6

Spectral Analysis II: Clustering

In the previous lecture we ended up with the formulation:

$$\max_{G_{m\times k}} trace(G^\top KG) \quad \text{s.t. } G^\top G = I \tag{6.1}$$

and showed the solution G is the leading eigenvectors of the symmetric positive semi definite matrix K. When $K = AA^\top$ (sample covariance matrix) with $A = [\mathbf{x}_1, ..., \mathbf{x}_m]$, $\mathbf{x}_i \in R^n$, those eigenvectors form a basis to a k-dimensional subspace of R^n which is the closest (in L_2 norm sense) to the sample points $\mathbf{x}_i$. The axes (called principal axes) $\mathbf{g}_1, ..., \mathbf{g}_k$ preserve the variance of the original data in the sense that the projection of the data points on the $\mathbf{g}_1$ has maximum variance, projection on $\mathbf{g}_2$ has the maximum variance over all vectors orthogonal to $\mathbf{g}_1$, etc. The spectral decomposition of the sample covariance matrix is a way to "compress" the data by means of linear super-position of the original coordinates $\mathbf{y} = G^\top \mathbf{x}$.

We also ended with a ratio formulation:

$$\max_{\mathbf{w}} \frac{\mathbf{w}^\top S_1 \mathbf{w}}{\mathbf{w}^\top S_2 \mathbf{w}}$$

where S_1, S_2 where scatter matrices defined such that $\mathbf{w}^\top S_1 \mathbf{w}$ is the variance of class centers (which we wish to maximize) and $\mathbf{w}^\top S_2 \mathbf{w}$ is the sum of within class variance (which we want to minimize). The solution $\mathbf{w}$ is the generalized eigenvector $S_1\mathbf{w} = \lambda S_2 \mathbf{w}$ with maximal λ.

In this lecture we will show additional applications where the search for leading eigenvectors plays a pivotal part of the solution. So far we have seen how spectral analysis relates to PCA and LDA and today we will focus on the classic Data Clustering problem of partitioning a set of points $\mathbf{x}_1, ..., \mathbf{x}_m$ into $k \geq 2$ classes, i.e., generating as output indicator variables $y_1, ..., y_m$ where $y_i \in \{1, ..., k\}$. We will begin with "K-means" algorithm for clustering and then move on to show how the optimization criteria relates

to grapth-theoretic approaches (like Min-Cut, Ratio-Cut, Normalized Cuts) and spectral decomposition.

6.1 K-means Algorithm for Clustering

The K-means formulation (originally introduced by [4]) assumes that the clusters are defined by the distance of the points to their class centers only. In other words, the goal of clustering is to find those k mean vectors $\mathbf{c}_1, ..., \mathbf{c}_k$ and provide the cluster assignment $y_i \in \{1, ..., k\}$ of each point $\mathbf{x}_i$ in the set. The K-means algorithm is based on an interleaving approach where the cluster assignments y_i are established given the centers and the centers are computed given the assignments. The optimization criterion is as follows:

$$\min_{y_1,...,y_m,\mathbf{c}_1,...,\mathbf{c}_k} \sum_{j=1}^{k} \sum_{y_i=j} \|\mathbf{x}_i - \mathbf{c}_j\|^2 \tag{6.2}$$

Assume that $\mathbf{c}_1, ..., \mathbf{c}_k$ are given from the previous iteration, then

$$y_i = \operatorname*{argmin}_j \|\mathbf{x}_i - \mathbf{c}_j\|^2,$$

and next assume that $y_1, .., y_m$ (cluster assignments) are given, then for any set $S \subseteq \{1, ..., m\}$ we have that

$$\frac{1}{|S|} \sum_{j \in S} \mathbf{x}_j = \operatorname*{argmin}_{\mathbf{c}} \sum_{j \in S} \|\mathbf{x}_j - \mathbf{c}\|^2.$$

In other words, given the estimated centers in the current round, the new assignments are computed by the closest center to each point $\mathbf{x}_i$, and then given the updated assignments the new centers are estimated by taking the mean of each cluster. Since each step is guaranteed to reduce the optimization energy the process must converge — to some local optimum.

