

A SURVEY ON GRAPH PARTITIONING APPROACH TO SPECTRAL CLUSTERING

SUBHANSHU GOYAL^{1,a}, SUSHIL KUMAR^{1,b}, M. A. ZAVERI², and A.K.SHUKLA^{1,c}

¹*Department of Applied Mathematics & Humanities, S. V. National Institute of Technology, Surat, Gujarat 395007, India,*

^asubhanshugoyal@gmail.com; ^bskumar.iitr@gmail.com; ^cajayshukla2@rediffmail.com

²*Department of Computer Science & Engineering, S. V. National Institute of Technology, Surat, Gujarat 395007, India*

mazaveri@coed.svnit.ac.in

Abstract. Cluster analysis is an unsupervised technique of grouping related objects without considering their label or class. The objects belonging to the same cluster are relatively more homogeneous in comparison with other clusters. The application of cluster analysis is in areas like gene expression analysis, galaxy formation, natural language processing and image segmentation etc. The clustering problem can be formulated as a graph cut problem where a suitable objective function has to be optimized. This study uses different graph cluster formulations based on graph cut and partitioning problems. A special class of graph clustering algorithm known as spectral clustering algorithms is used for the study. Two widely used spectral clustering algorithms are applied to explaining solution to these problems. These algorithms are generally based on the Eigen-decomposition of Laplacian matrices of either weighted or non-weighted graphs.

Keywords. Eigenvectors, graph cut, laplacian matrix, normalized cut, spectral clustering

1. INTRODUCTION

This survey presented a framework of spectral clustering, a method which utilizes an eigenvector from the so-called data similarity matrix. Computing eigenvectors of such matrices could be potentially a very expensive operation. Thus, faster approximation algorithms for spectral clustering have appeared in the literature. This survey tries to summarize and experimentally evaluate such approximation algorithms.

Cluster analysis has been applied to many areas e.g. gene expression analysis [1], natural language processing [2], galaxy formation [3] and image segmentation [4]. Clustering techniques are divided into two different categories: hierarchical and partitioning techniques. Hierarchical clustering techniques [5, 6, 7] are used to find structure which can be further divided into substructures and so on iteratively. This results is a hierarchical structure of groups which are known as dendrograms. Partitioning clustering methods seek to achieve a single partition of data without any other sub-partition. They are often based on the optimization of an appropriate objective function.

Spectral clustering offers an attractive alternative which clusters data in which eigenvectors of a similarity/affinity matrix are derived from the original data set. In certain cases, spectral clustering even becomes the only option. For instance, when different data points are represented using feature vectors of variable lengths, mixture models or K-means cannot be applied [8], while spectral clustering can still be employed as long as a pair-wise similarity measure can be defined for the data.

Spectral clustering methods arise from concepts of spectral graph theory. The main idea is to construct a weighted graph from the given data set where each node represents a pattern and each weighted edge simply takes into account the similarity between two patterns. In this framework the clustering problem is seen as a graph cut problem, which can be handled by means of the spectral graph theory. The core of this theory is the eigenvalue decomposition of the Laplacian matrix of the weighted graph obtained from data. It has been observed that there is a close relationship between the graph cut and the second smallest eigenvalue of the Laplacian [9, 10].

This paper focuses on main spectral clustering algorithms found in research papers for graph cut and graph partitioning problems. All the spectral graph theory necessary to understand these algorithms will be presented either before or during their descriptions. Moreover, important basic graph concepts are presented for those who are not familiar with graph notations and representations.

The survey is subdivided into various sections in which Section 2 presents spectral graph concepts, definitions, and construction of similarity graphs and function related with spectral algorithms covered in this study. Properties of graph Laplacian matrices important for the comprehension of the spectral clustering algorithms is presented in Section 3. Section 4 presents the different graph cut and partitioning problems for which spectral methods can be important in the literature. Section 5 shows experimental comparisons of two spectral clustering algorithms. In the last section conclusions are drawn.

2. SPECTRAL GRAPH THEORY

Spectral clustering methods [11] for clustering make use of the spectrum of the similarity matrix of the data. Many algorithms have been proposed in the literature [12, 13], each using the eigenvectors in somewhat different ways. A comparison of various spectral clustering methods has been recently proposed by Verma et al. [14].

