CSE 176 Introduction to Machine Learning

Lecture 7: Graph Clustering

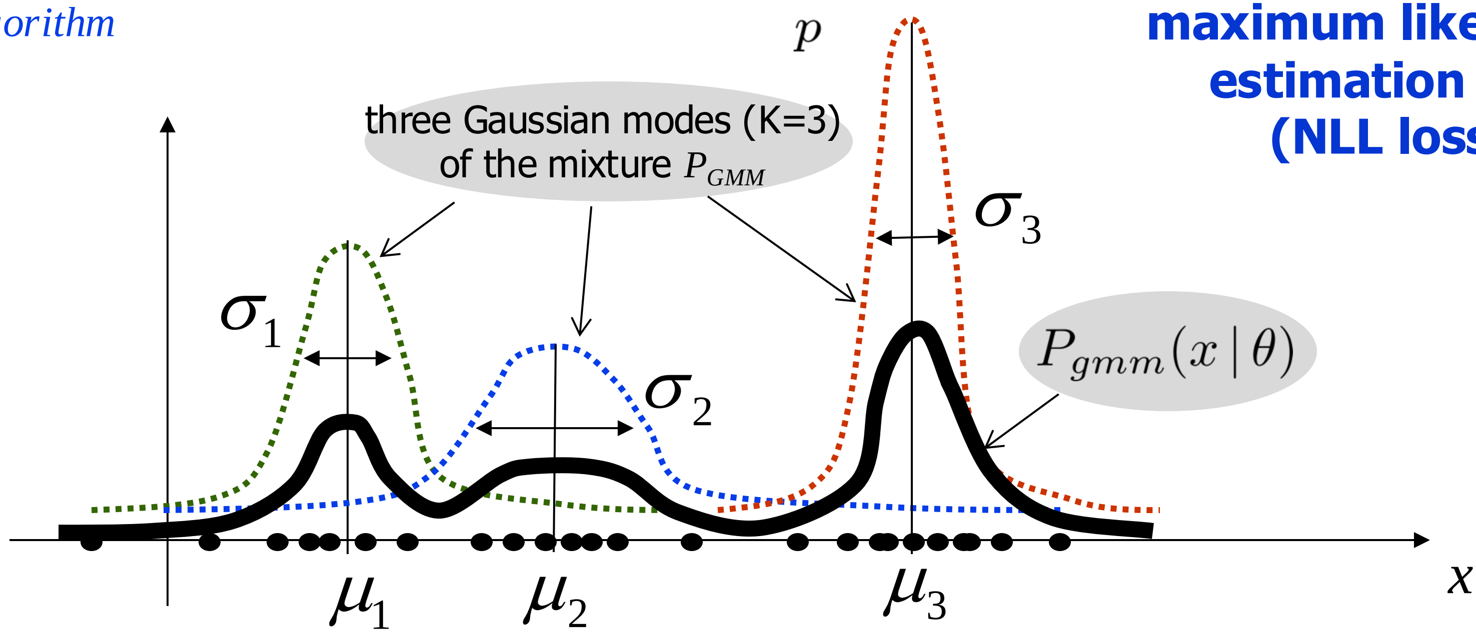
Some slides from Ahmed Elgammal and Gopalkrishna Veni

Recap: Gaussian Mixture Models (GMM)

approximate optimization via *EM algorithm*

$$E_{gmm}(\theta) \;:=\; -\sum_p \log P_{gmm}(x_p \,|\, \theta)$$

maximum likelihood estimation of θ (NLL loss)

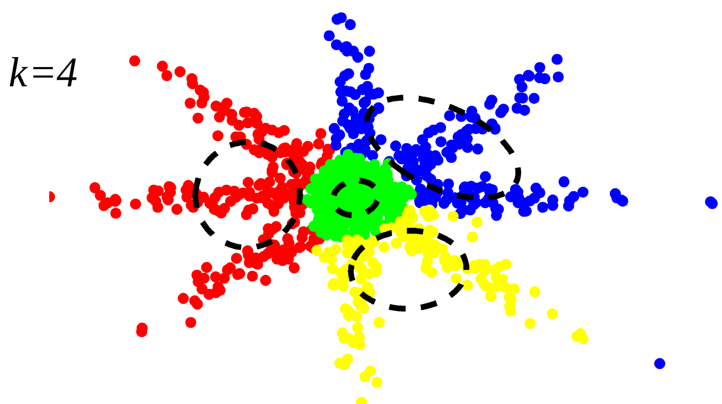


GMM distribution: $P_{gmm}(x \,|\, \theta) \;:=\; \sum_k \rho_k \, P(x \,|\, \mu_k, \sigma_k)$

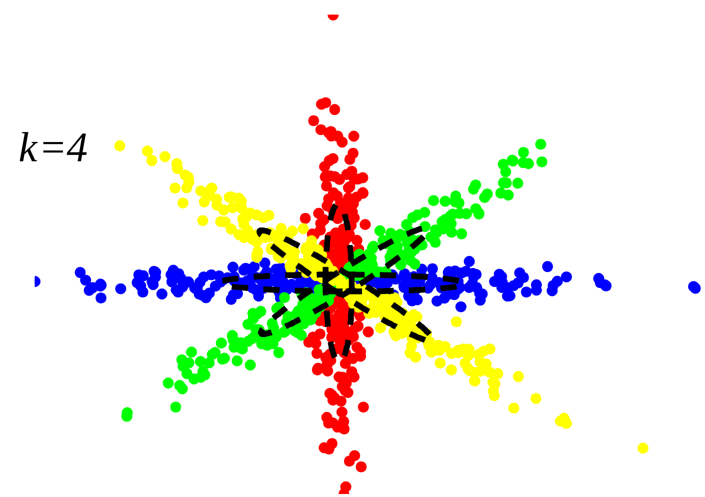
Recap: Gaussian clusters/modes in:

(basic) K-means vs. GMM (or fuzzy K-means)

- *hard* assignment to clusters
  - separates data points into multiple Gaussian blobs
- only estimates means μ_i
  - Σ_i can also be added as a cluster parameter (*elliptic K-means*)



- *soft* mode searching
  - estimates data distribution with multiple Gaussian modes
- estimates both mean μ_i and (co)variance Σ_i for each mode



Today’s topics

- ❑Graph representation
- ❑Normalized Cut
- ❑Graph Clustering and (kernel) K-means

