JOINT STRUCTURED GRAPH LEARNING AND CLUSTERING BASED ON CONCEPT FACTORIZATION

Yong Peng[1,2,*], *Rixin Tang*[1], *Wanzeng Kong*[1], *Jianhai Zhang*[1], *Feiping Nie*[2] *and Andrzej Cichocki*[3,1]

[1] School of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Center for OPTIMAL, Northwestern Polytechnical University, Xi'an 710072, China
[3]Skolkovo Institute of Science and Technology (SKOLTECH), Moscow 143026, Russia
yongpeng@hdu.edu.cn

ABSTRACT

As one of the matrix factorization models, concept factorization (CF) achieved promising performance in learning data representation in both original feature space and reproducible kernel Hilbert space (RKHS). Based on the consensuses that 1) learning performance of models can be enhanced by exploiting the geometrical structure of data and 2) jointly performing structured graph learning and clustering can avoid the suboptimal solutions caused by the two-stage strategy in graph-based learning, we developed a new CF model with self-expression. Our model has a combined coefficient matrix which is able to learn more efficiently. In other words, we propose a CF-based joint structured graph learning and clustering model (JSGCF). A new efficient iterative method is developed to optimize the JSGCF objective function. Experimental results on representative data sets demonstrate the effectiveness of our new JSGCF algorithm.

Index Terms— Structured graph learning, joint learning, concept factorization, clustering

1. INTRODUCTION

Non-negative matrix factorization (NMF) [1] is one of the famous matrix factorization models [2, 3, 4, 5, 6] to learn efficient data representation. Mathematically, NMF approximates the target matrix with the product of two non-negative matrices. The learned parts-based representation not only has biological plausibility but also shows excellent performance in pattern recognition problems. However, it can only be performed in original data feature space and thus cannot characterize the possible nonlinear structure of data. As an extension of NMF, concept factorization (CF) was proposed which can be performed in both original feature space and RKHS [7].

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).

The learning performance of models can be considerably enhanced by exploiting the geometrical structure of data, locally consistent CF (LCCF) [8] was proposed to constrain the coefficient matrix to preserve the local invariance property. In LCCF, the geometrical structure of data was characterized by a nearest neighbor graph to be used to regularize the learning process. After obtaining the coefficient matrix, K-means was used to obtain the final clustering results. In our approach, we learn an adaptive graph from data which has obvious clustering structure and then the post-processing such as K-means clustering would be unnecessary [9, 10]. In this paper, we treat CF as a non-negative self-expression model. Then the combined coefficient matrix can be viewed as a graph affinity matrix based on which we propose to learn a high level one with more suitable properties such as non-negativity, normalization and constrained rank. The clustering tasks are performed jointly with the graph learning process. This results in the proposed JSGCF model which is implemented as an efficient algorithm. Extensive experiments are conducted to demonstrate the high performance of our JSGCF approach.

2. THE PROPOSED JSGCF MODEL

2.1. Model Formulation

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, NMF minimizes the approximation error between the data matrix and two non-negative factor matrices $\mathbf{U} \in \mathbb{R}^{d\times c}$, $\mathbf{V} \in \mathbb{R}^{n\times c}$ (c is the number of classes/clusters) as

$$\min_{\mathbf{U},\mathbf{V}} \|\mathbf{X} - \mathbf{U}\mathbf{V}^T\|_F^2,\ s.t.\ \mathbf{U} \geq \mathbf{0}, \mathbf{V} \geq \mathbf{0}. \tag{1}$$

However, standard NMF can only be performed in the original feature space. To characterize the possible nonlinear structure of data, Xu and Gong [7] proposed CF as an extension of NMF. Concretely, each basis $\mathbf{u}_k$ is required to be a non-negative linear combination of samples as $\mathbf{u}_k = \sum_{j=1}^{n} \mathbf{x}_j w_{jk}, w_{jk} \geq 0$. Therefore, the objective of CF is

$$\min_{\mathbf{W},\mathbf{V}} \|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{V}^T\|_F^2,\ s.t.\ \mathbf{W} \geq \mathbf{0}, \mathbf{V} \geq \mathbf{0}. \tag{2}$$

Obviously, CF can be performed in both original feature space and reproducible kernel Hilbert space (RKHS).

