Eigenvectors for clustering: Unipartite, bipartite, and directed graph cases

Andri Mirzal and Masashi Furukawa
Graduate School of Information Science and Technology,
Hokkaido University, Kita 14 Nishi 9, Kita-Ku,
Sapporo 060-0814, Japan

Abstract: This paper presents a concise tutorial on spectral clustering for broad spectrum graphs which include unipartite (undirected) graph, bipartite graph, and directed graph. We show how to transform bipartite graph and directed graph into corresponding unipartite graph, therefore allowing a unified treatment to all cases. In bipartite graph, we show that the relaxed solution to the K-way co-clustering can be found by computing the left and right eigenvectors of the data matrix. This gives a theoretical basis for K-way spectral co-clustering algorithms proposed in the literatures. We also show that solving row and column co-clustering is equivalent to solving row and column clustering separately, thus giving a theoretical support for the claim: “column clustering implies row clustering and vice versa”. And in the last part, we generalize the Ky Fan theorem—which is the central theorem for explaining spectral clustering—to rectangular complex matrix motivated by the results from bipartite graph analysis.

Keywords: eigenvectors, graph clustering, Ky Fan theorem, spectral methods.

1 Introduction

Many papers have been written to reveal the secret and power of spectral clustering; the using of eigenvectors of an affinity matrix induced from a graph to find natural grouping of the vertices. Some noteworthy works are [1, 2, 3, 4, 5], and a comprehensive tutorial can be found in [6]. Despite being intensively studied, it is quite hard to find an intuitive and concise explanation on how and why the spectral clustering works. So, the logic behind the spectral clustering will be explained first.

Most works deal with bipartite data clustering since many real datasets such as a collection of documents, movie ratings, and experimental samples are bipartite. The usual approach for this case is to transform the feature-by-item rectangular matrix induced from a bipartite dataset into a corresponding symmetric matrix by using a kernel function. Then, a similar treatment as in unipartite graph can be employed to this symmetric matrix to find the clusters.

However, simultaneous row and column clustering (co-clustering) works in the original data matrix, hence, the above approach will not work. In subsection 4.2 we show that the co-clustering problem can be restated into the clustering of bipartite graph with two type of vertices—item vertices and feature vertices—where the induced affinity matrix is symmetric. Thus, various clustering algorithms built for unipartite graph can be employed directly.

In directed graph, usually edge directions are ignored to get an equivalent unipartite graph representation. However as noted in [7], ignoring the edge directions can lead to a poor result, and a significant improvement can be achieved by counting for the edge directions into the model. As rows and columns of the induced affinity matrix of a directed graph correspond to the same set of vertices with the same order, as long as the clustering problem is concerned, a symmetric matrix can be formed by simply adding the matrix to its transpose. Therefore, allowing similar treatment as in unipartite graph. We will discuss this more details in subsection 4.3.

A note on notation. $\mathbb{C}^{N\times K}$ denotes an $N\times K$ complex matrix, $\mathbb{R}^{N\times K}$ denotes an $N\times K$ real matrix, $\mathbb{R}_{+}^{N\times K}$ denotes an $N\times K$ nonnegative real matrix, $\mathbb{B}_{+}^{N\times K}$ denotes an $N\times K$ binary matrix, $k\in[1,K]$ denotes $k=1,\ldots,K$, and whenever complex matrix is concerned, transpose operation refers to conjugate transpose.

2 The Ky Fan theorem

The Ky Fan theorem [8] relates eigenvectors of a Hermitian matrix to the trace maximization problem of the matrix.

Theorem 1. *The optimal value of the following prob-*

lem:

$$\max_{\mathbf{X}^T\mathbf{X}=\mathbf{I}_K} \operatorname{tr}(\mathbf{X}^T\mathbf{H}\mathbf{X}) \tag{1}$$

is equal to $\sum_{k=1}^{K}\lambda_k$ *if*

$$\mathbf{X} = [\mathbf{u}_1, \ldots, \mathbf{u}_K]\mathbf{Q}, \tag{2}$$

where $\mathbf{H} \in \mathbb{C}^{N\times N}$ *denotes a full rank Hermitian matrix with eigenvalues* $\lambda_1 \geq \ldots \geq \lambda_N \in \mathbb{R}_+$, $1 \leq K \leq N$, $\mathbf{X} \in \mathbb{C}^{N\times K}$ *denotes a unitary matrix,* $\mathbf{I}_K$ *denotes a* $K \times K$ *identity matrix,* $\mathbf{u}_k \in \mathbb{C}^N$ *denotes* k*-th eigenvector corresponds to* λ_k*, and* $\mathbf{Q} \in \mathbb{C}^{K\times K}$ *denotes an arbitrary unitary matrix.*

The solution to eq. 1 is not unique since $\mathbf{X}$ remains equally good for arbitrary rotation and reflection due to the existence of unitary matrix $\mathbf{Q}$. However, since $[\mathbf{u}_1, \ldots, \mathbf{u}_K]$ is one of the optimal solution, setting $\mathbf{X} = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$ eventually leads to the optimal value.

If $\mathbf{H} \leftarrow \mathbf{W} \in \mathbb{R}_+^{N\times N}$ where $\mathbf{W}$ denotes a symmetric affinity matrix induced from a graph, and $\mathbf{X}$ is constrained to be nonnegative while preserving the orthogonality, i.e., $\mathbf{X}^T\mathbf{X} = \mathbf{I}_K$, then problem in eq, 1 turns into K-way graph cuts problem. Therefore, the Ky Fan theorem can be viewed as a relaxed version of the graph cuts. This relationship explains the logic behind the spectral clustering, where an orthogonal nonnegative clustering indicator matrix is derived by computing the first K eigenvectors of $\mathbf{W}$.

