Semidefinite Programming을 통한 그래프의 동시 분할법

김재환[0] , 최승진 , 방승양
포항공과대학교 컴퓨터공학과

K-Way Graph Partitioning: A Semidefinite Programming Approach

Jaehwan Kim[0] , Seungjin Choi , Sung-Yang Bang
Department of Computer Science, POSTECH, Korea
{dingkim[O], seungjin, sybang}@postech.ac.kr

Abstract

Despite many successful spectral clustering algorithm(based on the spectral decomposition of Laplacian[1], or stochastic matrix[2]), there are several unsolved problems. Most spectral clustering problems are based on the *normalized cut* algorithm[3], are close to the classical graph paritioning problem which is NP-hard problem. To get good solution in polynomial time, it needs to establish its convex form by using relaxation. In this paper, we apply a novel optimization technique, *semidefinite programming*(SDP), to the unsupervised clustering problem, and present a new multiple partitioning method. Experimental results confirm that the proposed method improves the clustering performance, especially in the problem of being mixed with non-compact clusters, compared to the previous multiple spectral clustering methods.

1. Introduction

Graph partitioning or cutting in an unsupervised way has been the common goal of considerable research in machine learning and pattern recognition. Recently, spectral methods have become popular for clustering tasks. This methods use eigenvectors of an affinity matrix, derived from the distance between points. Such methods, based on the spectral clustering, have been successfully used in many applications such as computer vision including image segmentation, perceptual grouping, and circuit layout design etc. Most of all spectral clustering methods are similar to the *normalized cut* algorithm[3]. And they are close to the classical graph partitioning problem from spectral graph theory[4]. The normalized cut algorithm is specified in terms of a ratio between intra-cluster and inter-cluster similarities of vertices, it is not only a heuristic algorithm, but also NP-hard problem. In this paper, we apply a novel optimization technique, *Semidefinite programming*, to the unsupervised graph partitioning problems, in order to obtain a relaxation form of NP-hard graph partitioning problems. *Semidefinte programming* has been used to obtain approximation results for various combinatorial problems, such as Max-Cut or Graph Coloring. But this approach is defined over binary decision variables. We present a new method that is able to partition over multiple decision variables as well as binary variables simultaneously. Our optimization task has three tasks : First we obtain a convex optimization problem through the semidefinite relaxation. Second we use interior-point methods to solve semidefinite programming in polynomial time. Third, to achieve multiple graph partitioning, we consider an algebraic property of the obtained primal variable matrix. We compare our method with the multiple spectral clustering algorithms described in the paper[2], and confirm that our method improves the clustering performance for non-compact data as well as the other data.

2. Two-Way Grouping As Graph Partitioning

Consider a graph $G(V,E)$ with vertices V and pairwise similarity values as edge weights $w: E \subseteq V\times V \rightarrow R_o^+$. The set V can be partitioned into two coherent groups, S, S', $S \cup S' = V$, $S \cap S' = \emptyset$, by simply cutting edges connecting the two parts. Represnting such a partitioned group by an indicator vector $x \in \{-1,+1\}^n$, a criterion for the bipartitioning can be defined as the degree of dissimilarity between these two pieces, can be computed as total weight of the edges that have been removed. In graph theoretic term, it is called the cut :

$$Cut(S,S')=\sum_{i\in S, j\in S'} w(i,j)=\frac{1}{4}x^TLx \quad (1)$$

Here, $L=D-W$ is the Laplacian matrix of the graph G, D is the diagonal matrix with the elements $d(i,i)$ $\sum_{j\in V} w(i,j)$ and the matrix $W=\{w(i,j)\}$ is the affinity matrix for G. If $w(i,j)=0$ then G has no edge ij. The optimal bipartitioning of a graph is equivalent to the minimization of cut value (1). So coherent groups correspond to low value of the weight function $Cut(S,S')$. But, there is a problem like unbalanced cut. Recently Shi and Malik[3] proposed the following normalized cut criterion, in order to avoid unbalanced partitions which are likely when minimizing $Cut(S,S')$:

$$Ncut(S,S')=(\frac{1}{volS}+\frac{1}{volS'})\sum_{i\in S, j\in S'} w(i,j) \quad (2)$$

where the volume of a set S is $volS=\sum_{i\in S} d(i,i)$[2]. Above approach can be expressed to the following classical combinatorial minimization problem with balancing and integer constraints :

$$\min_x x^TLx,\ e^Tx=0,\ x\in\{-1,+1\}^n. \quad (3)$$

where $e=(1,\ldots,1)^T$. But problem (3) is NP-hard problem. So it is necessary to establish its convex problem through relaxation.

