Part 2. Spectral Clustering from Matrix Perspective

A brief tutorial emphasizing recent developments

(More detailed tutorial is given in ICML'04)

From PCA to spectral clustering using generalized eigenvectors

Consider the kernel matrix: $W_{ij} = \left\langle \phi(x_i), \phi(x_j) \right\rangle$

In Kernel PCA we compute eigenvector: $Wv = \lambda v$

Generalized Eigenvector: $Wq = \lambda Dq$

$D = diag(d_1, \cdots, d_n) \qquad d_i = \sum_j w_{ij}$

This leads to Spectral Clustering !

Indicator Matrix Quadratic Clustering Framework

Unsigned Cluster indicator Matrix $H=(h_1, \cdots, h_K)$

Kernel K-means clustering:

$$\max_H \operatorname{Tr}(H^T W H), \quad s.t.\ H^T H = I, H \ge 0$$

K-means: $W = X^T X$; Kernel K-means $W = (<\phi(x_i), \phi(x_j)>)$

Spectral clustering (normalized cut)

$$\max_H \operatorname{Tr}(H^T W H), \quad s.t.\ H^T D H = I, H \ge 0$$

Brief Introduction to Spectral Clustering
(Laplacian matrix based clustering)

Some historical notes

- Fiedler, 1973, 1975, graph Laplacian matrix
- Donath & Hoffman, 1973, bounds
- Hall, 1970, Quadratic Placement (embedding)
- Pothen, Simon, Liou, 1990, Spectral graph partitioning (many related papers there after)
- Hagen & Kahng, 1992, Ratio-cut
- Chan, Schlag & Zien, multi-way Ratio-cut
- Chung, 1997, Spectral graph theory book
- Shi & Malik, 2000, Normalized Cut

Spectral Gold-Rush of 2001

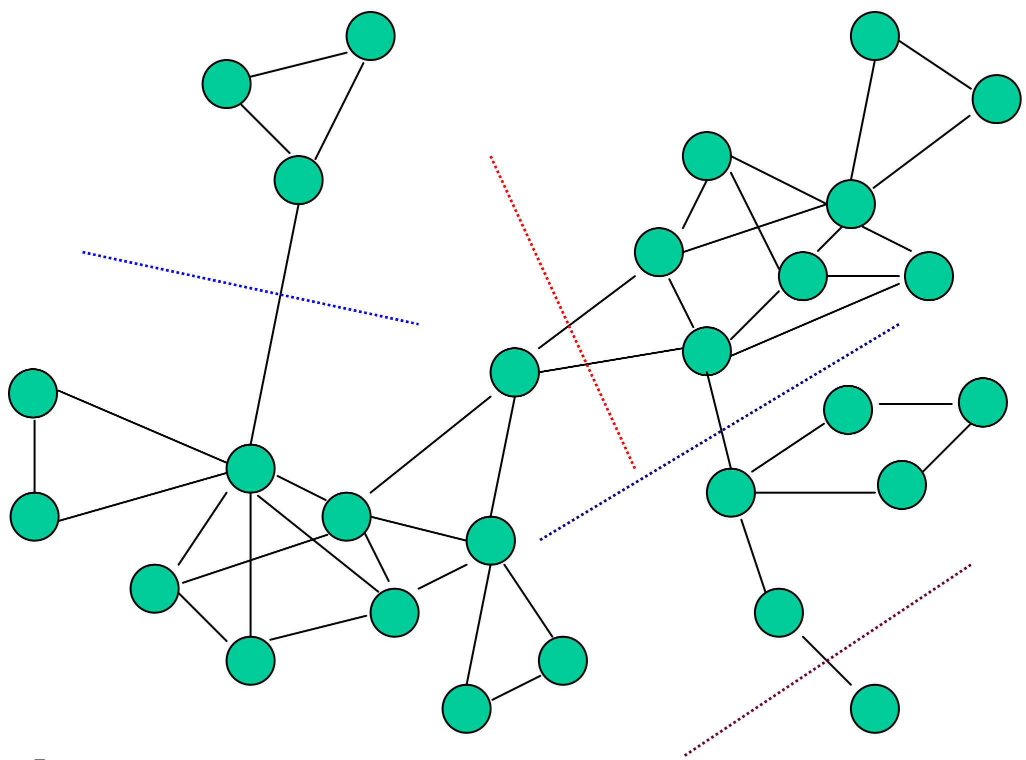
9 papers on spectral clustering

- Meila & Shi, AI-Stat 2001. Random Walk interpreation of Normalized Cut
- Ding, He & Zha, KDD 2001. Perturbation analysis of Laplacian matrix on sparsely connected graphs
- Ng, Jordan & Weiss, NIPS 2001, K-means algorithm on the embeded eigen-space
- Belkin & Niyogi, NIPS 2001. Spectral Embedding
- Dhillon, KDD 2001, Bipartite graph clustering
- Zha et al, CIKM 2001, Bipartite graph clustering
- Zha et al, NIPS 2001. Spectral Relaxation of K-means
- Ding et al, ICDM 2001. MinMaxCut, Uniqueness of relaxation.
- Gu et al, K-way Relaxation of NormCut and MinMaxCut

Spectral Clustering

min cutsize , without explicit size constraints

But where to cut ?