The drawback of the K-means algorithm is that the quality of the local optimum strongly depends on the initial guess (either the centers or the assignments). If we start with a wild guess for the centers it would be fairly unlikely that the process would converge to a good local minimum (i.e. one that is close to the global optimum). An alternative approach would be to define an approximate but simpler problem which has a closed form solution (such as obtained by computing eigenvectors of some matrix). The global optimum of the K-means is an NP-Complete problem (mentioned briefly in the next section).

Next, we will rewrite the K-means optimization criterion in matrix form and see that it relates to the spectral formulation (eqn. 6.1).

6.1.1 Matrix Formulation of K-means

We rewrite eqn. 6.2 as follows [7]. Instead of carrying the class variables y_i we define class sets $\psi_1, ..., \psi_k$ where $\psi_i \subset \{1, ..., n\}$ with $\bigcup \psi_j = \{1, ..., n\}$ and $\psi_i \bigcap \psi_j = \emptyset$. The K-means optimization criterion seeks for the centers and the class sets:

$$\min_{\psi_1,...,\psi_k,\mathbf{c}_1,...,\mathbf{c}_k} \sum_{j=1}^{k} \sum_{i \in \psi_j} \|\mathbf{x}_i - \mathbf{c}_j\|^2.$$

Let $l_j = |\psi_j|$ and following the expansion of the squared norm and dropping $\mathbf{x}_i^\top \mathbf{x}_i$ we end up with an equivalent problem:

$$\min_{\psi_1,...,\psi_k,\mathbf{c}_1,...,\mathbf{c}_k} \sum_{j=1}^{k} l_j \mathbf{c}_j^\top \mathbf{c}_j - 2 \sum_{j=1}^{k} \sum_{i \in \psi_j} \mathbf{x}_i^\top \mathbf{c}_j.$$

Next we substitute $\mathbf{c}_j$ with its definition: $(1/l_j) \sum_{i \in \psi_j} \mathbf{x}_j$ and obtain a new equivalent formulation where the centers $\mathbf{c}_j$ are eliminated form consideration:

$$\min_{\psi_1,...,\psi_k} -\sum_{j=1}^{k} \frac{1}{l_j} \sum_{r,s \in \psi_j} \mathbf{x}_r^\top \mathbf{x}_s$$

which is more conveniently written as a maximization problem:

$$\max_{\psi_1,...,\psi_k} \sum_{j=1}^{k} \frac{1}{l_j} \sum_{r,s \in \psi_j} \mathbf{x}_r^\top \mathbf{x}_s. \tag{6.3}$$

Since the resulting formulation involves only inner-products we could have replaced $\mathbf{x}_i$ with $\phi(\mathbf{x}_i)$ in eqn. 6.2 where the mapping $\phi(\cdot)$ is chosen such that $\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$ can be replaced by some non-linear function $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ — known as the "kernel trick" (discussed in previous lectures). Having the ability to map the input vectors onto some high-dimensional space before K-means is applied provides more flexibility and increases our chances of getting out a "good" clustering from the global K-means solution (again, the local optimum depends on the initial conditions so it could be "bad"). The RBF kernel is quite popular in this context $\kappa(\mathbf{x}_i, \mathbf{x}_j) = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma^2}$ with σ some pre-determined parameter. Note that $\kappa(\mathbf{x}_i, \mathbf{x}_j) \in (0, 1]$ which can be interpreted loosely as the probability of $\mathbf{x}_i$ and $\mathbf{x}_j$ to be clustered together.

Let $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ making K a $m \times m$ symmetric positive-semi-definite matrix often referred to as the "affinity" matrix. Let F be an $n \times n$ matrix whose entries are $F_{ij} = 1/l_r$ if $(i, j) \in \psi_r$ for some class ψ_r and $F_{ij} = 0$

otherwise. In other words, if we sort the points $\mathbf{x}_i$ according to cluster membership, then F is a block diagonal matrix with blocks $F_1, ..., F_k$ where $F_r = (1/l_r)\mathbf{1}\mathbf{1}^\top$ is an $l_r \times l_r$ block of 1's scaled by $1/l_r$. Then, Eqn. 6.3 can be written in terms of K as follows:

