augments the approximation error of the second term in (4). Therefore, minimizing the FNMFG objective function will result in a large γ and a relatively small λ.

To verify the convergence analysis given in section 2.3, we experimentally show the convergence curves of FNMFG on example COIL20 and PIE data sets in Figure 2 in which the x-axis denotes the iteration number and the y-axis is the value of objective function. We can see the iterative optimization method implemented in FNMFG has fast convergence speed, usually within 20 iterations.



(a) COIL20 (b) PIE

Fig. 2. Convergence curves of FNMFG on COIL20 and PIE.

4. CONCLUSION

In this paper, we proposed a Flexible NMF model regularized by adaptively learned Graph (FNMFG), which can effectively learn an adaptive graph for regularizing the coefficient factor matrix in NMF. Instead of performing regularization on a fixed graph, FNMFG successfully avoids the limitations caused by the two-stage problem in many graph based learning models. We presented an efficient iterative algorithm to optimize the FNMFG objective function. Extensive experiments were conducted to demonstrate the superior performance of the FNMFG in comparison to the state-of-the-art methods. In the future work, we will consider extending the adaptive graph learning into tensor factorization methods.

5. REFERENCES

[1] Daniel D Lee and H Sebastian Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[2] Dennis L Sun and Cedric Fevotte, "Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 6201–6205.

[3] Tatsuya Komatsu, Yuzo Senda, and Reishi Kondo, "Acoustic event detection based on non-negative matrix factorization with mixtures of local dictionaries and activation aggregation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 2259–2263.

[4] Wei He, Hongyan Zhang, and Liangpei Zhang, "Sparsity-regularized robust non-negative matrix factorization for hyperspectral unmixing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 9, pp. 4267–4279, 2016.

[5] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-ichi Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*, John Wiley & Sons, 2009.

[6] Guoxu Zhou, Andrzej Cichocki, and Shengli Xie, "Fast nonnegative matrix/tensor factorization based on low-rank approximation," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2928–2940, 2012.

[7] Naiyang Guan, Tongliang Liu, Yangmuzi Zhang, Dacheng Tao, and Larry Steven Davis, "Truncated cauchy non-negative matrix factorization for robust subspace learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 246–259, 2019.

[8] Sara Mandelli, Nicolò Bonettini, Paolo Bestagini, Vincenzo Lipari, and Stefano Tubaro, "Multiple JPEG compression detection through task-driven non-negative matrix factorization," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 2106–2110.

[9] Chuang Lin, Binghui Wang, Ning Jiang, and Dario Farina, "Robust extraction of basis functions for simultaneous and proportional myoelectric control via sparse non-negative matrix factorization," *Journal of Neural Engineering*, vol. 15, no. 2, pp. 026017, 2018.

[10] Sam T Roweis and Lawrence K Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[11] Joshua B Tenenbaum, Vin De Silva, and John C Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[12] Mikhail Belkin and Partha Niyogi, "Laplacian Eigenmaps and spectral techniques for embedding and clustering," in *Proceedings of Advances in Neural Information Processing Systems*, 2001, pp. 585–591.

[13] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, no. 11, pp. 2399–2434, 2006.

[14] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[15] Feiping Nie, Xiaoqian Wang, Michael I Jordan, and Heng Huang, "The constrained Laplacian rank algorithm for graph-based clustering.," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2016, pp. 1969–1976.

[16] Patrik O Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, no. Nov, pp. 1457–1469, 2004.

[17] Ky Fan, "On a theorem of Weyl concerning eigenvalues of linear transformations," *Proceedings of the National Academy of Sciences*, vol. 35, no. 11, pp. 652–655, 1949.

[18] Jin Huang, Feiping Nie, and Heng Huang, "A new simplex sparse learning model to measure data similarity for clustering," in *Proceedings of International Joint Conference on Artificial Intelligence*, 2015, pp. 3569–3575.

[19] Wei Xu and Yihong Gong, "Document clustering by concept factorization," in *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004, pp. 202–209.

[20] Deng Cai, Xiaofei He, and Jiawei Han, "Locally consistent concept factorization for document clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 902–913, 2011.