Eigenvectors of a matrix can be computed by using singular value decomposition (SVD). Hence, SVD will be discussed in the following section as many algorithms and software for computing SVD are available for use.

3 Singular value decomposition

SVD is a matrix decomposition technique that factorizes a matrix into a combination of left eigenvectors, right eigenvectors, and eigenvalues. SVD of a full rank matrix $\mathbf{A} \in \mathbb{C}^{M\times N}$ is defined as:

$$\mathbf{A} = \sum_{k=1}^{\min(M,N)} \sigma_k \mathbf{u}_k \mathbf{v}_k^T. \tag{3}$$

Or, in a more compact form can be written as:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{4}$$

where $\mathbf{U} \in \mathbb{C}^{M\times M} = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$ denotes an orthogonal matrix contains the left singular vectors of $\mathbf{A}$, $\mathbf{V} \in \mathbb{C}^{N\times N} = [\mathbf{v}_1, \ldots, \mathbf{v}_N]$ denotes an orthogonal matrix contains the right singular vectors of $\mathbf{A}$, and $\mathbf{\Sigma} \in \mathbb{R}_+^{M\times N}$ denotes a diagonal matrix contains the singular values $\sigma_1 \geq \ldots \geq \sigma_{\min(M,N)}$ of $\mathbf{A}$ along its diagonal.

In practice, usually rank-K approximation of $\mathbf{A}$ is used instead:

$$\mathbf{A}_K = \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^T, \tag{5}$$

where usually $K \ll \min(M, N)$, $\mathbf{U}_K$ and $\mathbf{V}_K$ contain the first K columns of $\mathbf{U}$ and $\mathbf{V}$ respectively, and $\mathbf{\Sigma}_K$ denotes a $K \times K$ principal submatrix of $\mathbf{\Sigma}$. According to Eckart-Young theorem, $\mathbf{A}_K$ is the closest rank-K approximation of $\mathbf{A}$ [9].

In the following section we show how to modify graph clustering objectives into trace maximization of corresponding symmetric matrices. And by relaxing the nonnegativity constraints, according to the Ky Fan theorem, clustering problems eventually become the tasks of finding the first K eigenvectors of the matrices, which are exactly the SVD problems.

4 Graph clustering

Graphs usually can be represented by symmetric, rectangular, or square affinity matrices. A collection of items connected by weighted edges describing similarities between item pairs like a friendship network can be modeled by a unipartite graph, then a symmetric affinity matrix can be induced from this graph. A collection of documents (and in general any bipartite dataset) can be modeled by a bipartite graph, and a term-by-document rectangular matrix containing (adjusted) frequencies of those terms in the documents can be constructed. And a square affinity matrix can be induced from a (unipartite) directed graph like WWW network.

Let $\mathcal{G}(\mathbf{A}) \equiv \mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{A})$ be the graph representation of a collection with $\mathcal{V}$ denotes the set of vertices, $\mathcal{E}$ denotes the set of edges connecting vertex pairs, and $\mathbf{A}$ denotes the induced affinity matrix. The K-way graph clustering is the problem of finding the best cuts on $\mathcal{G}(\mathbf{A})$ that maximize within cluster association, or equivalently, minimize inter cluster cuts to produce K clusters of $\mathcal{V}$.

Here we state two assumptions to allow the graph cuts be employed in clustering.

Assumption 1. *Let* e_{ij} *be an edge connecting vertex* v_i *to* v_j*, the weight value* $|e_{ij}|$ *denotes the similarity between* v_i *and* v_j *linearly, i.e., if* $|e_{ij}| = n|e_{ik}|$ *then* v_i *is* n *times more similar to* v_j *than to* v_k*. And zero weight means no similarity.*

Note that similarity term has many interpretations depending on the domain. For example, in the city road network the similarity can refer to the distance; the closer the distance between two points, the more similar those points are. And in the movie ratings, the similarity can refer to the number of common movies rated by the users.

Assumption 2. *Graph clustering refers to hard clustering, i.e., for* $\{\mathcal{V}_k\}_{k=1}^K \subset \mathcal{V}$, $\cup_{k=1}^K \mathcal{V}_k = \mathcal{V}$, *and* $\mathcal{V}_k \cap \mathcal{V}_l = \emptyset$ $\forall k \neq l$.

Proposition 1. *Assumption 1 and 2 lead to the grouping of similar vertices in* $\mathcal{G}(\mathbf{A})$.

Proof. Consider $\mathcal{G}(\mathbf{A})$ to be clustered into K groups by initial random assignments. Since assumption 1 guarantees $|e_{ij}|$ to be comparable, and assumption 2 guarantees each vertex to be assigned only to a single cluster, cluster assignment for v_i (z_{ik}) can be found by finding a cluster's center that is most similar to v_i.

$$z_{ik} = \arg\max_k \left(\sum_{j_k = v_{j_k} \in \mathcal{V}_k} \frac{|e_{ij_k}|}{|\mathcal{V}_k|} \;\middle|\; k \in [1, K] \right), \quad (6)$$

where $|\mathcal{V}_k|$ denotes the size of cluster k. The objective in eq. 6 is the K-means clustering applied to $\mathcal{G}(\mathbf{A})$, therefore leads to the grouping of similar vertices. □

Note that assumption 1 is an ideal situation which generally doesn't hold. For example, in bipartite representation of a term-by-document matrix, usually the relationships between term-document pairs are not linear to the corresponding term frequencies. Therefore, preprocessing steps (e.g., feature selection and term weighting) are usually necessary before applying the graph cuts. The preprocessing steps seem to be very crucial for obtaining good results [10], and many works are devoted to find more accurate similarity measures schemes [10, 11, 12, 13].

