The Ninth International Symposium on Operations Research and Its Applications (ISORA'10)
Chengdu-Jiuzhaigou, China, August 19–23, 2010

An Optimization Model for Fuzzy Binary Clustering

Xianwen Ren Yong Wang Jiguang Wang
Xiang-Sun Zhang

Academy of Mathematics and Systems Science, Chinese Academy of Science, Beijing 100190

Abstract Clustering has been a powerful tool to visualize complex data with extensive applications in many disciplines. In this paper, we propose an optimization-based solution to the fuzzy binary clustering problem by grouping all the data points into two clusters. Our model are based on two assumptions. One is that the similar objects are labeled similarly, which is known as the "cluster assumption" in semi-supervised learning. The other assumption is that the most dissimilar two objects belong to different clusters. The problem is formulated as a quadratic programming model and can seek the optimal fuzzy labels for the objects. Our model can be solved efficiently by designing a fast algorithm. In addition, it can be reformulated as a linear programming solved efficiently if the similarity matrix is sparse. Furthermore, this model can then be extended to explore the hard-binary-clustering and multiple-clustering problems by a few modifications. Experiments on both simulated and real data sets demonstrate the effectiveness of our method.

Keywords Fuzzy binary clustering; Quadratic programming; Linear programming; Spectral clustering; Semi-supervised clustering

1 Introduction

Clustering has been a basic tool for researchers to explore the structures of the data in various disciplines including data mining, document retrieval, image segmentation, bioinformatics, and so on. As a result a number of methods have been proposed. For instance, objects can be grouped into clusters by parametric models (e.g. k-means algorithm [4]) or algorithms based on some distance or similarity measure (e.g. graph theoretic methods [7], density estimation based methods [6] and physically motivated methods [1]). In this paper, we propose an optimization-based approach for the fuzzy binary clustering problem. It is well known that binary clustering is a basic problem in clustering analysis and it means that all the objects are exactly grouped into two clusters by labeling the objects either zero or one. When binary clustering is properly addressed, it can be used recursively to unravel the multiple-cluster structure of the data. Here, we extend it to fuzzy binary clustering by assigning a fuzzy label between zero and one to every data object and don't require that the data object is classified definitely to some cluster. We think fuzzy binary clustering can utilize more information from data than hard binary clustering and is more reasonable for the real situation. So we pay more attention to fuzzy binary clustering in this paper.

Let X be a dataset of N objects and x_i is the i-th object or data point. A similarity matrix S is defined according to the similarity between any two objects and s_{ij} denotes the similarity between x_i and x_j. Then the fuzzy binary clustering at hand is to assign a real-valued label to each data point so that the information containing in the similarity matrix S is extracted as much as possible.

Usually the objective is the "cluster assumption" [9] by requiring that similar objects should belong to similar cluster, which can be expressed as a quadratic function of the labels. In this case the clustering task is unsupervised. To obtain a proper clustering result, prior information is needed. Here we introduce an intuitive assumption to add the prior information that the most dissimilar two data points must belong to different clusters. This is reasonable because all the data points would belong the same cluster otherwise. Then we construct a quadratic programming model for fuzzy binary clustering. If the similarity matrix S is sparse, the problem can be further reformulated as a linear programming model and can be solved efficiently for large scale problems. Our new clustering model is closely related to the semi-supervised learning and spectral clustering, which will be discussed in detail in the algorithm section and the discussion section. Furthermore, the proposed model can be easily extended to hard binary clustering by choosing a proper threshold and multiple clustering by recursively solving the binary clustering model. Finally, the applications of this new model are illustrated in the experiment section.

2 Method

Given the data set X, our fuzzy binary clustering procedure consists of two steps: (1) calculating the similarity matrix; (2) solving the optimization model. Starting from the basic fuzzy binary clustering model, we can further implement the hard binary clustering or multiple clustering by presenting two algorithms in which the fuzzy binary clustering serves as the elementary operations.

2.1 Calculating the similarity matrix

Clustering model is based on the similarity matrix of the data points and calculating the similarity matrix is the pre-processing step. The method to calculate the similarity matrix is very important because it determines what type of and how much information is extracted from the original data set. In general, measuring the similarity of data points depends on the particular application. So it is hard to give out a prevailing similarity measure that fits everywhere. In this paper the similarity of data points in R^n is calculated by their Gaussian kernels, which can be easily extended to other similarity measures.