Graph Representation

Real-world graphs

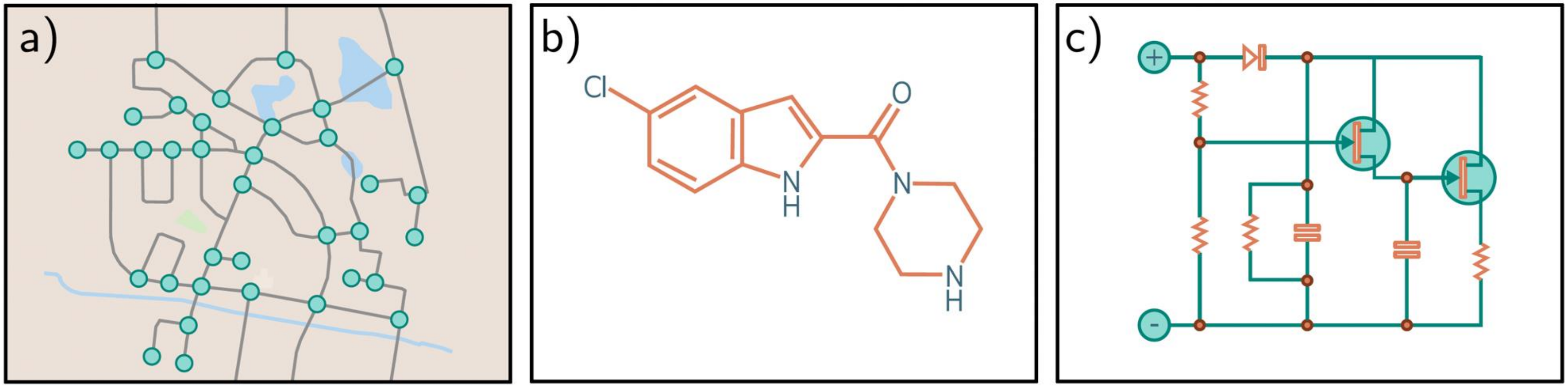
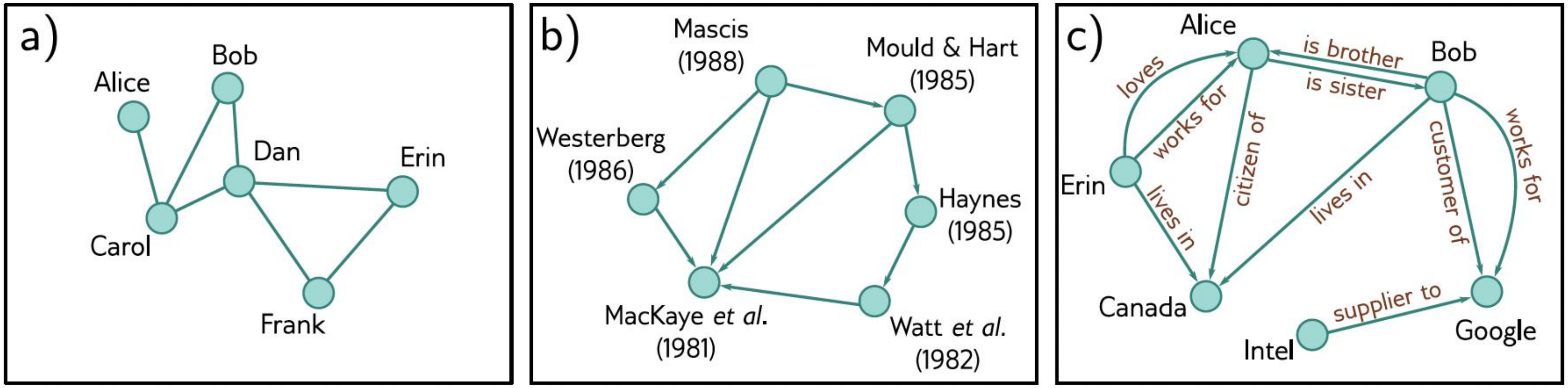


Figure 13.1 Real-world graphs. Some objects, such as a) road networks, b) molecules, and c) electrical circuits, are naturally structured as graphs.

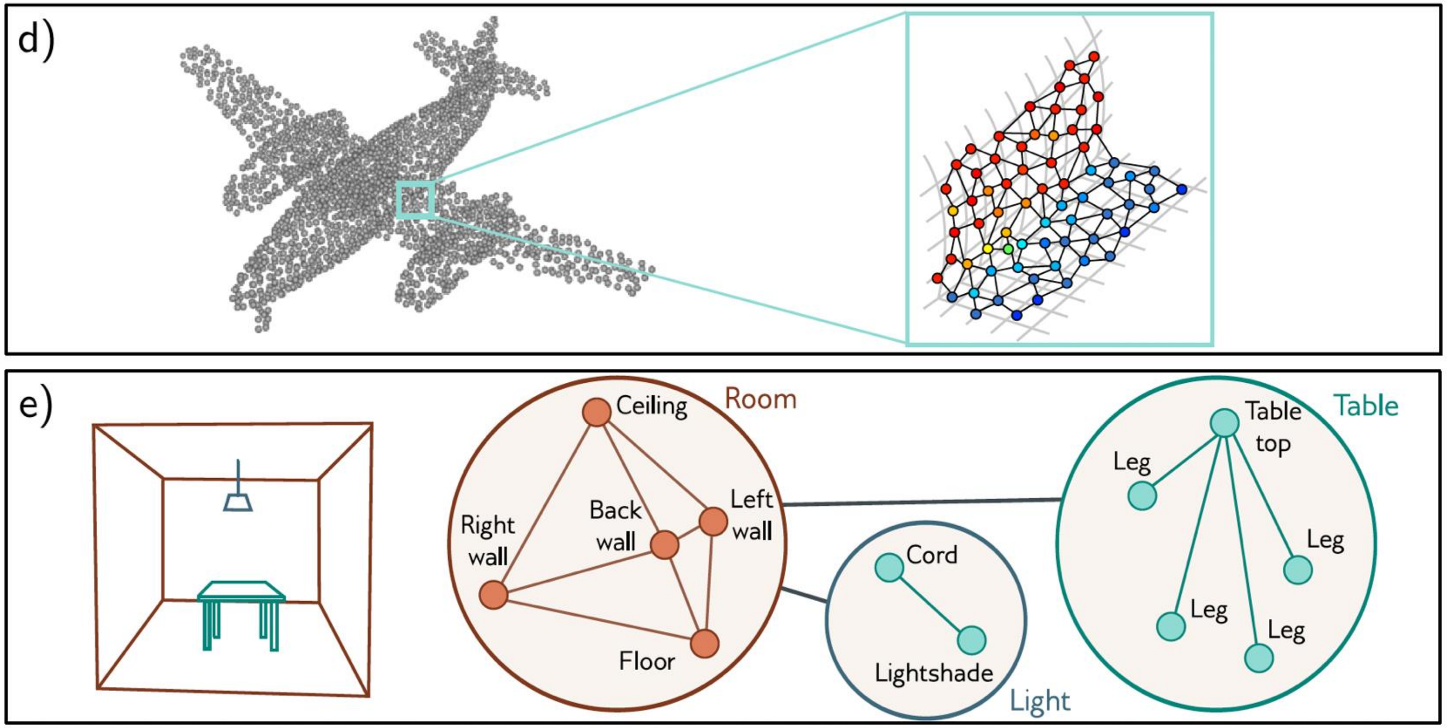
Types of graphs

- a) social network is an undirected graph
- b) citation network is a directed graph
- c) Knowledge graph is a directed heterogeneous multigraph