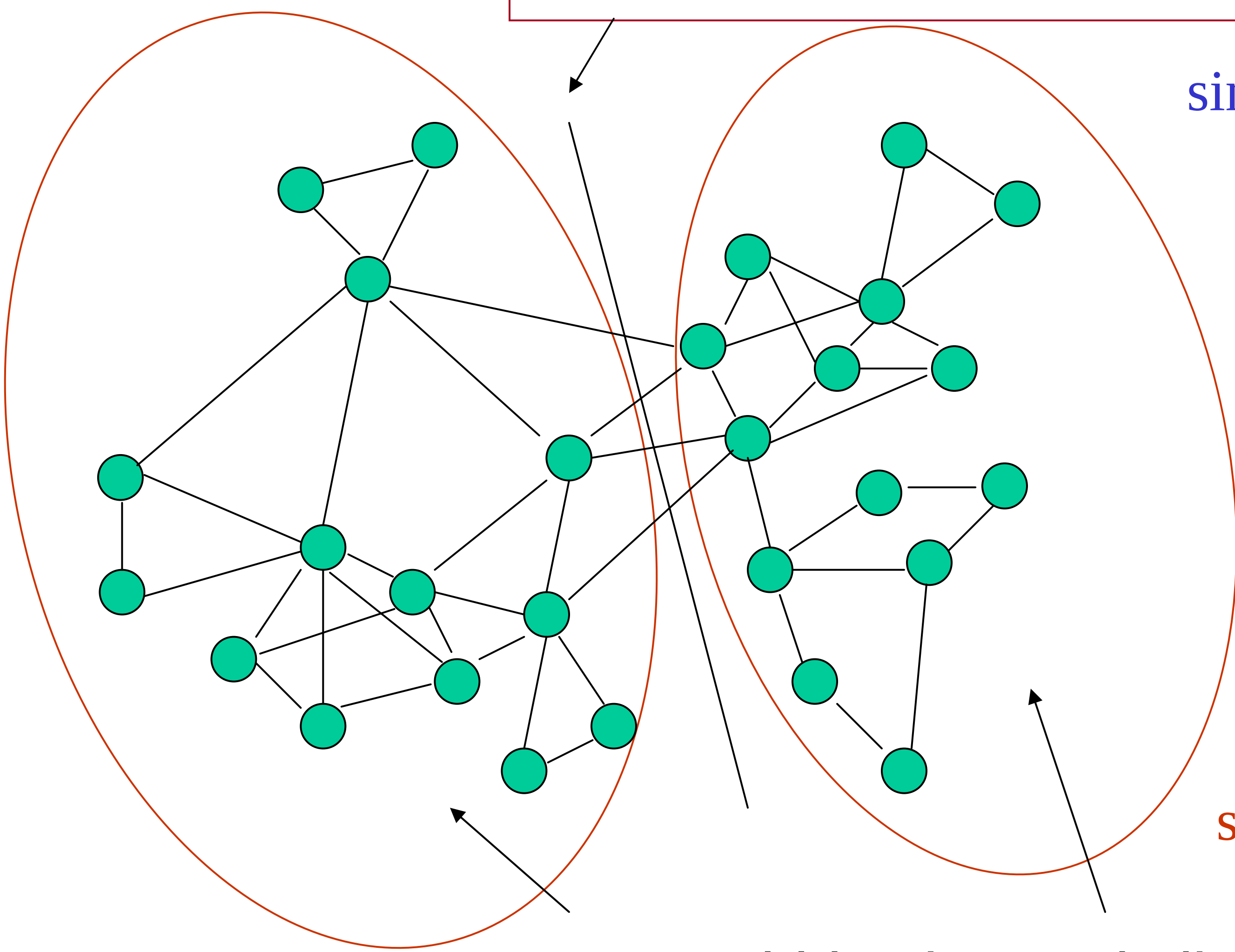


Need to balance sizes

Graph Clustering

min between-cluster similarities (weights)

$$sim(A,B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$



Balance weight

Balance size

Balance volume

$$sim(A,A) = \sum_{i \in A} \sum_{j \in A} w_{ij}$$

max within-cluster similarities (weights)

Clustering Objective Functions

$$s(A,B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$

- Ratio Cut

$$J_{Rcut}(A,B) = \frac{s(A,B)}{|A|} + \frac{s(A,B)}{|B|}$$

- Normalized Cut

$$d_A = \sum_{i \in A} d_i$$

$$J_{Ncut}(A,B) = \frac{s(A,B)}{d_A} + \frac{s(A,B)}{d_B}$$

$$= \frac{s(A,B)}{s(A,A)+s(A,B)} + \frac{s(A,B)}{s(B,B)+s(A,B)}$$

- Min-Max-Cut

$$J_{MMC}(A,B) = \frac{s(A,B)}{s(A,A)} + \frac{s(A,B)}{s(B,B)}$$

Normalized Cut (Shi & Malik, 2000)

Min similarity between A & B: $s(A,B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$

Balance weights $J_{Ncut}(A,B) = \frac{s(A,B)}{d_A} + \frac{s(A,B)}{d_B}$ $\quad d_A = \sum_{i \in A} d_i$

Cluster indicator: $q(i) = \begin{cases} \sqrt{d_B / d_A d} & \textbf{if } i \in A \\ -\sqrt{d_A / d_B d} & \textbf{if } i \in B \end{cases}$ $\quad d = \sum_{i \in G} d_i$

Normalization: $q^T D q = 1, q^T D e = 0$

Substitute q leads to $J_{Ncut}(q) = q^T (D - W) q$

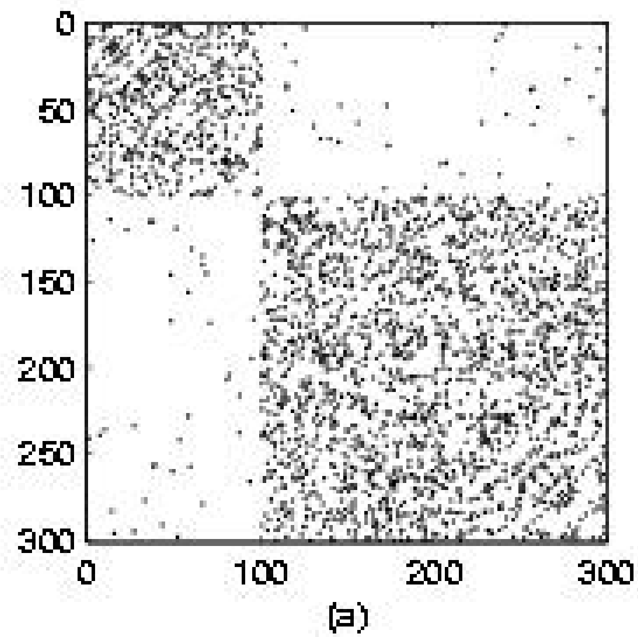
$$\mathbf{min}_{\mathbf{q}}\, q^T (D-W) q + \lambda (q^T D q - 1)$$

Solution is eigenvector of $(D-W) q = \lambda D q$

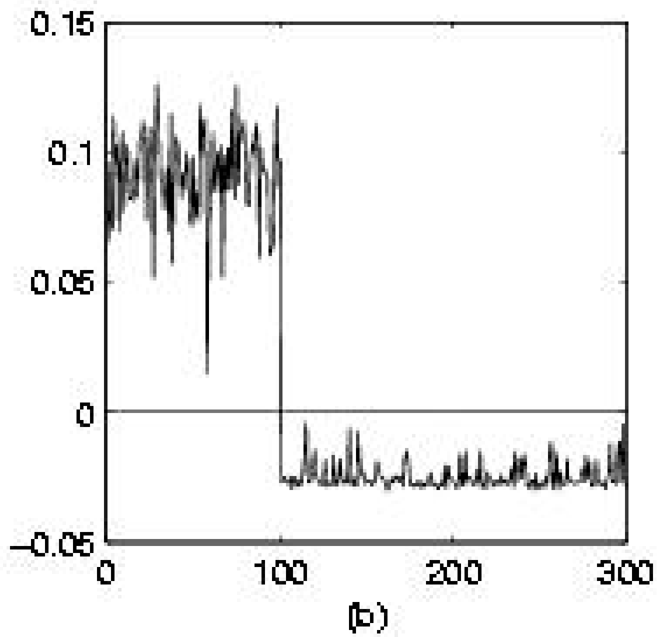
A simple example

2 dense clusters, with sparse connections between them.