Usually the calculated similarity matrix is dense. To facilitate the computation, the full similarity matrix often needs to be converted to a sparse similarity matrix that can be represented by a similarity graph. Two common strategies are the ε-neighborhood graph and the k-nearest neighbor graph [3]. In the ε-neighborhood graph, Two points are connected if their pairwise distance is smaller than ε. Because all the distances of the connected points are roughly of the same scale (less than ε), the ε-neighborhood graph is usually unweighted. In the k-nearest neighbor graph, x_i is connected with x_j if x_j is among the k nearest neighbors of x_i and the edge e_{ij} is weighted as the similarity score of x_i and x_j. This results a directed graph. To guarantee the symmetry of the similarity matrix, x_i and x_j are connected if both x_i is among the k nearest neighbors of x_j and x_j is

among the k nearest neighbors of x_i. The resulting graph is named the mutual k-nearest neighbor graph.

2.2 The optimization models

Given the similarity matrix S, we want to construct an optimization model to assign a real-valued label f_i to each data point x_i so that, 1) similar data points have similar labels, and 2) the most dissimilar two data points belong to two different clusters. The optimization model can be written as followings:

$$\min_{f} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} s_{ij}(f_i - f_j)^2 \tag{1}$$

$$\text{subject to} \quad f_a = 0 \tag{2}$$

$$f_b = 1 \tag{3}$$

$$f_i \le 1 \qquad i \in \{1,2,\cdots,N\} \tag{4}$$

$$f_i \ge 0 \qquad i \in \{1,2,\cdots,N\} \tag{5}$$

Here N is the total number of data points; s_{ij} is the known similarity score of data points x_i and x_j; and f_i is the unknown variable to be determined and represents the label of data point x_i. a and b are the most dissimilar two data points in the N data points, i. e., $s_{ab} = \min\{s_{ij} : i, j \in \{1\cdots N\}\}$. The objective function (1) requires the similar data points have similar labels. Constraints (2) and (3) force data points a and b belonging to two different clusters. Constraints (4) and (5) restrict the labels f_i to be between 0 and 1.

The objective function (1) can be further written in the vector form as:

$$f^T L f \tag{6}$$

where L is the Laplacian matrix of S, i.e., $L = D - S$ and D is a diagonal matrix with $d_{ii} = \sum_{j=1}^{N} s_{ji}$. If s_{ij} are all non-negative, L is positive semi-definite. Then the model (1-5) is a convex optimization problem and the global optimal solution is guaranteed.

If we replace the squared loss function by the absolute loss function, the model is reformulated as follows:

$$\min_{f} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} s_{ij}|f_i - f_j| \tag{7}$$

$$\text{subject to} \quad f_a = 0 \tag{8}$$

$$f_b = 1 \tag{9}$$

$$f_i \le 1 \qquad i \in \{1,2,\cdots,N\} \tag{10}$$

$$f_i \ge 0 \qquad i \in \{1,2,\cdots,N\} \tag{11}$$

Let $l_{ij} = s_{ij}(f_i - f_j)$, the model (7-11) can be further written as follows:

$$\min_{l,f} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} |l_{ij}| \tag{12}$$

$$\text{subject to} \quad l_{ij} = s_{ij}(f_i - f_j) \quad i,j \in \{1,2,\cdots,N\} \tag{13}$$

$$f_a = 0 \tag{14}$$

$$f_b = 1 \tag{15}$$

$$f_i \leq 1 \qquad i \in \{1,2,\cdots,N\} \tag{16}$$

$$f_i \geq 0 \qquad i \in \{1,2,\cdots,N\} \tag{17}$$

if all s_{ij} are non-negative.

By replacing l_{ij} by the difference of its positive part and negative part, the model can be further converted to a linear programming problem as follows:

$$\min_{u,v,f} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} (u_{ij} + v_{ij}) \tag{18}$$

$$\text{subject to} \quad u_{ij} - v_{ij} = s_{ij}(f_i - f_j) \quad i,j \in \{1,2,\cdots,N\} \tag{19}$$

$$f_a = 0 \tag{20}$$

$$f_b = 1 \tag{21}$$

$$f_i \leq 1 \qquad i \in \{1,2,\cdots,N\} \tag{22}$$

$$f_i \geq 0 \qquad i \in \{1,2,\cdots,N\} \tag{23}$$

$$u_{ij} \geq 0 \qquad i,j \in \{1,2,\cdots,N\} \tag{24}$$

$$v_{ij} \geq 0 \qquad i,j \in \{1,2,\cdots,N\} \tag{25}$$

In the quadratic programming model (1-5), there are N variables. If the similarity graph has M edges, the linear programming model has $N+2*M$ variables. Although it increases the number of variables, it makes the objective function a linear function and facilitates the solving of the model when the similarity graph is sparse.

