Networks/Graphs

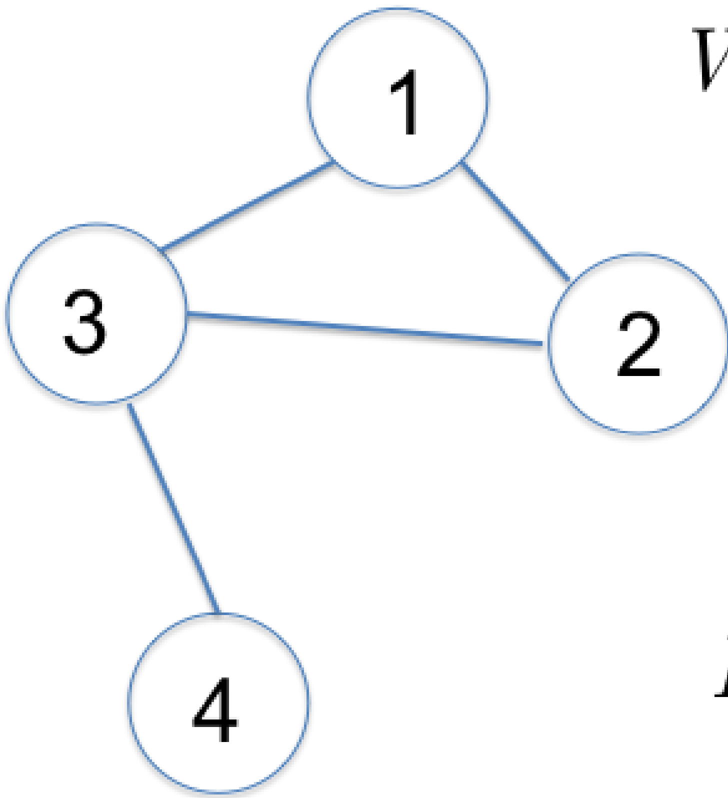
A graph G contains

- a vertex set $V = \{v_i\}_{i=1}^n$.

  (Just refer to node v_i as node i.)

- an edge set W: $W_{ij} = 1$ if v_i and v_j are connected, i.e., (i, j) is in the edge set.

  W_{ij} may take any positive real value, indicating the strength of the connection. Think of W as an $n \times n$ symmetric matrix.



$$W = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

$$D = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Similarity Graphs

- $V = \{x_i\}_{i=1}^n$
- $W = \{w_{ij}\}, \quad w_{ij} = \text{similarity}(x_i, x_j)$.
- Ex1: $w_{ij} =$ correlation or $\exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$.
- Ex2: ϵ-neighborhood graph:

$$w_{ij} = 0 \text{ if } d(x_i, x_j) \geq \epsilon.$$

- Ex3: kNN graph: $w_{ij} = 0$ if neither i nor j is among the other's kNN. In a mutual kNN graph, $w_{ij} = 0$ if at least one of i, j is not among the other's kNN.

Graph Partition

Consider bi-partitions.

- $V = A \cup B$

- What is a good partition?

- How to find such a partition?

Clustering Objectives

- Goodness of clustering
  - Points in the same cluster should be similar.
  - Points in different clusters should be dissimilar.
- In the similarity graph representation, we want to
  - Maximize within-cluster weights;
  - Minimize between-cluster weights.

MinCut

- Objective function

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

- Partition membership indicator

$$q_i = 1, i \in A; \quad -1, i \in B \quad (\|q\|^2 = n)$$

- Objective function

$$J_{\mathsf{Mcut}} = \frac{1}{8}\sum_{i,j} w_{ij}(q_i - q_j)^2 = \frac{1}{4} q^t (D - W) q.$$

- Approximate q by the 2nd eigenvector of

$$(D-W)q = \lambda q, \qquad D = \mathrm{diag}(W\mathbf{1}).$$

- Laplacian matrix of a graph: $L = D - W$
  - L is symmetric and semi-positive definite.

$$\alpha^t L \alpha = \frac{1}{2}\sum_{i,j} w_{ij}(\alpha_i - \alpha_j)^2 \geq 0.$$

  - 1st eigenvector is $\mathbf{1}_{n\times 1}$ with $\lambda_1 = 0$.
  - 2nd eigenvector is the desired solution (Let's exclude the trivial case where L has a block diagonal form).
  - Higher eigenvectors can also be used for further partition.

- Recovering the partition: choose the splitting point for q.
  - split at 0

$$A = \{i; q(i) < 0\}, \quad B = \{i; q(i) \geq 0\}.$$

  - split at median, if requires $|A| = |B|$ (balance sizes);
  - choose the optimal splitting point to minimize $s(A, B)$.
- Drawback of MinCut.

Normalized Cut

- Normalized cut (Shi & Malik, 1997)

$$
\begin{aligned}
J_{\mathsf{Ncut}} &= \frac{s(A,B)}{s(A,V)} + \frac{s(B,A)}{s(B,V)} \\
&= 2 - \frac{s(A,A)}{s(A,V)} - \frac{s(B,B)}{s(B,V)}
\end{aligned}
$$

$v_A = s(A,V)$: sum of weights on edges originating from set A (Volume)

- $\min J_{\mathsf{Ncut}}$ is equivalent to
  - minimize normalized between-cluster weights;
  - maximize normalized within-cluster weights.