Adjacency matrix



Eigenvector q_2



K-way Spectral Clustering

$K \geq 2$

K-way Clustering Objectives

- Ratio Cut

$$J_{\text{Rcut}}(C_1,\cdots,C_K) = \sum_{<k,l>}\left(\frac{s(C_k,C_l)}{|C_k|}+\frac{s(C_k,C_l)}{|C_l|}\right)=\sum_k \frac{s(C_k,G-C_k)}{|C_k|}$$

- Normalized Cut

$$J_{\text{Ncut}}(C_1,\cdots,C_K) = \sum_{<k,l>}\left(\frac{s(C_k,C_l)}{d_k}+\frac{s(C_k,C_l)}{d_l}\right)=\sum_k \frac{s(C_k,G-C_k)}{d_k}$$

- Min-Max-Cut

$$J_{\text{MMC}}(C_1,\cdots,C_K) = \sum_{<k,l>}\left(\frac{s(C_k,C_l)}{s(C_k,C_k)}+\frac{s(C_k,C_l)}{s(C_l,C_l)}\right)=\sum_k \frac{s(C_k,G-C_k)}{s(C_k,C_k)}$$

K-way Spectral Relaxation

Unsigned cluster indicators:

$$h_1 = (1\cdots 1, 0\cdots 0, 0\cdots 0)^T$$
$$h_2 = (0\cdots 0, 1\cdots 1, 0\cdots 0)^T$$
$$\cdots\cdots\cdots$$
$$h_k = (0\cdots 0, 0\cdots 0, 1\cdots 1)^T$$

Re-write:

$$J_{\mathbf{Rcut}}(h_1,\cdots,h_k) = \frac{h_1^T(D-W)h_1}{h_1^T h_1} + \cdots + \frac{h_k^T(D-W)h_k}{h_k^T h_k}$$

$$J_{\mathbf{Ncut}}(h_1,\cdots,h_k) = \frac{h_1^T(D-W)h_1}{h_1^T D h_1} + \cdots + \frac{h_k^T(D-W)h_k}{h_k^T D h_k}$$

$$J_{\mathbf{MMC}}(h_1,\cdots,h_k) = \frac{h_1^T(D-W)h_1}{h_1^T W h_1} + \cdots + \frac{h_k^T(D-W)h_k}{h_k^T W h_k}$$

K-way Normalized Cut Spectral Relaxation

Unsigned cluster indicators:

$$y_k = D^{1/2}(0\cdots0,\overbrace{1\cdots1}^{n_k},0\cdots0)^T / \| D^{1/2}h_k \|$$

Re-write:

$$J_{\mathbf{Ncut}}(y_1,\cdots,y_k) = y_1^T(I-\tilde{W})y_1 + \cdots + y_k^T(I-\tilde{W})y_k$$

$$= \mathbf{Tr}(Y^T(I-\tilde{W})Y) \qquad \tilde{W} = D^{-1/2}WD^{-1/2}$$

$$\textbf{Optimize : } \min_Y \mathbf{Tr}(Y^T(I-\tilde{W})Y), \textbf{subject to } \; Y^TY = I$$

By K. Fan’s theorem, optimal solution is eigenvectors: $Y=(v_1,v_2, \ldots, v_k)$, $\quad (I-\tilde{W})v_k = \lambda_k v_k$

$$(D-W)u_k = \lambda_k D u_k, \quad u_k = D^{-1/2}v_k$$

$$\lambda_1 + \cdots + \lambda_k \le \min J_{\mathbf{Ncut}}(y_1,\cdots,y_k)$$

(Gu, et al, 2001)

K-way Spectral Clustering is difficult

- Spectral clustering is best applied to 2-way clustering
  - positive entries for one cluster
  - negative entries for another cluster
- For K-way (K>2) clustering
  - Positive and negative signs make cluster assignment difficult
  - Recursive 2-way clustering
  - Low-dimension embedding. Project the data to eigenvector subspace; use another clustering method such as K-means to cluster the data (Ng et al; Zha et al; Back & Jordan, etc)
  - Linearized cluster assignment using spectral ordering and cluster crossing

Scaled PCA: a Unified Framework for clustering and ordering

- Scaled PCA has two optimality properties
  - Distance sensitive ordering
  - Min-max principle Clustering
- SPCA on contingency table $\Rightarrow$ Correspondence Analysis
  - Simultaneous ordering of rows and columns
  - Simultaneous clustering of rows and columns

Scaled PCA

similarity matrix $S=(s_{ij})$ (generated from XX^T)

$$D = \text{diag}(d_1, \cdots, d_n) \qquad d_i = s_{i.}$$

Nonlinear re-scaling: $\tilde{S} = D^{-\frac{1}{2}} S D^{-\frac{1}{2}}, \tilde{s}_{ij} = s_{ij} / (s_{i.} s_{j.})^{1/2}$

Apply SVD on $\tilde{S} \Rightarrow$

$$S = D^{\frac{1}{2}} \tilde{S} D^{\frac{1}{2}} = D^{\frac{1}{2}} \sum_k z_k \lambda_k z_k^T D^{\frac{1}{2}} = D \left[\sum_k q_k \lambda_k q_k^T \right] D$$

$q_k = D^{-1/2} z_k$ is the scaled principal component

Subtract trivial component $\lambda_0 = 1, z_0 = d^{1/2}/s.., q_0 = 1$

$$\Rightarrow \qquad S - dd^T/s.. = D \sum_{k=1} q_k \lambda_k q_k^T D$$

(Ding, et al, 2002)

Scaled PCA on a Rectangle Matrix $\Rightarrow$ Correspondence Analysis

Nonlinear re-scaling: $\tilde{P} = D_r^{-\frac{1}{2}} P D_c^{-\frac{1}{2}}, \tilde{p}_{ij} = p_{ij} / (p_{i.} p_{j.})^{1/2}$

Apply SVD on $\tilde{P}$ Subtract trivial component

$$P - rc^T / p.. = D_r \sum_{k=1} f_k \lambda_k g_k^T D_c$$

$$r = (p_{1.}, \cdots, p_{n.})^T$$

$$c = (p_{.1}, \cdots, p_{.n})^T$$

$$f_k = D_r^{-\frac{1}{2}} u_k, \; g_k = D_c^{-\frac{1}{2}} v_k$$

are the scaled row and column principal component (standard coordinates in CA)

Correspondence Analysis (CA)

- Mainly used in graphical display of data
- Popular in France (Benzécri, 1969)
- Long history
  - Simultaneous row and column regression (Hirschfeld, 1935)
  - Reciprocal averaging (Richardson & Kuder, 1933; Horst, 1935; Fisher, 1940; Hill, 1974)
  - Canonical correlations, dual scaling, etc.
- Formulation is a bit complicated ("convoluted" Jolliffe, 2002, p.342)
- "A neglected method", (Hill, 1974)

