# CSI 436/536
# Introduction to Machine Learning

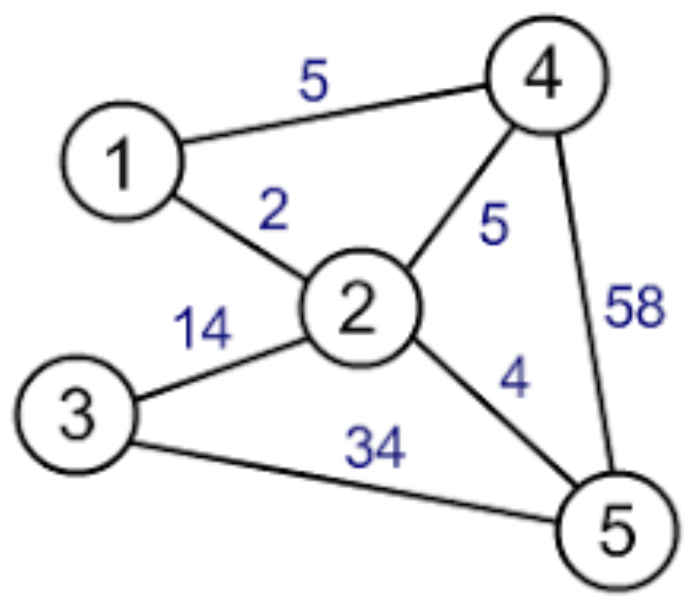
## **Spectral clustering**

Professor Siwei Lyu
Computer Science
University at Albany, State University of New York

# Spectral clustering

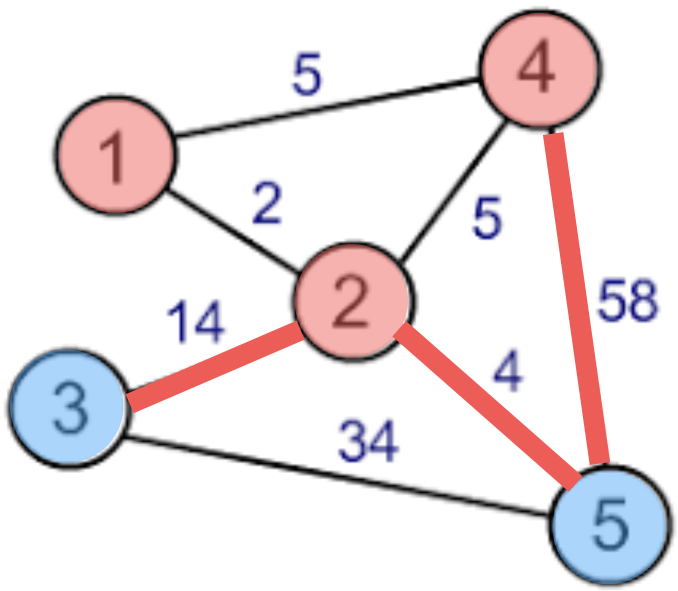
- K-means clustering focuses on the closeness of elements **within** the same cluster
- Spectral clustering focuses on distinctiveness of elements **across** different clusters
  - represents relation between data using an undirected weighted graph (similarity graph)
  - weights on the graph correspond to data similarities

$$W_{ij} = \exp(-d(x_i, x_j)^2/\sigma^2)$$



# Spectral clustering for two clusters

- use a binary indicator $v_i$ for each vertex, $v_i = 1$ if vertex i is in cluster 1, $v_i = 0$ if vertex i is in cluster 2
- for any edge connecting vertex i and vertex j, $W_{ij}(v_i - v_j)^2$ measures the cost of putting them into different cluster