Even though the similarities have been reflected by the weights in (almost) linear fashion, a normalization scheme on $\mathbf{A}$ generally is preferable to produce balance-size clusters. In fact, normalized association/cuts objectives are proven to offer better results compared to their unnormalized counterparts, ratio association/cuts objectives [3, 5, 14].

Table 1 shows the most popular graph clustering objectives with the first two objectives are from the work of Dhillon et al. [14]. *GWAssoc* (*GWCuts*) refers to general weighted association (cuts), *NAssoc* (*NCuts*) refers to normalized association (cuts), and *RAssoc* (*RCuts*) refers to ratio association (cuts). Since all other objectives can be derived from *GWAssoc* [14], we will only consider *GWAssoc* for the rest of this paper.

4.1 Unipartite graph clustering

Unipartite graph is the framework for deriving a unified treatment for the three graphs, so we discuss it first. The following proposition summarizes the effort of Dhillon et al. [14] in providing a general unipartite graph clustering objective.

Proposition 2. *Unipartite graph clustering can be stated in the trace maximization problem of a symmetric matrix.*

Proof. Let $\mathbf{W} \in \mathbb{R}_+^{N \times N}$ be the symmetric affinity matrix induced from a unipartite graph, K-way partitioning on $\mathcal{G}(\mathbf{W})$ using *GWAssoc* can be found by:

$$\max\ J_u = \frac{1}{K} \sum_{k=1}^{K} \frac{z_k^T \mathbf{W} z_k}{z_k^T \mathbf{\Phi} z_k} \quad (7)$$

where $\mathbf{\Phi} \in \mathbb{R}_+^{N \times N}$ denotes a diagonal matrix with Φ_{ii} associated with weight of v_i, and $\mathbf{z}_k \in \mathbb{B}_+^N$ denotes a binary indicator vector for cluster k with its i-th entry is 1 if v_i in cluster k, and 0 otherwise.

The objective above can be rewritten more compactly in the trace maximization as:

$$\begin{aligned} \max\ J_u &= \frac{1}{K} \operatorname{tr}\left(\frac{\mathbf{Z}^T \mathbf{W} \mathbf{Z}}{\mathbf{Z}^T \mathbf{\Phi} \mathbf{Z}} \right) \\ &= \frac{1}{K} \operatorname{tr}\left(\bar{\mathbf{Z}}^T \mathbf{\Phi}^{-1/2} \mathbf{W} \mathbf{\Phi}^{-1/2} \bar{\mathbf{Z}} \right) \end{aligned} \quad (8)$$

where $\mathbf{Z} \in \mathbb{B}_+^{N \times K} = [\mathbf{z}_1, \ldots, \mathbf{z}_K]$ denotes the clustering indicator matrix, and $\bar{\mathbf{Z}} \in \mathbb{R}_+^{N \times K} = \mathbf{Z} / \left(\mathbf{Z}^T \mathbf{Z}\right)^{1/2}$ denotes its orthonormal version. □

By relaxing the strict nonnegativity constraints, i.e., allowing $\bar{\mathbf{Z}}$ to contain negative values while preserving its orthonormality, according to the Ky Fan theorem, the global optimum of J_u can be obtained by assigning

$$\hat{\mathbf{Z}} = [\mathbf{u}_1, \ldots, \mathbf{u}_K] \mathbf{Q}, \quad (9)$$

where $\mathbf{u}_1, \ldots, \mathbf{u}_K \in \mathbb{C}^N$ denote the first K eigenvectors of $\mathbf{\Phi}^{-1/2} \mathbf{W} \mathbf{\Phi}^{-1/2}$, $\hat{\mathbf{Z}} \in \mathbb{C}^{N \times K}$ denotes a relaxed version of $\bar{\mathbf{Z}}$, and $\mathbf{Q} \in \mathbb{R}^{K \times K}$ denotes an arbitrary orthonormal matrix. Hence, eq. 9 presents a tractable solution for NP-hard problem in eq. 8.

The *GWAsssoc* objective in eq. 8 can be replaced by any objective in table 1 by substituting $\mathbf{W}$ and $\mathbf{\Phi}$ with corresponding affinity and weight matrices. Note that $\mathbf{I}$ denotes the identity matrix, $\mathbf{D} \in \mathbb{R}_+^{N \times N}$ denotes a diagonal matrix with its diagonal entries defined as $D_{ii} = \sum_j W_{ij}$, and $\mathbf{L} = \mathbf{D} - \mathbf{W}$ denotes the Laplacian of $\mathcal{G}(\mathbf{W})$.

4.2 Bipartite graph clustering

Bipartite graph clustering generally refers to the clustering of bipartite datasets—collections of items that are characterized by some shared features. A feature-by-item rectangular data matrix $\mathbf{A} \in \mathbb{R}_+^{M \times N}$ contains entries that describe the relationships between items and features.

Table 1: Graph clustering objectives.

Objective	Affinity matrix	Weight matrix
GWAssoc	$\mathbf{W}$	$\mathbf{\Phi}$
GWCuts	$\mathbf{\Phi}-\mathbf{L}$	$\mathbf{\Phi}$
NAssoc	$\mathbf{W}$	$\mathbf{D}$
NCuts	$\mathbf{D}-\mathbf{L}$	$\mathbf{D}$
RAssoc	$\mathbf{W}$	$\mathbf{I}$
RCuts	$\mathbf{I}-\mathbf{L}$	$\mathbf{I}$

Bipartite graph clustering can be done in two different ways; direct and indirect way. The former method applies the graph cuts directly to $\mathcal{G}(\mathbf{A})$ resulting in partitions that contain both item and feature vertices. And the latter method first transforms $\mathcal{G}(\mathbf{A})$ into an equivalent unipartite graph (either item or feature graph) by calculating similarities between vertex pairs from either item or feature set, and then applies the graph cuts on this unipartite graph. Both methods lead to symmetric affinity matrices, thus *GWAssoc* objective can be applied as in the unipartite graph case equivalently.