Clustering of Bipartite Graphs (rectangle matrix)

Simultaneous clustering of rows and columns of a contingency table (adjacency matrix B)

Examples of bipartite graphs

- Information Retrieval: word-by-document matrix
- Market basket data: transaction-by-item matrix
- DNA Gene expression profiles
- Protein vs protein-complex

Bipartite Graph Clustering

Clustering indicators for rows and columns:

$$f(i) = \begin{cases} 1 & \text{if } r_i \in R_1 \\ -1 & \text{if } r_i \in R_2 \end{cases} \qquad g(i) = \begin{cases} 1 & \text{if } c_i \in C_1 \\ -1 & \text{if } c_i \in C_2 \end{cases}$$

$$B = \begin{pmatrix} B_{R_1,C_1} & B_{R_1,C_2} \\ B_{R_2,C_1} & B_{R_2,C_2} \end{pmatrix} \qquad W = \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix} \qquad \mathbf{q} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

Substitute and obtain

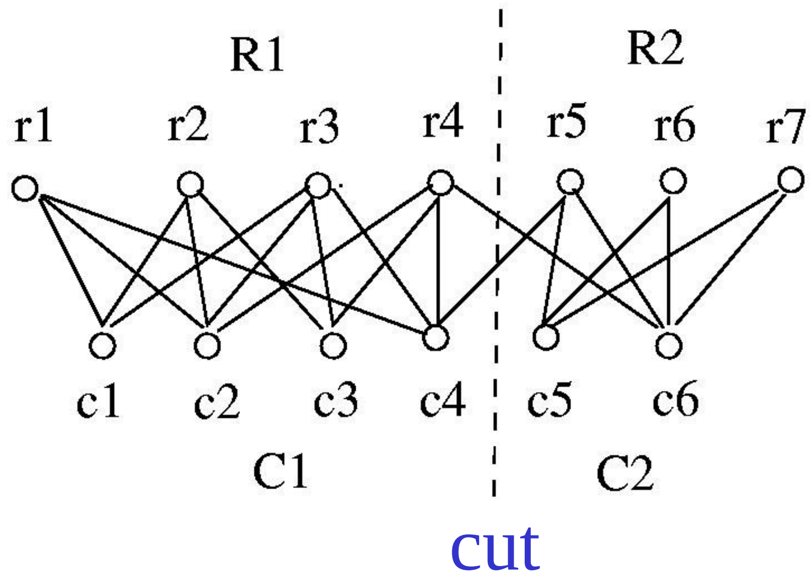
$$J_{MMC}(C_1, C_2; R_1, R_2) = \frac{s(W_{12})}{s(W_{11})} + \frac{s(W_{12})}{s(W_{22})}$$

f,g are determined by

$$\left[\begin{pmatrix} D_r & \\ & D_c \end{pmatrix} - \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix} \right] \begin{pmatrix} f \\ g \end{pmatrix} = \lambda \begin{pmatrix} D_r & \\ & D_c \end{pmatrix} \begin{pmatrix} f \\ g \end{pmatrix}$$

Spectral Clustering of Bipartite Graphs

Simultaneous clustering of rows and columns (adjacency matrix B)



$$s(B_{R_1,C_2}) = \sum_{r_i \in R_1} \sum_{c_j \in C_2} b_{ij}$$

min between-cluster sum of weights: $s(R_1,C_2)$, $s(R_2,C_1)$

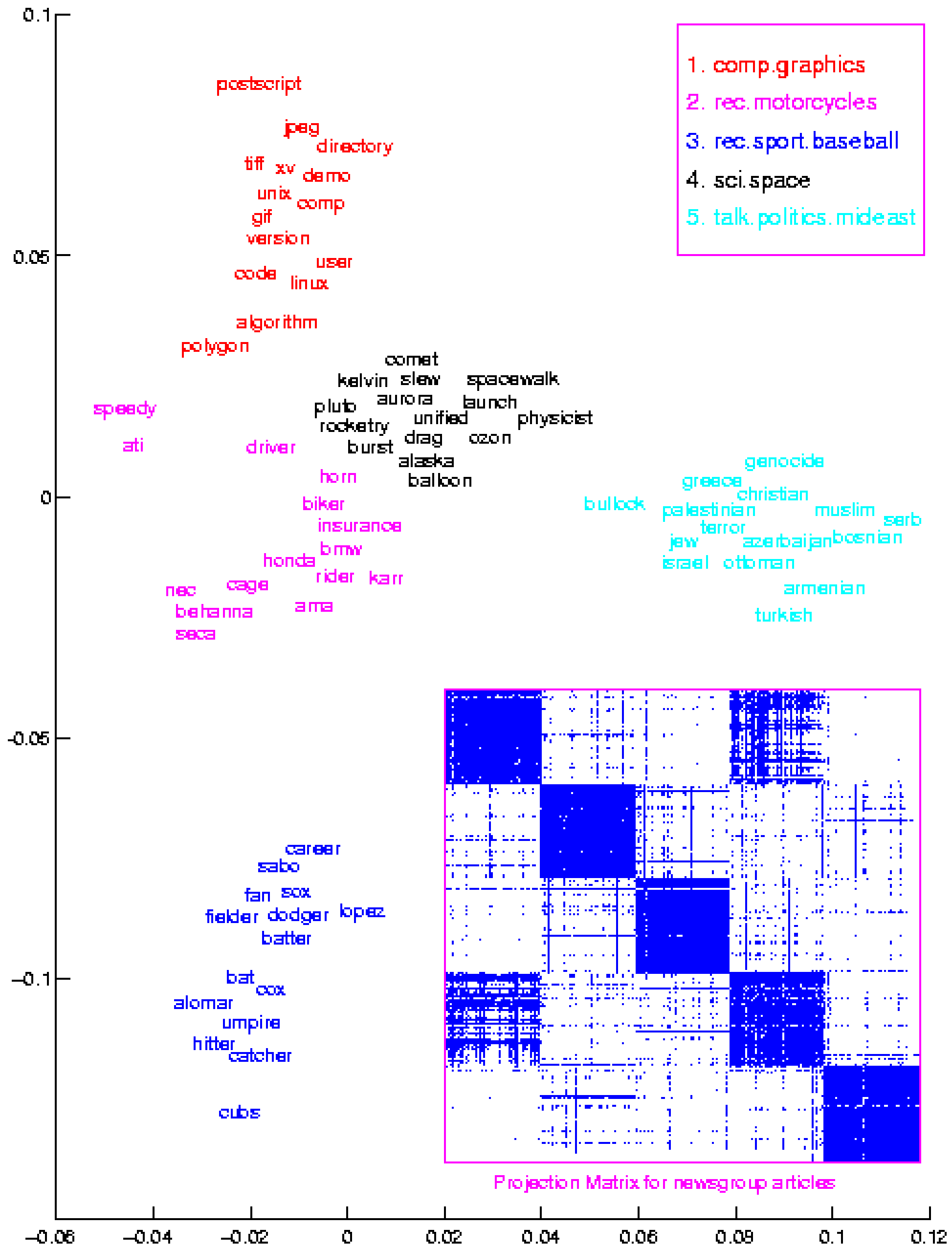
max within-cluster sum of weights: $s(R_1,C_1)$, $s(R_2,C_2)$

$$J_{MMC}(C_1,C_2;R_1,R_2) = \frac{s(B_{R_1,C_2}) + s(B_{R_2,C_1})}{2s(B_{R_1,C_1})} + \frac{s(B_{R_1,C_2}) + s(B_{R_2,C_1})}{2s(B_{R_2,C_2})}$$