$$\max_F \sum_{i,j=1}^{n} K_{ij}F_{ij} = trace(KF) \tag{6.4}$$

In order to form this as an optimization problem we need to represent the structure of F in terms of constraints. Let G be an $n \times k$ column-scaled indicator matrix: $G_{ij} = (1/\sqrt{l_j})$ if $i \in \psi_j$ (i.e., $\mathbf{x}_i$ belongs to the j'th class) and $G_{ij} = 0$ otherwise. Let $\mathbf{g}_1, ..., \mathbf{g}_k$ be the columns of G and it can be easily verified that $\mathbf{g}_r\mathbf{g}_r{}^\top = diag(0, .., F_r, 0, .., 0)$ therefore $F = \sum_j \mathbf{g}_j\mathbf{g}_j^\top = GG^\top$. Since $trace(AB) = trace(BA)$ we can now write eqn. 6.4 in terms of G:

$$\max_G trace(G^\top KG)$$

under conditions on G which we need to further spell out.

We will start with the necessary conditions. Clearly $G \geq 0$ (has non-negative entries). Because each point belongs to exactly one cluster we must have $G^\top G_{ij} = 0$ when $i \neq j$ and $G^\top G_{ii} = (1/l_i)\mathbf{1}^\top\mathbf{1} = 1$, thus $G^\top G = I$. Furthermore we have that the rows and columns of $F = GG^\top$ sum up to 1, i.e., $F\mathbf{1} = \mathbf{1}, F^\top\mathbf{1} = \mathbf{1}$ which means that F is *doubly stochastic* which translates to the constraint $GG^\top\mathbf{1} = \mathbf{1}$ on G. We have therefore three necessary conditions on G: (i) $G \geq 0$, (ii) $G^\top G = I$, and (iii) $GG^\top\mathbf{1} = \mathbf{1}$. The claim below asserts that these are also sufficient conditions:

Claim 4 *The feasibility set of matrices G which satisfy the three conditions $G \geq 0$, $GG^\top\mathbf{1} = \mathbf{1}$ and $G^\top G = I$ are of the form:*

$$G_{ij} = \left\{ \begin{array}{cc} \frac{1}{\sqrt{l_j}} & \mathbf{x}_i \in \psi_j \\ 0 & otherwise \end{array} \right\}$$

Proof: From $G \geq 0$ and $\mathbf{g}_r^\top\mathbf{g}_s = 0$ we have that $G_{ir}G_{is} = 0$, i.e., G has a single non-vanishing element in each row. It will be convenient to assume that the points are sorted according to the class membership, thus the columns of G have the non-vanishing entries in consecutive order and let l_j be the number of non-vanishing entries in column $\mathbf{g}_j$. Let $\mathbf{u}_j$ the vector of l_j entries holding only the non-vanishing entries of $\mathbf{g}_j$. Then, the doubly stochastic constraint $GG^\top\mathbf{1} = \mathbf{1}$ results that $(\mathbf{1}^\top\mathbf{u}_j)\mathbf{u}_j = \mathbf{1}$ for $j = 1, ..., k$. Multiplying $\mathbf{1}$ from both sides yields $(\mathbf{1}^\top\mathbf{u}_j)^2 = \mathbf{1}^\top\mathbf{1} = l_j$, therefore $\mathbf{u}_j = (1/\sqrt{l_j})\mathbf{1}$. □

This completes the *equivalence* between the matrix formulation:

$$\max_{G \in R^{m \times k}} trace(G^\top K G) \quad \text{s.t. } G \geq 0, G^\top G = I, G G^\top \mathbf{1} = \mathbf{1} \tag{6.5}$$

and the original K-means formulation of eqn. 6.2.

We have obtained the same optimization criteria as eqn. 6.1 with additional two constraints: G should be non-negative and $GG^\top$ should be doubly stochastic. The constraint $G^\top G = I$ comes from the requirement that each point is assigned to one class only. The doubly stochastic constraint comes from a "class balancing" requirement which we will expand on below.