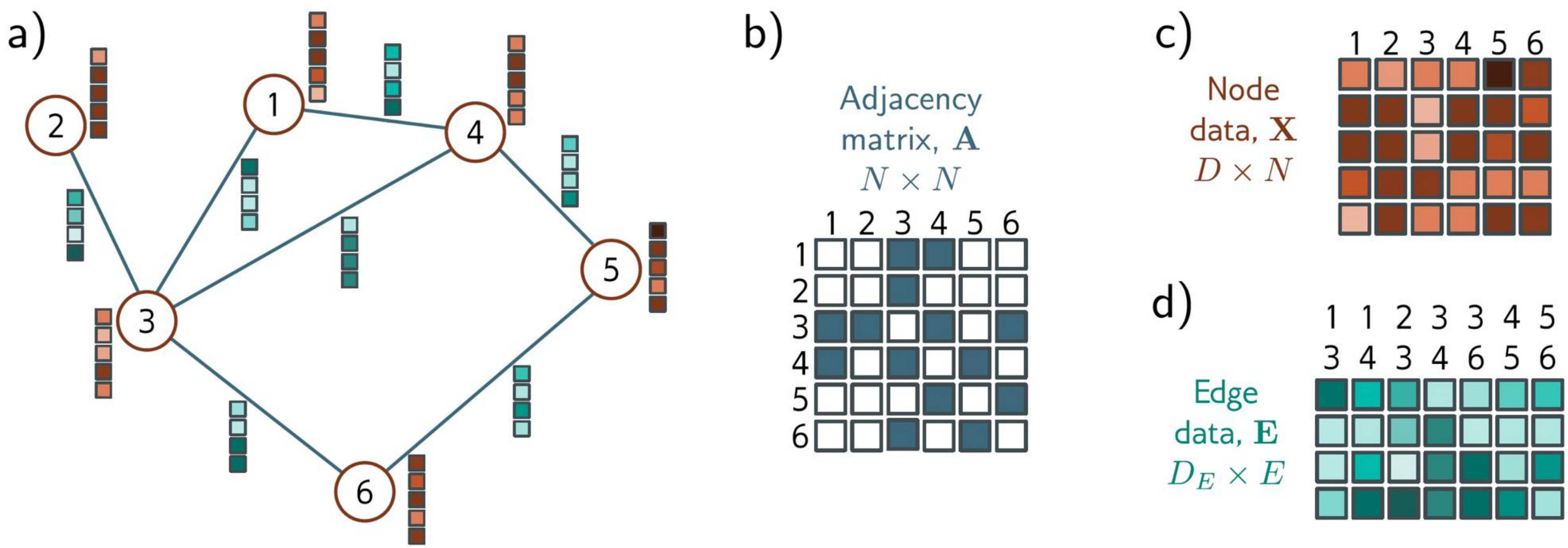
Types of graphs

- d) point cloud as a geometric graph
- e) Scene graph is hierarchical



Graph representation

- A graph is defined as a tuple G = (V, E)
  - where V is a set of nodes
  - and E is a set of edges
- An example graph with 6 nodes and 7 edges



Representing an image as a graph

- A vertex for each pixel
- Edges between pixels
- Weights on edges reflect similarity (affinity) in:
  - Brightness
  - Color
  - Texture
  - Distance
  - …
- Connectivity:
  - Fully connected: edges between every pair of pixels
  - Partially connected: edges between neighboring pixels

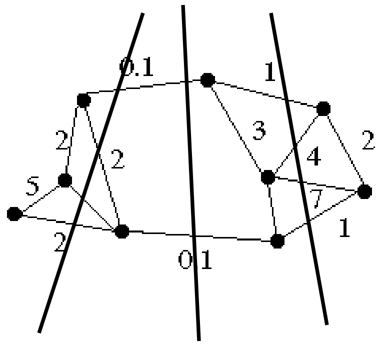
Normalized Cut

Normalized Cut for Image Segmentation

$$w_{ij} = e^{\frac{-\|\boldsymbol{F}_{(i)} - \boldsymbol{F}_{(j)}\|_2^2}{\sigma_I^2}} *
\begin{cases} e^{\frac{-\|\boldsymbol{X}_{(i)} - \boldsymbol{X}_{(j)}\|_2^2}{\sigma_X^2}} & \text{if } \|\boldsymbol{X}(i) - \boldsymbol{X}(j)\|_2 < r \\ 0 & \text{otherwise.} \end{cases}$$

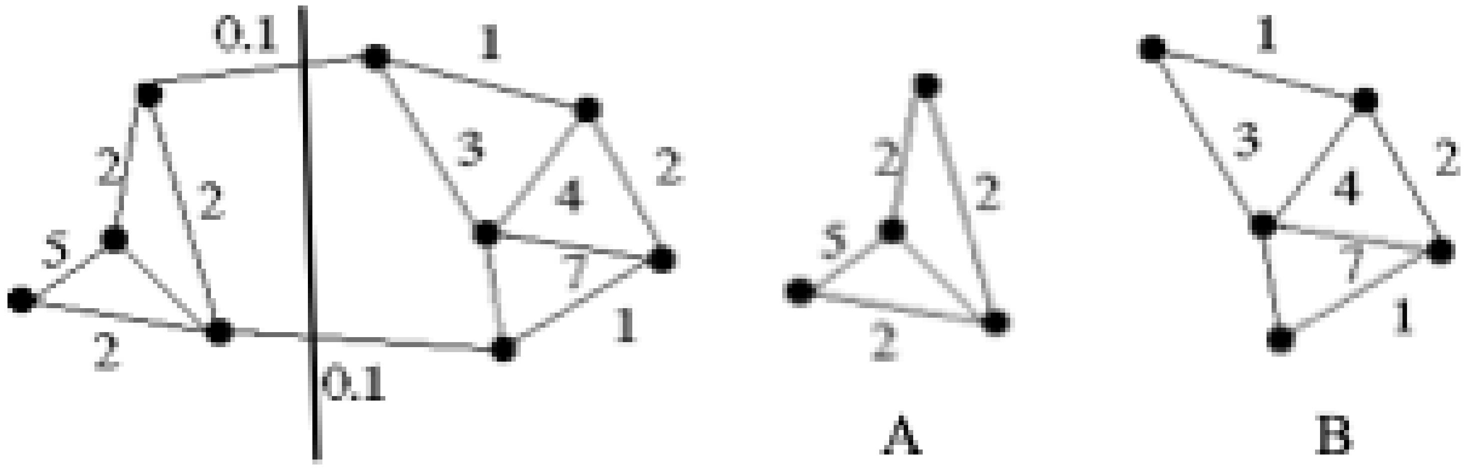
What is Graph Cut?