Because the objective function is a graph Laplacian function, the quadratic programming model is closely related to spectral clustering and semi-supervised learning. In spectral clustering, the original data points are first embedded in a new Euclidean space R^p and then k-means is utilized while the model we propose in this paper tries to group the data points without k-means. k-means is a classical algorithm for clustering and can give the correct clustering results in most cases. But it can not guarantee that the solution is globally optimal because the objective function of the k-means algorithm is not convex and it can only get local optimum depending on the choice of initial values. Different from the semi-supervised learning, we introduce label information for two data points by making the assumption that the most dissimilar two data points should belong to two different clusters. Another difference regarding to semi-supervised learning is that the labels of the unlabeled data points in our model are only required to be between 0 and 1, whereas the labels of the unlabeled data points are forced to escape from the unlabeled status in many semi-supervised models.

2.3 Fast algorithm to solve the optimization model

If all s_{ij} are non-negative, the Laplacian matrix L is positive semi-definite. Then, the quadratic programming is a convex programming problem. Generally speaking, it can be solved by the sequential minimal optimization (SMO) algorithm which can deal with large scale data sets and has been applied successfully to solve the optimization problems in support vector machine applications [5]. In addition, the quadratic model can be converted to the linear programming model as we mentioned above and solved by the linear programming solvers which can deal with very large scale problems.

Here we propose an additional specific algorithm to solve the optimization problem efficiently. It is very fast and can identify the clusters for large scale datasets with about 10,000 nodes. The Lagrange function of optimization model (1-5) is:

$$L = \sum_{\mathrm{i}} \sum_{\mathrm{j}} S_{ij}(f_i - f_j)^2 + \alpha f_a + \beta(f_b - 1) - \sum_i \gamma_i f_i + \sum_i \xi_i(f_i - 1)$$

Then the KKT condition is:

$$\begin{array}{lll} \frac{\partial L}{\partial f_i} = 0 \Rightarrow \gamma_i - \xi_i = 2\sum_{\mathrm{j}} S_{ij}(f_i - f_j), & & i = 1,2,\cdots,N \\ \gamma_i f_i = 0, & \xi_i(f_i - 1) = 0 & i = 1,2,\cdots,N \\ f_a = 0, & f_b = 1, & 0 \le f_i \le 1 \\ \gamma_i \ge 0, & \xi_i \ge 0 & \end{array}$$

These conditions can be further reduced as

$$\begin{array}{llll} f_i = 0, & \text{or,} & f_i = \frac{\sum_j S_{ij} f_j}{\sum_j S_{ij}}, & 1 \le i \le N, i \ne a, i \ne b \\ f_a = 0, & f_b = 1 & & \end{array}$$

Then we can use the following iterative algorithm to quickly find the solution from a predetermined initial solution ($f_a = 0, f_b = 1$, and $f_i = 0, i \le 1 \le N, i \ne a, i \ne b$):

$$f_i^{t+1} = \frac{\sum_j S_{ij} f_j^t}{\sum_j S_{ij}}$$

It can be proven that the algorithm is convergent and the convergent solution satisfies the constraints and the KKT condition. Finally the zero and non-zero entries in solution f_i (determined in practice as entries that are greater than a cutoff) define the final clusters.

2.4 Extension to hard binary clustering

If we impose the integer constraints on the label variables $f_i, i \in \{1,2,\cdots,N\}$, the optimization model will be an quadratic integer problem and will lead to a hard binary clustering result. However, the integer programming is NP-hard. To get the hard binary clustering result, we come out a two step method by solving fuzzy binary clustering first and then choosing a threshold to convert the fuzzy labels to integer labels. Because the fuzzy labels extract more information from the similarity matrix than the integer labels,

the fuzzy binary clustering result is expected to be sufficient to give out the hard binary clustering. Here we propose three types of criteria. The first and also the most straightforward threshold is 0.5. Considering the balance of the sizes of the resultant clusters, the median of all the fuzzy labels is another type of threshold. In addition, we propose a third type of criterion based on calculating the gaps among the fuzzy labels. The pseudo-code of this strategy is described as follows:

1. Calculate the similarity matrix S; Identify a pair of data points a and b such that $s_{ab} = \min\{s_{ij} : i, j \in \{1 \cdots N\}\}$; Construct the optimization model (1-5) and solve it.
2. Rank the $\{f_i, i \in \{1 \cdots N\}/\{a,b\}\}$ from the smallest to the largest. Let $h(k) = i$ if f_i is the k-th least, $k \in \{1 \cdots N-2\}$; Calculate gap statistics $\{g_k = \|f_{h(k+1)} - f_{h(k)}\| : k \in \{1 \cdots N-2\}\}$; Identity $k^* = \arg\max_k \{g_k : k \in \{1 \cdots N-2\}\}$.
3. Set $f^* = \frac{1}{2}(f_{h(k^*)} + f_{h(k^*+1)})$; For $i = 1 \cdots N$, if $f_i < f^*$, let $f_i = 0$; if $f_i > f^*$, let $f_i = 1$.

In the real application, the formation of clusters may be determined by more than one factors. So multiple criteria can be considered simultaneously in the conversion from the fuzzy labels to the binary labels.

2.5 Extension to multiple clustering

The data sets in real applications may have more than two clusters. For these cases, we recursively solve our model to classify the data points into two clusters on each step until some stopping criterion is applied. Here we propose two criteria. The pseudo-codes are as follows:

- Minimum similarity score criterion:
  1. **Step 1**: group the data points into two clusters by the method we proposed for hard binary clustering.
  2. **Step 2**: calculate the inner minimum similarity score for each cluster. If the minimum similarity score of some cluster is less than a predefined threshold, then repeat to **Step 1** to group the data points of this cluster into two new clusters.
  3. **Step 3**: repeat **Step 2** until all the inner minimum similarity scores of the clusters are above the threshold.
- Cluster size criterion:
  1. **Step 1**: group the data points into two clusters by the method we proposed for hard binary clustering.
  2. **Step 2**: calculate the numbers of data points in each cluster. If the number of data points in some cluster is above a predefined threshold, then repeat **Step 1** to group the data points of this cluster into two new clusters.
  3. **Step 3**: repeat **Step 2** until all the numbers of data points in each cluster are less than the predefined threshold.

Both the algorithms do not require the number of clusters but requires to define the least tolerant similarity score within clusters or the the most tolerant number of data points in each cluster. Compared with the number of clusters, these two thresholds are relatively

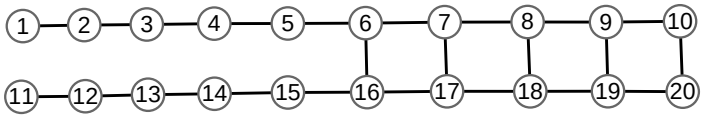

Figure 1: A ladder graph with twenty nodes

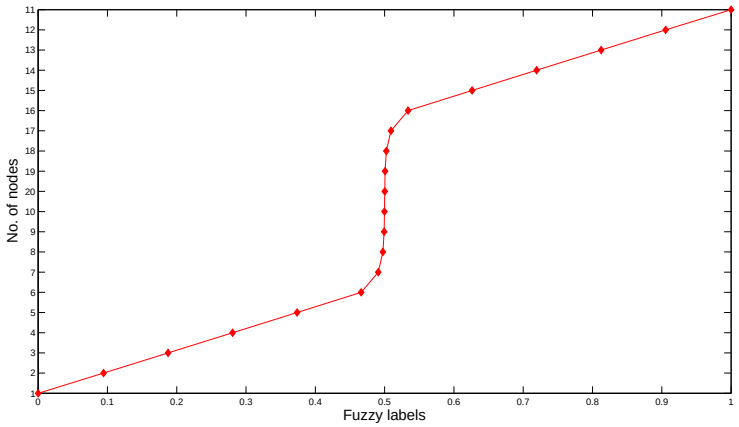

Figure 2: The distribution of the fuzzy labels of the ladder graph

easy to be defined. Furthermore, the algorithm we propose can be applied to the cases in which the data set consists of more than two clusters but the clusters are not organized hierarchically. This will be illustrated by examples in the next section.

3 Experiments

In this section, we show the computational results on both simulated and real datasets. The simulated datasets are used to exemplify how the model groups the data points into clusters. The real data sets are used to test if the model can reveal the true associations underlying the real data. The optimization model is implemented and solved by MATLAB on a PC with 2.4G Hz Pentium 4 processor.

3.1 Experimental results on simulated datasets

We constructed two simulated datasets. One is a ladder graph which is used to show how the fuzzy binary clustering works and how the hard binary clustering is generated based on the fuzzy binary clustering result. The other is a graph with three connected components used to show how non-hierarchically-organized multiple clusters are revealed by our algorithm.

