JOINT STRUCTURED GRAPH LEARNING AND UNSUPERVISED FEATURE SELECTION

Yong Peng[1,2,*], *Leijie Zhang*[1], *Wanzeng Kong*[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

The central task in graph-based unsupervised feature selection (GUFS) depends on two folds, one is to accurately characterize the geometrical structure of the original feature space with a graph and the other is to make the selected features well preserve such intrinsic structure. Currently, most of the existing GUFS methods use a two-stage strategy which constructs graph first and then perform feature selection on this fixed graph. Since the performance of feature selection severely depends on the quality of graph, the selection results will be unsatisfactory if the given graph is of low-quality. To this end, we propose a joint graph learning and unsupervised feature selection (JGUFS) model in which the graph can be adjusted to adapt the feature selection process. The JGUFS objective function is optimized by an efficient iterative algorithm whose convergence and complexity are analyzed in detail. Experimental results on representative benchmark data sets demonstrate the improved performance of JGUFS in comparison with state-of-the-art methods and therefore we conclude that it is promising of allowing the feature selection process to change the data graph.

Index Terms— Unsupervised feature selection, structured graph learning, non-negativity, joint learning, clustering

1. INTRODUCTION

We are often confronted with high dimensional data in research fields such as machine learning and signal processing, which consumes a lot of computing and storage resources. However, it is usually unnecessary to represent data with such high dimensional feature space which may contain redundant and irrelevant information. Generally, there are two different kinds of approaches to obtain the low dimensional representation of data corresponding to its intrinsic dimensionality, feature extraction and feature selection [1]. In this paper, we focus our topic on feature selection, especially the GUFS, in order to simultaneously improve the learning performance, provide faster and more cost-effective predictors, and provide a better understanding of the underlying data generation [2].

The performance of GUFS severely depends on the quality of the predefined graph. As pointed by Wight et al. [3], an informative graph should satisfy three properties: high discriminative power, low sparsity and adaptive neighborhood. Though a lot of studies were conducted to improve the quality of constructed graphs [4, 5, 6, 7, 8, 9, 10, 11], there still exists a serious drawback in most of the existing GUFS methods, that is, they employ a two-stage strategy in which the feature selection is conducted on a fixed graph. This may cause that the constructed graph cannot well adapt to the objective with respect to feature selection. To alleviate or partially solve such limitation in GUFS, we propose a new JGUFS model to jointly learn an optimal graph and perform feature selection. The main contributions of this paper can be summarized as follows. 1) In contrast to most existing GUFS methods which divide the graph construction and feature selection into separate stages, JGUFS can jointly learn the data affinity matrix and perform feature selection. Therefore, the feature selection process is allowed to adjust the graph. The resultant graph can well adapt to the feature selection, leading to better clustering performance. 2) Develop and test an efficient iterative algorithm to optimize the JGUFS objective function with convergence and complexity analyzed. 3) Experiments by comparing JGUFS with state-of-the-art methods show that it can significantly improve the clustering results.

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

2. THE PROPOSED JGUFS MODEL

2.1. Model Formulation

In JGUFS, we learn an optimal structured graph $\mathbf{S}$ based on an initial graph $\mathbf{A}$. Concretely, we expect $\mathbf{S}$ to approximate $\mathbf{A}$ but with some desirable properties including **non-negativity**, **row-sum-to-one** and **constrained rank** [12]. The second constraint means the sum of entries in each row of $\mathbf{S}$ should be equal to one. The third constraint means the graph Laplacian $\mathbf{L_S}$ should satisfy rank($\mathbf{L_S}$) $= n - c$ if $\mathbf{S}$ is expected to be exactly c block diagonals (n samples should be clustered into c clusters). The objective of structured graph learning is

