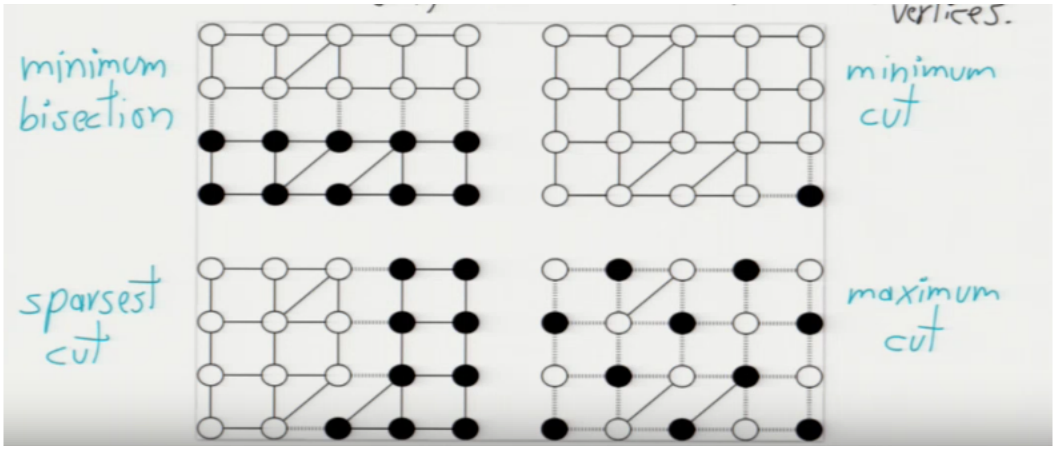
04/18/2016

Spectral Graph Clustering

- Input: Weighted, undirected graph $G = (V, E)$. No self-edges. w_{ij} = weight of edge, $(i,j) = (j,i)$; zero if (i,j) not in E.
- Goal: Cut G into 2 (or more) pieces G_i of similar sizes, but don't cut too much edge weight. e.g. Minimize the sparsity $\frac{Cut(G_1,G_2)}{Mass(G_1)\cdot Mass(G_2)}$ aka cut ratio, where $Cut(G_1, G_2)$ = total weight of cut edges, $Mass(G_1)$ =# of vertices in G_i or assign masses to vertices.



- Let $n = |V|$. Let $y \in \mathbb{R}^n$ be an indicator vector:

$$y_i = \begin{cases} 1 & \text{vertex } i \in G_1 \\ -1 & \text{vertex } i \in G_2 \end{cases}$$

then

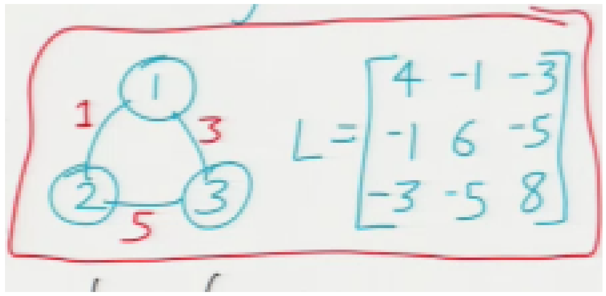
$$\frac{w_{ij}}{4}(y_i - y_j)^2 = \begin{cases} w_{ij} & (i,j) \text{ is a cut,} \\ 0 & (i,j) \text{ is not a cut.} \end{cases}$$

$$\begin{aligned} Cut(G_1, G_2) &= \sum_{(i,j)\in E} \frac{w_{ij}}{4}(y_i - y_j)^2 \\ &= \frac{1}{4}\sum_{(i,j)\in E}(w_{ij}y_i^2 - 2w_{ij}y_iy_j + w_{ij}y_j^2) \\ &= \frac{1}{4}\left(\sum_{(i,j)\in E} -2w_{ij}y_iy_j + \sum_{i=1}^{n} y_i^2 \sum_{k\neq i} w_{ik}\right) \\ &= \frac{y^T L y}{4} \end{aligned}$$

where,

$$L_{ij} = \begin{cases} -w_{ij} & i \neq j \\ \sum_{k\neq i} w_{ik} & i = j \end{cases}$$

- L is symmetric, n−by−n Laplacian matrix for G.

