INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Outline

Part 1: Basic Concepts of data clustering

- Non-Supervised Learning and Clustering
  - Problem formulation – cluster analysis
  - Taxonomies of Clustering Techniques
  - Data types and Proximity Measures
  - Difficulties and open problems

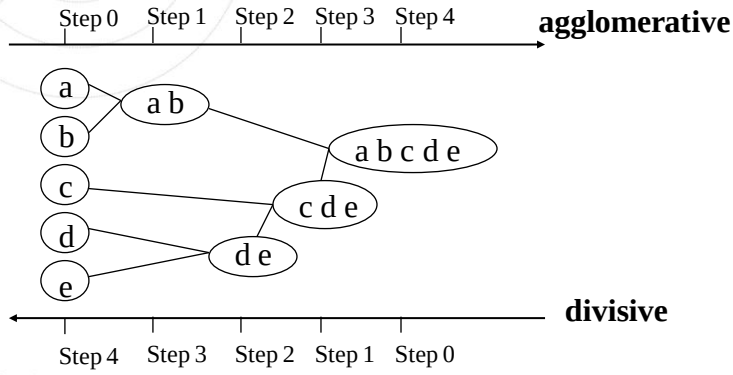
Part 2: Clustering Algorithms

- Hierarchical methods
  - Single-link
  - Complete-link
  - Clustering Based on Dissimilarity Increments Criteria

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering

- Use proximity matrix: *nxn*
  - D(i,j): proximity (similarity or distance) between patterns i and j



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

1. Start with *n* clusters containing one object
2. Find the most similar pair of clusters Ci e Cj from the proximity matrix and merge them into a single cluster
3. Update the proximity matrix (reduce its order by one, by replacing the individual clusters with the merged cluster)
4. Repeat steps (2) e (3) until a single cluster is obtained (i.e. N-1 times)

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

Similarity measures between clusters:

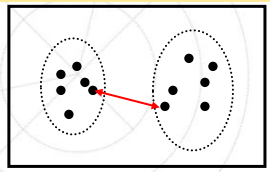
Well known similarity measures can be written using the Lance-Williams formula, expressing the distance between cluster *k* and cluster *i+j*, obtained by the merging of clusters *i* and *j*:

$$d(i+j,k) = a_i d(i,k) + a_j d(j,k) + b d(i,j) + c\left|d(i,k) - d(j,k)\right|$$

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroid	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

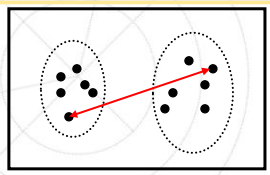


Single Link: Distance between two clusters is the distance between the closest points. Also called "neighbor joining."

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroid	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

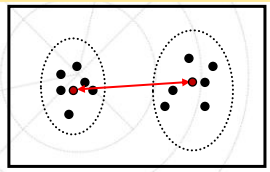


Complete Link: Distance between clusters is distance between farthest pair of points.

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroide	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

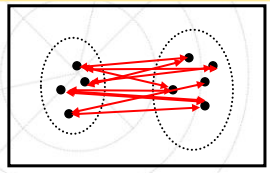


Centroid: Distance between clusters is distance between centroids.

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroide	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods

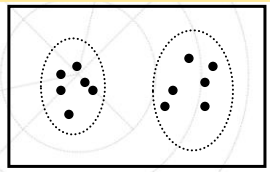


Average Link: Distance between clusters is average distance between the cluster points.

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroid	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering: Agglomerative Methods



Ward's Link: Minimizes the sum-of-squares criterion (measure of heterogenity)

$$ESS = \sum_{k=1}^{K} \sum_{x_i \in C_k} \sum_{j=1}^{d} (x_{ij} - \bar{x}_{kj})^2$$

Single-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = -0.5 \qquad d(i+j,k) = \min\{d(i,k), d(j,k)\}$
Complete-link	$a_i = a_j = 0.5\ ;\ b = 0\ ;\ c = 0.5 \qquad d(i+j,k) = \max\{d(i,k), d(j,k)\}$
Centroid	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = -\frac{n_i n_j}{(n_i+n_j)^2} \quad c = 0 \quad d(i+j,k) = d(\mu_{i+j}, \mu_k)$
Median	$a_i = a_j = 0.5\ ;\ b = -0.25\ ;\ c = 0$
(Average link)	$a_i = \frac{n_i}{n_i+n_j} \quad a_j = \frac{n_j}{n_i+n_j} \quad b = c = 0 \quad d(C_i, C_j) = \frac{1}{n_i n_j} \sum_{a \in C_i, b \in C_j} d(a,b)$
Ward's Method (minimum variance)	$a_i = \frac{n_k+n_i}{n_k+n_i+n_j} \quad a_j = \frac{n_k+n_j}{n_k+n_i+n_j} \quad b = -\frac{n_k}{n_k+n_i+n_j} \quad c = 0$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

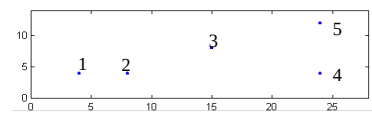
Single Linkage:

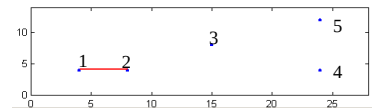
$$d(C_i, C_j) = \min_{a \in C_i, b \in C_j} \{d(a,b)\}$$

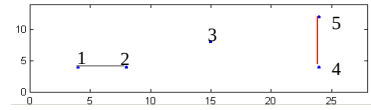
	x	y
1	4	4
2	8	4
3	15	8
4	24	4

	1	2	3	4	5
1	-	4	11.7	20	21.5
2	4	-	8.1	16	17.9
3	11.7	8.1	-	9.8	9.8
4	20	16	9.8	-	8
5	21.5	17.9	9.8	8	-

	1,2	3	4	5
1,2	-	8.1	16	17.9
3	8.1	-	9.8	9.8
4	16	9.8	-	8
5	17.9	9.8	8	-







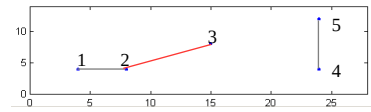
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

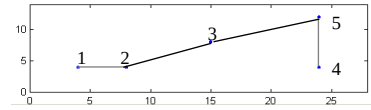
Single Linkage:

$$d(C_i, C_j) = \min_{a \in C_i, b \in C_j} \{d(a,b)\}$$

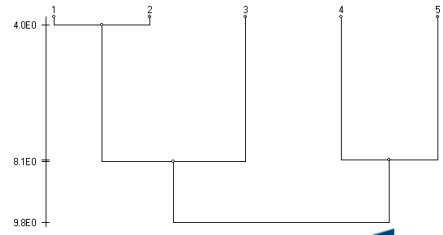
	1,2	3	4,5
1,2	-	8.1	16
3	8.1	-	9.8
4,5	16	9.8	-

	1,2,3	4,5
1,2,3	-	9.8
4,5	9.8	-





Dendrogram



instituto de telecomunicações

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

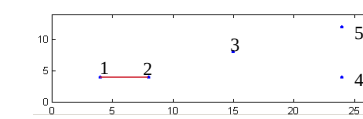
Complete-Link:

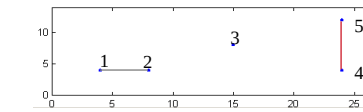
$$d(C_i, C_j) = \max_{a \in C_i, b \in C_j} \{d(a,b)\}$$

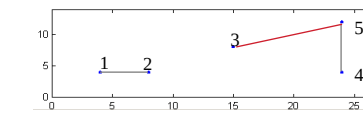
	1	2	3	4	5
1	-	4	11.7	20	21.5
2	4	-	8.1	16	17.9
3	11.7	8.1	-	9.8	9.8
4	20	16	9.8	-	8
5	21.5	17.9	9.8	8	-

	1,2	3	4	5
1,2	-	11.7	20	21.5
3	11.7	-	9.8	9.8
4	20	9.8	-	8
5	21.5	9.8	8	-

	1,2	3	4,5
1,2	-	11.7	21.5
3	11.7	-	9.8
4,5	21.5	9.8	-





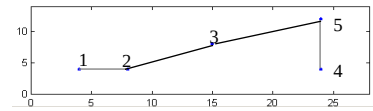


instituto de telecomunicações

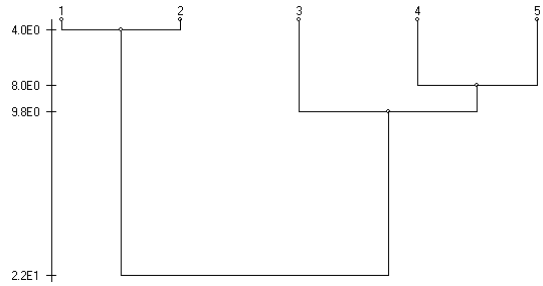
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Complete-Link: $d(C_i, C_j) = \max_{a \in C_i, b \in C_j} \{d(a,b)\}$

	1,2,3	4,5
1,2,3	-	21.5
4,5	21.5	-



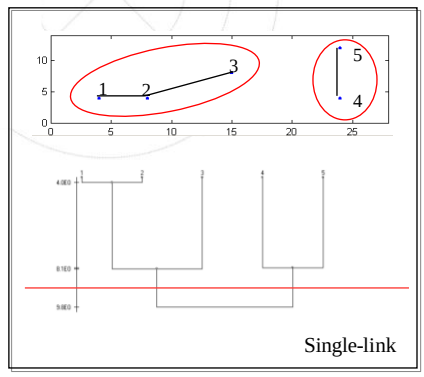
Dendrogram



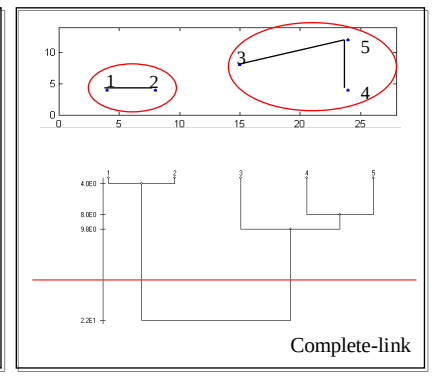
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

: A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster.