- Remove a subset of edges to partition the graph into two disjoint sets of vertices *A*,*B* (two sub graphs):
- $A \cup B = V, A \cap B = \Phi$



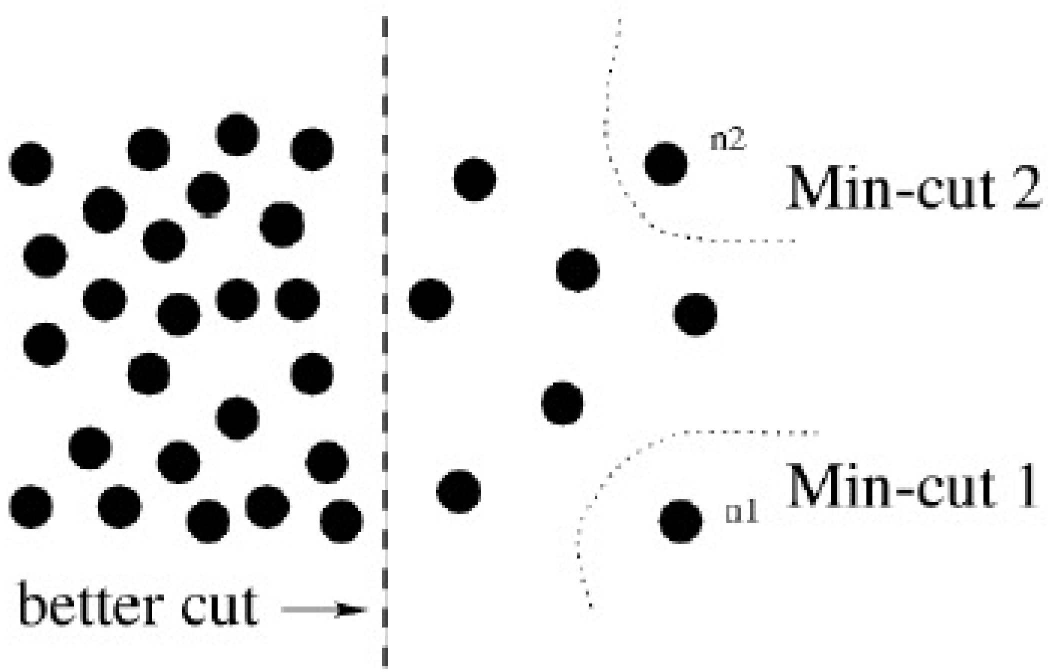
Minimum Cut

- In many applications it is desired to find the cut with minimum cost: *minimum cut*
- Well studied problem in graph theory, with many applications
- There exists efficient algorithms for finding minimum cuts



$$cut(A,B) = \sum_{u \in A, v \in B} w(u,v)$$

Mincut is not always the best cut



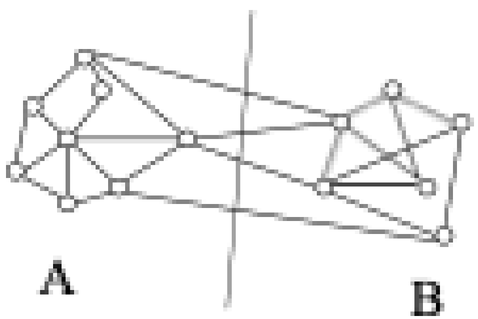
Association between sets

- Consider two sets A and B

$$assoc(A,B) = \sum_{u \in A, t \in B} w(u,t)$$

$$assoc(A,V) = \sum_{u \in A, t \in V} w(u,t)$$

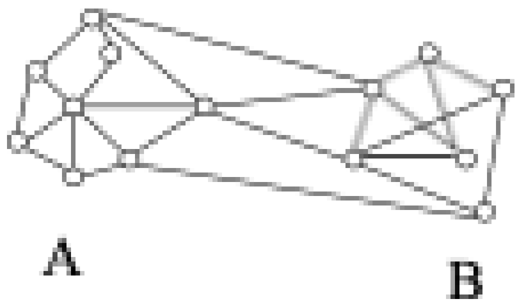
$$assoc(B,V) = \sum_{u \in B, t \in V} w(u,t)$$

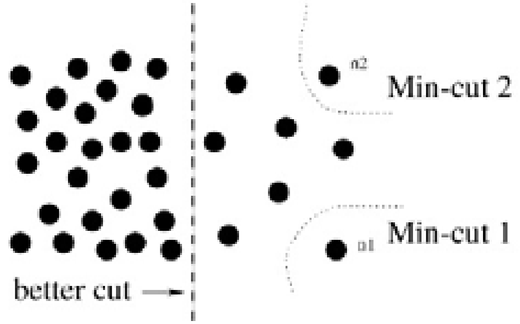


Normalized Cut

- Normalize *cut cost* by volume of clusters
- Compute the cut cost as a fraction of the total edge connections to all nodes in the graph

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$





Computation of optimum partition using $_{min}N_{cut}$

$$Ncut(A,B) = \frac{cut(A,B)}{asso(A,V)} + \frac{cut(B,A)}{asso(B,V)} = \frac{\sum_{(\boldsymbol{x}_i>0,\boldsymbol{x}_j<0)} -w_{ij}\boldsymbol{x}_i\boldsymbol{x}_j}{\sum_{\boldsymbol{x}_i>0} \boldsymbol{d}_i} + \frac{\sum_{(\boldsymbol{x}_i<0,\boldsymbol{x}_j>0)} -w_{ij}\boldsymbol{x}_i\boldsymbol{x}_j}{\sum_{\boldsymbol{x}_i<0} \boldsymbol{d}_i}$$

where, x is an N dimensional indicator vector such that x_i=1 if 'i' is in A and -1 if 'i' is in B. $d(i) = \sum_j w(i,j)$, total , total connection from node i to all other nodes.

Let D be an N x N diagonal matrix, W be an N x N symmetric matrix with W(i,j) = w_{ij} , $k = \frac{\sum_{x_i>0} d_i}{\sum_i d_i}$, and 1 be an N x 1 vector of all ones.

$$4[Ncut(\boldsymbol{x})] = \frac{(\mathbf{1}+\boldsymbol{x})^T(\mathbf{D}-\mathbf{W})(\mathbf{1}+\boldsymbol{x})}{k\mathbf{1}^T\mathbf{D}\mathbf{1}} + \frac{(\mathbf{1}-\boldsymbol{x})^T(\mathbf{D}-\mathbf{W})(\mathbf{1}-\boldsymbol{x})}{(1-k)\mathbf{1}^T\mathbf{D}\mathbf{1}}$$