If treating $\mathbf{WV}^T$ as a whole, CF will be a self-expression model [11, 12, 13, 14] in which $\mathbf{WV}^T$ acts as a combined graph affinity matrix. We expect such graph to be close to an ideal one based on which we can partition the data points into c clusters, without performing any post-processing. Based on the equivalence between such exact c block diagonals prior of a graph affinity matrix and the multiplicity of the eigenvalue zero of the corresponding Laplacian matrix [10], we can impose the rank constraint on the Laplacian matrix of $\mathbf{WV}^T$, $\mathbf{L}_{\mathbf{WV}^T}$, as $\text{rank}(\mathbf{L}_{\mathbf{WV}^T}) = n - c$. Therefore, we formulate the following optimization problem

$$\min_{\mathbf{W}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0}} \|\mathbf{X}-\mathbf{XWV}^T\|_F^2,\ s.t.\ \text{rank}(\mathbf{L}_{\mathbf{WV}^T}) = n-c. \quad (3)$$

Here we show how to convert the rank constraint in (3) into an equivalent mathematical expression to make it more tractable. Denote $\sigma_i(\mathbf{L}_{\mathbf{WV}^T})$ as the i-th smallest eigenvalue of $\mathbf{L}_{\mathbf{WV}^T}$. It is obvious that $\sigma_i(\mathbf{L}_{\mathbf{WV}^T}) \geq 0$ since $\mathbf{L}_{\mathbf{WV}^T}$ is positive semidefinite. If $\sum_{i=1}^{c}\sigma_i(\mathbf{L}_{\mathbf{WV}^T})$ approaches zero and then the constraint $\text{rank}(\mathbf{L}_{\mathbf{WV}^T}) = n - c$ in (3) will be approximately satisfied. Therefore, we can incorporate $\sum_{i=1}^{c}\sigma_i(\mathbf{L}_{\mathbf{WV}^T})$ into CF model as a regularization term and apply a sufficiently large regularization parameter α. Furthermore, according to Ky Fan's Theorem [15], we have the following minimization problem

$$\sum\nolimits_{i=1}^{c}\sigma_i(\mathbf{L}_{\mathbf{WV}^T}) = \min_{\mathbf{F}\in\mathbb{R}^{n\times c},\mathbf{F}^T\mathbf{F}=\mathbf{I}} \text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{WV}^T}\mathbf{F}), \quad (4)$$

where each row $\mathbf{f}_i$ of $\mathbf{F}$ can be seen as a vector connected to data point $\mathbf{x}_i$ on the graph $\mathbf{WV}^T$ [16, 17]. Therefore, (3) is equivalent to

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{V},\mathbf{F}} \|\mathbf{X}-\mathbf{XWV}^T\|_F^2 + \alpha\text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{WV}^T}\mathbf{F}).\\ &s.t.\ \mathbf{W}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0},\mathbf{F}\in\mathbb{R}^{n\times c},\mathbf{F}^T\mathbf{F}=\mathbf{I}.\end{aligned} \quad (5)$$

We can see that both $\mathbf{W}$ and $\mathbf{V}$ are involved in $\mathbf{L}_{\mathbf{WV}^T}$ which makes the above objective function difficult to minimize. Therefore, by introducing an auxiliary variable $\mathbf{S}$ w.r.t. $\mathbf{WV}^T$, we obtain

$$\begin{aligned}\min_{\mathbf{W},\mathbf{V},\mathbf{S},\mathbf{F}} &\|\mathbf{X}-\mathbf{XWV}^T\|_F^2 + \alpha\text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F}) + \beta\|\mathbf{S}-\mathbf{WV}^T\|_F^2.\\ &s.t.\ \mathbf{W}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0},\mathbf{F}\in\mathbb{R}^{n\times c},\mathbf{F}^T\mathbf{F}=\mathbf{I}.\end{aligned}$$