4.2.1 Direct treatment

If *GWAssoc* is applied to a bipartite graph, similar items will be grouped together with relevant features. This is known as simultaneous feature and item clustering or *co-clustering.*

Proposition 3. *Bipartite graph co-clustering can be stated in the trace maximization problem of a symmetric matrix.*

Proof. Let $\mathbf{M}\in\mathbb{R}_+^{P\times P}$ $(P=M+N)$ be the symmetric affinity matrix induced from a bipartite graph. $\mathbf{M}$ is defined as:

$$\mathbf{M}=\begin{bmatrix}\mathbf{0} & \mathbf{A}\\ \mathbf{A}^T & \mathbf{0}\end{bmatrix}. \tag{10}$$

Taking *GWAssoc* as the objective, K-way co-clustering can be found by:

$$\max\ J_b=\frac{1}{K}\sum_{k=1}^{K}\frac{z_k^T\mathbf{M}z_k}{z_k^T\mathbf{\Phi}z_k}. \tag{11}$$

Then, eq. 11 can be rewritten as:

$$\max\ J_b=\frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T\mathbf{\Phi}^{-1/2}\mathbf{M}\mathbf{\Phi}^{-1/2}\bar{\mathbf{Z}}\right), \tag{12}$$

where $\mathbf{\Phi}\in\mathbb{R}_+^{P\times P}$, $\mathbf{z}_k\in\mathbb{B}_+^P$, and $\bar{\mathbf{Z}}\in\mathbb{R}_+^{P\times K}$ are defined equivalently as in the unipartite graph case. □

By relaxing the nonnegativity constraints on $\bar{\mathbf{Z}}$, the optimum value of eq. 12 can be found by computing the first K eigenvectors of $\mathbf{\Phi}^{-1/2}\mathbf{M}\mathbf{\Phi}^{-1/2}$.

Instead of constructing $\mathbf{M}$ which is bigger and sparser than the original matrix $\mathbf{A}$, we provide a way to co-cluster bipartite graph directly from $\mathbf{A}$.

Theorem 2. *A relaxed solution to the bipartite graph co-clustering problem in eq. 12 can be found by computing the left and right eigenvectors of normalized version of* $\mathbf{A}$*.*

Proof. Let

$$\bar{\mathbf{Z}}=\begin{bmatrix}\bar{\mathbf{X}}\\ \bar{\mathbf{Y}}\end{bmatrix}, \text{ and } \mathbf{\Phi}=\begin{bmatrix}\mathbf{\Phi}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{\Phi}_2\end{bmatrix} \tag{13}$$

be rearranged into two smaller matrices that correspond to $\mathbf{A}$ and $\mathbf{A}^T$ respectively. Then, eq. 12 can be rewritten as:

$$\max\ J_b=\frac{1}{K}\mathrm{tr}\left(\begin{bmatrix}\bar{\mathbf{X}}\\ \bar{\mathbf{Y}}\end{bmatrix}^T\underbrace{\begin{bmatrix}\mathbf{0} & \bar{\mathbf{A}}\\ \bar{\mathbf{A}}^T & \mathbf{0}\end{bmatrix}}_{\bar{\mathbf{M}}}\begin{bmatrix}\bar{\mathbf{X}}\\ \bar{\mathbf{Y}}\end{bmatrix}\right), \tag{14}$$

where $\bar{\mathbf{A}}=\mathbf{\Phi}_1^{-1/2}\mathbf{A}\mathbf{\Phi}_2^{-1/2}$. Denoting $\hat{\mathbf{X}}\in\mathbb{C}^{M\times K}$ and $\hat{\mathbf{Y}}\in\mathbb{C}^{N\times K}$ as the relaxed version of $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$, by the Ky Fan theorem, the global optimum solution to eq. 14 is given by the first K eigenvectors of $\bar{\mathbf{M}}$:

$$\begin{bmatrix}\hat{\mathbf{X}}\\ \hat{\mathbf{Y}}\end{bmatrix}=\begin{bmatrix}\hat{\mathbf{x}}_1,\ldots,\hat{\mathbf{x}}_K\\ \hat{\mathbf{y}}_1,\ldots,\hat{\mathbf{y}}_K\end{bmatrix}\mathbf{Q}. \tag{15}$$

Therefore,

$$\begin{bmatrix}\mathbf{0} & \bar{\mathbf{A}}\\ \bar{\mathbf{A}}^T & \mathbf{0}\end{bmatrix}\begin{bmatrix}\hat{\mathbf{x}}_k\\ \hat{\mathbf{y}}_k\end{bmatrix}=\lambda_k\begin{bmatrix}\hat{\mathbf{x}}_k\\ \hat{\mathbf{y}}_k\end{bmatrix}, \tag{16}$$

where $k\in[1,K]$ and λ_k denotes k-th eigenvalue of $\bar{\mathbf{M}}$. Then,

$$\bar{\mathbf{A}}\hat{\mathbf{y}}_k=\lambda_k\hat{\mathbf{x}}_k,\ \text{ and} \tag{17}$$

$$\bar{\mathbf{A}}^T\hat{\mathbf{x}}_k=\lambda_k\hat{\mathbf{y}}_k. \tag{18}$$

Thus, a relaxed global optimum solution to the problem in eq. 12 can be found by computing the first K left and right eigenvectors of $\bar{\mathbf{A}}$. □

Theorem 2 generalizes the work of Dhillon [15] where the author only gives a theoretical explanation for 2-way bipartite graph co-clustering. And the multipartitioning algorithm proposed by the author [15] that derived from the bipartitioning algorithm by induction, now has a theoretical explanation.

The following theorem gives a support for an interesting claim in co-clustering: row clustering implies column clustering and vice versa.

