5.2. Cluster Analysis

- Aim: group n objects into k classes
- $k \ll n$, k often unknown
- Objects within classes are "close" to each other
  objects in different classes are well discriminable.
- Needed: metric of closeness & discriminable

5.2.1 k-means clustering

Given data $x_1, \dots, x_n \in \mathbb{R}^p$

Aim: Partition data into clusters

$S_1, \dots, S_k$ (a partition of $\{1, \dots, n\}$) and

centers $\mu_1, \dots, \mu_k \in \mathbb{R}^p$

as a solution to

$$\min_{\substack{S_1, \dots, S_k \text{ part. of } \{1, \dots, n\} \\ \mu_1, \dots, \mu_k \in \mathbb{R}^p}} \sum_{\ell=1}^{k} \sum_{i \in S_\ell} \|x_i - \mu_\ell\|^2 \qquad \text{(NP-hard)}$$

- Given the partition, the optimum centers are

$$\mu_\ell = \bar{x}_\ell = \frac{1}{n_\ell} \sum_{i \in S_\ell} x_i$$

$$n_\ell = \# S_\ell$$

RWTH AACHEN UNIVERSITY

- The k-means algorithm, Lloyd's alg.

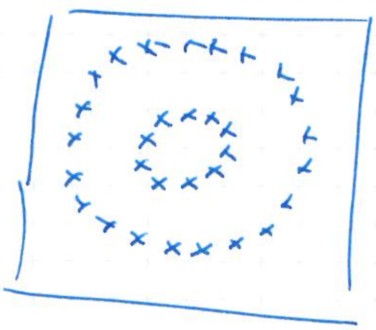
Alternate between

- Given centers $\mu_1, \dots, \mu_k$, assign x_i to the cluster $\ell = \arg\min_j \|x_i - \mu_j\|$
- Update the centers $\mu_\ell = \frac{1}{n_\ell} \sum_{i \in S_\ell} x_i$, $\ell = 1, \dots, k$

Remarks:

- Needs the no of clusters k a-priori.
- Needs a Euclidean space.
- May end in a suboptimal solution
- Solution has always "convex" clusters

How to cluster?



Thinking the Future
Zukunft denken

RWTH AACHEN UNIVERSITY

5.2.2. Spectral clustering

Given $x_1, \dots, x_n$

Construct a weighted graph $G = (V, E, W)$

Each points x_i represents a ~~node~~ vertex v_i.

Edges weights w_{ij} are $w_{ij} = K_\varepsilon(\|x_i - x_j\|)$

with a kernel K_ε, e.g., $K_\varepsilon(u) = \exp\left\{-\frac{1}{2\varepsilon} u^2\right\}$

Note: $\|x_i - x_j\|$ can be substituted by any dissimilarity measure.

Consider a random walk with transition matrix

$$M = D^{-1} W$$

$$P(X_{t+1} = j \mid X_t = i) = \frac{w_{ij}}{\deg(i)} = M_{ij}$$

$$D = \operatorname{diag}(\deg(i)), \quad \deg(i) = \sum_{l=1}^{n} w_{il}.$$

Decompose $M = \Phi \Lambda \Psi^T = \sum_{k=1}^{n} \lambda_k \varphi_k \psi_k^T$

(see 4.3 "diffusion maps")

$\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$, $\lambda_1 \geq \dots \geq \lambda_n$

$\Phi = (\varphi_1, \dots, \varphi_n)$, $\Psi = (\psi_1, \dots, \psi_n)$ biorthonormal, i.e.,

$$\Phi^T \Psi = I_n \Leftrightarrow \varphi_i^T \psi_j = \delta_{ij}.$$

φ_i, ψ_j right and left eigenvectors of M.

Thinking the Future
Zukunft denken

Then $\quad M^t = \sum_{k=1}^{n} \lambda_k^t \varphi_k \psi_k^T$

$$v_i \Leftrightarrow e_i M^t = \sum_{k=1}^{n} \lambda_k^t \underbrace{e_i \varphi_k}\psi_k^T$$

$$= \sum_{k=1}^{n} \underbrace{\lambda_k^t \varphi_{k,i}}_{\text{coefficients}} \underbrace{\psi_k^T}_{\text{orthonormal basis}}$$

Diffusion map truncated to d dimensions

$$\phi_t^{(d)}(i) = \begin{pmatrix} \lambda_2^t \varphi_{2i} \\ \vdots \\ \lambda_{d+1}^t \varphi_{d+1,i} \end{pmatrix} \qquad (\text{see } 4.3)$$

Aim: cluster the vertices of the graph in k clusters.

<u>Algorithm 5.3</u> (spectral clustering)

Given a graph $G=(V,E,W)$, no. of clusters k, t.

Compute the $(k-1)$-dim. diffusion map

$$\phi_t^{(k-1)}(i) = \begin{pmatrix} \lambda_2^t \varphi_{1,i} \\ \vdots \\ \lambda_k^t \varphi_{k,i} \end{pmatrix}$$

and cluster $\phi_t^{(k-1)}(1), \dots, \phi_t^{(k-1)}(n) \in \mathbb{R}^{k-1}$ using, e.g., k-means clustering. $\lrcorner$

Thinking the Future
Zukunft denken

5.2.3. Hierarchical Clustering

Method is based on dissimilarities between objects.