Before presenting the detailed optimization procedure, we further explain the role of $\mathbf{S}$. Mathematically, it is an auxiliary variable to make the objective function separable. In fact, we can see $\mathbf{WV}^T$ as a low-level graph and $\mathbf{S}$ as a high-level one which has better desired properties. From this perspective, we impose an additional constraint that the sum of elements in each row of $\mathbf{S}$ to be one. Finally, we achieve the objective of JSGCF as

$$\begin{aligned}&\min \|\mathbf{X}-\mathbf{XWV}^T\|_F^2 + \alpha\text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F}) + \beta\|\mathbf{S}-\mathbf{WV}^T\|_F^2\\ &s.t.\mathbf{W}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0},\mathbf{S1}=\mathbf{1},\mathbf{S}\geq\mathbf{0},\mathbf{F}\in\mathbb{R}^{n\times c},\mathbf{F}^T\mathbf{F}=\mathbf{I}\end{aligned} \quad (6)$$

2.2. Optimization

Obviously, directly minimizing (6) is intractable. Therefore, we update one variable while the others are fixed under the alternating direction method framework.

1) Update $\mathbf{F}$. The objective $\mathcal{O}(\mathbf{F})$ is

$$\min_{\mathbf{F}} \text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F}),\ s.t.\ \mathbf{F}\in\mathbb{R}^{n\times c}, \mathbf{F}^T\mathbf{F}=\mathbf{I}. \quad (7)$$

The optimal solution of $\mathbf{F}$ is formed as the c eigenvectors of $\mathbf{L}_{\mathbf{S}}$ corresponding to the c smallest eigenvalues.

2) Update $\mathbf{W}$ and $\mathbf{V}$. The objective $\mathcal{O}(\mathbf{W},\mathbf{V})$ is

$$\min_{\mathbf{W}\geq\mathbf{0},\mathbf{V}\geq\mathbf{0}} \|\mathbf{X}-\mathbf{XWV}^T\|_F^2 + \beta\|\mathbf{S}-\mathbf{WV}^T\|_F^2. \quad (8)$$

Let ψ_{jk} and ϕ_{jk} be Lagrange multipliers for $w_{jk}\geq 0$ and $v_{jk}\geq 0$ respectively. We define $\mathbf{\Psi}=[\psi_{jk}]$ and $\mathbf{\Phi}=[\phi_{jk}]$. The corresponding Lagrangian function $\mathcal{L}$ is

$$\min \|\mathbf{X}-\mathbf{XWV}^T\|^2+\beta\|\mathbf{S}-\mathbf{WV}^T\|^2+\text{Tr}(\mathbf{\Psi W}^T)+\text{Tr}(\mathbf{\Phi V}^T).$$

The derivatives of $\mathcal{L}$ w.r.t. $\mathbf{W}$ and $\mathbf{V}$ can be expressed as

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\mathbf{W}} &= -2(\mathbf{K}+\beta\mathbf{S})\mathbf{V} + 2(\mathbf{K}+\beta\mathbf{I})\mathbf{WV}^T\mathbf{V} + \mathbf{\Psi},\\ \frac{\partial\mathcal{L}}{\partial\mathbf{V}} &= -2(\mathbf{K}+\beta\mathbf{S}^T)\mathbf{W} + 2\mathbf{VW}^T(\mathbf{K}+\beta\mathbf{I})\mathbf{W} + \mathbf{\Phi}.\end{aligned}$$