Theorem 3. *Solving simultaneous row and column clustering is equivalent to solving row and column clustering separately, and consequently, row clustering implies column clustering and vice versa.*

Proof. By substituting $\hat{\mathbf{y}}_k$ from eq. 18 into eq. 17, and similarly, substituting $\hat{\mathbf{x}}_k$ from eq. 17 into eq. 18, we get:

$$\bar{\mathbf{A}}\bar{\mathbf{A}}^T\hat{\mathbf{x}}_k = \lambda_k^2\hat{\mathbf{x}}_k, \text{ and} \tag{19}$$

$$\bar{\mathbf{A}}^T\bar{\mathbf{A}}\hat{\mathbf{y}}_k = \lambda_k^2\hat{\mathbf{y}}_k, \tag{20}$$

where $\bar{\mathbf{A}}\bar{\mathbf{A}}^T$ and $\bar{\mathbf{A}}^T\bar{\mathbf{A}}$ respectively denote row and column affinity matrices. After some manipulations, we get:

$$\max \text{ tr}\left(\hat{\mathbf{X}}^T\bar{\mathbf{A}}\bar{\mathbf{A}}^T\hat{\mathbf{X}}\right) = \sum_{k=1}^{K}\lambda_k^2, \text{ and} \tag{21}$$

$$\max \text{ tr}\left(\hat{\mathbf{Y}}^T\bar{\mathbf{A}}^T\bar{\mathbf{A}}\hat{\mathbf{Y}}\right) = \sum_{k=1}^{K}\lambda_k^2, \tag{22}$$

where $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_K]$ and $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{x}}_K]$ respectively denote the relaxed row and column clustering indicator matrices. As shown above, $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ can be computed separately, and since $\hat{\mathbf{Y}}$ can be derived from $\hat{\mathbf{X}}$ and vice versa (see eq. 17 and eq. 18), row clustering implies column clustering and vice versa. □

Theorem 2 provides a "shortcut" to computing $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ which are usually be constructed by computing the first K eigenvectors of $\bar{\mathbf{A}}\bar{\mathbf{A}}^T$ and $\bar{\mathbf{A}}^T\bar{\mathbf{A}}$ respectively.

Theorem 4. $\hat{\mathbf{X}}$ *and* $\hat{\mathbf{Y}}$ *can be constructed by computing the first* K *left and right eigenvectors of* $\bar{\mathbf{A}}$.

Proof. As shown in the proof of theorem 3, $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_K]$ and $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_K]$, where according to the proof of theorem 2, $\hat{\mathbf{x}}_k$ and $\hat{\mathbf{y}}_k$ are the k-th left and right eigenvectors of $\bar{\mathbf{A}}$. □

4.2.2 Indirect treatment

There are cases where the data points are inseparable in the original space or clustering can be done more effectively by first transforming $\mathbf{A}$ into a corresponding symmetric matrix $\mathbf{V} \in \mathbb{R}_+^{N\times N}$ (we assume item clustering for the rest of this subsection, feature clustering can be done similarly). Then the graph cuts can be applied to $\mathcal{G}(\mathbf{V})$ to obtain the item clustering.

There are two common approaches to learn $\mathbf{V}$ from $\mathbf{A}$. The first approach is to use kernel functions. Table 2 lists the most widely used kernel functions according to Dhillon et al. [14] with $\mathbf{a}_i$ is i-th column of $\mathbf{A}$, and the unknown parameters (c, d, α, and θ) are either directly determined based on previous experiences or learned from sample datasets.

Table 2: Examples of popular kernel functions [14].

Polynomial kernel	$\kappa(\mathbf{a}_i, \mathbf{a}_j) = (\mathbf{a}_i \cdot \mathbf{a}_j + c)^d$
Gaussian kernel	$\kappa(\mathbf{a}_i, \mathbf{a}_j) = \exp(-\|\mathbf{a}_i - \mathbf{a}_j\|^2/2\alpha^2)$
Sigmoid kernel	$\kappa(\mathbf{a}_i, \mathbf{a}_j) = \tanh(c(\mathbf{a}_i \cdot \mathbf{a}_j) + \theta)$

The second approach is to make no assumption about the data domain nor the possible similarity structure between item pairs. $\mathbf{V}$ is learned directly from the data, thus avoiding some inherent problems associated with the first approach, e.g., (1) no standard in choosing the kernel function and (2) similarities between item pairs are computed independently without considering interactions among items. Some recent works on this approach can be found in [11, 12, 13].

Proposition 4. *Clustering on* $\mathcal{G}(\mathbf{V})$ *can be stated in the trace maximization of* $\mathbf{V}$.

Proof. If the first approach to be used, entries of $\mathbf{V}$ can be determined using a kernel function,

$$V_{ij} = \begin{cases} \kappa(\mathbf{a}_i, \mathbf{a}_j) & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \tag{23}$$

Similarly, if the second approach to be used, $\mathbf{V}$ can be learned directly from the data. Then, by using *GWAssoc* as the objective, K-way clustering on $\mathcal{G}(\mathbf{V})$ can be computed by:

$$\max\ J_b = \frac{1}{K}\text{tr}\left(\bar{\mathbf{Z}}^T\mathbf{\Phi}^{-1/2}\mathbf{V}\mathbf{\Phi}^{-1/2}\bar{\mathbf{Z}}\right) \tag{24}$$

where $\bar{\mathbf{Z}}$ and $\mathbf{\Phi}$ are defined equivalently as in the unipartite graph case. □

If asymmetric metrics like Bregman divergences are used as the kernel functions, the resulting $\mathbf{V}'$ will be asymmetric. Accordingly, $\mathcal{G}(\mathbf{V}')$ is a directed graph, and therefore it must be treated as a directed graph.