Derivations

Derivations

let us consider there are 3 nodes such that

2 nodes in group A

1 node in group B

so,

$$\begin{array}{cccc} & A & A & B \\ x = & [\,1 & 1 & -1\,] \end{array}$$

$$\sum_{(x_i>0,\, x_j<0)} -w_{ij}\, x_i x_j \Rightarrow -w_{13}x_1x_3 = w_{13}$$

$$-w_{23}x_2x_3 = w_{23}$$

$$4N_{cut}\,(1^{st}\text{ part numerator}) = 4(w_{13}+w_{23}) \text{ ——— } (1)$$

$$(1+x)^T(D-W)(1+x) \Rightarrow [2\;\;2\;\;0]\begin{bmatrix} w_{12}+w_{13} & -w_{12} & -w_{13} \\ -w_{21} & w_{21}+w_{23} & -w_{23} \\ -w_{31} & -w_{32} & w_{31}+w_{32} \end{bmatrix}\begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix}$$

$$= 4[w_{13}+w_{23}] \text{ ——— } (2)$$

$$\underline{(1) = (2)}$$

$$k = \frac{\sum_{x_i>0} d_i}{\sum_i d_i} = \frac{w_{11}+w_{21}+w_{22}+w_{13}+w_{23}+w_{12}}{w_{11}+w_{12}+w_{13}+w_{21}+w_{22}+w_{23}+w_{31}+w_{32}+w_{33}}$$

$$k 1^T D 1 = \underline{w_{11}+w_{21}+w_{22}+w_{13}+w_{23}+w_{12}} \;(\qquad)$$

$$= w_{11}+w_{21}+w_{22}+w_{13}+w_{23}+w_{12} = \underline{\sum_{x_i>0} d_i}\;[\text{denominator part}]$$

Computation of optimum partition using $_{min}N_{cut}$

- Letting y=(1+x)-b(1-x),
- Solution:

$$min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\boldsymbol{D}-\boldsymbol{W})\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{D}\boldsymbol{y}}$$

with the condition $y^T D1 = 0$

$$D = \begin{bmatrix} \sum_j w(1,j) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_j w(N,j) \end{bmatrix}, \quad W = \begin{bmatrix} w(1,1) & \cdots & w(1,N) \\ \vdots & \ddots & \vdots \\ w(N,1) & \cdots & w(N,N) \end{bmatrix}$$

Derivations

$$y^T D 1 = \sum_{x_i>0} d_i - b \sum_{x_i<0} d_i = 0$$

$$b = \frac{k}{1-k} = \frac{w_{11}+w_{12}+w_{21}+w_{22}+w_{13}+w_{23}}{w_{31}+w_{32}+w_{33}}$$

$$\sum_{x_i>0} d_i = w_{11}+w_{12}+w_{13}+w_{22}+w_{23}+w_{21}$$

$$\cancel{\therefore y^T D 1 =} \qquad \sum_{x_i<0} d_i = w_{31}+w_{32}+w_{33}$$

$$\therefore y^T D 1 = w_{11}+w_{12}+w_{13}+w_{22}+w_{23}+w_{21} - \left(\frac{\quad}{\quad}\right)(w_{31}+w_{32}+w_{33})$$

$$= \underline{0}$$

$(D-w)y = \lambda D y$ [Generalized Eigen System]

$(D-w)D^{-1/2}z = \lambda D D^{-1/2} z$ (since $y = D^{-1/2} z$)

$D^{-1/2}(D-w)D^{-1/2} z = \lambda D^{-1/2} D D^{-1/2} z$

$\underline{D^{-1/2}(D-w)D^{-1/2} z = \lambda z}$ [Standard Eigen System]

Summary of Normalized cut algorithm

- Given a set of features, construct a weighted graph by computing weight on each edge and then placing the data into W and D.
- Solve $(D-W)x=\lambda Dx$ for eigen vectors with the smallest eigenvalues.
- Use the eigen vector corresponding to the second smallest eigenvalue to bipartition the graph into two groups.
- Recursively repartition the segmented parts if necessary.

Normalized Cut for Image Segmentation

$$w_{ij} = e^{\frac{-\|\boldsymbol{F}_{(i)} - \boldsymbol{F}_{(j)}\|_2^2}{\sigma_I^2}} *$$

$$\begin{cases} e^{\frac{-\|\boldsymbol{X}_{(i)} - \boldsymbol{X}_{(j)}\|_2^2}{\sigma_X^2}} & \text{if } \|\boldsymbol{X}(i) - \boldsymbol{X}(j)\|_2 < r \\ 0 & \text{otherwise.} \end{cases}$$

