ICANN19

28th International Conference on Artificial Neural Networks

Adaptive Graph Fusion for Unsupervised Feature

Sijia Niu, Pengfei Zhu*, Qinghua Hu, Hong Shi

College of Intelligence and Computing, Tianjin University, China

Contact Author: narska_0919@tju.edu.cn

Abstract

Most methods rely on spectral clustering to generate pseudo labels to guide feature selection in unsupervised setting. The construction of adjacency graphs could be affected by the parameters of kernel functions, the number of nearest neighbors or the size of the neighborhood. However, it is difficult to evaluate the effectiveness of different graphs in unsupervised feature selection. In this paper, we propose a novel adaptive multi-graph fusion based unsupervised feature selection model (GFFS). The proposed model is free of graph selection and can combine the complementary information of different graphs. Experiments on benchmark datasets show that GFFS outperforms the state-of-the-art unsupervised feature selection algorithms.

Method

We denote $X = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^{d \times n}$as the data matrix, where d is the dimension of features and n is the number of samples. For a given X ,$U = \{u_{ij}\} \in \mathbb{R}^{n \times n} (\forall i, j \in 1,2,\ldots,n)$ is the affinity matrix and the corresponding degree matrix can be constructed to$D$. And as a description of graph, Laplacian matrix is L. For multiple graphs, let m be the number of graphs and L_v means any matrix of the set of Laplacian matrices. So we define the cluster indicator matrix $F = [f_1, f_2, \ldots, f_n]^T \in \mathbb{R}^{n \times c}$, where c is the number of classes. We propose the objective function for feature selection:

$$\min_{F,W} \sum_{v=1}^{m} \sqrt{Tr(F^T L_v F)} + \alpha\left(\|X^T W - F\|_F^2 + \beta \|W\|_{2,1}\right)$$

$$s.t. F^T F = I, F \geq 0$$

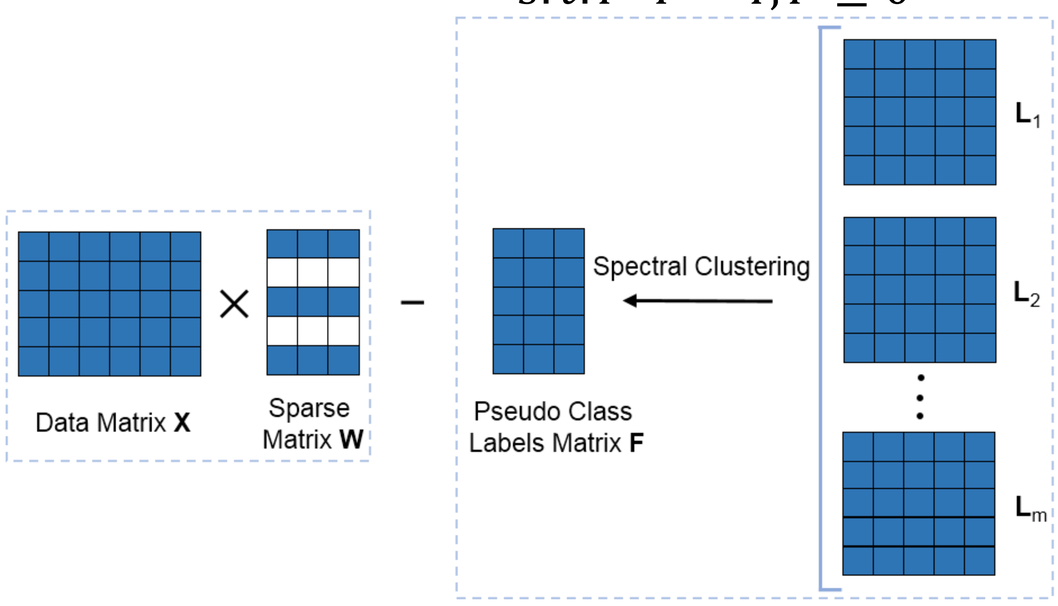

Fig.1: The framework of the proposed adaptive graph fusion for unsupervised feature selection.

Algorithm 1 Graph Fusion Based Unsupervised Feature Selection (when $d > n$)

Input:

Data matrix $\mathbf{X} \in \mathbb{R}^{d \times n}$; Laplacian matrices set $\mathbf{L}$; Parameter α, β, μ, c, and m.