4.3 Directed graph clustering

The researches on directed graph clustering come from complex network studies conducted mainly by physicists. Different from conventional method of ignoring the edge directions, complex network researchers preserve this information in their proposed methods. As shown in [7, 16], accomodating it can be very useful in improving clustering quality. In some cases, ignoring the edge directions can lead to the clusters detection failure [17].

The directed graph clustering usually is done by mapping the original square affinity matrix into another square matrix which entries are adjusted to emphasize the importance of the edge directions. Some mapping functions can be found in, e.g., [7, 16, 17]. To make use of the available clustering methods for unipartite graph, some works [7, 17] construct a symmetric matrix representation of the directed graph without ignoring the edge directions.

Here we describe the directed graph clustering by naturally following the previous discussions on the unipartite and bipartite graph cases.

Proposition 5. *Directed graph clustering can be stated in the trace maximization problem of a symmetric matrix.*

Proof. Let $\mathbf{B} \in \mathbb{R}_+^{N\times N}$ be the affinity matrix induced from a directed graph, and $\mathbf{\Phi}_i$ and $\mathbf{\Phi}_o$ be diagonal weight matrices associated with indegree and outdegree of vertices in $\mathcal{G}(\mathbf{B})$ respectively. We define a diagonal weight matrix of $\mathcal{G}(\mathbf{B})$ with:

$$\mathbf{\Phi}_{io} = \sqrt{\mathbf{\Phi}_i \mathbf{\Phi}_o}. \tag{25}$$

Since both rows and columns of $\mathbf{B}$ correspond to the same set of vertices with the same order, the row and column clustering indicator matrices are the same, matrix $\bar{\mathbf{Z}}$. By using *GWAssoc*, K-way clustering on $\mathcal{G}(\mathbf{B})$ and $\mathcal{G}(\mathbf{B}^T)$ can be found by:

$$\max\ J_{d1} = \frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T \mathbf{\Phi}_{io}^{-1/2}\mathbf{B}\mathbf{\Phi}_{io}^{-1/2}\bar{\mathbf{Z}}\right), \text{ and} \tag{26}$$

$$\max\ J_{d2} = \frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T \mathbf{\Phi}_{io}^{-1/2}\mathbf{B}^T\mathbf{\Phi}_{io}^{-1/2}\bar{\mathbf{Z}}\right) \tag{27}$$

respectively. By adding the two objectives above, we obtain:

$$\max\ J_d = \frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T \mathbf{\Phi}_{io}^{-1/2}\left(\mathbf{B}+\mathbf{B}^T\right)\mathbf{\Phi}_{io}^{-1/2}\bar{\mathbf{Z}}\right), \tag{28}$$

which is the trace maximization problem of a symmetric matrix $\mathbf{\Phi}_{io}^{-1/2}\left(\mathbf{B}+\mathbf{B}^T\right)\mathbf{\Phi}_{io}^{-1/2}$. □

The directed graph clustering raises an interesting issue in the weight matrix formulation which doesn't appear in the unipartite and bipartite graph cases as the edges are undirected. As explained in the original work [14], $\mathbf{\Phi}$ is introduced with two purposes: *first* to provide a general form of graph cuts objective which other objectives can be derived from it, and *second* to provide compatibility with weighted kernel K-means objective so that eigenvector-free K-means algorithm can be utilized to solve the graph cuts problem.

However, as information of the edge directions appears, defining a weight for each vertex is no longer adequate. To see the reason, let's apply *NAssoc* to $\mathcal{G}(\mathbf{B})$ and $\mathcal{G}(\mathbf{B}^T)$. By using table 1:

$$\max\ J_{d1} = \frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T\mathbf{D}^{-1/2}\mathbf{B}\mathbf{D}^{-1/2}\bar{\mathbf{Z}}\right) \text{ and} \tag{29}$$

$$\max\ J_{d2} = \frac{1}{K}\mathrm{tr}\left(\bar{\mathbf{Z}}^T\mathbf{D}^{*-1/2}\mathbf{B}^T\mathbf{D}^{*-1/2}\bar{\mathbf{Z}}\right), \tag{30}$$

where D and D^* are diagonal weight matrices with $D_{ii} = \sum_j B_{ij}$ and $D^*_{ii} = \sum_i B_{ij}$ respectively. But now $J_{d1} + J_{d2}$ won't end up in a nice trace maximization of a symmetric matrix as in eq. 28. Therefore, we cannot apply the Ky Fan theorem to find a relaxed global optimum solution.

This motivates us to define a more general form of the weight matrix, $\mathbf{\Phi}_{io}$, which allows directed graph clustering be stated in the trace maximization of a symmetric matrix, yet still turns into $\mathbf{\Phi}$ if the corresponding affinity matrix is symmetric.

In the case of *NAssoc* and *NCuts*, $\mathbf{\Phi}_i$ and $\mathbf{\Phi}_o$ are defined as:

$$\mathbf{\Phi}_i = \mathrm{diag}\left(\sum_i B_{i1}, \ldots, \sum_i B_{iN}\right) \text{ and} \tag{31}$$

$$\mathbf{\Phi}_o = \mathrm{diag}\left(\sum_j B_{1j}, \ldots, \sum_j B_{Nj}\right). \tag{32}$$

Note that there is no need to define weight matrix for *RAssoc* and *RCuts* since $\mathbf{I}$ is used.

5 Extension to the Ky Fan Theorem

Theorem 2 implies an extension to the Ky Fan theorem for more general rectangular complex matrix.