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

Based on Ky Fan's Theorem [13], the rank constraint on $\mathbf{L_S}$ can be converted to the optimization on pseudo clustering indicator matrix $\mathbf{F} \in \mathbb{R}^{n\times c}$ and then we can rewrite (1) as

$$\min_{\mathbf{S},\mathbf{F}} \|\mathbf{S}-\mathbf{A}\|_F^2+\alpha\text{Tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}),\ s.t.\ \mathbf{S1}=\mathbf{1}, \mathbf{S}\geq \mathbf{0}, \mathbf{F}^T\mathbf{F}=\mathbf{I}_c,$$

where $\alpha > 0$ is usually a large enough regularization parameter. Therefore, by simultaneously performing the structured graph learning and the $\ell_{2,1}$-norm based feature selection [14], we can formulate the objective function of JGUFS as

$$\begin{aligned}\min_{\mathbf{S},\mathbf{W},\mathbf{F}} &\|\mathbf{S}-\mathbf{A}\|_F^2+\alpha\text{Tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F})+\beta(\|\mathbf{XW}-\mathbf{F}\|_F^2+\gamma\|\mathbf{W}\|_{2,1}),\\ &s.t.\ \mathbf{S1}=\mathbf{1}, \mathbf{S}\geq \mathbf{0}, \mathbf{F}^T\mathbf{F}=\mathbf{I}_c, \mathbf{F}\geq\mathbf{0},\end{aligned} \quad (2)$$

where $\mathbf{X} \in \mathbb{R}^{n\times d}$ is the data matrix, $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the projection matrix, β and γ are regularization parameters. Similar to [15, 16], we impose the non-negativity on $\mathbf{F}$ here.

2.2. Optimization to JGUFS

Obviously, since the objective (2) is not jointly convex with respect to $\mathbf{S}$, $\mathbf{W}$ and $\mathbf{F}$, we cannot get the analytical closed-form solutions to them. Therefore, we propose an iterative algorithm to alternately update each of them.

1) Update $\mathbf{S}$. The objective associated with $\mathbf{S}$ is

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geq\mathbf{0}} \|\mathbf{S}-\mathbf{A}\|_F^2+\alpha\text{Tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}), \quad (3)$$

which can be decoupled in scalar form as

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

Denote $d_{ij} = \|\mathbf{f}_i-\mathbf{f}_j\|_2^2$ and $\mathbf{d}_i$ as a vector with the j-th element equal to d_{ij}. Similarly, we get $\mathbf{s}_i$ and $\mathbf{a}_i$ and then problem (4) can be rewritten in vector form as

$$\min_{\mathbf{s}_i\mathbf{1}=1,\mathbf{s}_i\geq\mathbf{0}} \|\mathbf{s}_i-(\mathbf{a}_i-\frac{\alpha}{2}\mathbf{d}_i)\|_F^2. \quad (5)$$

This optimization problem can be solved with a closed form solution by an efficient iterative algorithm [17].

2) Update $\mathbf{W}$. The objective associated with $\mathbf{W}$ is

$$\min_{\mathbf{W}} \|\mathbf{XW}-\mathbf{F}\|_F^2+\gamma\|\mathbf{W}\|_{2,1}. \quad (6)$$

By introducing a diagonal matrix $\mathbf{M}$ with its i-th diagonal entry defined as

$$m_{ii} = \frac{1}{2\|\mathbf{w}_i\|_2} \doteq \frac{1}{2\sqrt{\mathbf{w}_i\mathbf{w}_i^T+\varepsilon}}, \quad (7)$$

where $\mathbf{w}_i$ is the i-th row of $\mathbf{W}$ and ε is a small positive value, we can reformulate (6) as

$$\min_{\mathbf{W}} \|\mathbf{XW}-\mathbf{F}\|^2+\gamma\text{Tr}(\mathbf{W}^T\mathbf{M}\mathbf{W}). \quad (8)$$

Taking the derivative of (8) with respect to $\mathbf{W}$ and setting it to zero, we obtain a simple updating rule to $\mathbf{W}$ as

$$\mathbf{W} = (\mathbf{X}^T\mathbf{X}+\gamma\mathbf{M})^{-1}\mathbf{X}^T\mathbf{F}. \quad (9)$$