3. Semidefinte Programming Approach

To obtain a semidefinte relaxation of the problem (3) we linearize the cost function, and observe that the convex hull of feasible points of the linearized model is contained in the cone of semidefinite matrices. After we replace the constraints by quardratic ones, we obtain the relaxed problem with respect to problem (3) :

$$z_d^* = \max_{Z,y} e^Ty,\ Z=L-y_0ee^T-D(y)\in S^n_+. \quad (4)$$

where S^n_+ denotes the set of symmetric $n\times n$ matrices which are positive semidefinite. Through the *selfdual* property and *Minimax* inequality[5], we can take the following convex problem :

$$z_p^* = \min_{X\in S^n_+} L\bullet X,\ ee^T\bullet X=0,\ D(X)=I. \quad (5)$$

where X is a Lagrangian multiplier, $X\in S^n_+$. Hence, although the vector x is limited to the binary integer values at the object function(3), matrix X is not limited to the such a constraint (only thing we must consider is that the matrix X is *positive semidefinite*). Both optimization problems which are primal(5) and dual(4), are convex problem. There they yield *no duality gap*: $z_p-z_d=L\bullet X^*-e^Ty^*=0$. In this paper, in order to obtain optimal primal and dual solutions (X^*,Z^*,y^*), we use interior-points method. The basic idea to find a solution is to solve this system iteratively using Newton's method. Typically, a sequence of minimizers (X,Z,y) is computed untill the duality gap falls below some threshold ε

4. K-Way Partitioning Algorithm via SDP

In this section, we describe our proposed multiple graph partitioning method. Algorithm of our method can be expressed in the following procedure :
Given a graph of vertices $V=\{v_1,\ldots,v_n\}$ in R^l that we want to cluster into k subgraphs :

①. Form the affinity matrix $W\in R^{n\times n}$ defined by $W_{ij}=\exp(-\|v_i-v_j\|^2/2\sigma^2)$ if $i\neq j$, $W_{ii}=0$. (where the scaling parameter σ^2 controls the slope of the decaying weight on an edge as a function of the distance.)

②. Construct D is the diagonal matrix with the elements $d(i,i)=\sum_{j\in V} w(i,j)$, and construct the matrix $L=I-P$. (where P is the stochastic matrix, $D^{-1}W$ in [2]. $I-P$ is non-symmetric matrix, so we make the matrix L to be symmetric.)

③. Obtain the primal solution $X^*\in S^n_+$ through the Sec. 3 and interior-points methods. Then X^* can be factored as $X^*=Q\Lambda Q^T$ by spectral decomposition. We form the matrix $C\in R^{n\times k}$ from M defined by $M=[\sqrt{\lambda^{(1)}}q^{(1)},\ldots,\sqrt{\lambda^{(k)}}q^{(k)}]$ in matrix the $Q\Lambda^{1/2}$, through the renormalization of M's rows to have unit length. (where $Q=[q^{(1)},\ldots,q^{(n)}]\in R^{n\times n}$ is orthogonal, i.e., satisfies $Q^TQ=I$, and $\Lambda=diag(\lambda^{(1)},\ldots,\lambda^{(n)})$ in order as $\lambda^{(1)}\leq\lambda^{(2)}\leq\ldots\leq\lambda^{(n)}$. The (real) numbers $\lambda^{(i)}$ are eigenvalues of X.)

④. Treating each row of X as a vertice defined in the k dimensional space, cluster them into k cluster via fuzzy C-means(with k known) or any other algorithm.

⑤. Finally, assign the original vetrice v_i to cluster j if and only if row i of the matrix C was assigned to cluster j.