Theorem 5. *The optimal value of the following problem:*

$$\max_{\mathbf{X}^T\mathbf{X}=\mathbf{Y}^T\mathbf{Y}=\mathbf{I}_K} \mathrm{tr}(\mathbf{X}^T\mathbf{R}\mathbf{Y}), \tag{33}$$

is equal to $\sum_{k=1}^K \lambda_k$ *if*

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_K]\mathbf{Q}, \text{ and} \tag{34}$$

$$\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_K]\mathbf{Q} \tag{35}$$

where $\mathbf{R} \in \mathbb{C}^{M\times N}$ *denotes a full rank rectangular complex matrix with eigenvalues* $\lambda_1 \geq \ldots \geq \lambda_{\min(M,N)} \in \mathbb{R}_+$, $0 \leq K \leq \min(M,N)$, $\mathbf{X} \in \mathbb{C}^{M\times K}$ *and* $\mathbf{Y} \in \mathbb{C}^{N\times K}$ *denote unitary matrices,* $\mathbf{x}_k$ *and* $\mathbf{y}_k$ $(k \in [1,K])$ *respectively denote k-th left and right eigenvectors correspond to* λ_k, *and* $\mathbf{Q} \in \mathbb{C}^{K\times K}$ *denotes an arbitrary unitary matrix.*

Proof. Eq. 33 can be rewritten as:

$$\max_{\mathbf{X}^T\mathbf{X}=\mathbf{Y}^T\mathbf{Y}=\mathbf{I}_K} \frac{1}{2}\mathrm{tr}\left(\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix}^T \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{R} \\ \mathbf{R}^T & \mathbf{0} \end{bmatrix}}_{\mathbf{\Psi}} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} \right). \tag{36}$$

Since $\mathbf{\Psi}$ is a Hermitian matrix, by the Ky Fan theorem, the global optimum solution is given by the first K eigenvectors of $\mathbf{\Psi}$:

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1, \dots, \mathbf{x}_K \\ \mathbf{y}_1, \dots, \mathbf{y}_K \end{bmatrix} \mathbf{Q}. \tag{37}$$

By following the proof of theorem 2, it can be shown that $\mathbf{x}_1, \dots, \mathbf{x}_K$ and $\mathbf{y}_1, \dots, \mathbf{y}_K$ are the first K left and right eigenvectors of $\mathbf{R}$. □

Interestingly, theorem 5 can also be proven by using the SVD definition.

Proof. Without loosing generality, let assume $N \leq M$

$$\begin{aligned} \mathbf{R} =& \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \\ =& \mathbf{U}_{1,\dots,K}\mathbf{\Sigma}_{1,\dots,K}\mathbf{V}_{1,\dots,K}^T + \\ & \mathbf{U}_{K+1,\dots,N}\mathbf{\Sigma}_{K+1,\dots,N}\mathbf{V}_{K+1,\dots,N}^T, \end{aligned} \tag{38}$$

where $\mathbf{U}$ and $\mathbf{V}$ defined as in section 3, $\mathbf{U}_{a,\dots,b}$ and $\mathbf{V}_{a,\dots,b}$ denote matrices built by taking column a to b from $\mathbf{U}$ and $\mathbf{V}$ respectively, and $\mathbf{\Sigma}_{a,\dots,b} = \mathrm{diag}\,[\lambda_a, \dots, \lambda_b]$. Then,

$$\mathbf{U}_{1,\dots,K}^T\mathbf{R}\mathbf{V}_{1,\dots,K} = \mathbf{\Sigma}_{1,\dots,K}, \tag{39}$$

or more conveniently,

$$\mathbf{U}_K^T\mathbf{R}\mathbf{V}_K = \mathbf{\Sigma}_K. \tag{40}$$

Therefore,

$$\mathrm{tr}\left(\mathbf{U}_K^T\mathbf{R}\mathbf{V}_K\right) = \sum_{k=1}^{K}\lambda_k = \max_{\mathbf{X}^T\mathbf{X}=\mathbf{Y}^T\mathbf{Y}=\mathbf{I}_K} \mathrm{tr}(\mathbf{X}^T\mathbf{R}\mathbf{Y}). \tag{41}$$

□

Theorem 5 is the general form of theorem 2 and gives a theoretical support for directly applying the graph cuts on the data matrix $\mathbf{A} \in \mathbb{R}_+^{M\times N}$ to get simultaneous row and column clustering:

$$\max_{\bar{\mathbf{X}}^T\bar{\mathbf{X}}=\bar{\mathbf{Y}}^T\bar{\mathbf{Y}}=\mathbf{I}_K} \mathrm{tr}(\bar{\mathbf{X}}^T\mathbf{A}\bar{\mathbf{Y}}), \tag{42}$$

where $\bar{\mathbf{X}} \in \mathbb{R}_+^{M\times K}$ and $\bar{\mathbf{Y}} \in \mathbb{R}_+^{N\times K}$ denote the row and column clustering indicator matrices respectively.

6 Related works

Zha et al. [1] and Ding et al. [2] mention the Ky Fan theorem in their discussions on the spectral clustering. However, the role of the theorem in the spectral clustering can be easily overlooked as it is not clearly described.

The equivalences between K-means clustering and several graph cuts objectives to the trace maximization objectives are well-known facts in the spectral clustering researches as many papers discuss about it with exception for the directed graph case, as this problem arises from complex network researches. Some representative works are [1, 2, 5, 14, 18].

Leicht et al. [7] discuss how to extend the so-called modularity—which is equivalent to the graph cuts objective—of unipartite graph to directed graph. They form an asymmetric modularity matrix $\mathbf{B}^* \in \mathbb{R}_+^{N\times N}$ by applying modularity function to emphasizes the importance of the edge directions to the original asymmetric affinity matrix $\mathbf{B} \in \mathbb{R}_+^{N\times N}$, and then transform $\mathbf{B}^*$ into a symmetric matrix by adding $\mathbf{B}^*$ to its transpose. The clustering is done by calculating the first K eigenvectors of this symmetric matrix. This is equivalent to applying *RAssoc* to $(\mathbf{B}^* + \mathbf{B}^{*T})$.