(Ding, AI-STAT 2003)

Internet Newsgroups

Simultaneous clustering of documents and words



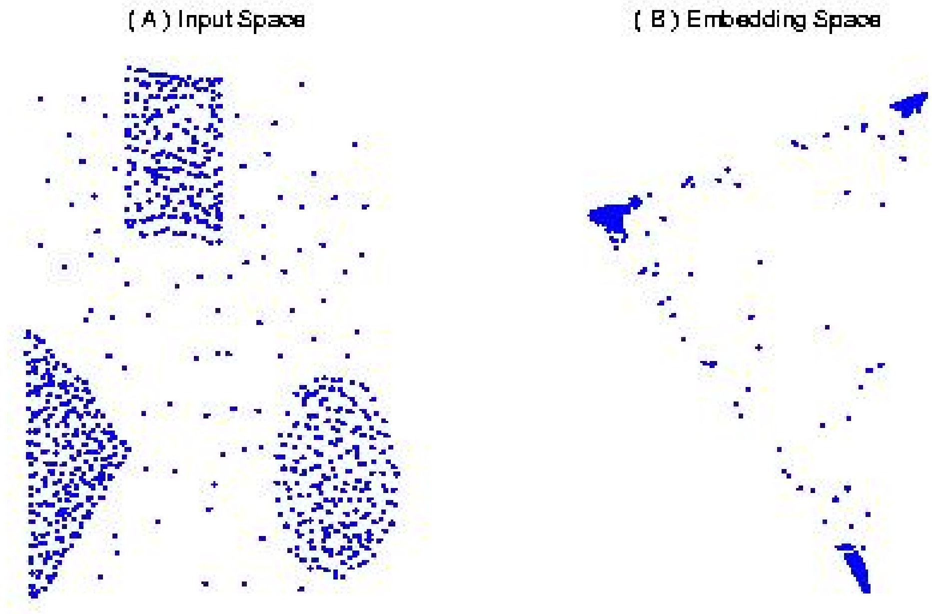
Embedding in Principal Subspace

Cluster Self-Aggregation
(proved in perturbation analysis)

(Hall, 1970, “quadratic placement” (embedding) a graph)

Spectral Embedding: Self-aggregation

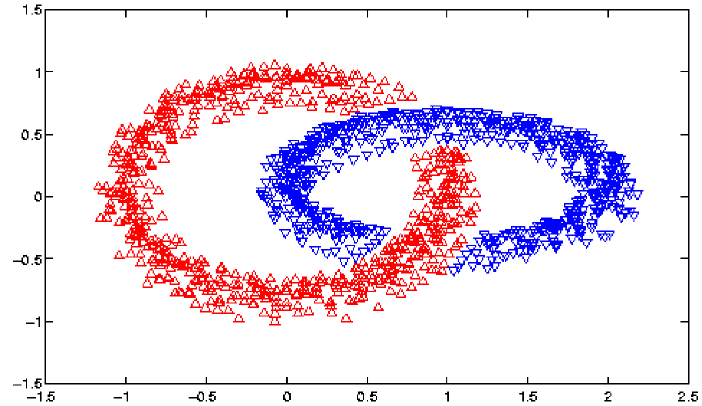
- Compute K eigenvectors of the Laplacian.
- Embed objects in the K-dim eigenspace



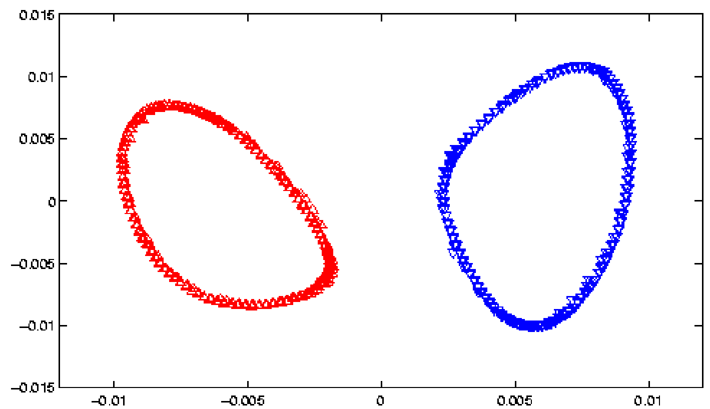
(Ding, 2004)

Spectral embedding is not topology preserving

700 3-D data points form 2 interlock rings



In eigenspace, they shrink and separate



Spectral Embedding

Simplex Embedding Theorem.

Objects self-aggregate to K centroids

Centroids locate on K corners of a simplex

- Simplex consists K basis vectors + coordinate origin
- Simplex is rotated by an orthogonal transformation T
- T are determined by perturbation analysis

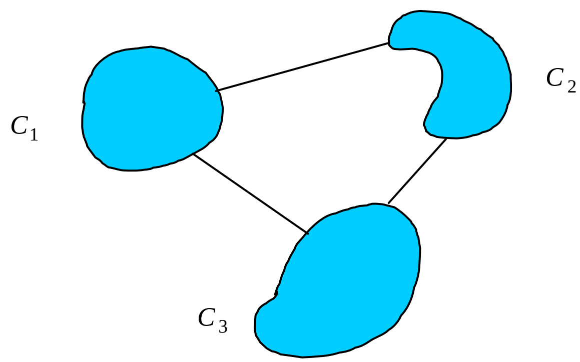
(Ding, 2004)

Perturbation Analysis

$$Wq = \lambda Dq \qquad \hat{W}z = (D^{-1/2} W D^{-1/2}) z = \lambda z \quad q = D^{-1/2} z$$

Assume data has 3 dense clusters sparsely connected.