Setting the derivatives to be zero and using the KKT conditions $\psi_{jk}w_{jk}=0$ and $\phi_{jk}v_{jk}=0$, we get

$$\begin{aligned}&-[(\mathbf{K}+\beta\mathbf{S})\mathbf{V}]_{jk}w_{jk} + [(\mathbf{K}+\beta\mathbf{I})\mathbf{WV}^T\mathbf{V}]_{jk}w_{jk} = 0,\\ &-[(\mathbf{K}+\beta\mathbf{S}^T)\mathbf{W}]_{jk}v_{jk} + [\mathbf{VW}^T(\mathbf{K}+\beta\mathbf{I})\mathbf{W}]_{jk}v_{jk} = 0,\end{aligned}$$

which lead to the following simple updating rules

$$w_{jk} \leftarrow w_{jk}\frac{[(\mathbf{K}+\beta\mathbf{S})\mathbf{V}]_{jk}}{[(\mathbf{K}+\beta\mathbf{I})\mathbf{WV}^T\mathbf{V}]_{jk}} \quad (9)$$

$$v_{jk} \leftarrow v_{jk}\frac{[(\mathbf{K}+\beta\mathbf{S}^T)\mathbf{W}]_{jk}}{[\mathbf{VW}^T(\mathbf{K}+\beta\mathbf{I})\mathbf{W}]_{jk}} \quad (10)$$

In particular, in order to make $\mathbf{W}$ and $\mathbf{V}$ unique, we use the same simple method in [8] to normalize them.

3) Update $\mathbf{S}$. The objective $\mathcal{O}(\mathbf{S})$ is

$$\min_{\mathbf{S}\geq\mathbf{0},\mathbf{S1}=\mathbf{1}} \alpha\text{Tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F}) + \beta\|\mathbf{S}-\mathbf{WV}^T\|_F^2. \quad (11)$$

The matrix form representation in (11) can be decomposed as

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geq 0} \frac{\alpha}{\beta}\sum_{i,j=1}^{n}\|\mathbf{f}_i-\mathbf{f}_j\|_2^2 s_{ij} + \sum_{i,j=1}^{n}(s_{ij}-(\mathbf{WV})_{ij})^2. \quad (12)$$

Denoting $p_{ij}=\|\mathbf{f}_i-\mathbf{f}_j\|^2$ and denoting $\mathbf{p}_i$ as a vector with the j-th element equal to p_{ij} (and similarly for $\mathbf{s}_i$ and $(\mathbf{wv})_i$), and thus (12) can be rewritten in vector form as

$$\min_{\mathbf{s}_i\mathbf{1}=1,\mathbf{s}_i\geq 0} \|\mathbf{s}_i - ((\mathbf{wv})_i - \frac{\alpha}{2\beta}\mathbf{p}_i)\|_2^2. \quad (13)$$

This problem can be solved with a closed form solution or alternatively solved by an efficient iterative algorithm [18, 10].

We summarize the detailed procedure to JSGCF optimization in (6) in Algorithm 1.

Algorithm 1 The optimization to JSGCF objective in (6)

Input: Data $\mathbf{X} \in \mathbb{R}^{d\times n}$, the number of clusters c, regularization parameters α and β.

Output: Variable $\mathbf{W}$, $\mathbf{V}$, $\mathbf{F}$ and $\mathbf{S}$ with exactly c connected components.

1: Initialize $\mathbf{W}$ and $\mathbf{V}$ randomly; Initialize $\mathbf{F}$ as the c eigenvectors of $\mathbf{L_A} = \mathbf{D_A} - \frac{\mathbf{A}^T+\mathbf{A}}{2}$ corresponding to the c smallest eigenvalues, where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is an affinity matrix constructed based on $\mathbf{X}$ with 'HeatKernel' function.
2: **while** not converged **do**
3: Update $\mathbf{F}$ according to (7);
4: Update $\mathbf{W}$ according to (9);
5: Update $\mathbf{V}$ according to (10);
6: Update $\mathbf{S}$ according to (13);
7: **end while**