3) Update $\mathbf{F}$. In order to eliminate the orthogonal constraint, we add a penalty term $\frac{\lambda}{2}\|\mathbf{F}^T\mathbf{F}-\mathbf{I}\|_F^2$ in which λ is usually a large value (we set it to 10^7 in all the following experiments). Therefore, we have the objective related to $\mathbf{F}$ as

$$\min_{\mathbf{F}\geq\mathbf{0}} \alpha\text{Tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F})+\beta\|\mathbf{XW}-\mathbf{F}\|_F^2+\frac{\lambda}{2}\|\mathbf{F}^T\mathbf{F}-\mathbf{I}\|_F^2. \quad (10)$$

Since $\mathbf{W}$ is also related to $\mathbf{F}$, by substituting (9) into (10), we have the following optimization problem

$$\min_{\mathbf{F}\geq\mathbf{0}} \text{Tr}(\mathbf{F}^T\mathbf{R}\mathbf{F})+\frac{\lambda}{2}\|\mathbf{F}^T\mathbf{F}-\mathbf{I}\|_F^2, \quad (11)$$

where $\mathbf{R} = \alpha\mathbf{L}+\beta(\mathbf{I}_n-2\mathbf{X}(\mathbf{X}^T\mathbf{X}+\gamma\mathbf{M})^{-1}\mathbf{X}^T)$. The Lagrangian function $\mathcal{L}$ of (11) is

$$\text{Tr}(\mathbf{F}^T\mathbf{R}\mathbf{F})+\frac{\lambda}{2}\|\mathbf{F}^T\mathbf{F}-\mathbf{I}\|_F^2+\text{Tr}(\mathbf{\Phi}^T\mathbf{F}), \quad (12)$$

where $\mathbf{\Phi}$ is the Lagrange multiplier for the inequality constraint. Taking the derivative of $\mathcal{L}$ w.r.t. $\mathbf{F}$ and setting to zero, we have

$$\mathbf{RF}+\lambda\mathbf{F}(\mathbf{F}^T\mathbf{F}-\mathbf{I})+\mathbf{\Phi} = \mathbf{0}. \quad (13)$$

Based on the KKT condition $\phi_{ij}f_{ij} = 0$, we can get following updating rule for entries in $\mathbf{F}$ as

$$f_{ij} \leftarrow f_{ij}\frac{(\lambda\mathbf{F})_{ij}}{(\mathbf{RF}+\lambda\mathbf{F}\mathbf{F}^T\mathbf{F})_{ij}}. \quad (14)$$

After obtaining the updated $\mathbf{F}$, we normalize it to satisfy the orthogonal constraint $\mathbf{F}^T\mathbf{F} = \mathbf{I}_c$.

Based on the above analysis, we summarize our new JGUFS in Algorithm 1.

2.3. Complexity and Convergence Analysis

The complexity of Algorithm 1 is mainly caused by the three blocks in the loop. We need $O(nt_1)$ operations to obtain the affinity matrix $\mathbf{S}$ by an efficient iterative method in which t_1 is the number of iterations of the Newton method. We need $O(d^3+nd^2+ndc)$ operations to update $\mathbf{W}$ by (9) and $O(cn^2)$ operations to update $\mathbf{F}$ in each iteration. Since $c \ll d$ and t_1 is usually relatively small, the overall complexity of Algorithm 1 is $O(t(d^3+nd^2))$ where t is the number of iterations.

We show that the derived updating rules in Algorithm 1 make the objective function monotonically decrease. With