$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}$$



Off-diagonal blocks are between-cluster connections, assumed small and are treated as a perturbation

(Ding et al, KDD'01)

Spectral Perturbation Theorem

Orthogonal Transform Matrix $T = (\mathbf{t}_1, \cdots, \mathbf{t}_K)$

T are determined by: $\Gamma \mathbf{t}_k = \lambda_k \mathbf{t}_k$

Spectral Perturbation Matrix $\Gamma = \Omega^{-\frac{1}{2}} \overline{\Gamma} \Omega^{-\frac{1}{2}}$

$$\overline{\Gamma} = \begin{bmatrix} h_{11} & -s_{12} & \cdots & -s_{1K} \\ -s_{21} & h_{22} & \cdots & -s_{2K} \\ \vdots & \vdots & \cdots & \vdots \\ -s_{K1} & -s_{K2} & \cdots & h_{KK} \end{bmatrix}$$

$$s_{pq} = s(C_p, C_q)$$

$$h_{kk} = \sum\nolimits_{p|p\neq k} s_{kp}$$

$$\Omega = \mathbf{diag}[\rho(C_1), \cdots, \rho(C_k)]$$

Connectivity Network

$$C_{ij} = \begin{cases} 1 & \text{if } \; i, j \; \text{belong to same cluster} \\ 0 & \text{otherwise} \end{cases}$$

Scaled PCA provides
$$C \cong D \sum_{k=1}^{K} q_k \lambda_k \, q_k^T \, D$$

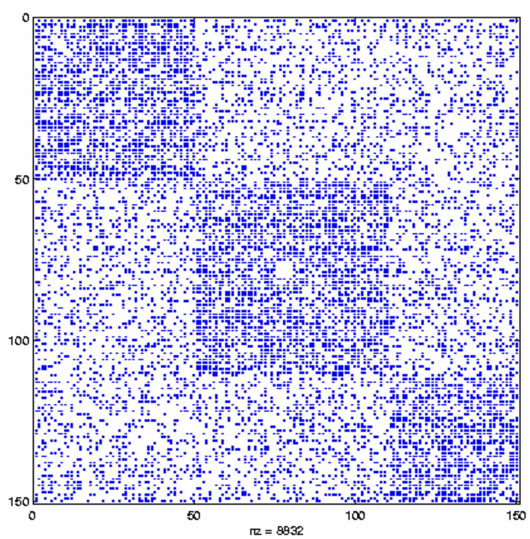
Green's function :
$$C \approx G = \sum_{k=2}^{K} q_k \frac{1}{1-\lambda_k} q_k^T$$

Projection matrix:
$$C \approx P \equiv \sum_{k=1}^{K} q_k \, q_k^T$$

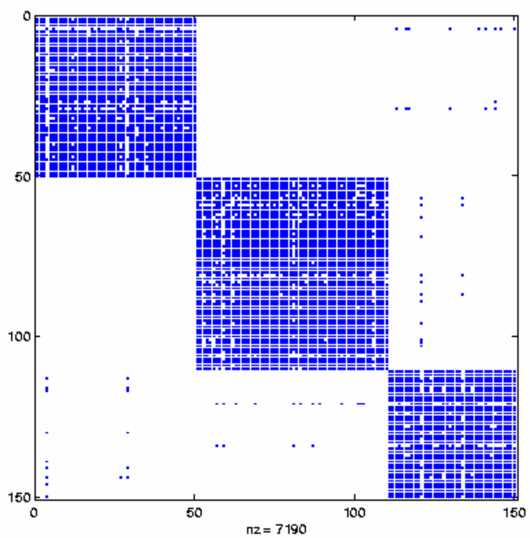
(Ding et al, 2002)

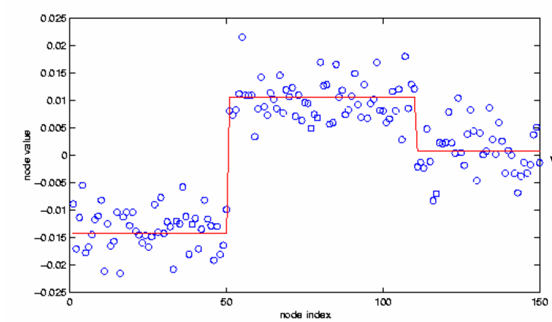
1^{st} order Perturbation: Example 1

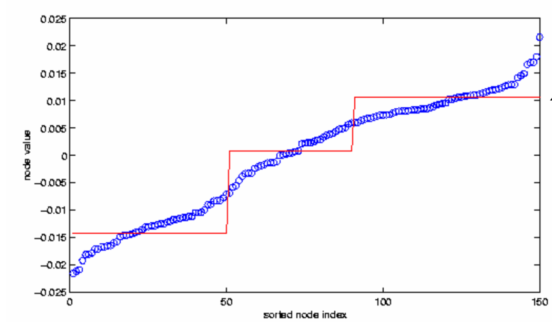
Similarity matrix W



Connectivity matrix







1^{st} order solution

$$\lambda_2 = 0.300, \bar{\lambda}_2 = 0.268$$

Between-cluster connections suppressed

Within-cluster connections enhanced

Effects of self-aggregation

Optimality Properties of Scaled PCA

Scaled principal components have optimality properties:

Ordering

- Adjacent objects along the order are similar
- Far-away objects along the order are dissimilar
- Optimal solution for the permutation index are given by scaled PCA.

Clustering

- Maximize within-cluster similarity
- Minimize between-cluster similarity
- Optimal solution for cluster membership indicators given by scaled PCA.

Spectral Graph Ordering

(Barnard, Pothen, Simon, 1993), envelop reduction of sparse matrix: find ordering such that the envelop is minimized

$$\min \sum_i \max_j |i-j| w_{ij} \quad \Rightarrow \quad \min \sum_{ij} (x_i - x_j)^2 w_{ij}$$

(Hall, 1970), "quadratic placement of a graph":

Find coordinate x to minimize

$$J = \sum_{ij} (x_i - x_j)^2 w_{ij} = x^T (D - W) x$$

Solution are eigenvectors of Laplacian

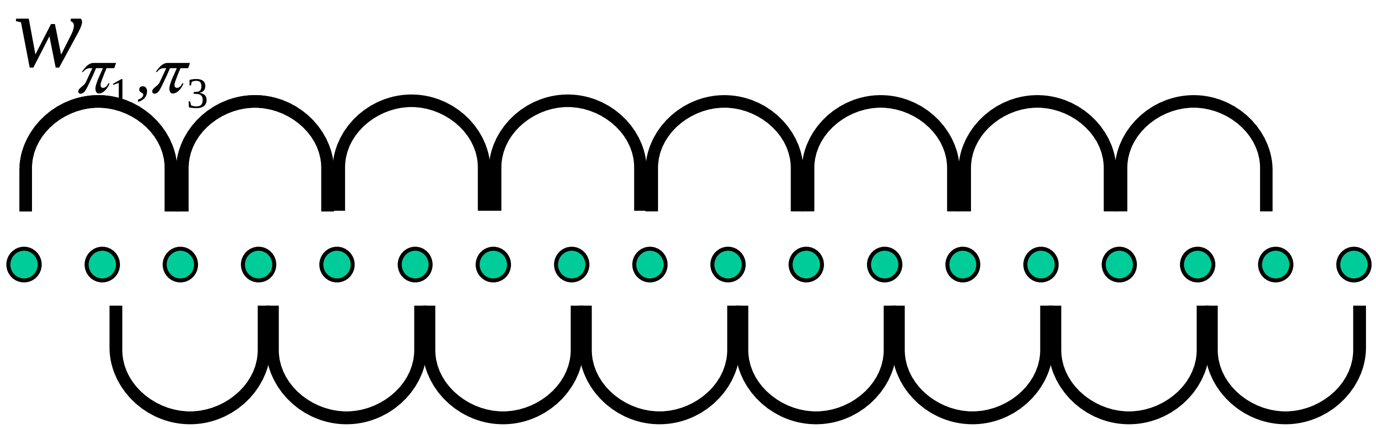
Distance Sensitive Ordering

Given a graph. Find an optimal Ordering of the nodes.