- Partition membership indicator

$$q_i = \sqrt{\frac{v_B}{v_A}\frac{1}{v}}, \quad i \in A; \quad q_i = -\sqrt{\frac{v_A}{v_B}\frac{1}{v}}, \quad i \in B$$

where $v = s(V, V)$. $(q^t D q = 1, q^t D \mathbf{1} = 0)$

- Objective function

$$J_{\text{Ncut}} = \sum_{i,j} w_{ij}(q_i - q_j)^2 = \frac{1}{2} q^t (D - W) q.$$

Solve q by the generalized eigenvalue problem (2nd eigenvector)

$$(D - W) q = \lambda D q,$$

or

$$(I - D^{-1} W) q = \lambda q.$$

Multi-partition ($K > 2$)

- Recursive bi-partition
- Use multi-partition objective functions
- Spectral clustering
  - Embed data in a subspace of eigenvectors
  - Cluster embedded data points using another algorithm, such as K-means.

Spectral Clustering

- Define

$$D = \text{diag}(W_{i+}), \quad L = I - D^{-1}W$$

where L is known as the (normalized) graph Laplacian.

- The first m (non-constant) eigen-vectors (corresponding to the lowest eigen-values) of L, $X_1, \ldots, X_k$, form a new data matrix

$$\mathbf{X}_{n\times k} = [X_1, \ldots, X_m],$$

i.e., each node corresponds to a data point in $\mathbb{R}^m$.

- Run k-means based on the data matrix $\mathbf{X}$.

- Next, we justify the use of spectral clustering via the so-called Stochastic Blockmodel.

- Stochastic Blockmodel, a probabilistic model for graphs, assumes a grouping structure among the nodes. We can show that if we run spectral clustering on the corresponding W matrix, we can correctly retrieve the underlying grouping structure.

Statistical Models for Graphs/Networks

- Node set V is given; $\mathcal{W}$ is a stochastic binary matrix.
- The simplest (one parameter) model (aka Erdos-Renyi model)

$$\mathbb{P}(\mathcal{W}_{ij} = 1) = p.$$

- The most complicated (many parameters) model

$$\mathbb{P}(\mathcal{W}_{ij} = 1) = w_{ij}.$$

- Something in the middle: cluster w_{ij}'s.

Stochastic Blockmodel (Holland et al., 1983)

- Suppose n nodes are from k groups/blocks.
- $B_{k\times k}$: connecting probability between nodes from the k blocks.
- Conditioning on node i in l-th block and node j in h-th block,

$$\mathbb{P}(\mathcal{W}_{ij} = 1) = w_{ij} = B_{lh}.$$

- Block/cluster membership $\mathbf{z}_i \in (0,1)^k$:

$$\text{node } i \text{ in } l\text{th block, iff } z_{ih} = 0, \text{ except } z_{il} = 1.$$

  Stack all the $\mathbf{z}_i$'s as a $n \times k$ matrix $\mathbf{Z}$. Then, the stochastic blockmodel implies

$$W_{n\times n} = \mathbf{Z}_{n\times k} B \mathbf{Z}^T.$$

- Each row of $\mathbf{Z}$ has one and only one value equal to 1 and all others 0.

Spectral Clustering at the Population Level

- Assume $W = \mathbf{Z}B\mathbf{Z}^T$ is known. Define the population laplacian L:

$$\begin{aligned} D_{ii} &= \sum_k W_{ik} \\ L &= D^{-1/2} W D^{-1/2} \end{aligned}$$

- It can be shown that the top k eigen-vectors of L are equal to $\mathbf{Z}\mu$ where μ is a $k \times k$ matrix. Therefore spectral clustering can retrieve the k clusters correctly[a].

---

[a] Lemma 3.1 from "Spectral clustering and the high-dimensional stochastic block-model" by Karl Rohe, Sourav Chatterjee, & Bin Yu.

1) Show that L can be re-expressed as $L = (\mathbf{Z} B_L \mathbf{Z}^T)$ where

$$\begin{aligned} D_B &= \text{diag}(B\mathbf{Z}^T\mathbf{1}_n) \in \mathbb{R}^{k\times k} \\ B_L &= D_B^{-1/2} B D_B^{-1/2} \end{aligned}$$

2) Assume $(\mathbf{Z}^T\mathbf{Z})^{1/2} B_L (\mathbf{Z}^T\mathbf{Z})^{1/2}$ has the following SVD,

$$(\mathbf{Z}^T\mathbf{Z})^{1/2} B_L (\mathbf{Z}^T\mathbf{Z})^{1/2} = V\Gamma V^T.$$

3) Left-multiply $\mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1/2}$ and right-multiply its transpose on both sides,

$$\mathbf{Z} B_L \mathbf{Z}^T = \mathbf{Z}\mu\Gamma(\mathbf{Z}\mu)^T,$$

where $\mu = (\mathbf{Z}^T\mathbf{Z})^{-1/2}V$. Note that $(\mathbf{Z}\mu)^T(\mathbf{Z}\mu) = \mathbf{I}_k$. So $\mathbf{Z}\mu$ are the eigen-vectors.