2.1. Graphs notations

Let $X = \{x_1, x_2, \dots, x_n\}$ be the set of data points or patterns to cluster. Starting from X , a complete, weighted and undirected graph $G = (V, E)$, is built, where V is a non empty set of n nodes (or vertices) and E is a set of m edges. Each edge in E can be defined by the pair (v_i, v_j) , where v_i and v_j are nodes of G , i.e., elements from V . A subgraph of G is a graph $G', G' = (V', E')$, where $V' \subset V$ and $E' \subset E$.

The adjacency matrix of G is a binary matrix, given by $A = [a_{ij}]_{n \times n}$, where $a_{ij} = 1$, if there is an edge connecting nodes v_i, v_j and $a_{ij} = 0$, otherwise. Moreover, weights may be associated with the graph's edges, resulting in weighted graphs. The edge weights are represented by a non negative weight matrix $W = [w_{ij}]_{n \times n}$, where $w_{ij} \geq 0$, $w_{ij} \in \mathbb{R}$ and represents the edge weight between nodes v_i and v_j . If $w_{ij} = 0$, this means that the vertices v_i and v_j are not connected by an edge. If the edges of a graph have no weight, the graph is known as an unweighted graph. As G is undirected, $w_{ij} = w_{ji}$ is required.

In an undirected graph, the degree of a vertex $v_i \in V$ is defined as $d_{ii} = \sum_{j=1}^n a_{ij}$ i.e. the number of its adjacent edges. In an undirected weighted graph, the degree of a node can also be defined as $d_{ii} = \sum_{j=1}^n w_{ij}$ i.e. the sum of weights of its adjacent edges. The following section shows some constructions of similarity graphs and functions from datasets.

2.2. Construction of similarity graph and function

There are cases where data are not originally structured in graphs. In these cases, a similarity graph can be constructed from these data. There are many popular methods to transform a given set x_1, x_2, \dots, x_n of data points with pair-wise similarities s_{ij} or pair-wise distances d_{ij} into a graph. At the time of construction of similarity graphs, the aim is to find out and develop a model for the local neighborhood relationships between the given data points. For such, an undirected weighted graph $G = (V, E)$ where each node v_i is represented by the i^{th} object from a given dataset is considered. The edges of G are defined according to a similarity measure between pairs of objects from this dataset. One of the most frequently used similarity measures is given by the sigmoid function. The weight matrix W of a similarity graph G from the given dataset can be calculated by making $w_{ij} = s(x_i, x_j) = \exp(-d(x_i, x_j)/(2\sigma_d^2))$ if $i \neq j$, and 0, otherwise, where d measures the dissimilarity between patterns and σ controls the rapidity of decay of h . This particular choice has the property wherein W has only few terms considerably different from 0 and is sparse. The parameter σ has a high impact on the clustering partition obtained.

The similarity graph resulting from this strategy can be either used as a complete graph or processed in order to eliminate some of its edges. An alternative for the latter case is the elimination of the edges of a similarity graph whose weights are lower than a predefined threshold. Further details and more options for constructing similarity graphs can be found in [15]. Moreover, [16] is a good source for additional information about the impact of different graph constructions on graph clustering results. Since spectral clustering algorithms are based on the Eigen-decomposition of graph Laplacian matrices, these matrices will be discussed in this study.

3. SPECTRAL GRAPH PARTITIONING

The study of spectral graph theory started in Quantum Chemistry, with a theoretic model of non saturated hydrocarbon molecules [17, 18]. These molecules have chemical linkages with many electron energy levels. Some of these energy levels can be represented by the eigenvalues of a graph. The study of eigenvectors and eigenvalues of a square matrix is the essence of the spectral theory. Since spectral clustering algorithms are based on the Eigen-decomposition of graph Laplacian matrices, so in this section, the different graph Laplacian and their most significant properties are characterized.