Kim et al. [17] propose a method for transforming the affinity matrix induced from a directed graph into a symmetric matrix without ignoring the edge directions. So, clustering algorithms built for unipartite graph can be applied unchanged.

7 A note on spectral clustering algorithms

There are many spectral clustering algorithms available. They are different in many aspects, from the chosen affinity matrices to the postprocessing methods to derive clustering from eigenvectors. According to Luxburg [6], the most popular ones are algorithms by Shi et al. [3] and by Ng et al. [4], with the former is more favorable because the computed eigenvectors are more related to the clustering indicator vectors.

Here we like to note that according to Dhillon et al. [14], a state-of-the-art spectral clustering algorithm based on the work of Yu et al. [5] empirically performed the best among various spectral algorithms that were tested in the terms of optimizing the objective function values. Furthermore, the multilevel algorithm proposed in [14]—which exploits the equivalences of various graph clustering objectives to weighted kernel K-means objective to eliminate the need for eigenvectors computation—shows very promising results which while moderately improving clustering quality, drastically improving computational speed (up to 2000 times faster than the spectral method) and memory usage.

8 Conclusion

We presented a concise explanation on the logic behind the spectral clustering. Unlike K-means clustering and graph cuts which are very intuitive and straightforward, the spectral clustering tends to be incomprehensible. By using the Ky Fan theorem, we showed that the spectral clustering has a simple explanation and is also intuitive.

We showed how to treat K-way clustering on unipartite, bipartite and directed graphs as the trace maximization problems on the corresponding symmetric matrices, thus a unified treatment can be applied to those graphs.

In bipartite graph, we proved that the co-clustering can be obtained by computing the left and right eigenvectors of the corresponding feature-by-item data matrix, thus generalizing the result of Dhillon [15] and providing a theoretical basis for spectral co-clustering algorithms proposed in, e.g., [15, 19]. We also proved that solving simultaneous row and column clustering is equivalent to solving row and column clustering separately, thus giving a theoretical support for the claim: "column clustering implies row clustering and vice versa", and then gave a "shortcut" to compute the row and column clustering indicator matrices.

In directed graph, we described a new clustering objective by following the discussions on unipartite and bipartite graphs naturally.

By extending theorem 2 to complex domain, we generalized the Ky Fan theorem to rectangular complex matrix. The second proof of theorem 5 shows that this theorem is a corollary of the SVD formulation, and thus the Ky Fan theorem and its general form are the corollaries of the Eckart-Young theorem.

We must note that, however, as the mathematics behind the spectral clustering has a long story (the Ky Fan theorem itself was proposed in 50's), it is probable that the contributions in this paper are not new, or can be derived easily from other well-established facts, theorems, or definitions.

References

[1] H. Zha, X. He, C. Ding, H. Simon, and M. Gu, "Spectral relaxation for K-means clustering," Proc. 14th Advances in Neural Information Processing Systems, pp. 1057-64, 2001.

[2] C. Ding and X. He, "K-means clustering via principal component analysis," Proc. 21st International Conference on Machine Learning, pp. 29-37, 2004.

[3] J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 888-905, 2000.

[4] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," Proc. 14th Advances in Neural Information Processing Systems, pp. 849-56, 2001.

[5] S. X. Yu and J. Shi, "Multiclass spectral clustering," Proc. 9th IEEE International Conference on Computer Vision, pp. 313-9, 2003.

[6] U. von Luxburg, "A tutorial on spectral clustering," Statistics and Computing, pp. 395-416, 2007.

[7] E. A. Leicht and M. E. J. Newman, "Community structure in directed networks," Phys. Rev. Lett., Vol. 10, No. 11, pp. 118703-6, 2008.

[8] I. Nakić and K. Veselić, "Wielandt and Ky-Fan theorem for matrix pairs," Linear Algebra and its Applications, Vol. 369, No. 17, pp. 77-93, 2003.

[9] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," Psychometrika, Vol. 1, pp. 211-8, 1936.

[10] F. Bach and M. I. Jordan, "Learning spectral clustering," Proc. 16th Advances in Neural Information Processing Systems, 2003.

[11] R. Jin, S. Wang, and Y. Zhou, "Regularized distance metric learning: Theory and algorithm," Proc. 22nd Advances in Neural Information Processing Systems, 2009.

[12] L. Wu, R. Jin, S. C. H. Hoi, J. Zhu, and N. Yu, "Learning Bregman distance functions and its application for semi-supervised clustering," Proc. 22nd Advances in Neural Information Processing Systems, 2009.

[13] P. Jain, B. Kulis, I. S. Dhillon, K. Grauman, "Online metric learning and fast similarity search," Proc. 21st Advances in Neural Information Processing Systems, 2008.

[14] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors: A multilevel approach," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, No. 11, pp. 1944-57, 2007.

[15] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," Proc. 7th ACM SIGKDD Int'l Conference on

Knowledge Discovery and Data Mining, pp. 269-74, 2001.

[16] Y. Kim, S. W. Son, and H. Jeong, "Finding communities in directed networks," Phys. Rev. E, Vol. 81, No. 1, pp. 016103-11, 2010.

[17] Y. Kim, S. W. Son, and H. Jeong, "Communities identification in directed networks," Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Springer Berlin Heidelberg, Vol. 5, pp. 2050-3, 2009.

[18] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel K-means, spectral clustering and normalized cuts," Proc. 10th ACM Knowledge Discovery and Data Mining Conference, pp. 551-6, 2004.

[19] Y. Klugar, R. Basri, J. T. Chang, and M. Gerstein. "Spectral biclustering of microarray data: coclustering genes and conditions," Genome Research, Vol. 13, pp. 703-16, 2003.