Single-link



Complete-link

Favours connectedness

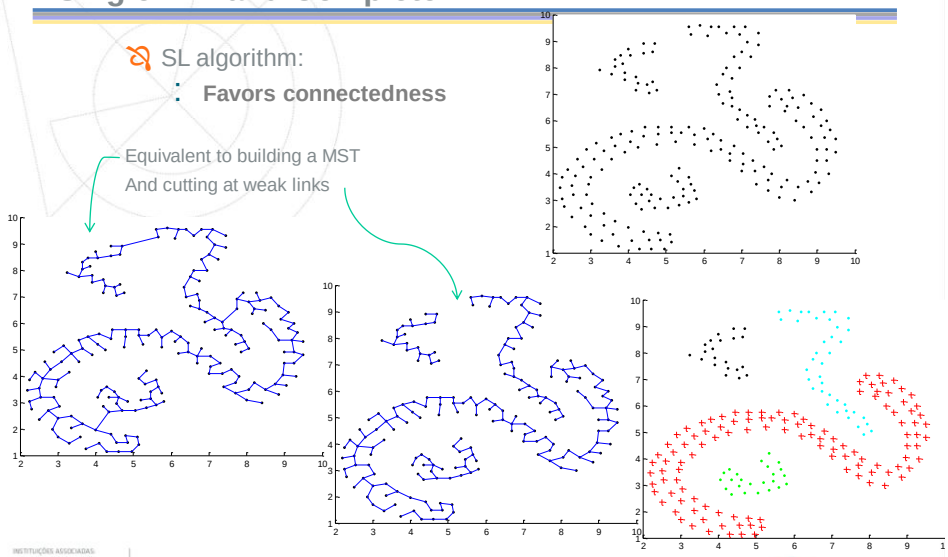
Favours compactness

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

- SL algorithm:
  - **Favors connectedness**

Equivalent to building a MST
And cutting at weak links

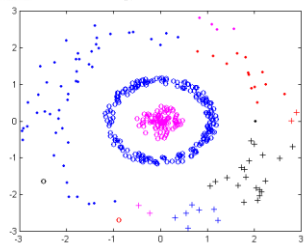


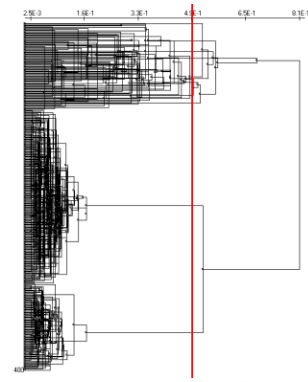
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

- SL algorithm:
  - Favors connectedness
  - Detects arbitrary-shaped clusters with even densities
  - **Cannot handle distinct density clusters**

Single-link method, th=0.49

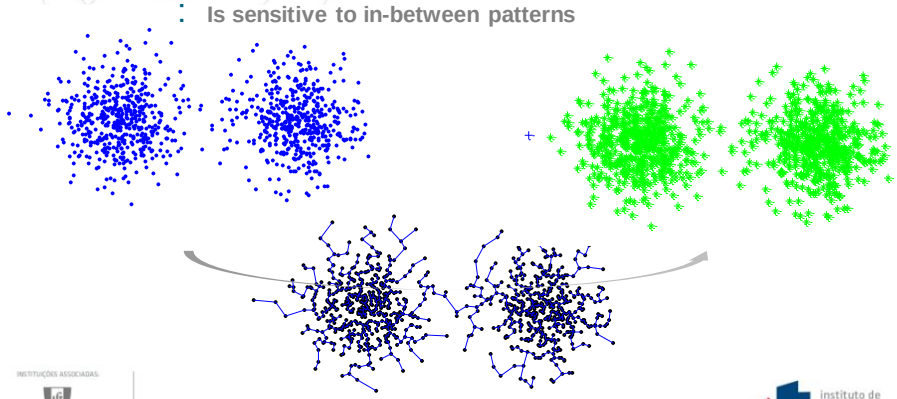




INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

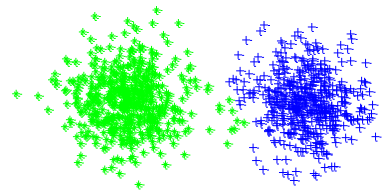
- SL algorithm:
  - Favors connectedness
  - Detects arbitrary-shaped clusters with even densities
  - Cannot handle distinct density clusters
  - **Is sensitive to in-between patterns**



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

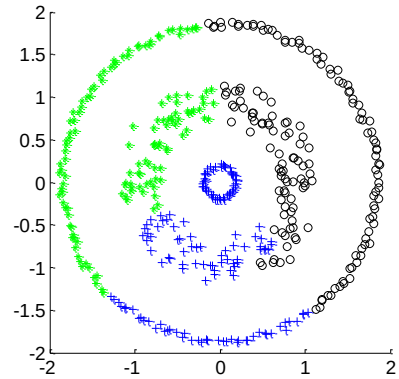
- SL algorithm:
  - Favors connectedness
  - Detects arbitrary-shaped clusters with even densities
  - Cannot handle distinct density clusters
  - Is sensitive to in-between patterns
  - Needs criteria to set the final number of clusters
- CL algorithm:
  - Favors compactness