$\mathbf{W}^t$ and $\mathbf{S}^t$ fixed, by introducing an auxiliary function of (10) similar to [18], we can prove that $\mathcal{O}(\mathbf{F}^{t+1}) \leq \mathcal{O}(\mathbf{F}^t)$. Thus, we have $\mathcal{O}(\mathbf{F}^{t+1}, \mathbf{W}^t, \mathbf{S}^t) \leq \mathcal{O}(\mathbf{F}^t, \mathbf{W}^t, \mathbf{S}^t)$. If $(\mathbf{F}^{t+1}, \mathbf{S}^t)$ is the fixed point, we have $\mathcal{O}(\mathbf{F}^{t+1}, \mathbf{W}^{t+1}, \mathbf{S}^t) \leq \mathcal{O}(\mathbf{F}^{t+1}, \mathbf{W}^t, \mathbf{S}^t)$ based on the definition of $\mathbf{M}$ and derivations as in [14]. Since the updating to $\mathbf{S}$ is a closed form solution, it is obvious that $\mathcal{O}(\mathbf{F}^{t+1}, \mathbf{W}^{t+1}, \mathbf{S}^{t+1}) \leq \mathcal{O}(\mathbf{F}^{t+1}, \mathbf{W}^{t+1}, \mathbf{S}^t)$ if $(\mathbf{F}^{t+1}, \mathbf{W}^{t+1})$ is a fixed point. We conclude that JGUFS objective function monotonically decreases under the optimization in Alg. 1.

Algorithm 1 Optimization to JGUFS objective function

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, α, β, and γ, c, the dimension of projected subspace c;

Output: Rank features based on the values of $\|\mathbf{w}_i\|_2|_{i=1}^d$ in descending order and then select the top-ranked ones.

1: Initialization. Construct the initial graph affinity matrix $\mathbf{A}$ based on the 'HeatKernel' function; Calculate $\mathbf{F} \in \mathbb{R}^{n \times c}$ by the c eigenvectors of the graph Laplacian $\mathbf{L_A} = \mathbf{D_A} - \frac{\mathbf{A}^T + \mathbf{A}}{2}$ corresponding to the c smallest eigenvalues; Initialize $\mathbf{M} \in \mathbb{R}^{d \times d}$ as an identity matrix;
2: **while** not converged **do**
3: Update $\mathbf{S}$ by solving (5);
4: Update $\mathbf{W}$ by (9);
5: Update $\mathbf{M}$ by (7);
6: Update $\mathbf{F}$ by (14);
7: **end while**

3. EXPERIMENT

3.1. Experimental Settings for Clustering Problems

Seven benchmark data sets were used in the experiments including JAFFE, UMIST, USPS, MNIST, COIL20, WebKB, and ISOLET. In the experiments, we normalized feature into [0,1]. Detailed information was demonstrated in Table 1.

Table 1. Dataset description.

Dataset	# Samples	# Features	# Clusters
JAFFE	213	676	10
UMIST	575	644	20
USPS	9298	256	10
MNIST	5000	784	10
COIL20	1440	1024	20
WebKB	814	4029	7
ISOLET	1560	617	26

In the experiments, we set the projected dimension of subspace to the number of clusters, that is, $\mathbf{W} \in \mathbb{R}^{d \times c}$. We set the number of selected features as $\{50, 100, \cdots, 300\}$ for all the data sets except USPS. Since the total features of USPS are 256, we set them as $\{50, 100, \cdots, 250\}$. Once obtaining the selected features, we run ten times K-means clustering from different staring points and report the average results with standard deviations. Two metrics, *i.e.*, Accuracy (ACC) and Normalized Mutual Information (NMI), were used to measure the clustering performance. We compare JGUFS with one baseline *All-Fea* and several state-of-the-art methods including *MaxVar*, *LapScore* [19], *MCFS* [20], *FSSL* [21], *UDFS* [15], *NDFS* [16], and *JELSR* [22]. To keep fair comparison, we tuned the parameters involved in each method from $\{10^{-3}, 10^{-2}, \cdots, 10^3\}$ and reported the best results under the optimal parameter combination.