For the ladder graph (Figure 1), the similarity score for vertex v_i and v_j is defined as

Table 1: The fuzzy labels of the nodes in the ladder graph.

No. of nodes	Fuzzy labels	No. of nodes	Fuzzy labels
1	0	6	0.4662
2	0.0944	7	0.4908
3	0.1875	8	0.4975
4	0.2806	9	0.4993
5	0.3737	10	0.4998
11	1	16	0.5338
12	0.9056	17	0.5092
13	0.8125	18	0.5025
14	0.7194	19	0.5007
15	0.6263	20	0.5002

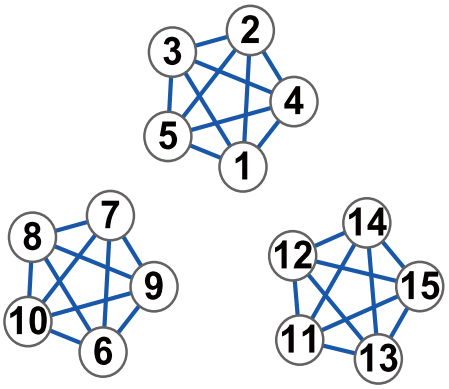

Figure 3: A graph with three connected components

follows:

$$s_{ij} = e^{-\frac{d_{ij}}{\sigma}} \tag{26}$$

where d_{ij} is the length of the shortest path from v_i to v_j and $\sigma = 5$. Applying our model to the ladder graph assigns fuzzy labels between 0 and 1 to each node (Table 1). Sorting the fuzzy labels in the ascending order illustrates the relationships among the nodes (Figure 2). Setting the threshold as 0.5, the nodes from v_1 to v_{10} form a cluster and the nodes from v_{11} to v_{20} form another cluster. Each cluster corresponds to a leg of the ladder. Considering the balance between clusters gets the same result. If the gap statistics is used to convert the fuzzy labels to the hard labels, the nodes nearest to 0.5 form a cluster while the nodes nearest to 0 or 1 form singleton clusters, that is, outliers. This is reasonable from the ladder graph, suggesting that the model can be applied to identify outliers if the gap statistics is used.

To illustrate that our model can deal with the cases with multiple clusters, we show

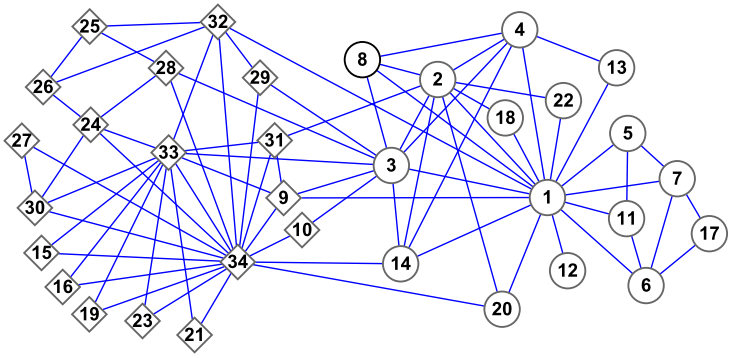

Figure 4: The Zachary's karate club friendship network (circle: Cluster 1; diamond: Cluster 2).

the results on a graph with three connected components of which each connected components is a clique (Figure 3). The adjacency matrix is just used as the similarity matrix. Obviously there are three clusters in the graph and it is intuitively hard to depict exactly the multiple-cluster structure by a binary clustering method. We apply our model on the graph and set the vertex v_1 and v_6 as the most dissimilar node pair. The result shows that the nodes from v_1 to v_5 and from v_{11} to v_{15} are all labeled with 0 while the nodes from v_6 to v_{10} are labeled with 1. That is, the dissimilarity information between $\{v_1, v_2, v_3, v_4, v_5\}$ and $\{v_{11}, v_{12}, v_{13}, v_{14}, v_{15}\}$ is omitted, so the cluster structure among them collapses and they form a meta-cluster. Meanwhile, the dissimilar information between v_1 and v_6 is utilized and the cluster structure with nodes $\{v_6, v_7, v_8, v_9, v_{10}\}$ is revealed. Applying our model further on the meta-cluster which is heterogeneous and of more nodes reveals the cluster structures among $\{v_1, v_2, v_3, v_4, v_5\}$ and $\{v_{11}, v_{12}, v_{13}, v_{14}, v_{15}\}$.