2.3. Convergence and Complexity Analysis

The updating of each variable in JSGCF approach is essentially iterative. Here we give a brief analysis on the updating rule to each variable. Since sub-objective associated with $\mathbf{F}$ is convex, the updating rule to $\mathbf{F}$ is in analytical form which leads to the non-increasing trend of the JSGCF objective function. The optimization procedure to update $\mathbf{W}$ and $\mathbf{V}$ follows the pipeline of CF and LCCF whose convergence has been already investigated in [7, 8] by introducing appropriate auxiliary functions. Since the objective on $\mathbf{S}$ is convex by fixing $\mathbf{W}$, $\mathbf{V}$ and $\mathbf{F}$, the updating rule for $\mathbf{S}$ also provides the convergence of JSGCF. In practice, we can investigate the convergence condition by checking whether $\|\mathbf{S} - \mathbf{W}\mathbf{V}^T\|_\infty < \varepsilon$ (ε is a small value) is satisfied or not.

The main complexity is caused by the loop in Algorithm 1 which contains four blocks. For the updating of variable $\mathbf{F}$, the main costs lie in calculating the c eigenvector of Laplacian matrix $\mathbf{L_S} \in \mathbb{R}^{n\times n}$ with complexity $O(n^2c)$. We need $O(t_1 n)$ operations to calculate $\mathbf{S}$ in each iteration by an efficient iterative method in which t_1 is the number of iterations of the Newton method. For the updating to $\mathbf{W}$ and $\mathbf{V}$, we count the arithmetic operations for CF, LCCF and JSGCF in Table 1 where p is the number of nearest neighbors in LCCF, *fladd*, *flmlt* and *fldiv* respectively mean the *floating-point addition*, *floating-point multiplication* and *floating-point division*. In LCCF, $\mathbf{S}$ is a p-sparse matrix while in JSGCF it is an $\frac{n}{c}$-sparse matrix on average (Each cluster has $\frac{n}{c}$ samples on average). In general, the complexity of JSGCF is $O(t(n^2c + nt_1))$, where t is the number of iterations.

3. EXPERIMENTS

3.1. Experimental Settings

Three representative benchmark data sets, COIL20, PIE and UMIST, were used in our experiments. The properties of them are the same as those in [19, 16]. We compare JSGCF with K-means, Normalized Cut (NCut) [20], NMF [1], CF [7] and LCCF [8] in terms of the clustering performance on the given data sets. Linear kernel was used in different CF variants. Two metrics, Accuracy (ACC) and Normalized Mutual Information (NMI), are used to evaluate the clustering performance. The parameters involved in respective algorithm were tuned in wide range from 10^{-3} to 10^3.

Table 1. Complexity analysis: operations of CF, LCCF and JSGCF in updating $\mathbf{W}$ and $\mathbf{V}$ in each iteration.

	CF	LCCF	JSGCF
fladd	$4n^2c + 4nc^2$	$4n^2c + 4nc^2 + n(p+3)c$	$4n^2c + 4nc^2 + 2(n\frac{n}{c}c + 3nc)$
flmlt	$4n^2c + 4nc^2 + 2nc$	$4n^2c + 4nc^2 + n(p+3)c$	$4n^2c + 4nc^2 + 2nc + 2(n\frac{n}{c}c + nc)$
fldiv	$2nc$	$2nc$	$2nc$
overall	$O(n^2c)$	$O(n^2c)$	$O(n^2c)$

3.2. Clustering Results

Tables 2, 3 and 4 show the clustering performance of COIL20, PIE and UMIST data sets. The clustering experiments were conducted with different numbers of clusters. For each given cluster numbers, 20 test runs are conducted on different randomly chosen clusters. The final results are reported by averaging the results for 20 runs. For the sake of fairness, we record the randomly chosen cluster indices and fix them for all competing algorithms. From the results, we can find that 1) the learning performance can be greatly enhanced by exploiting and considering the geometrical structure of data. This is reflected by the fact that the performance of both LCCF and JSGCF can get better performance than CF. 2) it is beneficial to jointly perform graph construction and learning task. Such obtained graph can well adapt to the structure of data sets and thus much better clustering performance can be achieved. This indicates that JSGCF which learns an adaptive graph is more competitive than LCCF which utilizes a fixed graph to depict the data structure and then imposes constraints on the coefficient matrix.