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Cor

- SL algorithm:
  - Favors connec
  - Detects arbitra
  - Cannot handle
  - Is sensitive to
  - Needs criteria
- CL algorithm:
  - Favors compa
  - **Imposes spherical-shaped clusters on data**



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Single-link and Complete-Link

- SL algorithm:
  - Favors connectedness
  - Detects arbitrary-shaped clusters with even densities
  - Cannot handle distinct density clusters
  - Is sensitive to in-between patterns
  - Needs criteria to set the final number of clusters
- CL algorithm:
  - Favors compactness
  - Imposes spherical-shaped clusters on data
  - Needs criteria to set the final number of clusters

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Hierarchical Clustering

- Weakness
  - do not scale well: time complexity of $O(n^2)$, where n is the number of total objects
  - can never undo what was done previously
- Integration of hierarchical with distance-based clustering
  - BIRCH (Zhang, Ramakrishnan, Livny, 1996): uses a ClusteringFeature-tree and incrementally adjusts the quality of sub-clusters
  - CURE (Guha, Rastogi & Shim, 1998): selects well-scattered points from the cluster and then shrinks them towards the center of the cluster by a specified fraction
  - CHAMELEON (G. Karypis, E.H. Han and V. Kumar, 1999): hierarchical clustering using dynamic modeling
    1. Use a graph partitioning algorithm: cluster objects into a large number of relatively small sub-clusters
    2. Use an agglomerative hierarchical clustering algorithm: find the genuine clusters by repeatedly combining these sub-clusters

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- Smoothness Hypothesis:
  - A cluster is a set of patterns sharing important characteristics in a given context
  - A dissimilarity measure encapsulates the notion of pattern resemblance
  - Higher resemblance patterns are more likely to belong to the same cluster and should be associated first
  - Dissimilarity between neighboring patterns within a cluster should not occur with abrupt changes
  - The merging of well separated clusters results in abrupt changes in dissimilarity values

A Fred, J Leitão, A New Cluster Isolation Criterion Based on Dissimilarity Increments, IEEE PAMI, 2003.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

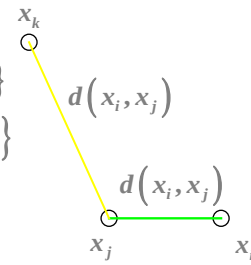
Clustering Based on Dissimilarity Increments Criteria

Dissimilarity Increments:

(x_i, x_j, x_k) - **nearest neighbors**

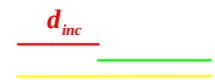
$$x_j : j = \arg\min_l \{d(x_l, x_i), l \neq i\}$$

$$x_k : k = \arg\min_l \{d(x_l, x_j), l \neq i\}$$



Dissimilarity increment:

$$d_{inc}(x_i, x_j, x_k) = |d(x_i, x_j) - d(x_j, x_k)|$$

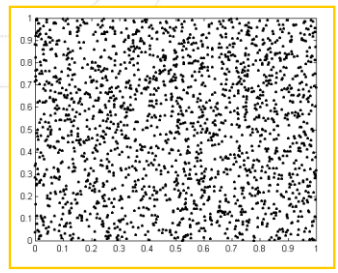


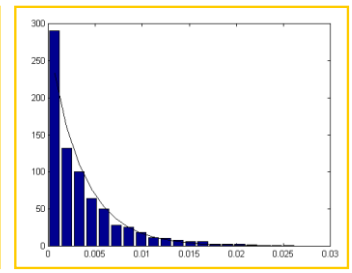
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

Distribution of Dissimilarity Increments:

: Uniformly distributed data

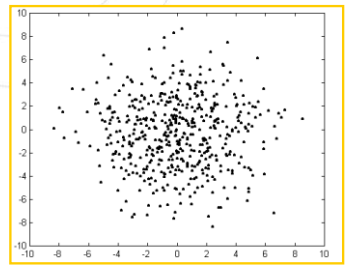
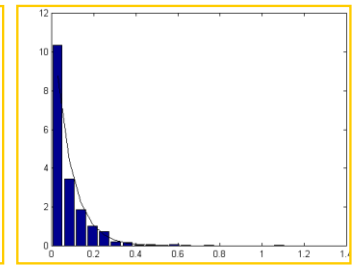




INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

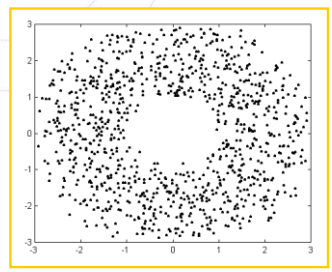
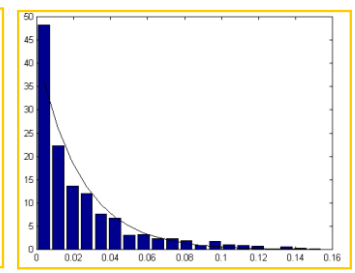
- Distribution of Dissimilarity Increments:
  - 2D Gaussian data

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

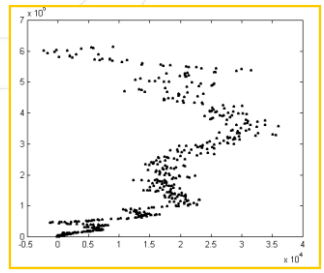
- Distribution of Dissimilarity Increments:
  - Ring-shaped data