Normalized Cut for Image Segmentation

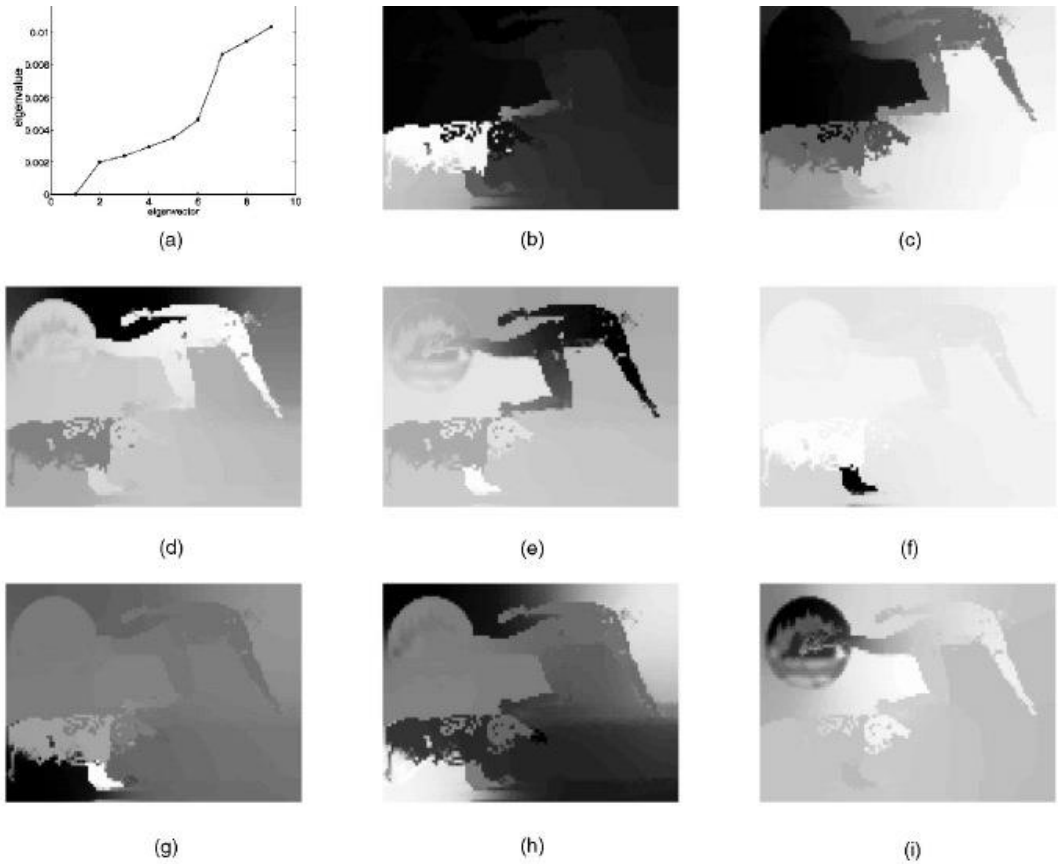


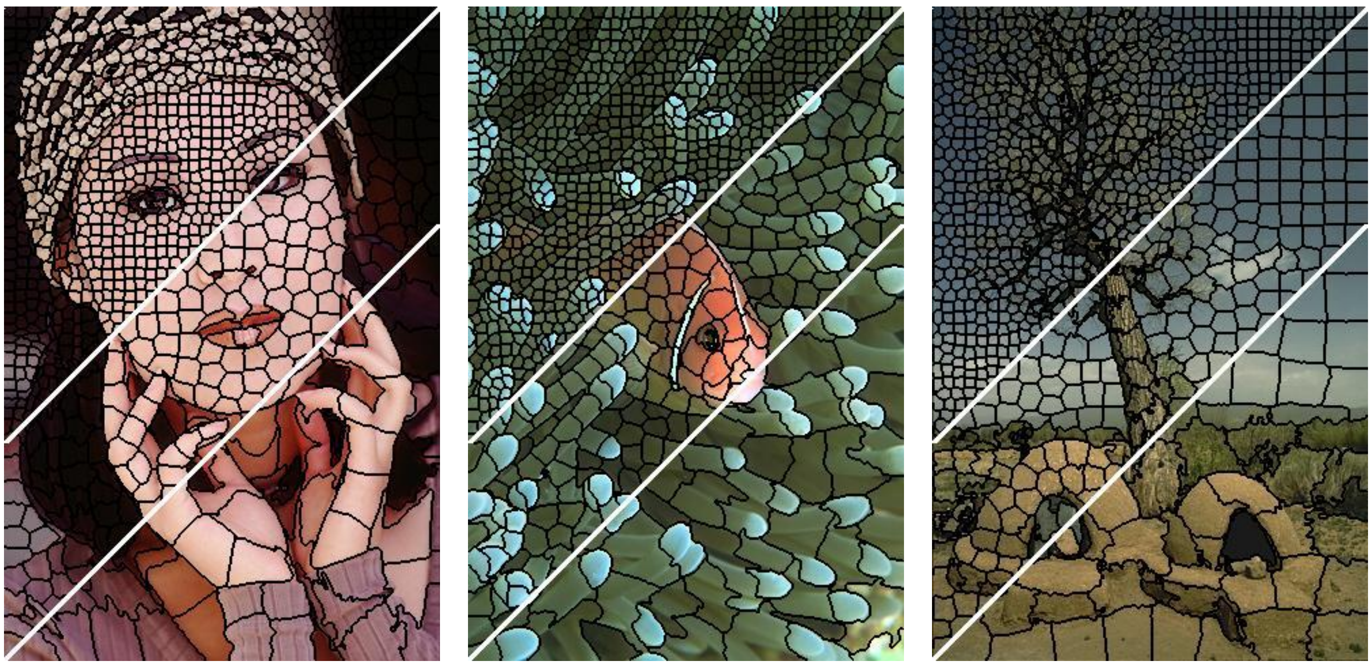
Figure from "Normalized cuts and image segmentation," Shi and Malik, copyright IEEE, 2000

Graph Clustering and (Kernel) K-means

Basic K-means examples: Superpixels

- Apply K-means to RGBXY features

[SLIC superpixels, Achanta et al., PAMI 2011]



“squared distance” as “log-likelihoods”

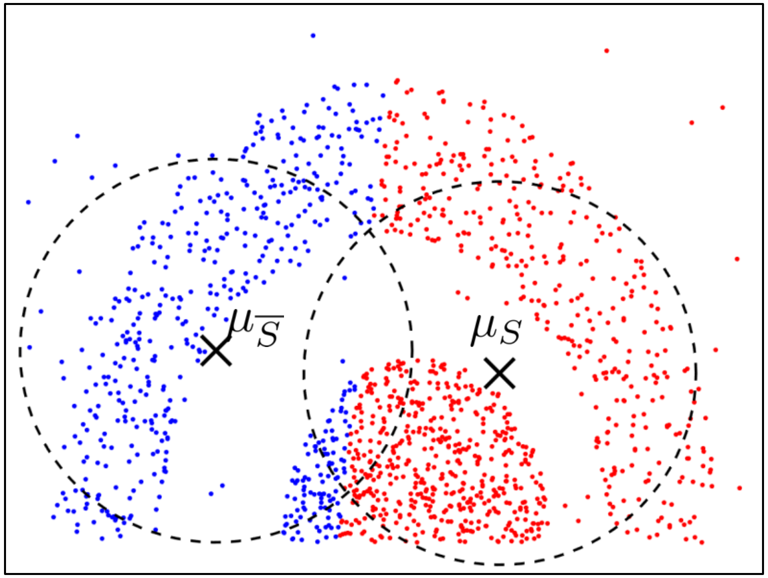
Assume K=2, $\Omega = S \cup \bar{S}$

$$\boxed{\sum_{p \in S} \|f_p - \mu_S\|^2 + \sum_{p \in \overline{S}} \|f_p - \mu_{\overline{S}}\|^2}$$

$$= \quad -\sum_{p \in \mathbf{S}} \ln \underline{\mathcal{N}(f_p|\mu_{\mathbf{S}})} \quad - \quad \sum_{p \in \bar{\mathbf{S}}} \ln \mathcal{N}(f_p|\mu_{\bar{\mathbf{S}}})$$

single Gaussian

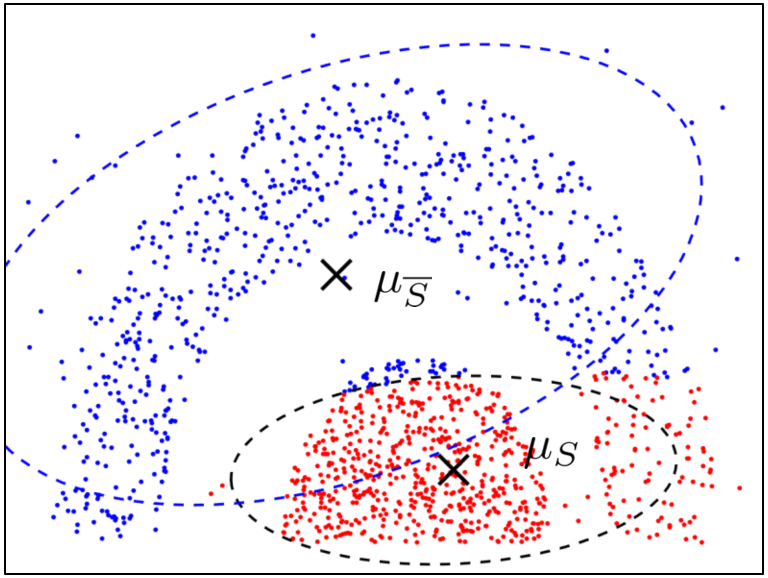
single Gaussian of ***fixed*** covariance



$$\theta_S = \{\mu_S\}$$

$$-\sum_{p \in S} \ln \Pr(f_p | \theta_S) - \sum_{p \in \bar{S}} \ln \Pr(f_p | \theta_{\bar{S}})$$