3.2. Experimental Results

Table 2 shows the results of all compared unsupervised feature selection methods. From the experimental results, we have several findings: 1) Feature selection is effective and necessary, which not only saves a lot of storage resources but also considerably improves the clustering performance. In most cases, feature selection methods can provide better performance by using the selected feature subset than directly using all features. 2) Both local structure and discriminative information are beneficial since they can effectively characterize the properties of data from two complementary perspectives which have been extensively investigated in both supervised and unsupervised learning. 3) Evaluating features jointly by employing the $\ell_{2,1}$-norm is more efficient than investigating features one after another based on certain criteria. Generally, the results obtained by FSSL, UDFS, NDFS, JELSR and JGUFS are better than those of MaxVar, LapScore and MCFS. 4) Learning an optimal structured graph for unsupervised feature selection is better than performing feature selection on a fixed graph. Existing methods such as UDFS and NDFS construct the graph using a predefined similarity measure (*i.e.*, 'Heat Kernel' function) which may not be appropriate for all data sets. JGUFS can jointly perform feature selection and graph learning in which the two sub-objectives can co-evolve towards the optimum. It can effectively avoid the limitations caused by the widely used two-stage strategy in graph-based learning, that is, firstly constructing a graph and then performing learning tasks on it. Therefore, JGUFS achieves significant performance improvement in comparison with state-of-the-art unsupervised feature selection methods.

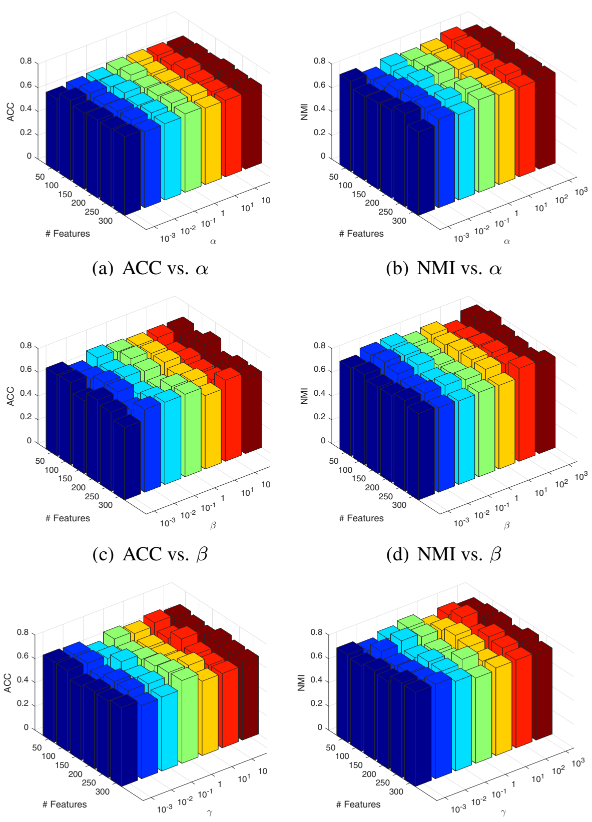
3.3. Parameter Sensitivity and Convergence Study

In JGUFS, there are three regularization parameters; respectively, α is to control the rank of the graph Laplacian matrix to let the learned graph have the desirable block diagonal property; β is to control the fitting error between the projected data and the estimated scaled cluster indicator, and γ measures the row sparsity of the projection matrix. Here we show the clustering performance of JGUFS on ACC and NMI to illustrate the impact of each parameter. We first fix two of the three parameters as one and then investigate the clustering performance in terms of the third one on different number of selected features. Figure 1 illustrates the clustering performance of JGUFS on COIL20 with different settings of parameters. From this figure, we find that JGUFS is not sensitive to the values of parameters in a wide range of variations. That is, JGUFS provides excellent performance when the parameters

Table 2. Comparison of performance of clustering for different feature selection methods (ACC/NMI±std%).