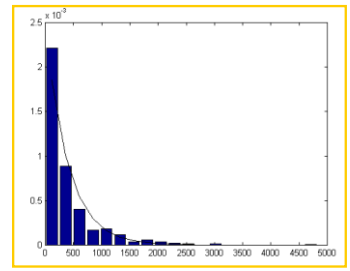



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- Distribution of Dissimilarity Increments:
  - Directional expanding data

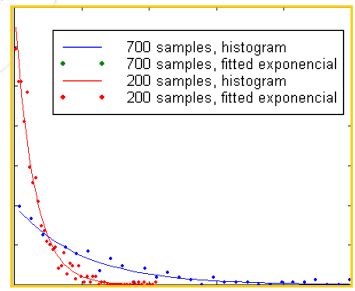




INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

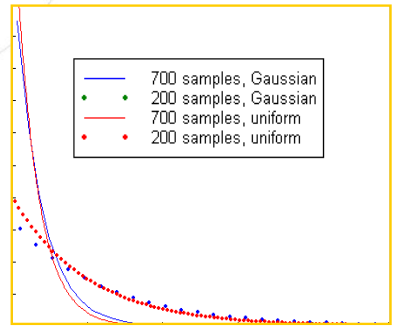
- Distribution of Dissimilarity Increments:
  - Exponential distribution: $p(x) = \beta \exp^{-\beta x}$



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

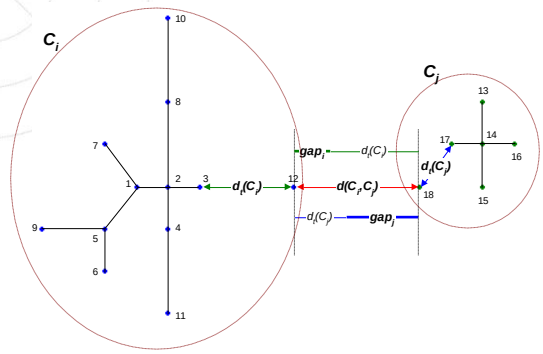
- Exponential distribution:
  - Higher density patterns -> higher β
  - Well separated clusters -> d_{inc} on the tail of $p(x)$



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

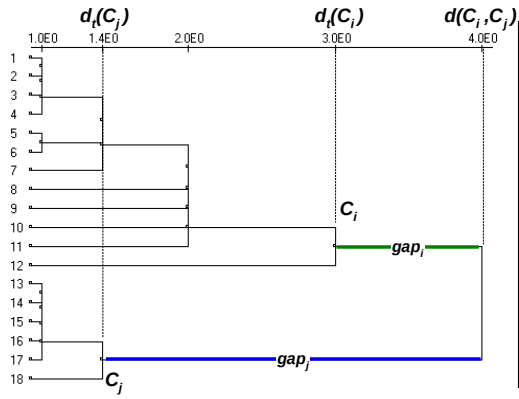
- *Gap* between clusters: $gap_i = d\left(C_i, C_j\right) - d_t\left(C_i\right)$



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- *Gap* between clusters: $gap_i = d(C_i, C_j) - d_t(C_i)$



Unsupervised Learning Clustering Algorithms

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

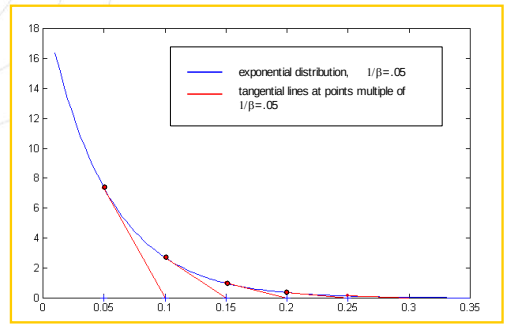
- Cluster Isolation criterion:

Let C_i, C_k be two clusters which are candidates for merging, and let μ_i , μ_k be the respective mean values of the dissimilarity increments in each cluster. Compute the increments for each cluster, gap_i and gap_k. If $gap_i \geq \alpha\ \mu_i$ ($gap_k \geq \alpha\ \mu_k$), isolate cluster C_i (C_k) and proceed the clustering strategy with the remaining patterns. If neither cluster exceeds the gap limit, merge them.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- Setting the Isolation Criterion Parameter α:
  - Result: the crossings of the tangential line, at points which are multiple of the distribution mean value, α/β, with the *x* axis, is given by $(\alpha+1)/\beta$



  - $\alpha \in [3,5]$ cover the significant part of the distribution

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

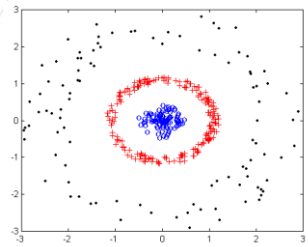
Clustering Based on Dissimilarity Increments Criteria

- Hierarchical Clustering Algorithm:
  - A statistic of the dissimilarity increments within a cluster is maintained and updated during cluster merging
  - Clusters are obtained by comparing dissimilarity increments with a dynamic threshold, $\alpha\mu_i$, based on cluster statistics

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- Results:
  - Ring-Shaped Clusters