The application of spectral theory to graph clustering problems is usually based on the relaxation of some graph partitioning problems. Spectral clustering algorithms are commonly based on fast iterative methods and can be promoted by the use of linear algebra packages, such as the linear algebra package (LAPACK) [19]. In the following, we show some properties of graph Laplacian matrices important for the understanding of spectral clustering algorithms presented in Section 4

3.1. Graph Laplacian Matrices and Their Properties

The main tools used for spectral clustering are graph Laplacian matrices. The study of those matrices is called spectral graph theory [9]. A graph $G = (V, E)$ and its weighted matrix W , such as $w_{ij} \geq 0$ for $i, j = 1, \dots, n$ is considered. Let $D = [d_{ij}]_{n \times n}$, with $d_{ij} \in \mathbb{R}$, be a diagonal matrix defined by $d_{ii} = \sum_{j=1}^n w_{ij}$, i.e., d_{ii} is the degree of node v_i , with $i = 1, \dots, n$. For simplicity reasons, d_{ii} will be referred to here as d_i . The unnormalized graph Laplacian matrix, defined by $L = [l_{ij}]_{n \times n}$ is given by

$$L_{UN} = D - W \tag{1}$$

If a graph is not weighted, its adjacency matrix A instead of the weight matrix W in Eq. (1) is given by

$$L_{UN} = D - A \quad (2)$$

The eigenvalues and eigenvectors of un-normalized graph Laplacian can be used to define various properties of graphs. The Laplacian matrix L is also famous as the Kirchhoff matrix, due to its role in the Matrix-Tree-Theorem [16]. In addition to this description of Laplacian, there are three substitute graph Laplacians given by the following equations.

$$\text{Symmetric Laplacian } L_{sym} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2} \quad (3)$$

$$\text{Generalized } L_G \text{ or Random walk Laplacian } L_{rw} = D^{-1} L = I - D^{-1} W \quad (4)$$

$$\text{Relaxed Laplacian } L_\rho = L - \rho D \quad (5)$$

Laplacian matrices are the heart of the majority of the spectral clustering algorithms. For this reason, some theorems and properties concerning the Laplacian matrix L , considered to be relevant for the spectral relaxation of graph partitioning problems [15, 20] are presented next.

3.1.1. Properties of un-normalized Laplacian L_{UN}

- i. For every vector $f \in \mathbb{R}^n$, results in $f' L f = \frac{1}{2} \sum_{i,j=1}^n w_{ij} (f_i - f_j)^2$ where f_i is the i^{th} component of f .
- ii. L is symmetric and positive semi-definite matrix.
- iii. The smallest eigenvalue of L is 0, the resultant eigenvector is the constant one vector $\mathbb{1}$, where $\mathbb{1}$ is the indicator vector $\mathbb{1} = (1, \dots, 1)^t$.
- iv. L has a non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.

3.1.2. Properties of normalized Laplacian L_N and L_G

The normalized graph Laplacian L_N satisfies the following properties:

- i. For every vector $f \in \mathbb{R}^n$, results in $f' L f = \frac{1}{2} \sum_{i,j=1}^n w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2$.
- ii. λ is an eigenvalue of L_{rw} with eigenvector u if and only if λ is an eigenvalue of L_{sym} with eigenvector $w = D^{1/2} u$.
- iii. λ is an eigenvalue of L_{rw} with eigenvector u iff λ and u solve the generalized eigen problem $L u = \lambda D u$.
- iv. 0 is an eigenvalue of L_{rw} with the constant one vector $\mathbb{1}$ as eigenvector. 0 is an eigenvalue of L_{sym} with eigenvector $D^{1/2} \mathbb{1}$.
- v. L_{sym} and L_{rw} are positive semi-definite and have n non-negative real eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.

The spectral decomposition of the Laplacian matrix gives practical information about the properties of the graph. Spectral approach to clustering has a powerful association with Laplacian eigenmaps [21].

4. GRAPH CUT AND PARTITIONING POINT OF VIEW

The objective of clustering is to separate points in dissimilar groups based on their similarities. For initial data given in the form of a similarity graph, the aim is to find a partition of the graph such that edges between different groups have a low weight and edges in a group have a high weight. In this case, spectral clustering can be defined as an approximation to graph partitioning problems. A large number of graph clustering algorithms are based on graph partitioning problems. This study concerns itself with a particular class of these algorithms, known as spectral clustering algorithms. Spectral clustering algorithms are mostly based on the solution to graph cut problems. For such, they use one or more eigenvectors from Laplacian matrices of a graph to be partitioned that are solutions to the relaxation of some graph cut problems. In this section, how spectral clustering can be used to derive an approximation for such graph partitioning problems is on trial.