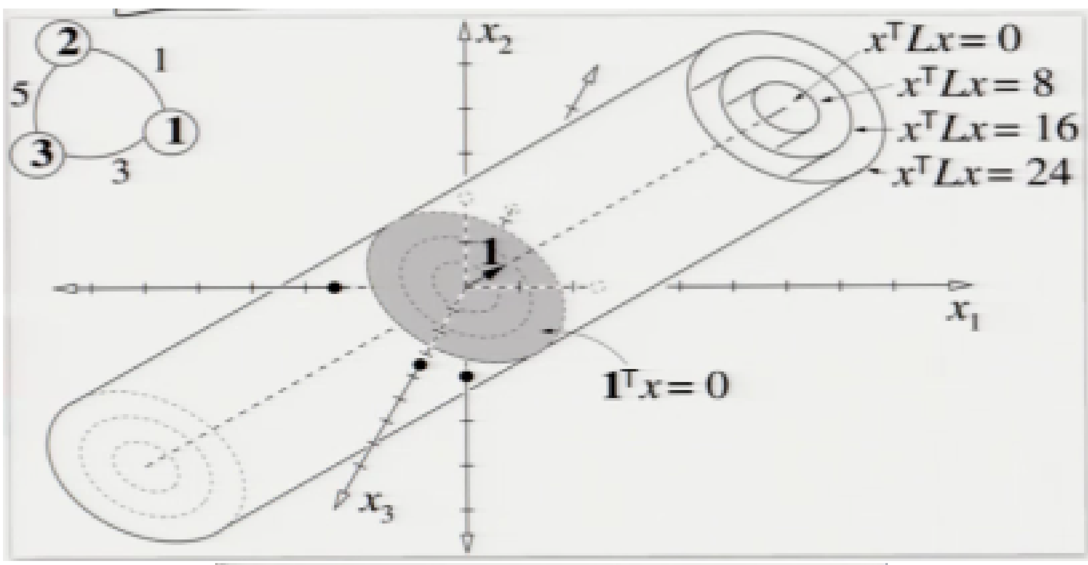


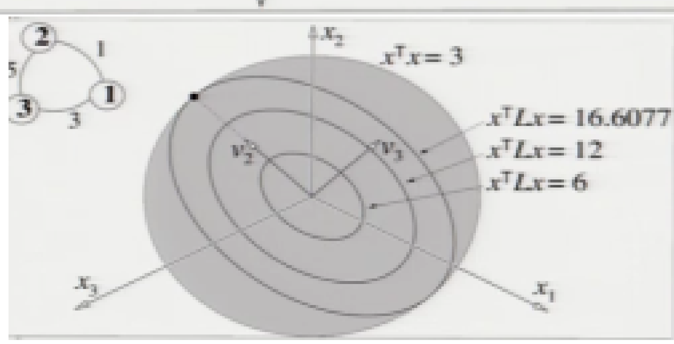
- If $y = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \end{bmatrix}^T$, then $Cut(G_1, G_2) = 0$ 1 is an eigenvector of L w/eigenvalue 0.
- Bisection: exactly $\frac{n}{2}$ vertices in G_1, $\frac{n}{2}$ in G_2. Write $\mathbf{1}^T y = 0$.

$$\boxed{\text{Find } y \text{ that minimizes } y^T L y \text{ s.t. } \forall i,\ y_i = 1 or y_i = -1 \text{ and } \mathbf{1}^T y = 0}$$

- NP-hard. We relax the binary constraint → fractional vertices.
- New constraint: y must lie on sphere of radius $\sqrt{n}$.
- Relaxed problem:

$$\boxed{\text{Minimize } y^T L y \text{ s.t. } y^T y = n \text{ and } \mathbf{1}^T y = 0}$$



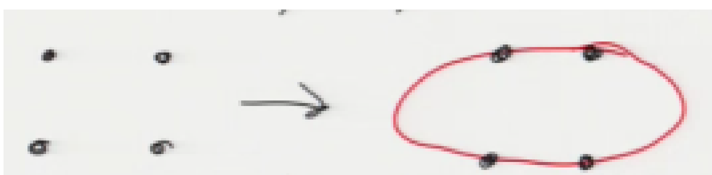


- Let λ_2 = second-smallest eigenvalue of L.

- Eigenvector v_2 is the fiedler vector.
- Spectral partitioning algorithm:
  - Compute Fiedler vector v_2 of L.
  - Round v_2 with a sweep cut:
    - Sort components of v_2
    - Try the $n-1$ cuts between successive components.
    - Choose sparsest cut.
- Fact: Sweep cut finds a cut w/sparsity $\leq \sqrt{2\lambda_2 \ \max_i \frac{L_{ii}}{M_{ii}}}$.
- Cheeger's inequality.
- The optimal cut has sparsity $\geq \frac{\lambda_2}{2}$.

Vertex Masses

- Let M be a diagonal matrix w/vertex masses on diagonal.
- New balance constraint: $\mathbf{1}^T My = 0$
- New ellipsoid constraint: $y^T My = Mass(G) = \sum_i M_{ii}$.



- Now we want Fiedler vector of generalized eigensystem $Lv = \lambda Mv$.

Greedy Divisive Clustering

- Partition G into 2 subgraphs; recursively cluster them.
- Can form a dendogram, but it may have inversions.

The Normalized Cut

- Set vertex i's mass $M_{ii} = L_{ii}$.
- Popular for image segmentation.
- For pixels with location w_i, brightness b_i

$$w_{ij} = exp\left(\frac{-|w_i - w_j|^2}{\alpha} - \frac{|b_i - b_j|^2}{\beta}\right)$$

  or zero if $|w_i - w_j|$ large.