6.2 Min-Cut

We will arrive to eqn. 6.5 from a graph-theoretic perspective. We start with representing the graph Min-Cut problem in matrix form, as follows. A convenient way to represent the data to be clustered is by an undirected graph with edge-weights where $V = \{1, ..., m\}$ is the vertex set, $E \subset V \times V$ is the edge set and $\kappa : E \rightarrow R_+$ is the positive weight function. Vertices of the graph correspond to data points $\mathbf{x}_i$, edges represent neighborhood relationships, and edge-weights represent the similarity (affinity) between pairs of linked vertices. The weight adjacency matrix K holds the weights where $K_{ij} = \kappa(i, j)$ for $(i, j) \in E$ and $K_{ij} = 0$ otherwise.

A *cut* in the graph is defined between two disjoint sets $A, B \subset V$, $A \cup B = V$, is the sum of edge-weights connecting the two sets: $cut(A, B) = \sum_{i \in A, j \in B} K_{ij}$ which is a measure of dissimilarity between the two sets. The Min-Cut problem is to find a minimal weight cut in the graph (can be solved in polynomial time through Max Network Flow solution). The following claim associates algebraic conditions on G with an indicator matrix:

Claim 5 *The feasibility set of matrices G which satisfy the three conditions $G \geq 0$, $G\mathbf{1} = \mathbf{1}$ and $G^\top G = D$ for some diagonal matrix D are of the form:*

$$G_{ij} = \left\{ \begin{array}{cc} 1 & x_i \in \psi_j \\ 0 & otherwise \end{array} \right\}$$

Proof: Let $G = [\mathbf{g}_1, ..., \mathbf{g}_k]$. From $G \geq 0$ and $\mathbf{g}_r^\top \mathbf{g}_s = 0$ we have that $G_{ir} G_{is} = 0$, i.e., G has a single non-vanishing element in each row. From $G\mathbf{1} = \mathbf{1}$ the single non-vanishing entry of each row must have the value of 1. □

In the case of two classes ($k = 2$), the function $tr(G^\top K G)$ is equal to $\sum_{(i,j) \in \psi_1} K_{ij} + \sum_{(i,j) \in \psi_2} K_{ij}$. Therefore $\max_G tr(G^\top K G)$ is equivalent to

minimizing the cut: $\sum_{i\in\psi_1, j\in\psi_2} K_{ij}$. As a result, the Min-Cut problem is equivalent to solving the optimization problem:

$$\max_{G\in R^{m\times 2}} tr(G^\top KG)\ s.t \quad G \geq 0,\ G\mathbf{1} = \mathbf{1},\ G^\top G = diag \tag{6.6}$$

We seem to be close to eqn. 6.5 with the difference that G is orthogonal (instead of orthonormal) and the doubly-stochasitc constraint is replaced by $G\mathbf{1} = \mathbf{1}$. The difference can be bridged by considering a "balancing" requirement. Min-Cut can produce an unbalanced partition where one set of vertices is very large and the other contains a spurious set of vertices having a small number of edges to the larger set. This is an undesirable outcome in the context of clustering. Consider a "balancing" constraint $G^\top\mathbf{1} = (m/k)\mathbf{1}$ which makes a strict requirement that all the k clusters have an equal number of points. We can relax the balancing constraint slightly by combining the balancing constraint with $G\mathbf{1} = \mathbf{1}$ into one single constraint $GG^\top\mathbf{1} = (m/k)\mathbf{1}$, i.e., $GG^\top$ is scaled doubly stochastic. Note that the two conditions $GG^\top\mathbf{1} = (m/k)\mathbf{1}$ and $G^\top G = D$ result in $D = (m/k)I$. Thus we propose the relaxed-balanced hard clustering scheme:

$$\max_G tr(G^\top KG)\ s.t \quad G \geq 0,\ GG^\top\mathbf{1} = \frac{m}{k}\mathbf{1},\ G^\top G = \frac{m}{k}I$$

The scale m/k is a global scale that can be dropped without affecting the resulting solution, thus the Min-Cut with a relaxed balancing requirement becomes eqn. 6.5 which we saw is equivalent to K-means:

$$\max_G tr(G^\top KG)\ s.t \quad G \geq 0,\ GG^\top\mathbf{1} = \mathbf{1},\ G^\top G = I.$$