ACC	JAFFE	UMIST	USPS	MNIST	COIL20	WebKB	ISOLET
All-Fea	72.1±3.3	42.9±2.8	63.7±4.1	51.8±4.7	61.7±2.4	55.9±3.1	57.4±3.9
MaxVar	76.3±2.9	46.7±2.4	64.9±3.1	53.0±2.9	61.1±2.8	54.8±2.3	56.9±2.7
LapScore	77.2±3.2	45.8±3.0	64.1±3.2	53.9±3.5	62.1±2.1	56.1±2.8	56.8±2.9
MCFS	79.5±2.7	46.7±3.1	65.1±4.7	55.9±3.7	60.9±2.3	61.5±2.3	60.9±2.5
FSSL	85.6±2.2	51.9±3.3	66.5±2.4	57.1±3.8	62.5±2.8	62.3±2.7	64.9±3.1
UDFS	84.7±2.3	48.9±3.8	66.3±3.0	56.7±3.2	60.8±2.7	61.9±2.9	64.7±3.6
NDFS	86.9±2.5	51.1±3.7	66.9±2.7	58.5±2.8	63.3±2.1	62.5±3.0	65.1±3.9
JELSR	86.5±2.3	53.7±3.2	67.8±2.9	58.1±3.1	64.8±1.9	61.8±2.9	63.7±2.8
JGUFS	**88.3±2.4**	**57.8±2.6**	**69.7±2.8**	**59.3±3.0**	**68.9±1.6**	**63.8±2.7**	**66.8±3.2**
NMI	JAFFE	UMIST	USPS	MNIST	COIL20	WebKB	ISOLET
All-Fea	78.9±2.1	63.5±2.2	59.7±1.8	46.3±2.1	73.5±2.8	11.7±4.2	73.9±1.7
MaxVar	80.3±2.0	65.1±2.0	60.9±1.5	47.9±2.3	71.8±3.1	16.9±2.1	73.7±1.8
LapScore	81.9±1.8	64.7±2.6	60.3±1.3	48.3±2.0	73.9±2.9	13.4±3.5	72.1±1.1
MCFS	82.3±1.8	65.6±1.8	61.7±1.5	50.3±1.7	74.8±2.3	18.3±3.7	74.9±1.6
FSSL	88.6±1.3	67.7±2.0	62.3±1.3	50.8±2.1	75.1±2.7	18.5±3.5	76.8±1.7
UDFS	85.3±2.0	66.5±2.1	61.8±1.5	50.1±1.5	75.7±1.9	17.1±2.9	76.3±1.9
NDFS	87.6±1.9	68.9±2.5	61.3±1.1	51.6±1.1	77.3±1.8	17.6±2.7	78.4±1.2
JELSR	86.9±2.1	70.3±1.7	62.0±1.3	51.1±1.4	77.9±1.7	18.0±3.1	75.8±1.1
JGUFS	**89.8±0.6**	**73.9±2.1**	**63.9±1.1**	**52.9±1.0**	**79.8±1.3**	**20.3±2.3**	**79.9±1.2**

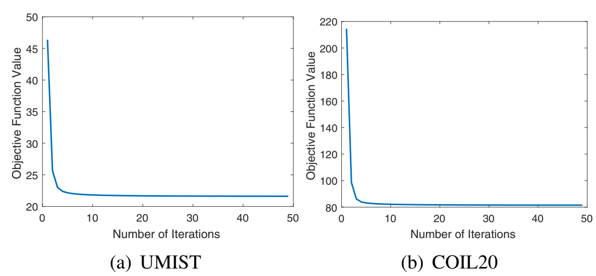
are set as different values in a wide range. Further, we can observe that even if a small number of features are selected, JGUFS can still achieve relatively good clustering results.



(a) ACC vs. α (b) NMI vs. α (c) ACC vs. β (d) NMI vs. β (e) ACC vs. γ (f) NMI vs. γ

Fig. 1. Performance of JGUFS algorithm for large variation of set of control parameters.

As an experimental verification, Figure 2 shows the convergence curves of the JGUFS objective function in terms of the number of iterations on UMIST and COIL20 from which we can observe that JGUFS has a relatively fast convergence speed. Typically, it converges in less than 10 iterations which reflects the proposed optimization method to JGUFS is effective. The converge curves for the other data sets share similar properties.



(a) UMIST (b) COIL20

Fig. 2. Convergence speed of JGUFS for UMIST and COIL20 data sets.