Given n objects $v_1, \dots, v_n$ and pairwise dissimilarities

$$\delta_{ij} = \delta_{ji} \quad , \quad \Delta = (\delta_{ij})_{1 \le i,j \le n}$$

Define a linkage function between clusters C_1, C_2

$$d(C_1, C_2) = \begin{cases} \min\limits_{i \in C_1, j \in C_2} \delta_{ij} & \text{single linkage} \\ \max\limits_{i \in C_1, j \in C_2} \delta_{ij} & \text{complete linkage} \\ \dfrac{1}{\# C_1 \, \# C_2} \sum\limits_{\substack{i \in C_1 \\ j \in C_2}} \delta_{ij} & \text{average linkage} \end{cases}$$

Algorithm: (agglomerative clustering)

Initialize clusters as singleton: for $i=1$ to n do $C_i \leftarrow \{i\}$.

Initialize the set of clusters for merging: $S \leftarrow \{1, \dots, n\}$

repeat

pick 2 most similar clusters to merge: ~~(j,k)~~

$$(j,k) \leftarrow \arg\min_{j,k \in S} d(C_j, C_k)$$

Create new cluster $C_\ell \leftarrow C_j \cup C_k$

Mark j and k as unavailable: $S \leftarrow S \setminus \{j,k\}$

if $C_\ell \neq \{1, \dots, n\}$ then

mark ℓ as available $S \leftarrow S \cup \{\ell\}$

for each $i \in S$ do

update the dissimilarity matrix $d(C_i, C_j)$

until no more clusters are available for merging.

Cut off, if a certain thershold is exceeded by $d(C_i, C_j)$

Visualization by dendrograms or binary trees.

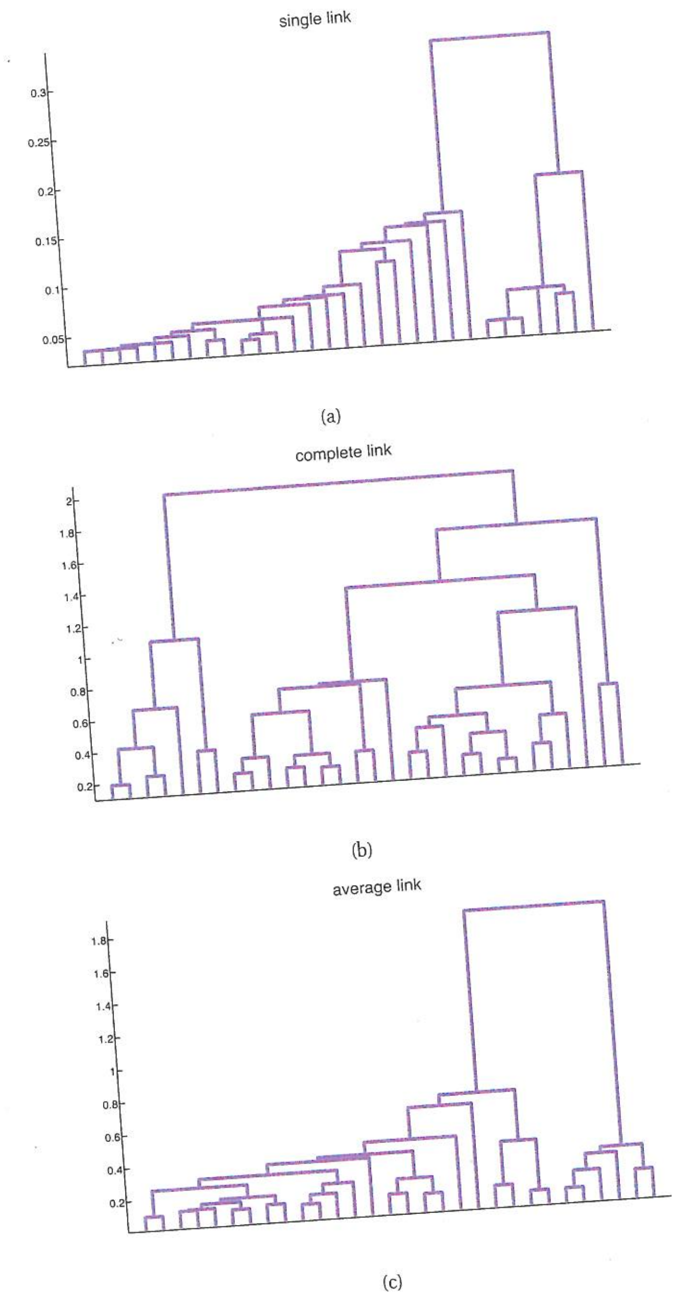


Figure 25.15 Hierarchical clustering of yeast gene expression data. (a) Single linkage. (b) Complete linkage. (c) Average linkage. Figure generated by `hclustYeastDemo`.

From: Murphy: Machine Learning