π permutation indexes

$$J_d(\pi) = \sum_{i=1}^{n-d} w_{\pi_i,\pi_{i+d}} \qquad \pi(1,\cdots,n) = (\pi_1,\cdots,\pi_n)$$

$$J_{d=2}(\pi):$$



$$\min_{\pi} J(\pi) = \sum_{d=1}^{n-1} d^2 J_d(\pi)$$

The larger distance, the larger weights, panelity.

Distance Sensitive Ordering

$$J(\pi) = \sum_{ij} (i-j)^2 w_{\pi_i,\pi_j} = \sum_{\pi_i,\pi_j} (i-j)^2 w_{\pi_i,\pi_j}$$

$$= \sum_{ij} (\pi_i^{-1} - \pi_j^{-1})^2 w_{i,j}$$

$$= \frac{n^2}{8} \sum_{ij} \left(\frac{\pi_i^{-1} - (n+1)/2}{n/2} - \frac{\pi_j^{-1} - (n+1)/2}{n/2}\right)^2 w_{i,j}$$

Define: shifted and rescaled inverse permutation indexes

$$q_i = \frac{\pi_i^{-1} - (n+1)/2}{n/2} = \left\{\frac{1-n}{n}, \frac{3-n}{n}, \cdots, \frac{n-1}{n}\right\}$$

$$J(\pi) = \tfrac{n^2}{8} \sum_{ij} (q_i - q_j)^2 w_{ij} = \tfrac{n^2}{4} q^T (D-W) q$$

Distance Sensitive Ordering

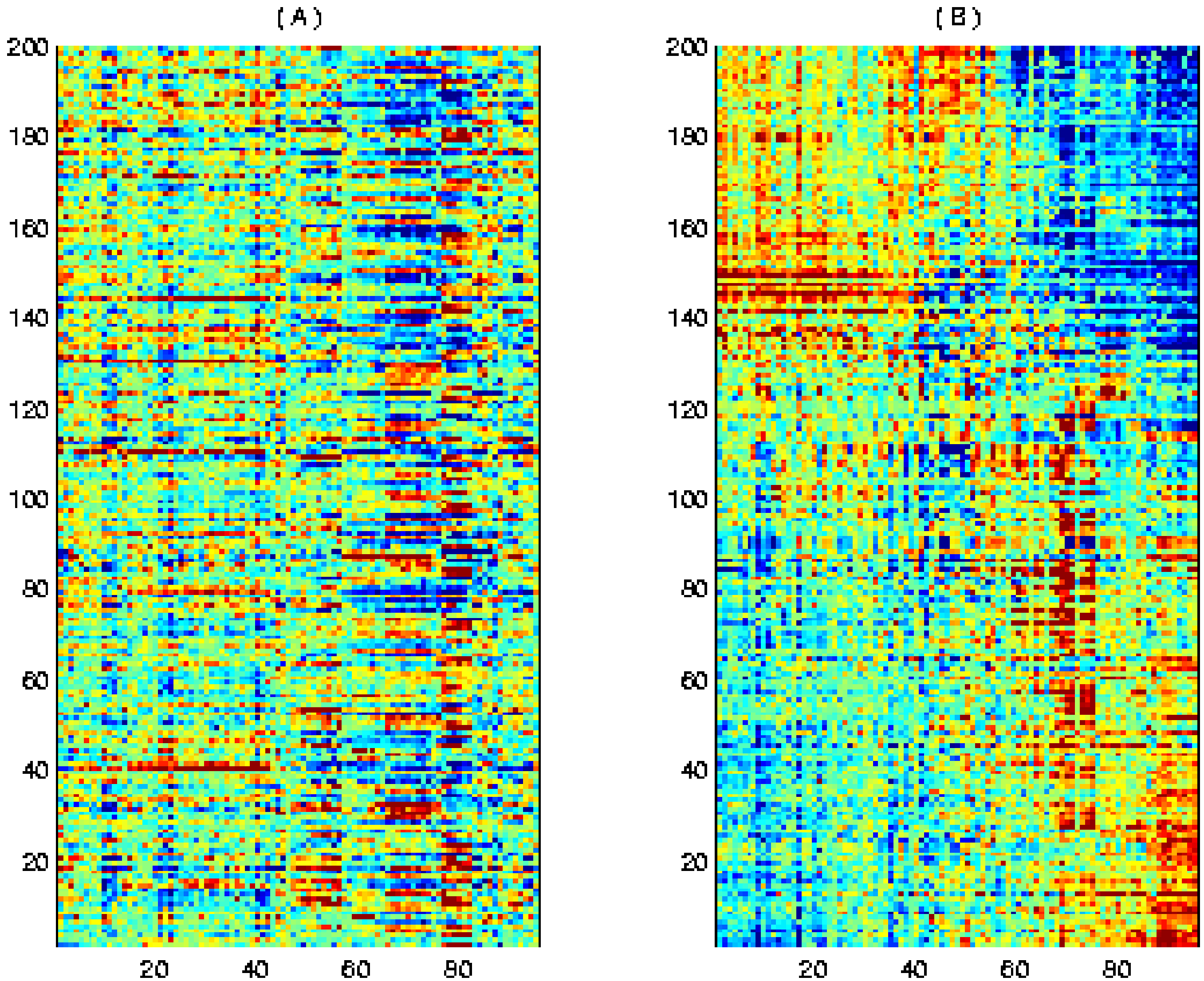
Once q_2 is computed, since

$$q_2(i) < q_2(j) \Rightarrow \pi_i^{-1} < \pi_j^{-1}$$

π_i^{-1} can be uniquely recovered from q_2

Implementation: sort $\boldsymbol{q}_2$ induces π

Re-ordering of Genes and Tissues



$$r = \frac{J(\pi)}{J(\text{random})}$$

$$r = 0.18$$

$$r_{d=1} = \frac{J_{d=1}(\pi)}{J_{d=1}(\text{random})}$$

$$r_{d=1} = 3.39$$

Spectral clustering vs Spectral ordering

- Continuous approximation of both integer programming problems are given by the same eigenvector

- Different problems could have the same continuous approximate solution.