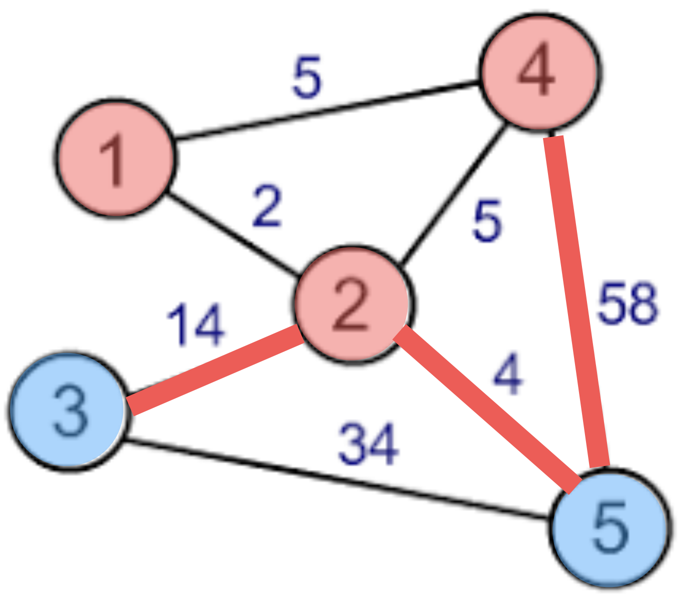


# Spectral clustering for two clusters

- total cost of a bi-section (cut) of the graph is then

$$cut(C_1, C_2) = \frac{1}{2}\sum_{i,j} W_{ij}(v_i - v_j)^2$$

  - two vertices in the same cluster has no cost
  - we aim to minimize this cost by searching optimal assignments for $v_i$
  - this is a NP-hard problem if solved precisely



# Spectral clustering for two clusters

- Expand the min-cut cost

$$\frac{1}{2}\sum_{i,j} W_{ij}(v_i - v_j)^2 = \frac{1}{2}\left(\sum_{ij} W_{ij}v_i^2 - 2\sum_{ij} W_{ij}v_iv_j + \sum_{ij} W_{ij}v_j^2\right) = \sum_i v_i^2 \sum_j W_{ij} - \sum_{ij} W_{ij}v_iv_j$$

- Introduce $v = (v_1, \cdots, v_n)^\top$ and a diagonal matrix

$D = diag(W1)$ as $D_{ii} = \sum_j W_{ij}$

- The min-cut cost becomes $\min_v v^\top L v$, s.t. $v_i \in \{0,1\}$
  - where L = D - W is the graph Laplacian matrix
  - an integer-programming problem
  - we find ***approximate*** solution by ***relaxation***

# Graph Laplacian

- Definition $L = diag(W1) - W$ where 1 is the all one vector, it has the following properties
  - L is symmetric and positive definite
  - any constant vector is an eigenvector with eigenvalue zero
- Graph Laplacian can be understood as the differential operator for functions on a graph
  - It is a very useful tool for graph data analysis
  - # of zero eigenvalues = # of connected components in a graph
  - smallest non-zero eigenvalue is known as the Fiedler number of the graph (spectral gap)

# Relaxation of the min-cut problem

- Original problem $\min_{v} v^\top L v, \quad \text{s.t.} \quad v_i \in \{0,1\}$ is intractable (exponential number of possible solutions)
- Approximation by relaxation
  - Solve $\min_{v} v^\top L v, \quad \text{s.t.} \quad \|v\| = 1, v \neq 1$
  - Thresholding the obtained v into binary vector
  - The approximate solution is an upper-bound of the actual objective

# Solving the relaxed objective

- This is a constrained optimization

$$\min_v v^\top L v, \quad \text{s.t.} \quad \|v\| = 1, v \neq 1$$

- Solve it by introducing Lagrangian multiplier

$$0 = \frac{\partial}{\partial v}(v^\top L v - 2\lambda(v^\top v - 1)) \Rightarrow Lv = \lambda v$$

- So optimal solution is necessarily an eigenvector of matrix L
- Selecting the eigenvector corresponding to the smallest non-zero eigenvalue (Fiedler number) to be the optimal v