Spectral Clustering

May 7, 2018

Quadratic Form

- Let $\mathbf{A}$ be some $n \times n$ matrix.
- What is $\mathbf{Ax}$? What's the type of the output? What may $\mathbf{x}$ represent?
  - Some numeric assignment on all nodes in G.
  - E.g., what if x_i $0, 1$? x_i $[0, 1]$? x_i ?
-

Quadratic Form

- Let $\mathbf{A}$ be some $n \times n$ matrix.
- What is $\mathbf{Ax}$? What's the type of the output? What may $\mathbf{x}$ represent?
  - Some numeric assignment on all nodes in G.
  - E.g., what if x_i $0, 1$? x_i $[0, 1]$? x_i ?
-
  -

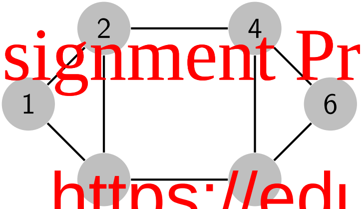
- Let **A** is the adjacency matrix of a "normal" (unweighted) undirected graph G. $\mathbb{V}$ are the vertices of G and $\mathbb{E}$ are the edges of G
  - An edge between v_i and v_j is modelled as (i, j) and (j, i), i.e., $A_{ij} = A_{ji} = 1$.
  -
-
- d

  $d_i = deg(v_i)$?
- What about $\mathbf{x}^\top \mathbf{A} \mathbf{x}$?
- Now what about $2(\mathbf{x}^\top \mathbf{D} \mathbf{x} - \mathbf{x}^\top \mathbf{A}$

Example

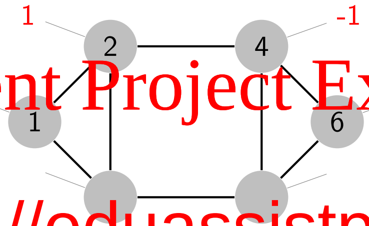
$$\mathbf{x}^\top \mathbf{L} \mathbf{x} = \frac{1}{2} \cdot \sum_{e_{ij} \in \mathbb{E}} (x_i - x_j)^2 \quad \text{, where } \mathbf{L} = \mathbf{D} - \mathbf{A}.$$

Example



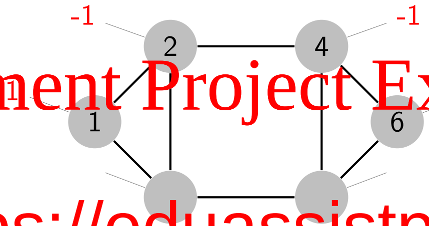
	n_1	n_2	n_3	n_4	n_5	n_6
n_1						
n_2						
n_3						

- $\mathbf{1}$ is the one vector.
- $\mathbf{L}\mathbf{1} =$ (NB: $\mathbf{L}^\top = \mathbf{L}$) $=$ $\mathbf{1}$
- $\mathbf{x}^\top \mathbf{L}\mathbf{x} =$



- $\mathbf{x} =$
- $\mathbf{x}^\top \mathbf{L} \mathbf{x} =$

Binary $\mathbf{x}$ induces a Clustering /2



- $\mathbf{x} =$
- $\mathbf{x}^\top \mathbf{L} \mathbf{x} =$
- $\mathbf{x}^\top \mathbf{x} =$

- Min cuts are not *always* desirable.
  - Biased towards cutting small sets of isolated nodes.

- Cut: $cut(A, B) = \sum_{v_i \in A, v_j \in B} w_{i,j}$.
- Normalized cut:

$$ncut(A, B) = \frac{cut(A, B)}{vol(A)} + \frac{cut(A, B)}{vol(B)},$$

where $vol(A) = \sum_{v_i \in A} d_i) = \sum_{v_i \in A, v_j \in \mathbb{V}} w_{i,j}$.

$$ncut(A, B) = cut(A, B) \left(\frac{1}{vol(A)} + \frac{1}{vol(B)} \right)$$

- Let $x_i = \frac{1}{vol(A)}$ if $v_i \in A$, and $= \frac{-1}{vol(B)}$ otherwise.
- $\mathbf{x}^\top \mathbf{L} \mathbf{x} = \sum_e w_{ij}(x_i - x_j)^2 = 0 + \sum_{v_i \in A, v_j \in B} \left(\frac{1}{vol(A)} + \frac{1}{vol(B)} \right)^2$
-

$$ncut(A, B) = \text{———}$$

$$\text{Minimize } ncut(A, B) = \frac{\mathbf{x}^\top \mathbf{L}\mathbf{x}}{\mathbf{x}^\top \mathbf{D}\mathbf{x}} \quad \text{Subject to} \quad x_i \in \left\{ \frac{1}{vol(A)}, \frac{-1}{vol(B)} \right\}$$

-
-
  - allow $\mathbf{x}$ to be a real vector?
  - Yes, but too large.
  - This gives the constraint: $\mathbf{x}^\top \mathbf{D}$ ... $\mathbf{1}$
    (You can verify this by plugging in any discr

- Solution: the second smallest eigenvector of the generalized eigen value problem $\mathbf{L}\mathbf{x} = \lambda \mathbf{D}\mathbf{x}$.
- Normalized Laplacian:
  $\mathbf{L}' = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$

- Algorithm $SC\_recursive\_bin\_cut(data, k)$
  - Construct the weighted graph G
  - Construct the special graph laplacian L for G.
  - Compute the smallest non-zero eigenvector for L. This is the new representation of vertices in a new 1-dimensional space
  -
  -

    are needed.

Spectral Clustering Algorithm Framework

- Algorithm $SC\_k\_way\_cut(data, k)$
  - Construct the weighted graph G
  - Construct the special graph laplacian L for G.
  - Compute the smallest t non-zero eigenvector for L. This is the new representation of vertices in a new t-dimensional space
  -

Notes on the Algorithms

- How to construct the weighted graph if only n objects are given?
  - Be based on the similarity or distance among objects.
  - E.g., $w_{ij} = \exp(-\frac{\|f(o_i)-f(o_j)\|}{2\sigma^2})$ where $f(o)$ is the feature vector of object o. One can also induce a sparse graph if one caps the

-
  -
  - Normalized graph laplacian $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}}$.

- Pros:
  - Usually better quality than other methods.
  - Can be thought of (non-linear) dimensionality reduction or embedding.
  - Freedom to construct a (sparse) G to preserve local
  -
  - al

  case).

- Cons:
  - Still need to determine k
  - Assumes clusters are of similar sizes.
  - Does not scale well with large datasets; but more scalable variants exist.
  - One of the relaxation of the original NP-hard problem – may not be the tightest relaxation.