4.1. Minimum cut problem

The first problem to be presented is the k-way minimum cut problem. Given a similarity graph with adjacency matrix W , the simplest and most direct way to construct a partition of the graph is to solve the mincut problem. For a given number k of subsets, the mincut approach simply consists of choosing a partition A_1, A_2, \dots, A_k which minimizes

$$\text{mincut}(A_1, A_2, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k (A_i, \bar{A}_i) \quad (6)$$

It aims at minimizing the sum of weights of the edges whose nodes come from different clusters. In many tested graphs, the solutions to this problem are partitions with isolated nodes in clusters [3]. This might be a drawback of many applications, such as in VLSI domain [22].

4.2. Minimum ratio cut problem

Another approach to avoid finding partitions with isolated nodes in clusters is to consider equation (6) divided by the number of elements in each cluster. This formulation was first proposed [23, 24] to solve the bi-partitioning problem also known as two-way ratio cut problem. Later, this formulation was generalized by Chan *et al.* [25] for the k-way ratio cut problem through its connection with the weighted quadratic placement problem formulated by Hall [26]. The k-way minimum ratio cut formulation is represented by following equation:

$$\text{ratiocut}(A_1, A_2, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{|A_i|} = \sum_{i=1}^k \frac{\text{cut}(A_i, \bar{A}_i)}{|A_i|}. \quad (7)$$

4.3. Normalized cut problem

The k-way Ncut problem was proposed by Shi and Malik [27, 12] and was derived from the relation between the normalized association and dissociation measures of a partition.

$$\text{Ncut}(A_1, A_2, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{\text{vol}(A_i)} = \sum_{i=1}^k \frac{\text{cut}(A_i, \bar{A}_i)}{\text{vol}(A_i)} \quad (8)$$

In Ncut, the size is measured by the weights of its edges $vol(a)$ while in Ratio cut, the size of a subset A of a graph is calculated by its number of vertices $|A_i|$ [27, 12]. It is important to note down that both objective functions take small values if the clusters A_i are not too small. Both objective functions attempt to make sure that the clusters are balanced, as measured by the number of edge weights or vertices respectively.

4.4. Min-max cut problem

A distinct cut problem proposed by Ding *et al.* [28] creates the min-max cut problem for clustering. The min-max cut formulation aims at minimizing the inter-cluster similarities maximizing the intra-cluster similarities, as can be seen in equation (9):

$$MinMaxcut(A_1, A_2, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{W(A_i, A_i)} \quad (9)$$

5. SPECTRAL CLUSTERING ALGORITHMS

As mentioned earlier, an option to solve graph partitioning problems is to use the spectral graph theory. A proper relaxation of graph partitioning problems enables the exploration of the eigenvalues and eigenvectors properties of their Laplacian and adjacency matrices. One positive aspect of these methods is the possibility of defining upper or lower bounds for the objective function of the graph partitioning problems. Some promising results of the application of spectral graph theory to data clustering can be found in computer vision [12, 29], VLSI design [30] and detection of clusters in protein structures [31].

Spectral clustering algorithms can be classified into two categories: recursive two-way spectral clustering algorithms and direct k-way spectral clustering algorithms. The first finds the Fiedler eigenvector of a Laplacian matrix of a graph G and recursively partitions G until a k-way partition is found and the second uses the largest $d \geq k$ eigenvectors for partition. This section defines two extensively used spectral clustering algorithms which uses the different graph laplacian and eigenvectors to split a data set into different clusters.

5.1. Unnormalized spectral clustering algorithm

The un-normalized spectral clustering algorithm given by Shi and Malik [12] uses the generalized eigenvectors of laplacian L . The algorithm is composed of these steps:

- Step 1. Compute the affinity matrix $W \in \mathbb{R}^{n \times n}$ defined by $W_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$ if $i \neq j$ and $w_{ij} = 0$.
- Step 2. Construct D to be the diagonal matrix whose (i, i) element is the sum of W 's i^{th} row, and construct the unnormalized Laplacian matrix $L = D - W$.
- Step 3. Perform the eigenvalue decomposition and compute the eigenvectors of L .
- Step 4. Select $\{E_1, E_2, \dots, E_k\}$ first k generalized eigenvectors of the generalized eigen problem $Lu = \lambda Du$.
- Step 5. Form the matrix $X = [E_1, E_2, \dots, E_k] \in \mathbb{R}^{n \times k}$ by stacking the first k-eigenvectors in columns.

Step 6. Treating each row of X as a point in \mathbb{R}^k , cluster them into k cluster using K-means.

Step 7. Finally assign the original point x_i to cluster j if and only if row i of the matrix Y was assigned to cluster j .