3.2 Experimental results on real data sets

We test our model on two real data sets. One is the Zachary's karate club friendship network [8]. The club consists of 34 members, and splits into two smaller clubs after a dispute happened during the course of Zachary's study (Figure 4). We calculate the Pearson correlation coefficients of the nodes as the similarity scores. Although some of the elements of the similarity matrix are negative, the Laplacian matrix is positive definite. So we calculate the fuzzy labels based on this similarity matrix. The fuzzy labels for all the members are depicted in Figure 5. If the threshold is chosen as 0.5, then the resultant two clusters are just consistent with the real division of the club. If the numbers of nodes in both clusters are the same, then v_9 is misclassified to Cluster 1. If the maximum gap is set to the threshold, then v_{14} and v_{20} are misclassified to Cluster 2. These three nodes are expected to be the intermediate nodes. This suggests that our model is valid to extract the true cluster structures of real data sets.

The other real data set is the well-known Iris data set which is used frequently in clustering and classification [2]. In the Iris data set, there are three clusters. Cluster

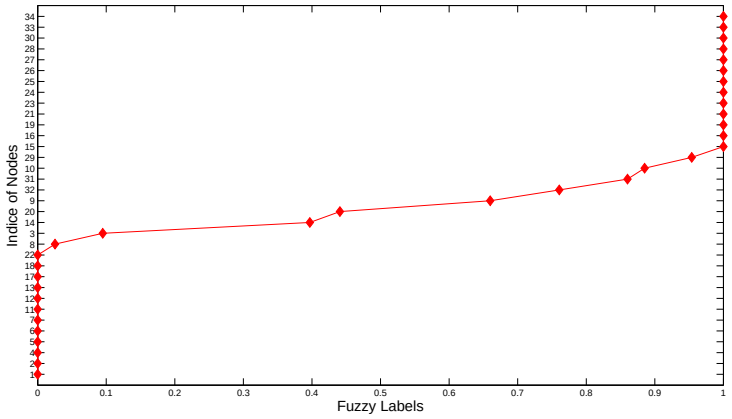

Figure 5: The fuzzy labels for the Zachary's karate club friendship network.

1 is linearly separated from Cluster 2 and Cluster 3, while Cluster 2 and Cluster 3 can not be classified linearly. Here, we use the Gaussian kernel to construct the similarity matrix ($\sigma = 1$). The first binary clustering separates Cluster 1 from Cluster 2 and Cluster 3 accurately. The second run separates Cluster 2 from Cluster 3 with only four data points misclassified, suggesting that our model can be used to reveal the multiple-cluster structure of real datasets.

4 Conclusion

In this paper we propose a new computational model to group the data points in some data set into clusters. A quadratic programming model is proposed for fuzzy binary clustering. In addition, the model has been extended to hard binary clustering and multiple clustering. Different from the classical k-means method, the model we propose is convex and the global optimal solution is guaranteed. It operates on the similarity matrix of the original data set. So the kernel trick can be applied here to deal with the nonlinear cases. Similar to the semi-supervised methods, "prior" information is introduced but in our model it is derived from the similarity matrix. It is closely related to the spectral clustering in which the second least eigenvector is a non-trivial solution of the objective function of our model but in our model the solution has concrete meanings, which is interpreted as the labels of the data points. Experiments on the simulated and real data sets suggest the validity of our method and we expect it provides a useful tool for exploring the structures of data sets in the real world.

Acknowledgements

YW and XSZ are supported by the Grant No. 2006CB503905 from the Ministry of Science and Technology, China. YW is also supported by the Grant No. 10801131 and No. 10701080 from the National Natural Science Foundation of China.

References

[1] Marcelo Blatt and Shai Wiseman and Eytan Domany. Data clustering using a model granular magnet. *Neural Computation*, 1997.

[2] Fisher, Ronald A. The use of multiple measurements in taxonomic problems. *Annals Eugen*,1936,7:179–188.

[3] U. von Luxburg. A tutorial on spectral clustering. 2006.

[4] Macqueen, J. B. Some methods of classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967.

[5] Platt, J. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998.

[6] Stephen J. Roberts. Parametric and Non-parametric Unsupervised Cluster Analysis. *Pattern Recognition*, 1996.

[7] Ron Shamir and Roded Sharan. Algorithmic Approaches to Clustering Gene Expression Data. *Current Topics in Computational Biology*, 2001.

[8] Zachary, W. W. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 1977, 33:452–473.

[9] Xiaojin Zhu. Semi-Supervised Learning Literature Survey. 2006.