single Gaussian of ***variable*** covariance



$$\theta_S = \{\mu_S, \sigma_S\}$$

<u>Probabilistic</u> K-means with Descriptive Models

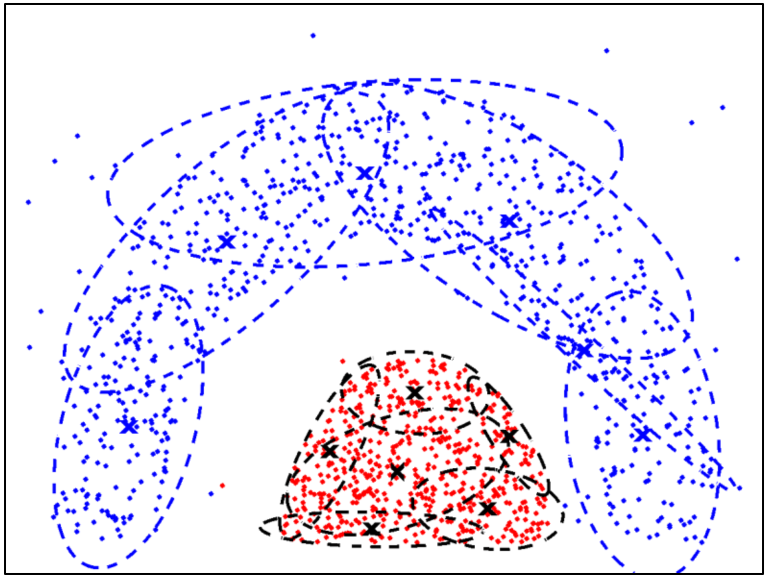
$$\theta$$

$$-\sum_{p \in S} \ln \Pr(f_p|\theta_S) - \sum_{p \in \bar{S}} \ln \Pr(f_p|\theta_{\bar{S}})$$

[Kearns, Mansour & Ng, UAI'97]

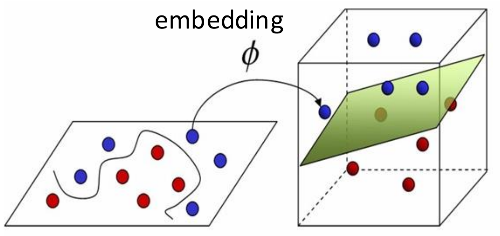
Examples of $\Pr(\cdot|\theta)$:
Normal, gamma, exponential, Gibbs, etc.

Gaussian Mixture Models (GMM)



$$\theta_S = \{..., \mu_S^i, \sigma_S^i, ...\}$$

Toward Kernel K-means



$$E(S, \mu) = \sum_{k=1}^{K} \sum_{p \in S^k} \| f_p - \mu_k \|^2$$

(Basic K-means)

$$E_k(S, \hat{\mu}) = \sum_{k=1}^{K} \sum_{p \in S^k} \| \phi(f_p) - \hat{\mu}_k \|^2$$

(Kernel K-means)

Explicit Kernel ⇔ *Implicit Embedding*

$$E_k(S,\hat{\mu}) = \sum_{k=1}^{K} \sum_{p \in S^k} \|\phi(f_p) - \hat{\mu}_k\|^2$$

just plug-in

$$\hat{\mu}_k = \frac{1}{|S^k|} \sum_{q \in S^k} \phi(f_q)$$

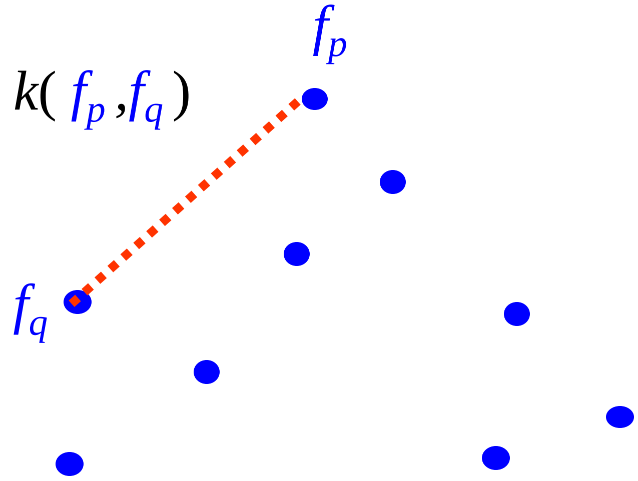
equivalent

$$E_k(S) = -\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} k(f_p, f_q)}{|S^k|}$$

kernel K-means

$$-\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} k(f_p, f_q)}{|S^k|}$$ - objective

$$A_{pq} = k(f_p, f_q)$$

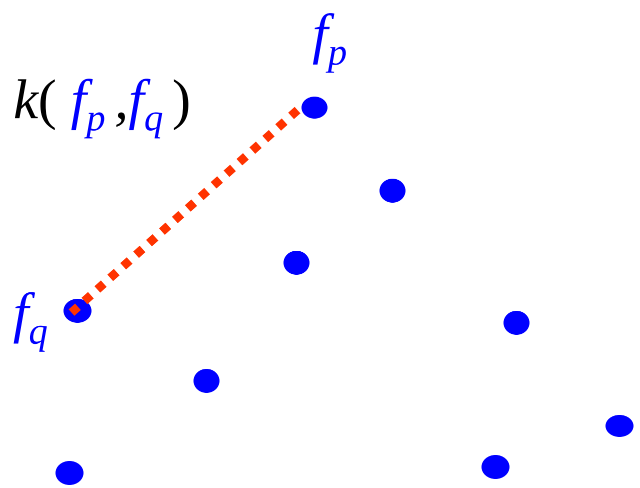


kernel K-means

$$-\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} A_{pq}}{|S^k|}$$

- objective

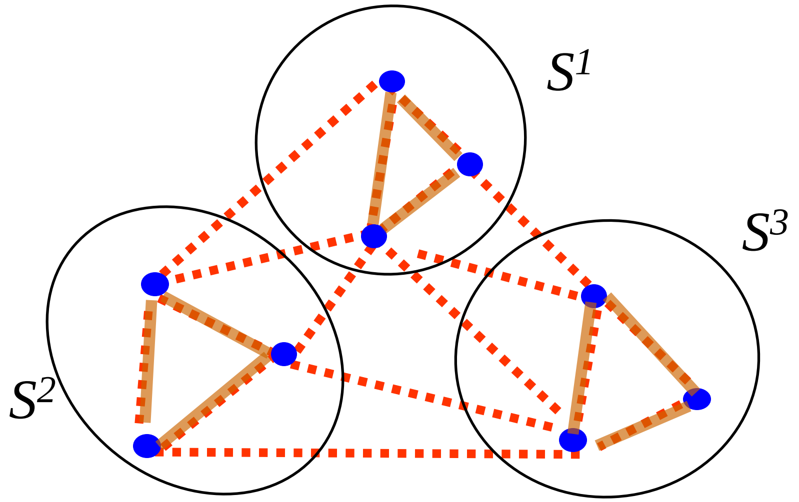
$$A_{pq} = k(f_p, f_q)$$



kernel K-means or ***average association***

$$E(S) = -\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} A_{pq}}{|S^k|}$$