- Quality of the approximation:

  Ordering: better quality: the solution relax from a set of evenly spaced discrete values

  Clustering: less better quality: solution relax from 2 discrete values

Linearized Cluster Assignment

Turn spectral clustering to 1D clustering problem

- Spectral ordering on connectivity network
- Cluster crossing
  - Sum of similarities along anti-diagonal
  - Gives 1-D curve with valleys and peaks
  - Divide valleys and peaks into clusters

Cluster overlap and crossing

Given similarity W, and clusters A,B.

- Cluster overlap

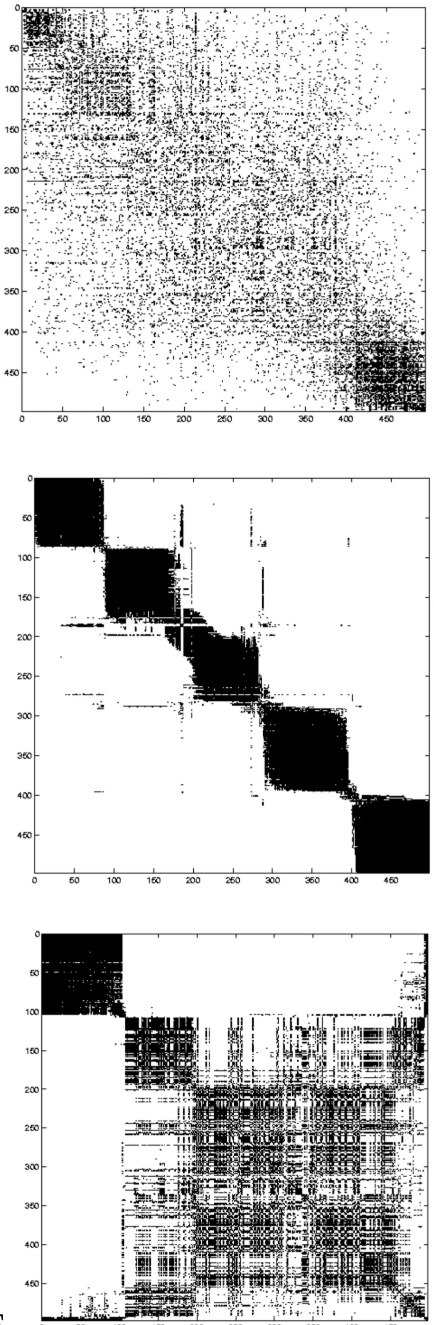
$$s(A,B) = \sum_{i \in A} \sum_{j \in B} w_{ij}$$

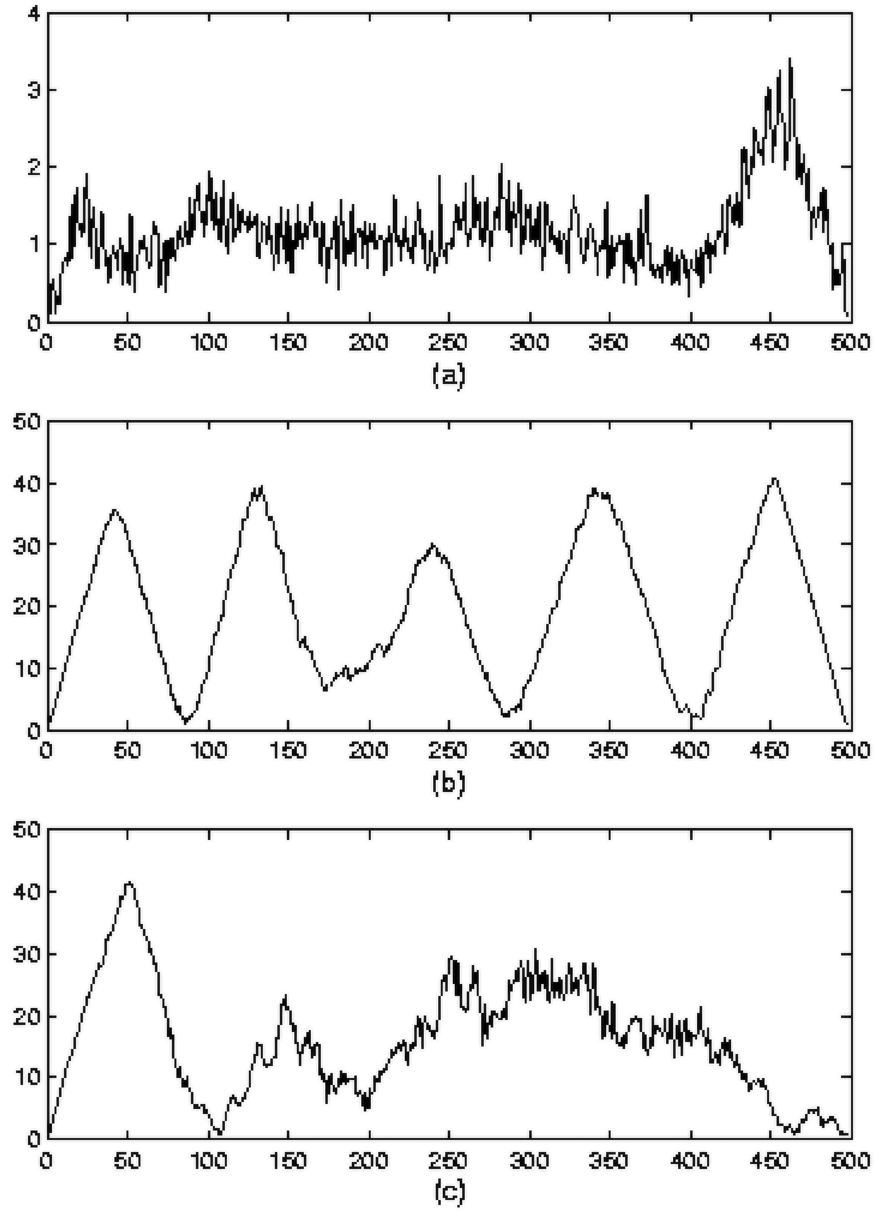
- Cluster crossing compute a smaller fraction of cluster overlap.
- Cluster crossing depends on an ordering $\boldsymbol{o}$. It sums weights cross the site i along the order

$$\rho(i) = \sum_{j=1}^{m} w_{o(i-j),o(i+j)}$$

- This is a sum along anti-diagonals of W.

cluster crossing





K-way Clustering Experiments

Accuracy of clustering results:

Method	Linearized Assignment	Recursive 2-way clustering	Embedding + *K*-means
Data A	89.0%	82.8%	75.1%
Data B	75.7%	67.2%	56.4%

Some Additional Advanced/related Topics

- Random talks and normalized cut
- Semi-definite programming
- Sub-sampling in spectral clustering
- Extending to semi-supervised classification
- Green's function approach
- Out-of-sample embeding