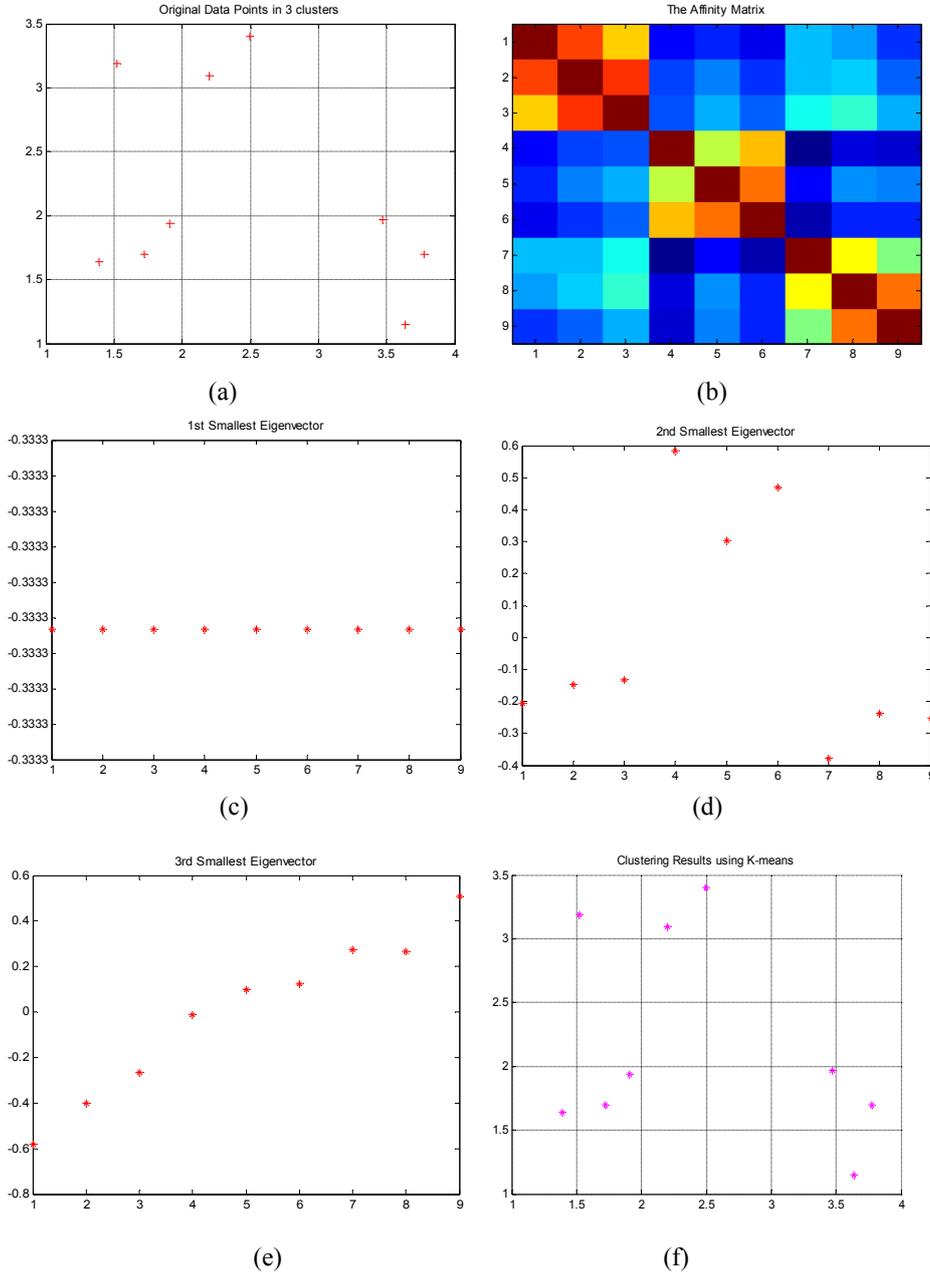


Figure 1: (a) shows a well separated 2-D data set consisting of well separated 3 clusters (b) the affinity matrix shows clear block structure (c)-(e) show that 3 smallest eigenvectors (f) the result of the un-normalized algorithm