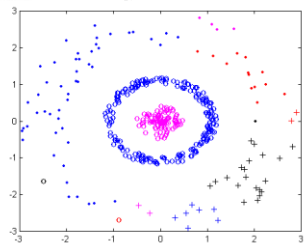


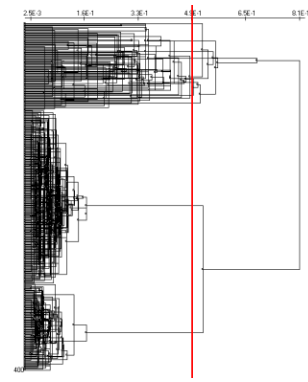
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

Results:

- Ring-Shaped Clusters
  - Single-link method, th=0.49



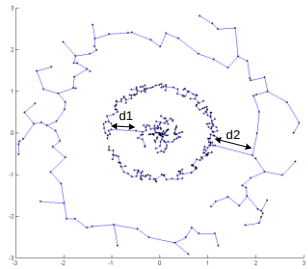


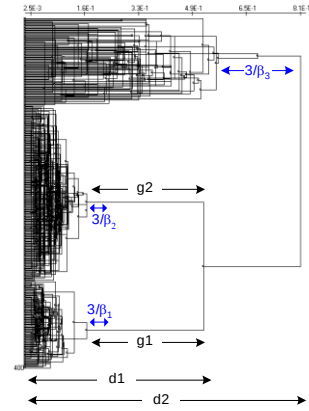
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

Results:

- Ring-Shaped Clusters
  - Dissimilarity Increments-base method:





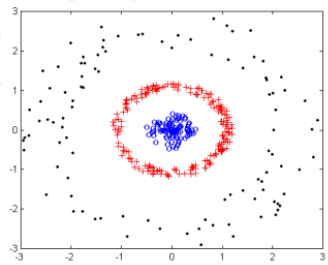
instituto de telecomunicações

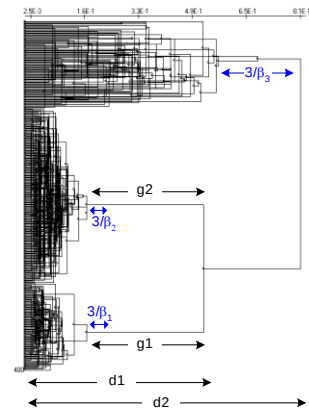
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

Results:

- Ring-Shaped Clusters
  - Dissimilarity Increments-base method:





instituto de telecomunicações

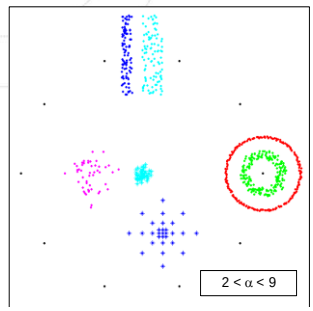
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

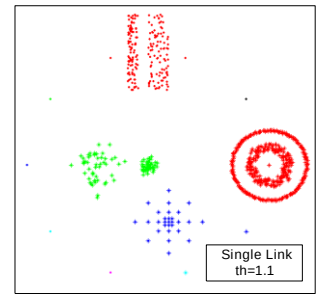
Results:

- 2-D Patterns with Complex Structure

Dissimilarity Increments-base method

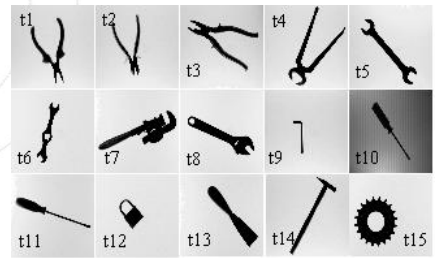


Single-link



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering of Contour Images

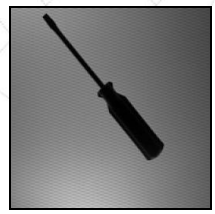


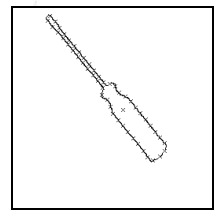
- The data set is composed by 634 contour images of 15 types of hardware tools: t1 to t15.
- When counting each pose as a distinct sub-class in the object type, we obtain a total of 24 classes.

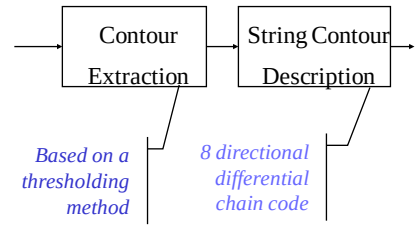
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering of Contour Images

Contour extraction







- the object boundary is sampled at 50 equally spaced points
- the angle between consecutive segments is quantized in 8 levels.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering of Contour Images