To illustrate how JSGCF constructs the optimal graph, we select the first 13 clusters from the PIE data set and visualize the data affinity matrix $\mathbf{S}$ learned by JSGCF in Figure 1. We can see that the block diagonal structure is gradually more clear as the number of iterations increases. The values within block diagonals increased means the within-cluster connections are enhanced. To verify the convergence analysis provided in section 2.3, we experimentally show the convergence curves of JSGCF on COIL20 and PIE data sets in Figure 2. The objective function value monotonically decreases till convergence during iteration process.

4. CONCLUSIONS

In this paper, we proposed a joint structured graph learning and clustering model, termed JSGCF. The motivation of JSGCF was to view $\mathbf{W}\mathbf{V}^T$ in CF as a whole and thus view CF as

Table 2. Comparison of clustering performance on COIL20.

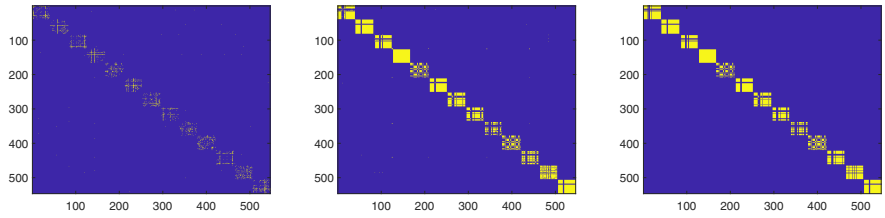
K	Accuracy (mean±std-dev%)						Normalized Mutual Information (mean±std-dev%)					
	Kmeans	NCut	NMF	CF	LCCF	JSGCF	Kmeans	NCut	NMF	CF	LCCF	JSGCF
6	75.9±12.4	82.5±14.5	70.2±11.7	69.3±11.3	86.9±9.3	**88.5±11.8**	72.9±11.9	86.2±9.4	68.7±12.5	67.3±11.5	85.2±9.0	**90.1±8.0**
8	72.7±9.2	79.0±15.6	71.0±10.2	67.1±11.0	82.1±11.0	**86.6±15.4**	73.0±9.0	84.8±10.8	70.7±9.2	67.9±9.9	83.1±9.3	**90.4±9.2**
10	70.3±6.1	76.7±10.3	69.9±9.1	64.8±5.1	78.2±8.9	**84.8±9.6**	73.6±5.6	84.5±6.8	72.0±7.4	69.2±4.6	82.5±6.4	**90.2±5.9**
12	66.9±7.8	75.3±12.1	67.9±8.1	63.7±8.7	77.3±6.1	**82.0±12.3**	73.4±5.9	84.5±8.2	72.5±6.7	70.4±6.2	82.1±5.7	**88.5±8.4**
14	66.3±4.7	66.3±9.0	65.6±5.0	62.5±5.1	76.4±6.5	**80.8±6.2**	73.9±3.3	81.4±4.8	72.4±4.1	71.0±3.4	83.4±4.1	**89.0±3.7**
16	65.6±5.9	65.0±10.7	66.4±4.8	61.9±5.1	74.3±5.1	**83.2±5.3**	73.6±4.1	79.5±7.7	73.0±3.5	71.6±4.1	82.5±3.9	**90.9±3.1**
18	65.9±3.6	64.1±7.2	65.1±3.7	62.7±3.9	75.2±5.4	**80.9±3.0**	75.7±2.3	80.5±4.7	74.2±2.6	73.4±2.4	84.1±3.3	**89.9±1.8**
20	60.5	62.6	63.8	63.8	74.7	**83.5**	73.9	89.0	72.8	74.0	80.9	**91.9**
Avg.	68.0	71.4	67.5	64.5	78.1	**83.8**	73.8	82.8	72.0	70.6	83.3	**90.1**

Table 3. Comparison of clustering performance on PIE.