5.2. Normalized spectral clustering algorithm

The normalized spectral clustering algorithm given by Ng et al. [13] uses a different Laplacian matrix L than the un-normalized spectral and normalizes the eigenvectors before using k-means to cluster them into k partitions. For a given set of points $S = \{x_1, x_2, \dots, x_n\}$ to be clustered into k subsets, the algorithm is composed of these steps:

- Step 1. Compute the affinity matrix $W \in \mathbb{R}^{n \times n}$ defined by $W_{ij} = \exp\left(-\|x_i - x_j\|^2 / 2\sigma^2\right)$ if $i \neq j$ and $w_{ij} = 0$.
- Step 2. Construct D to be the diagonal matrix whose (i, i) element is the sum of W 's i -th row, and construct the normalized Laplacian matrix $L = D^{-1/2}WD^{-1/2}$.
- Step 3. Perform the eigenvalue decomposition and find the eigenvectors of L .
- Step 4. Select $\{E_1, E_2, \dots, E_k\}$, k largest eigenvectors of L associated to the largest eigenvalues $\{\lambda_1, \lambda_2, \dots, \lambda_k\}$.
- Step 5. Form the matrix $X = [E_1, E_2, \dots, E_k] \in \mathbb{R}^{n \times k}$ by stacking the eigenvectors in columns.
- Step 6. Form the matrix Y from X by renormalizing each of X 's rows to have unit length i.e.

$$\left(Y_{ij} = X_{ij} / \left(\sum_j X_{ij}^2 \right)^{1/2} \right).$$
- Step 7. Treating each row of X as a point in \mathbb{R}^k , cluster them into k cluster using K-means.
- Step 8. Finally assign the original point x_i to cluster j if and only if row i of the matrix Y was assigned to cluster j .

As a condition to select σ , Ng *et al.* recommend to use the value that make sure minimum distortion when the clustering stage is performed on Y . This algorithm on artificial data sets is tested. The above steps of the algorithm are applied to the data points in Figure 2(a). Once the eigen value decomposition of L is computed, the top eigenvectors in Figure 2(c-e) (here obtained with $\sigma = 1.0$) can be seen. And the groups of points are clustered resulting in the Figure 2.

By this algorithm, the eigenvector associated with the largest eigenvalues can be considered as the best for clustering. It has a computational benefit since the principal eigenvectors can be computed for sparse matrices competently by using the power iteration technique. Like other spectral clustering methods one finds a new representation of patterns on the first k eigenvectors of the Laplacian of the graph.

6. CONCLUSION

In general, there are 'n' eigenvalues of similarity matrix of order 'N', which is equal to the total number of nodes in the similarity graph. The increasing ordered set of eigenvalues $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ is referred to as the spectrum of the graph. Normalized spectral clustering algorithm is used for partitioning data by using the largest k eigenvectors of the normalized affinity matrix derived from a dataset. But when the dataset is of complex structure, the affinity matrix constructed by Gaussian function are unable to reflect the real similarity among data points, then the choice of clustering