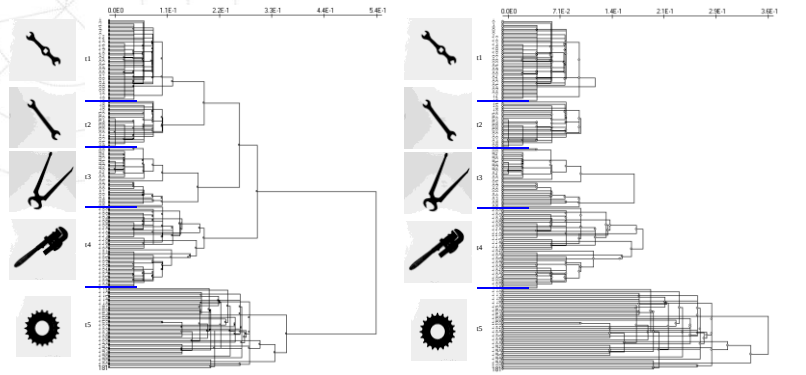
Class	Tool	Samples
1		6107262002627016370262002620726107260162702620026 2 7117262002620016270163702620726107361072611637026 2 7117262002627026200262073701637026200262002620016 2
2		0000000066000000171000006000001000000007500000000 3 0000000050000000000000760000016100000066000000 3 0000000050000000171000005000001610000007600000000 3
3		1000000005000000000720700070170000000005000000001 2 1000000005000000000107000702700000000050000000012 1000000005700000000710700770270000000005000000012

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering of Contour Images

Single-link

Dissimilarity Increments-based method



(string-edit distance)

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Clustering Based on Dissimilarity Increments Criteria

- Description:
  - Hierarchical agglomerative algorithm adopting a cluster isolation criterion based on dissimilarity increments
- Strength
  - Method is not conditioned by a particular dissimilarity measure (examples used Euclidean and string-edit distances)
  - Ability to identify arbitrary shaped and sized clusters
  - **The number of clusters is intrinsically found**
- Weakness
  - Sensitive to in-between points connecting touching clusters

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Outline

- **Partitional Methods**
  - K-Means
  - Spectral Clustering
  - EM-based Gaussian Mixture Decomposition

Part 3.: Validation of clustering solutions

- Cluster Validity Measures

Part 4.: Ensemble Methods

- Evidence Accumulation Clustering

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods

- K-Means
  - Minimizes the cost function: $H^{KM} = \sum_{i=1}^{n} ||x_i - y_{c_i}||^2$
  - Algorithm:
    - Input: *k*, the number of clusters; data set

1. Randomly select *k* seed points from the data set, and take them as initial centroids
2. Partition the data into *k* clusters by assigning each object to the cluster with the nearest centroid.

$$c_i = \arg \min_{\nu \in \{1,\ldots,k\}} ||x_i - y_{c_i}||^2$$

3. Compute centroids of the clusters of the current partition. The centroid is the center (mean point) of the cluster.

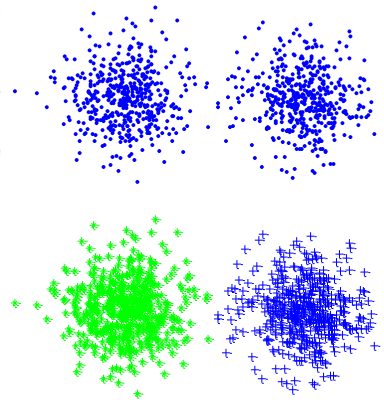
$$y_\nu = \frac{1}{n_\nu} \sum_{i:c_i=\nu} x_i \quad \text{with} \quad n_\nu = |\{i : c_i = \nu\}|$$

4. Go back to step 2 or stop when no more new assignment.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods: K-Means

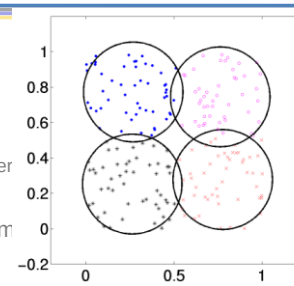
- Favors compactness



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods: K-Means

- Favors compactness
- Strength
  - Fast algorithm ($O(tkn)$ – t is the number
  - Scalability
  - Often terminates at a local optimum
- Weakness
  - **Imposes spherical-shaped clusters**

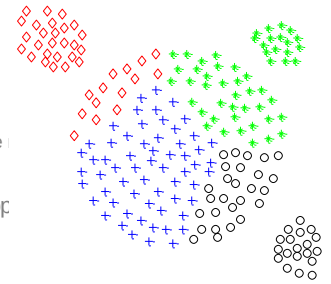


K-means clustering of uniform data (k=4)

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods: K-Means

- Favors compactness
- Strength
  - Fast algorithm ($O(tkn)$ – t is the
  - Scalability
  - Often terminates at a local op
- Weakness
  - Imposes spherical-shaped clusters
  - **Is sensitive to the number of objects in clusters**