K	Accuracy (mean±std-dev%)						Normalized Mutual Information (mean±std-dev%)					
	Kmeans	NCut	NMF	CF	LCCF	JSGCF	Kmeans	NCut	NMF	CF	LCCF	JSGCF
10	29.8±4.1	71.3±12.7	55.9±4.7	55.6±3.7	69.1±9.9	**78.7±11.7**	35.2±6.1	81.1±8.9	65.0±3.6	62.5±2.7	80.8±7.0	**85.6±6.9**
20	27.5±2.7	66.8±9.6	57.6±3.5	57.8±3.5	65.5±5.8	**74.6±8.5**	43.8±2.7	81.5±6.6	74.5±1.8	70.5±1.8	81.7±3.4	**85.7±5.7**
30	26.5±2.1	66.4±6.1	56.4±3.5	58.4±2.2	63.7±5.4	**73.6±5.2**	48.6±2.0	81.9±4.2	76.8±1.9	72.9±1.7	82.5±2.5	**86.6±2.8**
40	25.8±1.4	65.1±4.7	57.4±3.1	57.3±2.9	61.2±3.7	**72.1±5.0**	50.4±1.7	81.1±3.5	78.8±1.2	73.4±1.4	82.3±1.8	**85.8±2.7**
50	24.7±1.3	62.7±3.5	57.1±2.6	56.3±2.8	62.8±4.1	**71.9±4.0**	51.5±1.1	81.2±1.7	79.6±1.3	73.7±1.6	83.5±1.6	**85.3±2.1**
60	24.3±1.1	62.4±3.6	56.8±2.5	55.5±2.1	62.6±3.5	**70.1±2.6**	53.3±1.1	80.5±2.3	80.5±0.9	74.3±1.1	84.4±1.3	**85.0±1.4**
68	24.5	63.2	56.4	57.2	62.5	**70.0**	53.8	81.4	81.1	74.6	83.8	**84.9**
Avg.	26.2	65.4	56.8	56.9	63.9	**73.0**	48.1	81.2	76.6	71.7	82.7	**85.6**

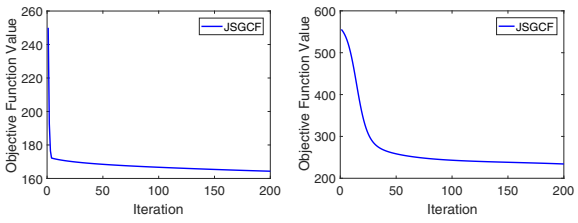
Table 4. Comparison of clustering performance on UMIST.