6.3 Spectral Clustering: Ratio-Cuts and Normalized-Cuts

We saw above that the doubly-stochastic constraint has to do with a "balancing" desire. A further relaxation of the balancing desire is to perform the optimization in two steps: (i) replace the affinity matrix K with the closest (under some chosen error measure) doubly-stochastic matrix K', (ii) find a solution to the problem:

$$\max_{G\in R^{m\times k}} tr(G^\top K'G)\ s.t \quad G \geq 0,\ G^\top G = I \tag{6.7}$$

because $GG^\top$ should come out close to K' ($tr(G^\top K'G) = tr(K'GG^\top)$) and K' is doubly-stochastic, then $GG^\top$ should come out close to satisfying a doubly-stochastic constraint — this is the motivation behind the 2-step approach. Moreover, we drop the non-negativity constraint $G \geq 0$. Note that the non-negativity constraint is crucial for the physical interpretation of

G; nevertheless, for $k = 2$ clusters it is possible to make an interpretation, as we shall next. As a result we are left with a spectral decomposition problem of eqn. 6.1:

$$\max_{G \in R^{m \times k}} tr(G^\top K' G) \; s.t \quad G^\top G = I,$$

where the columns of G are the leading eigenvectors of K'. We will refer to the first step as a "normalization" process and there are two popular normalizations in the literature — one leading to Ratio-Cuts and the other to Normalized-Cuts.

6.3.1 Ratio-Cuts

Let $D = diag(K\mathbf{1})$ which is a diagonal matrix containing the row sums of K. The Ratio-Cuts normalization is to look for K' as the closest doubly-stochastic matrix to K by minimizing the L_1 norm — this turns out to be $K' = K - D + I$.

Claim 6 (ratio-cut) *Let K be a symmetric positive-semi-definite whose values are in the range $[0, 1]$. The closest doubly stochastic matrix K' under the L_1 error norm is*

$$K' = K - D + I$$

Proof: Let $r = \min_F \|K - F\|_1 \;\; s.t. \;\; F\mathbf{1} = \mathbf{1}, \;\; F = F^\top$. Since $\|K - F\|_1 \geq \|(K - F)\mathbf{1}\|_1$ for any matrix F, we must have:

$$r \geq \|(K - F)\mathbf{1}\|_1 = \|D\mathbf{1} - \mathbf{1}\|_1 = \|D - I\|_1.$$

Let $F = K - D + I$, then

$$\|K - (K - D + I)\|_1 = \|D - I\|_1.$$

□

The Laplacian matrix of a graph is $D - K$. If $\mathbf{v}$ is an eigenvector of the Laplacian $D - K$ with eigenvalue λ, then $\mathbf{v}$ is also an eigenvector of $K' = K - D + I$ with eigenvalue $1 - \lambda$ and since $(D - K)\mathbf{1} = 0$ then the smallest eigenvector $\mathbf{v} = \mathbf{1}$ of the Laplacian is the largest of K', and the second smallest eigenvector of the Laplacian (the ratio-cut result) corresponds to the second largest eigenvector of K'. Because the eigenvectors are orthogonal, the second eigenvector must have positive and negative entries (because the inner-product with $\mathbf{1}$ is zero) — thus the sign of the entries of the second eigenvector determines the class membership.

Ratio-Cuts, the second smallest eigenvector of the Laplacian $D - K$, is an approximation due to Hall in the 70s [2] to the Min-Cut formulation. Let $\mathbf{z} \in R^m$ determine the class membership such that $\mathbf{x}_i$ and $\mathbf{x}_j$ would be clustered together if z_i and z_j have similar values. This leads to the following optimization problem:

$$\min_{\mathbf{z}} \frac{1}{2} \sum_{i,j} (z_i - z_j)^2 K_{ij} \quad s.t. \quad \mathbf{z}^\top \mathbf{z} = 1$$

The criterion function is equal to $(1/2)\mathbf{z}^\top (D-K)\mathbf{z}$ and the derivative of the Lagrangian $(1/2)\mathbf{z}^\top (D-K)\mathbf{z} - \lambda(\mathbf{z}^\top \mathbf{z} - 1)$ with respect to $\mathbf{z}$ gives rise to the necessary condition $(D-K)\mathbf{z} = \lambda \mathbf{z}$ and the Ratio-Cut scheme follows.