"self-association" of cluster S^k



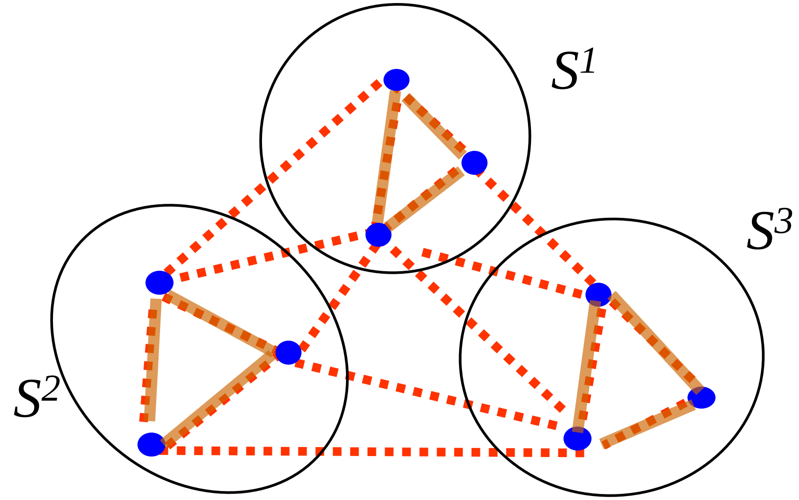
kernel K-means or ***average association***

$$E(S) = -\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} A_{pq}}{|S^k|} \quad \sum_{pq \in S_k} A_{pq} = {S^k}' A S^k$$

in matrix notation:

S^k - indicator vector

' means *transpose*



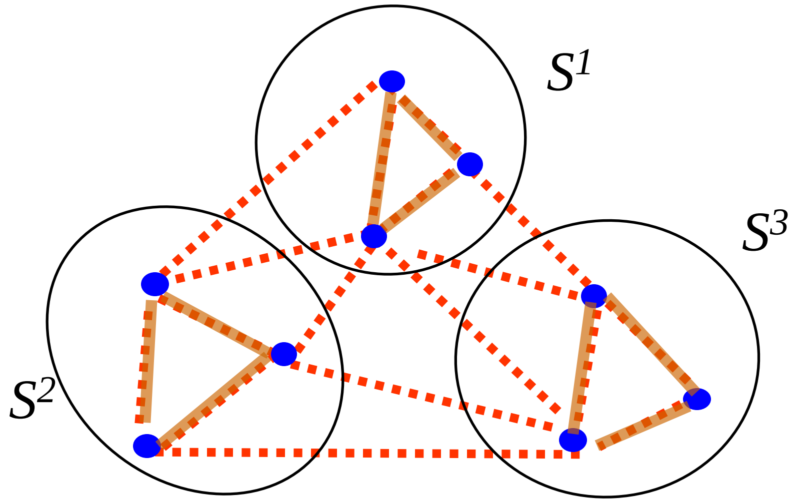
kernel K-means or ***average association***

$$E(S) = -\sum_{k=1}^{K} \frac{\phantom{S^{k\prime} A S^k}}{|S^k|} \qquad S^{k\prime} A S^k$$

in matrix notation:

S^k - indicator vector

‘ means *transpose*



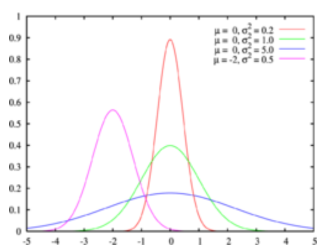
K-means

$$\boxed{\sum_{p\in\mathbf{S}} \|f_p - \mu_{\mathbf{S}}\|^2 + \sum_{p\in\overline{\mathbf{S}}} \|f_p - \mu_{\overline{\mathbf{S}}}\|^2}$$

probabilistic K-means

make models more complex

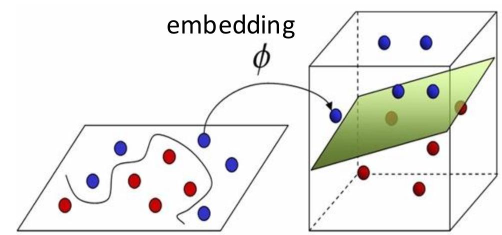
$$-\sum_{p\in\mathbf{S}} \ln \Pr(f_p|\theta_{\mathbf{S}}) - \sum_{p\in\bar{\mathbf{S}}} \ln \Pr(f_p|\theta_{\bar{\mathbf{S}}})$$



kernel K-means

make data more complex

$$\sum_{p\in\mathbf{S}} \|\phi(f_p) - \hat{\mu}_{\mathbf{S}}\|^2 + \sum_{p\in\overline{\mathbf{S}}} \|\phi(f_p) - \hat{\mu}_{\overline{\mathbf{S}}}\|^2$$



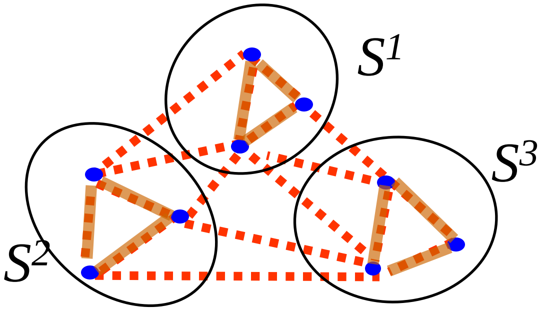
input space feature space

Other kernel (graph) clustering objectives

Average Association

“self-association” for S^k

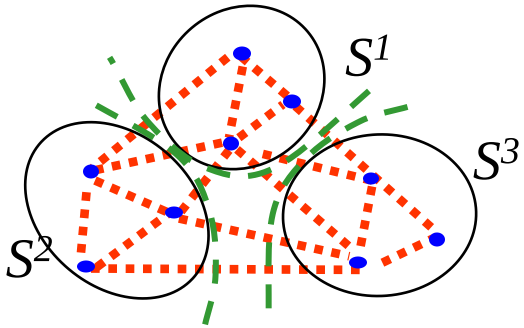
$$-\sum_{k=1}^{K} \frac{S^{k\prime} A\, S^k}{|S^k|}$$



Average Cut

“cut” for S^k

$$\sum_{k=1}^{K} \frac{S^{k\prime} A\, (\mathbf{1} - S^k)}{|S^k|}$$



kernel (graph) clustering objectives

- **Average Cut**
- **Average Association**

$$\sum_{k=1}^{K} \frac{{S^k}' A \, (\mathbf{1} - S^k)}{\mathbf{1}' S^k} \quad \not\Leftrightarrow \quad -\sum_{k=1}^{K} \frac{{S^k}' A \, S^k}{\mathbf{1}' S^k}$$

- **Normalized Cut**
- **Normalized Association**

$$\sum_{k=1}^{K} \frac{{S^k}' A \, (\mathbf{1} - S^k)}{d' S^k} \quad \equiv \quad K \; -\sum_{k=1}^{K} \frac{{S^k}' A S^k}{d' S^k}$$

normalization $d := A\mathbf{1}$