K	Accuracy (mean±std-dev%)						Normalized Mutual Information (mean±std-dev%)					
	Kmeans	NCut	NMF	CF	LCCF	JSGCF	Kmeans	NCut	NMF	CF	LCCF	JSGCF
6	54.2±7.5	67.2±13.5	53.1±6.4	52.5±7.3	56.3±11.3	**72.3±16.4**	55.3±10.5	71.8±13.4	54.4±10.7	51.5±8.7	56.2±15.9	**77.0±14.6**
8	50.7±5.5	62.8±10.9	49.3±7.6	49.3±5.1	56.5±10.0	**67.4±7.3**	57.5±6.3	70.7±8.4	56.1±8.0	54.4±5.9	64.9±10.6	**75.5±6.5**
10	48.9±7.0	61.5±7.7	47.3±4.9	46.0±4.2	50.4±8.1	**64.9±8.2**	58.4±6.9	72.0±7.3	55.7±6.1	55.2±4.7	62.1±9.7	**76.5±6.2**
12	46.0±4.8	59.9±7.3	45.4±4.2	45.1±5.7	51.0±7.5	**60.0±7.3**	58.4±5.3	71.6±5.6	57.4±4.8	55.2±4.6	63.7±7.7	**73.2±6.1**
14	44.1±3.0	54.7±5.9	41.9±3.2	42.3±4.5	51.0±8.4	**58.5±7.8**	59.1±3.6	69.7±5.8	56.4±3.4	54.7±4.7	65.2±8.3	**74.1±5.7**
16	42.9±2.9	53.6±5.3	40.2±2.4	40.5±2.8	50.4±5.1	**55.9±3.9**	59.3±2.8	69.5±3.8	55.9±2.3	55.1±2.8	66.3±4.8	**72.2±3.1**
18	40.8±2.0	55.2±5.0	40.4±2.6	39.9±2.8	51.5±6.4	**56.6±4.0**	59.6±1.7	70.9±2.8	57.6±2.6	56.1±2.1	68.3±5.2	**72.6±2.6**
20	39.3	50.1	40.7	39.8	53.7	**57.6**	60.0	66.1	57.5	54.9	70.7	**73.0**
Avg	45.9	58.1	44.8	44.4	52.6	**61.7**	58.5	70.3	56.4	54.6	64.7	**74.3**



(a) Iteration 1 (b) Iteration 3 (c) Iteration 5

Fig. 1. Visualization of the data affinity matrices for the PIE subset learned by our JSGCF.



(a) COIL20 (b) PIE

Fig. 2. Convergence curves of JSGCF for COIL20 and PIE data sets.

a self-expression model in which $\mathbf{W}\mathbf{V}^T$ played a crucial role as a graph affinity matrix. However, this graph was somewhat elementary and thus we proposed to learn a high level one with more desirable properties. To this end, a structured graph with **non-negativity**, **row-sum-to-one** and **constrained rank** was learned jointly with the clustering task completed. Extensive experiments showed the very good performance of JSGCF in data clustering. Furthermore, we have investigated how our algorithm evolves in time and gradually improves the quality of the graph affinity matrix.

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] Chris Ding, Xiaofeng He, and Horst D Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Proceedings of SIAM International Conference on Data Mining*, 2005, pp. 606–610.

[3] Andrzej Cichocki, Rafal Zdunek, and Shun-ichi Amari, "New algorithms for non-negative matrix factorization in applications to blind source separation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006, vol. 5, pp. 621–624.

[4] 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.

[5] 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.

[6] Yong Peng, Rixin Tang, Wanzeng Kong, Feiwei Qin, and Feiping Nie, "Parallel vector field regularized non-negative matrix factorization for image representation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 2216–2220.

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

[8] 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.

[9] Feiping Nie, Xiaoqian Wang, and Heng Huang, "Clustering and projected clustering with adaptive neighbors," in *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2014, pp. 977–986.

[10] 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.

[11] John Wright, Allen Y Yang, Arvind Ganesh, S Shankar Sastry, and Yi Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[12] Lei Zhang, Meng Yang, and Xiangchu Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *Proceedings of IEEE International Conference on Computer Vision*, 2011, pp. 471–478.

[13] Ehsan Elhamifar and Rene Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[14] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.

[15] 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.

[16] Minnan Luo, Feiping Nie, Xiaojun Chang, Yi Yang, Alexander G Hauptmann, and Qinghua Zheng, "Adaptive unsupervised feature selection with structure regularization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 944–956, 2018.

[17] Wei Wang, Yan Yan, Feiping Nie, Shuicheng Yan, and Nicu Sebe, "Flexible manifold learning with optimal graph for image and video representation," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2664–2675, 2018.

[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] 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.

[20] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.