6.3.2 Normalized-Cuts

Normalized-Cuts looks for the closest doubly-stochastic matrix K' in *relative entropy* error measure defined as:

$$RE(\mathbf{x} \mid\mid \mathbf{y}) = \sum_i x_i \ln \frac{x_i}{y_i} + \sum_i y_i - \sum_i x_i.$$

We will encounter the relative entropy measure in more detail later in the course. We can show that K' must have the form $\Lambda K \Lambda$ for some diagonal matrix Λ:

Claim 7 *The closest doubly-stochastic matrix F under the relative-entropy error measure to a given non-negative symmetric matrix K, i.e., which minimizes:*

$$\min_F \; RE(F||K) \;\; s.t. \;\; F \geq 0, \; F = F^\top, \; F\mathbf{1} = 1, \; F^\top \mathbf{1} = 1$$

has the form $F = \Lambda K \Lambda$ for some (unique) diagonal matrix Λ.

Proof: The Lagrangian of the problem is:

$$L() = \sum_{ij} f_{ij} \ln \frac{f_{ij}}{k_{ij}} + \sum_{ij} k_{ij} - \sum_{ij} f_{ij} - \sum_i \lambda_i (\sum_j f_{ij} - 1) - \sum_j \mu_j (\sum_i f_{ij} - 1)$$

The derivative with respect to f_{ij} is:

$$\frac{\partial L}{\partial f_{ij}} = \ln f_{ij} + 1 - \ln k_{ij} - 1 - \lambda_i - \mu_j = 0$$

from which we obtain:

$$f_{ij} = e^{\lambda_i} e^{\mu_j} k_{ij}$$

Let $D_1 = diag(e^{\lambda_1}, ..., e^{\lambda_n})$ and $D_2 = diag(e^{\mu_1}, ..., e^{\mu_n})$, then we have:

$$F = D_1 K D_2$$

Since $F = F^\top$ and K is symmetric we must have $D_1 = D_2$. □

Next, we can show that the diagonal matrix Λ can found by an iterative process where K is replaced by $D^{-1/2}KD^{-1/2}$ where D was defined above as $diag(K\mathbf{1})$:

Claim 8 *For any non-negative symmetric matrix $K^{(0)}$, iterating the process $K^{(t+1)} \leftarrow D^{-1/2}K^{(t)}D^{-1/2}$ with $D = diag(K^{(t)}\mathbf{1})$ converges to a doubly stochastic matrix.*

The proof is based on showing that the permanent increases monotonically, i.e. $perm(K^{(t+1)}) \geq perm(K^{(t)})$. Because the permanent is bounded the process must converge and if the permanent does not change (at the convergence point) the resulting matrix must be doubly stochastic. The resulting doubly stochastic matrix is the closest to K in relative-entropy.

Normalized-Cuts takes the result of the first iteration by replacing K with $K' = D^{-1/2}KD^{-1/2}$ followed by the spectral decomposition (in case of $k = 2$ classes the partitioning information is found in the second leading eigenvector of K' — just like Ratio-Cuts but with a different K'). Thus, K' in this manner is not the closest doubly-stochastic matrix to K but is fairly close (the first iteration is the dominant one in the process).

Normalized-Cuts, as the second leading eigenvector of $K' = D^{-1/2}KD^{-1/2}$, is an approximation to a "balanced" Min-Cut described first in [6]. Deriving it from first principles proceeds as follows:

Let $sum(V_1, V_2) = sum_{i \in V_1, j \in V_2} K_{ij}$ be defined for any two subsets (not necessarily disjoint) of vertices. The normalized-cuts measures the cut cost as a fraction of the total edge connections to all the nodes in the graph:

$$Ncuts(A, B) = \frac{cut(A, B)}{sum(A, V)} + \frac{cut(A, B)}{sum(B, V)}.$$

A minimal Ncut partition will no longer favor small isolated points since the cut value would most likely be a large percentage of the total connections from that small set to all the other vertices. A related measure $Nassoc(A, B)$ defined as:

$$Nassoc(A, B) = \frac{sum(A, A)}{sum(A, V)} + \frac{sum(B, B)}{sum(B, V)},$$

reflects how tightly on average nodes within the group are connected to each other. Given that $cut(A, B) = sum(A, V) - sum(A, A)$ one can easily verify

that:

$$Ncuts(A,B) = 2 - Nassoc(A,B),$$

therefore the optimal bi-partition can be represented as maximizing $Nassoc(A, V-A)$. The $Nassoc$ naturally extends to $k > 2$ classes (partitions) as follows: Let $\psi_1, ..., \psi_k$ be disjoint sets $\cup_j \psi_j = V$, then:

$$Nassoc(\psi_1, ..., \psi_k) = \sum_{j=1}^{k} \frac{sum(\psi_j, \psi_j)}{sum(\psi_j, V)}.$$

We will now rewrite $Nassoc$ in matrix form and establish equivalence to eqn. 6.7. Let $\bar{G} = [\mathbf{g}_1, ..., \mathbf{g}_k]$ with $\mathbf{g}_j = 1/\sqrt{sum(\psi_j, V)}(0, ..., 0, 1, ...1, 0., , , 0)$ with the 1s indicating membership to the j'th class. Note that

$$\mathbf{g}_j^\top K \mathbf{g}_j = \frac{sum(\psi_j, \psi_j)}{sum(\psi_j, V)},$$

therefore $trace(\bar{G}^\top K \bar{G}) = Nassoc(\psi_1, ..., \psi_k)$. Note also that $\mathbf{g}_i^\top D \mathbf{g}_i = (1/sum(\psi_i, V)) \sum_{r \in \psi_i} d_r = 1$, therefore $\bar{G}^\top D \bar{G} = I$. Let $G = D^{1/2}\bar{G}$ so we have that $G^\top G = I$ and $trace(G^\top D^{-1/2} K D^{-1/2} G) = Nassoc(\psi_1, ..., \psi_k)$. Taken together we have that maximizing $Nassoc$ is equivalent to:

$$\max_{G \in R^{m \times k}} trace(G^\top K' G) \quad \text{s.t. } G \geq 0, G^\top G = I, \tag{6.8}$$

where $K' = D^{-1/2} K D^{-1/2}$. Note that this is exactly the K-means matrix setup of eqn. 6.5 where the doubly-stochastic constraint is relaxed into the replacement of K by K'. The constraint $G \geq 0$ is then dropped and the resulting solution for G is the k leading eigenvectors of K'.

We have arrived via seemingly different paths to eqn. 6.8 which after we drop the constraint $G \geq 0$ we end up with a closed form solution consisting of the k leading eigenvectors of K'. When $k = 2$ (two classes) one can easily verify that the partitioning information is fully contained in the *second* eigenvector. Let $\mathbf{v}_1, \mathbf{v}_2$ be the first leading eigenvectors of K'. Clearly $\mathbf{v} = D^{1/2}\mathbf{1}$ is an eigenvector with eigenvalue $\lambda = 1$:

$$D^{-1/2} K D^{-1/2} (D^{1/2}\mathbf{1}) = D^{-1/2} K \mathbf{1} = D^{1/2}\mathbf{1}.$$

In fact $\lambda = 1$ is the largest eigenvalue (left as an exercise) thus $\mathbf{v}_1 = D^{1/2}\mathbf{1} > 0$. Since K' is symmetric the $\mathbf{v}_2^\top \mathbf{v}_1 = 0$ thus $\mathbf{v}_2$ contains positive and negative entries — those are interpreted as indicating class membership (positive to one class and negative to the other).

The case $k > 2$ is treated as an *embedding* (also known as Multi-Dimensional Scaling) by re-coordinating the points $\mathbf{x}_i$ using the rows of G. In other

words, the i'th row of G is a representation of $\mathbf{x}_i$ in R^k. Under *ideal* conditions where K is block diagonal (the distance between clusters is infinity) the rows associated with points clustered together are identical (i.e., the n original points are mapped to k points in R^k) [5]. In practice, one performs the iterative K-means in the embedded space.