Here, we compare the objective function of this final semidefinite relaxation (5) with that of the original problem (3). We write the latter as $\min_x x^T L x = \min_x L \cdot xx^T$. Note that xx^T is positive semidefinite and has rank one. A comparision with the relaxed problem (5) shows that xx^T is replaced by an arbitrary matrix $X \in S^n_+$ (rank one condition is dropped). Since the primal solution matrix $X^* \in S^n_+$ in the problem (5), X^* can be expressed as :

$$X^* = GG^T = Q\Lambda Q^T = \sum_i^n \lambda^{(i)} q^{(i)} q^{(i)^T} \qquad (6)$$

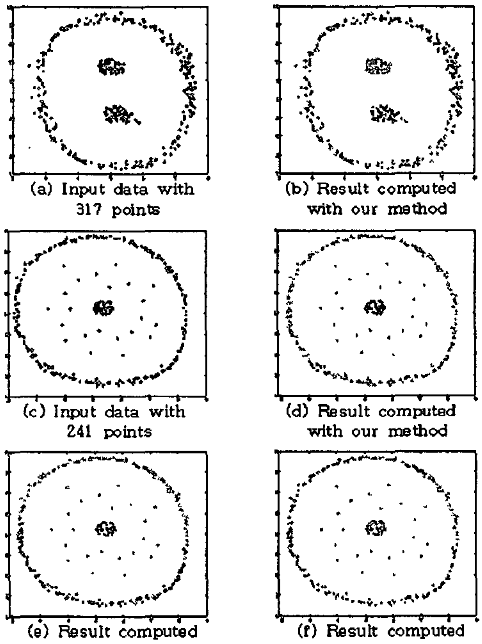
where $G \in R^{n \times n}$ with rank $G = r$ (here, we assume that the value r is equal to the value k the number of subgroups). Therefore, the problem (5) with the property (6) can be derived as follows :

$$\begin{aligned} L \cdot X^* &= tr(LGG^T) = tr(G^T LG) = \sum_i^n \lambda^{(i)} q^{(i)^T} L q^{(i)} \\ &= \Big(\sum_i^n \sqrt{\lambda^{(i)}} q^{(i)}\Big)^T L \Big(\sum_i^n \sqrt{\lambda^{(i)}} q^{(i)}\Big) \\ &= \sum_i^n \gamma^{(i)} \end{aligned}$$

We assume that the original graph is made up of a few subgraphs which have infinite distances among them, then matrix L will be block diagonal matrix with a few subblock matrices, $L^{(i)}$ for $i=1, \dots, m$ (the value m is the number of subgraphs, here we assume that the value m is equal to the value n). And its eigenvalues and eigenvectors are the union of the eigenvalues and eigenvectors of its blocks. That is, the term $(\sqrt{\lambda^{(i)}} q^{(i)})$ is can be shown as the second smallest eigenvector of $L^{(i)}$, and the $\gamma^{(i)}$ is can be also regarded as the second smallest eigenvalue of $L^{(i)}$, for $i=1, \dots, n$. This approach is reasonable because the sum of each second minimum values is also minimum beside the other values's sum except for the first smallest eigenvalues's that. Besides the first contraint in the problem (5) is satisfied when each column vector of the matrix G is the second smallest eigenvector of the matrix $L^{(i)}$.

5. Experimental Results

We presented a semidefinite programming algorithm for the multiple graph cut, which can be implemented using Matlab. In order to test our algorithm, we applied it to a few clustering problems. The results show that our approach is better than the previous methods, especially a multiple spectral method. We compare our method to the *MNCut(Modified NCut)* algorithm described in [2]. Their method is based on he spectral decomposition of the stochastic matrix P.



(a) Input data with 317 points
(b) Result computed with our method
(c) Input data with 241 points
(d) Result computed with our method
(e) Result computed with MNCut
(f) Result computed with K-means

Fig. 1. Comparision our method to the other methods.

6. References

[1] A. T. Ng, M. J. Jordan and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in Advances in Neural Information Processing System. 2002, vol. 14, MIT press.

[2] M. Meila and J. Shi, "A random walks view of spectral segmentation," in Int. Workshop on Artificial Intelligence and Statistics, 2001.

[3] J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp.888-905, 2000.

[4] B. Mohar and S. Poljak, "Eigenvalues in combinatorial optimization," in Combinatorial Graph-Theoretical Problems in Linear Algebra, IMA, S. Friedland and V. Klee, Eds., vol. 50, pp. 107-151. Springer, 1993.

[5] S. Boyd and L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.