1: Set the iteration step t = 1;
Initialize $\mathbf{F^t} \in \mathbb{R}^{n \times c}$ and set $\mathbf{G^t} \in \mathbb{R}^{d \times n}$ as an identity matrix; Initialize the weight vector ${\varphi_v}^t = \frac{1}{m}$ for each graph.
2: **repeat**
3: $\mathbf{A}^t = \alpha(\mathbf{I} - \mathbf{X}^T\mathbf{G}^{t-1}\mathbf{X}(\mathbf{X}^T\mathbf{G}^{-1}\mathbf{X} + \beta\mathbf{I})^{-1})$;
4: $F_{ij}^{t+1} = F_{ij}^t \frac{(\mu\mathbf{F})_{ij}}{(\sum_{v=1}^{m} \varphi_v \mathbf{L}_v \mathbf{F^t} + \mathbf{A^t}\mathbf{F^t} + \mu\mathbf{F}\mathbf{F}^T\mathbf{F})_{ij}}$;
5: $\mathbf{W}^{t+1} = \mathbf{G}^{t-1}\mathbf{X}(\mathbf{X}^T\mathbf{G}^{t-1}\mathbf{X} + \beta\mathbf{I})^{-1}\mathbf{F}^{t+1}$;
6: Update the diagonal matrix
$\mathbf{G}^{t+1} = \begin{bmatrix} \frac{1}{2\|\mathbf{w}_1\|_2} & & \\ & \ldots & \\ & & \frac{1}{2\|\mathbf{w}_d\|_2} \end{bmatrix}$;
7: ${\varphi_v}^{t+1} = \frac{1}{2\sqrt{Tr(\mathbf{F}^{t+1^T}\mathbf{L}_v\mathbf{F}^{t+1})}}$
8: $t = t + 1$;
9: **until** Converge.

Output:

The index by sorting all d features according $\|{\mathbf{w}_i}^t\|_2 (i = 1, \ldots, d)$ in descending order.

Experiment

We selected six datasets to measure the performance of our method and compared with five LSDR algorithms, i.e. LS, SPEC, MCFS, UDFS, NUFS, and EUFS. We apply three common evaluation metrics, i.e., classification accuracy (ACC), normalized mutual information , and clustering accuracy to evaluate the performance.

Table 1: Classification accuracy (ACC %) of different feature selection methods.

DATA	LS	SPEC	MCFS	UDFS	NDFS	EUFS	GFFS	GFFS(k1)	GFFS(k2)	GFFS(k3)
TOX-171	49.75	54.11	66.04	60.35	64.08	54.11	66.08	**66.10**	65.84	**66.74**
ALLAML	65.85	84.04	75.77	87.28	81.27	86.44	**89.25**	82.51	**88.10**	87.53
CLL-SUB-111	58.13	59.47	56.59	63.55	66.45	61.33	**68.88**	**66.97**	63.01	65.57
SMK-CAN-187	60.23	61.32	63.32	63.35	64.65	63.59	**68.21**	**67.35**	66.78	67.26
USPS	68.59	86.56	88.11	77.32	90.40	90.39	**91.39**	90.79	**91.19**	90.90
binalpha	24.03	34.73	54.89	50.54	58.48	49.30	**59.25**	58.86	58.22	**58.93**
PalmData25	96.92	95.04	98.90	98.88	98.90	98.16	**98.97**	98.93	**99.05**	98.96
Mpeg7	54.59	1.43	66.54	56.38	66.54	56.14	**68.48**	**66.62**	66.60	66.59

Table 2: Clustering performance (NMI %) of different feature selection methods.

DATA	LS	SPEC	MCFS	UDFS	NDFS	EUFS	GFFS	GFFS(k1)	GFFS(k2)	GFFS(k3)
TOX-171	9.41	9.83	12.44	19.51	31.29	15.25	**33.37**	28.96	**31.71**	30.98
ALLAML	7.89	20.11	11.34	19.55	30.00	11.01	**33.79**	**30.20**	24.15	27.12
CLL-SUB-111	10.51	19.67	20.20	21.29	21.25	25.05	**26.17**	21.07	20.90	**25.10**
SMK-CAN-187	2.05	1.75	0.25	4.25	6.89	2.44	**10.13**	6.89	**9.38**	7.76
USPS	37.31	52.48	55.49	44.91	57.48	54.10	**58.63**	57.53	57.26	**57.93**
binalpha	36.83	36.93	52.64	53.39	53.38	46.08	**54.67**	**54.78**	54.20	54.18
PalmData25	85.42	83.88	89.33	88.69	89.19	87.74	**89.78**	89.24	**89.62**	89.37
Mpeg7	53.60	27.21	58.65	55.93	63.67	55.93	**64.78**	63.74	**63.89**	63.89

Conclusion

In this paper, we proposed a novel adaptive graph fusion based unsupervised feature selection (GFFS) algorithm. Different from the existing models that use either kernel similarity or self-representation to generate the affinity matrix, GFFS avoids graph selection by automatically learning the weights of graphs and fusing them in a parameter-free way. Extensive experiments on benchmark datasets validate that the proposed model outperforms the state-of-the-art unsupervised feature selection methods. In the future work, we will extend the proposed model to semi-supervised feature selection tasks.