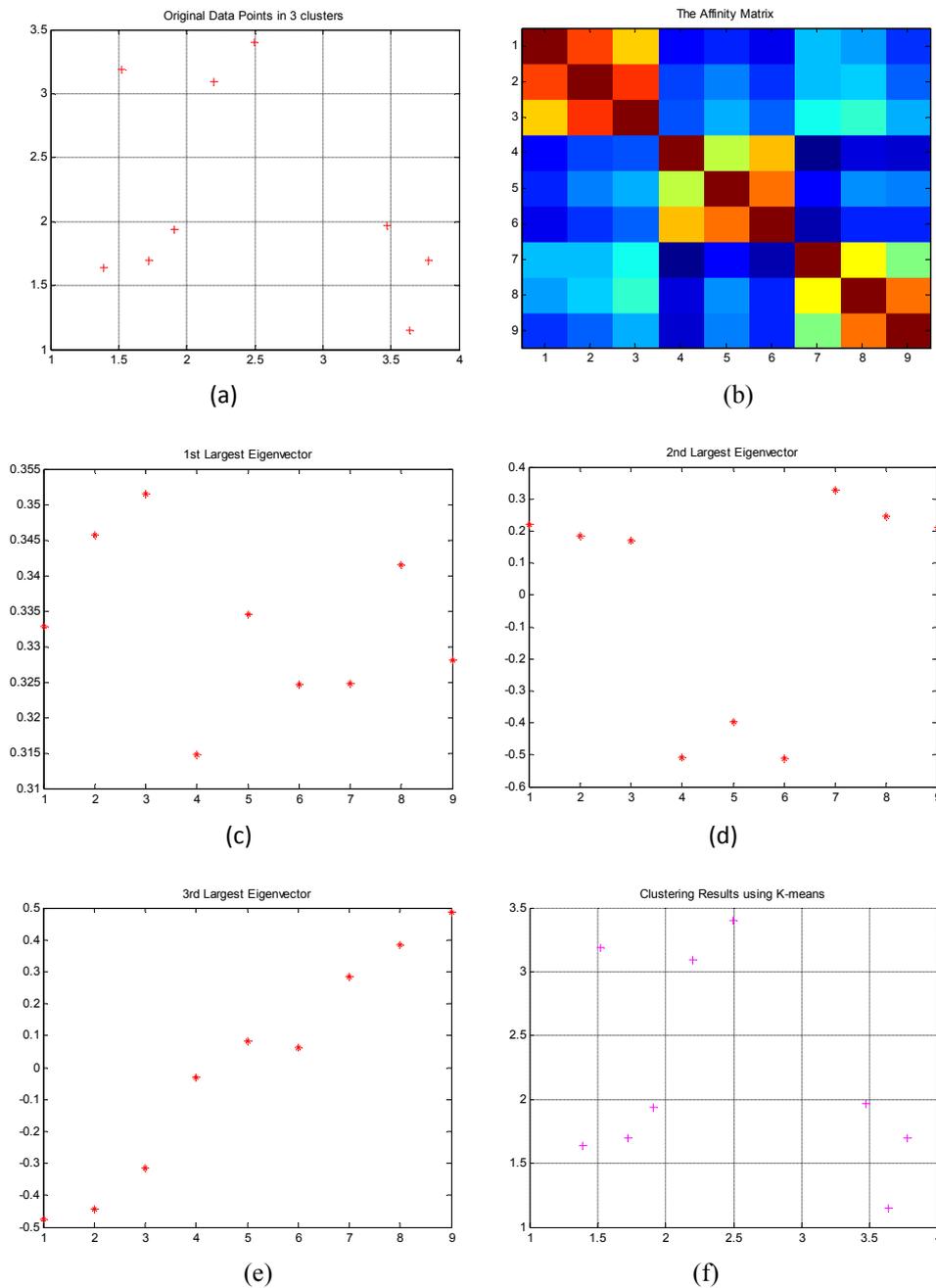


Figure 2: (a) shows a well separated 2-D data set consisting of three clusters (b) the affinity matrix shows clear block structure (c)-(e) show that top 3 eigenvectors contain useful information about the natural grouping of the data (f) the outcome of the normalized algorithm

number and selection of k largest eigenvectors are not always effective. The spectrum of a graph gives a natural clustering of its nodes {fig 2(b)}. The components of eigenvectors denote the node participation in the associated cluster, while the corresponding eigenvalue denotes the coherency of the cluster.

In this survey, it is found that Eigen-decomposition of a similarity matrix aims at reducing the dimensionality of the feature space for data representation. One data point can be represented by each row of an affinity matrix. By doing so N data points are represented in an N -dimensional feature space. After Eigen-decomposition, if all N eigenvectors are used for data representation, the problem will be same. Although no information is lost in this representation, still the clustering process may suffer from the error of dimensionality. Therefore, all spectral clustering algorithms must perform eigenvector selection to cluster the data points.

In a computational study, some effective results using this approach show how eigenvalues and eigenvectors can significantly contribute to define the cluster structure.

ACKNOWLEDGMENT

The first author thanks to Prof. Unni Krishnan A. for helpful conversations about this study.