4. CONCLUSION

In this paper, we proposed a novel GUFS method, termed JGUFS, which simultaneously performs graph construction and feature selection. Instead of performing feature selection on a fixed graph, JGUFS successfully avoided the disadvantages caused by the two-stage strategy. In JGUFS, the sub-objectives respectively corresponding to graph construction and unsupervised feature selection could co-evolve towards the optimum. An efficient iterative optimization method with convergence guarantee was presented to optimize the JGUFS objective. Extensive experiments were conducted on representative data sets to demonstrate the excellent performance of JGUFS in comparison with state-of-the-art methods.

5. REFERENCES

[1] Huan Liu and Hiroshi Motoda, *Feature selection for knowledge discovery and data mining*, vol. 454, Springer Science & Business Media, 2012.

[2] Isabelle Guyon and André Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[3] John Wright, Yi Ma, Julien Mairal, Guillermo Sapiro, Thomas S. Huang, and Shuicheng Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, pp. 1031–1044, 2010.

[4] Bing Cheng, Jianchao Yang, Shuicheng Yan, Yun Fu, and Thomas S. Huang, "Learning with ℓ^1-graph for image analysis," *IEEE Transactions on Image Processing*, vol. 19, pp. 858–866, 2010.

[5] Liansheng Zhuang, Haoyuan Gao, Zhouchen Lin, Yi Ma, Xin Zhang, and Nenghai Yu, "Non-negative low rank and sparse graph for semi-supervised learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2328–2335.

[6] Xiaofeng Zhu, Xindong Wu, Wei Ding, and Shichao Zhang, "Feature selection by joint graph sparse coding," in *SIAM International Conference on Data Mining*, 2013, pp. 803–811.

[7] Yong Peng, Bao-Liang Lu, and Suhang Wang, "Enhanced low-rank representation via sparse manifold adaption for semi-supervised learning," *Neural Networks*, vol. 65, pp. 1–17, 2015.

[8] Zhou Zhao, Xiaofei He, Deng Cai, Lijun Zhang, Wilfred Ng, and Yueting Zhuang, "Graph regularized feature selection with data reconstruction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 689–700, 2016.

[9] Pengfei Zhu, Wencheng Zhu, Qinghua Hu, Changqing Zhang, and Wangmeng Zuo, "Subspace clustering guided unsupervised feature selection," *Pattern Recognition*, vol. 66, pp. 364–374, 2017.

[10] Xiaofeng Zhu, Xuelong Li, Shichao Zhang, Chunhua Ju, and Xindong Wu, "Robust joint graph sparse coding for unsupervised spectral feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 6, pp. 1263–1275, 2017.

[11] Shiping Wang and William Zhu, "Sparse graph embedding unsupervised feature selection," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 3, pp. 329–341, 2018.

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

[13] 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.

[14] Feiping Nie, Heng Huang, Xiao Cai, and Chris Ding, "Efficient and robust feature selection via joint ℓ2,1-norms minimization," in *Advances in Neural Information Processing Systems*, 2010, pp. 1813–1821.

[15] Yi Yang, Heng Tao Shen, Zhigang Ma, Zi Huang, and Xiaofang Zhou, "$\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning," in *International Joint Conference on Artificial Intelligence*, 2011, vol. 22, pp. 1589–1594.

[16] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, and Hanqing Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *AAAI Conference on Artificial Intelligence*, 2012, vol. 2, pp. 1026–1032.

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

[18] Daniel D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, 2001, pp. 556–562.

[19] Xiaofei He, Deng Cai, and Partha Niyogi, "Laplacian score for feature selection," in *Advances in Neural Information Processing Systems*, 2005, pp. 507–514.

[20] Deng Cai, Chiyuan Zhang, and Xiaofei He, "Unsupervised feature selection for multi-cluster data," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010, pp. 333–342.

[21] Quanquan Gu, Zhenhui Li, and Jiawei Han, "Joint feature selection and subspace learning," in *International Joint Conference on Artificial Intelligence*, 2011, pp. 1294–1299.

[22] Chenping Hou, Feiping Nie, Xuelong Li, Dongyun Yi, and Yi Wu, "Joint embedding learning and sparse regression: a framework for unsupervised feature selection," *IEEE Transactions on Cybernetics*, vol. 44, no. 6, pp. 793–804, 2014.