REFERENCES

- [1] C. Huttenhower, A. I. Flamholz, J. N. Landis, S. Sahi, C. L. Myers, K. L. Olszewski, M. A. Hibbs, N. O. Siemers, O. G. Troyanskaya, and H. A. Collier, "Nearest neighbor networks: clustering expression data based on gene neighborhoods," *BMC Bioinformatics*, vol. 8, no. 1, p. 250, 2007.
- [2] A. Ushioda and J. Kawasaki, "Hierarchical clustering of words and application to NLP tasks," in *Proceedings of the Fourth Workshop on Very Large Corpora*, 1996, pp. 28–41.
- [3] S. D. White and C. S. Frenk, "Galaxy formation through hierarchical clustering," *The Astrophysical Journal*, vol. 379, pp. 52–79, 1991.
- [4] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 11, pp. 1101–1113, 1993.
- [5] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.
- [6] P. H. Sneath, R. R. Sokal *et al.*, *Numerical taxonomy. The principles and practice of numerical classification*. W.H. Freeman, San Francisco, 1973.
- [7] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *Journal of the American statistical association*, vol. 58, no. 301, pp. 236–244, 1963.
- [8] T. Xiang and S. Gong, "Spectral clustering with eigenvector selection," *Pattern Recognition*, vol. 41, no. 3, pp. 1012–1029, 2008.
- [9] F. R. Chung, "Spectral graph theory (cbms regional conference series in mathematics, no. 92)(cbms regional conference series in mathematics)," *American Mathematical Society*, vol. 5, no. 2, pp. 5–3, 1997.
- [10] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298–305, 1973.
- [11] N. Cristianini, J. Shawe-Taylor, and J. S. Kandola, "Spectral kernel methods for clustering," in *Advances in neural information processing systems*, 2001, pp. 649–655.

- [12] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888–905, 2000.
- [13] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems 14*, vol. 2, pp. 849–856, 2002.
- [14] D. Verma and M. Meila, "A comparison of spectral clustering algorithms," *University of Washington Tech Rep UWCSE030501*, vol. 1, pp. 1–18, 2003.
- [15] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [16] B. Mohar and Y. Alavi, "The laplacian spectrum of graphs," *Graph theory, combinatorics, and applications*, vol. 2, pp. 871–898, 1991.
- [17] E. Hückel, "Quantentheoretische beiträge zum benzolproblem," *Zeitschrift für Physik*, vol. 72, no. 5-6, pp. 310–337, 1931.
- [18] D. M. Cvetkovic, M. Doob, and H. Sachs, *Spectra of graphs: Theory and application*. Academic press New York, 1979.
- [19] E. Anderson, Z. Bai, J. Dongarra, A. Greenbaum, A. McKenney, J. Du Croz, S. Hammerling, J. Demmel, C. Bischof, and D. Sorensen, "LAPACK: A portable linear algebra library for high-performance computers," in *Report 90-105, Computer Science Department, University of Tennessee, Knoxville, TN*, 1990.
- [20] L. Hagen and A. B. Kahng, "New spectral methods for ratio cut partitioning and clustering," *Computer-aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 11, no. 9, pp. 1074–1085, 1992.
- [21] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [22] C. J. Alpert and A. B. Kahng, "Multi-way partitioning via spacefilling curves and dynamic programming," in *Proceedings of the 31st annual Design Automation Conference*. ACM, 1994, pp. 652–657.
- [23] T. Leighton and S. Rao, "An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms," in *IEEE Symposium on Foundations of Computer Science*, 1988, pp. 422–431.
- [24] Y.-C. Wei and C.-K. Cheng, "Towards efficient hierarchical designs by ratio cut partitioning," in *Digest of Technical Papers., 1989 IEEE International Conference on Computer-Aided Design*, 1989, pp. 298–301.
- [25] P. K. Chan, M. D. Schlag, and J. Y. Zien, "Spectral k-way ratio-cut partitioning and clustering," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 13, no. 9, pp. 1088–1096, 1994.
- [26] K. M. Hall, "An r-dimensional quadratic placement algorithm," *Management science*, vol. 17, no. 3, pp. 219–229, 1970.
- [27] J. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 731–737.

- [28] C. H. Ding, X. He, H. Zha, M. Gu, and H. D. Simon, “A min-max cut algorithm for graph partitioning and data clustering,” in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*. IEEE, 2001, pp. 107–114.
- [29] B. Luo, R. C. Wilson, and E. R. Hancock, “Spectral embedding of graphs,” *Pattern recognition*, vol. 36, no. 10, pp. 2213–2230, 2003.
- [30] C. J. Alpert, A. B. Kahng, and S.-Z. Yao, “Spectral partitioning with multiple eigenvectors,” *Discrete Applied Mathematics*, vol. 90, no. 1, pp. 3–26, 1999.
- [31] N. Kannan and S. Vishveshwara, “Identification of side-chain clusters in protein structures by a graph spectral method,” *Journal of molecular biology*, vol. 292, no. 2, pp. 441–464, 1999.

Received on June 12 - 2014

Revised on March 16 - 2015