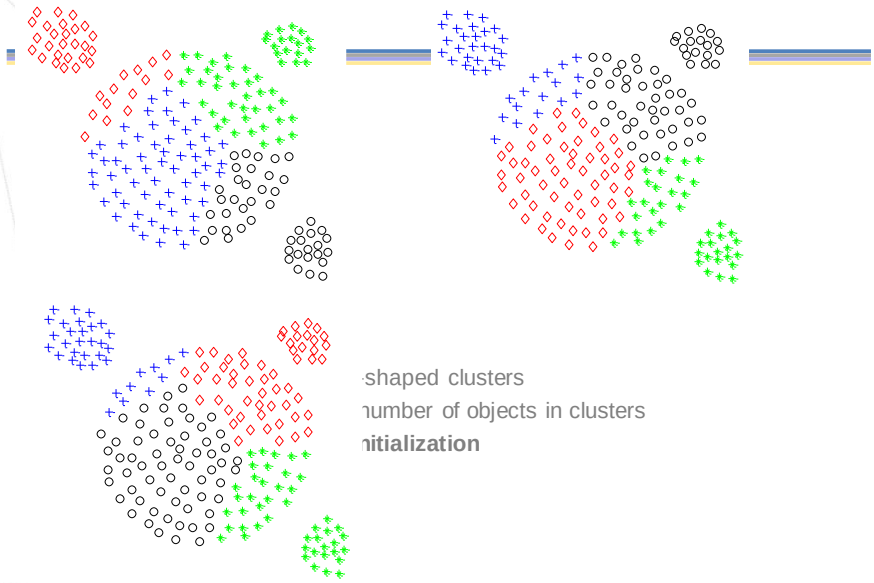


INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods: K-Means

- Favors compactness
- Strength
  - Fast algorithm ($O(tkn)$ – t is the number of iterations; normally, k, $t << n$.)
  - Scalability
  - Often terminates at a local optimum.
- Weakness
  - Imposes spherical-shaped clusters
  - Is sensitive to the number of objects in clusters
  - **Dependence on initialization**

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa



-shaped clusters
number of objects in clusters
nitialization

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Partitional Methods: K-Means

- Favors compactness
- Strength
  - Fast algorithm ($O(tkn)$ – t is the number of iterations; normally, k, $t << n$.)
  - Scalability
  - Often terminates at a local optimum.
- Weakness
  - Imposes spherical-shaped clusters
  - Is sensitive to the number of objects in clusters
  - Dependence on initialization
  - Needs criteria to set the final number of clusters
  - Applicable only when *mean* is defined (what about categorical data?)

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

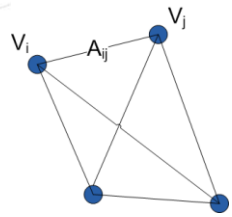
Variations of K-Means Method

- K-Means(MacQueen'67): each cluster is represented by the center of the cluster
- A few variants of the k-means which differ in
  - Selection of the initial k means
  - Dissimilarity calculations (Mahalanobis distance -> elliptic clusters)
  - Strategies to calculate cluster means
    - Medoid - each cluster is represented by one of the objects in the cluster
  - Fuzzy version: *Fuzzy K-Means*
- Handling categorical data: *k-modes* (Huang'98)
  - Replacing means of clusters with modes
  - Using new dissimilarity measures to deal with categorical objects
  - Using a frequency-based method to update modes of clusters

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering

- For a given data set, ***X***, Spectral Clustering finds a set of data clusters on the basis of spectral analysis of a similarity graph
- The clustering problem is defined in terms of a complete graph, G, with vertices $V=\{1, \ldots, N\}$, corresponding to the data points in the data set, and each edge between two vertices is weighted by the similarity between them.
- The weight matrix is also called the ***affinity matrix*** or the similarity matrix.



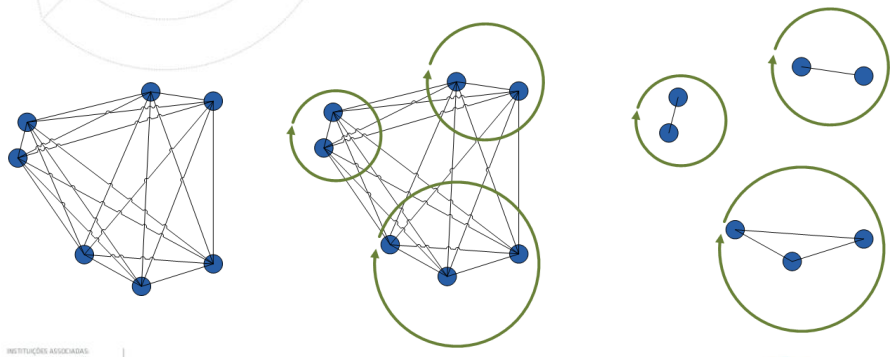
$$A_{ij} = e^{\frac{\left(\|x_i - x_j\|\right)^2}{2\sigma^2}}$$

Gaussian Kernel

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering

- Cutting edges of *G* we obtain disjoint subgraphs of *G as the clusters of X*
- The goal of clustering is to organize the dataset into disjoint subsets with high intra-cluster similarity and low inter-cluster similarity
  - The resulting clusters should be as **compact** and **isolated** as possible



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

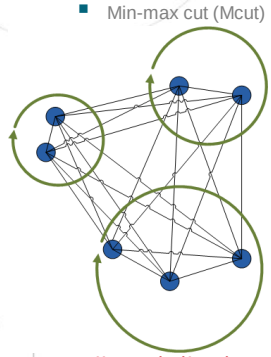
Spectral Clustering

- The graph partitioning for data clustering can be interpreted as a minimization problem of an objective function, in which the compactness and isolation are quantified by the subset sums of edge weights.
- Common objective functions
  - Ratio cut (Rcut)
  - Normalised cut (Ncut)
  - Min-max cut (Mcut)

$$\mathrm{Rcut}(C_1,\ldots,C_k) := \sum_{l=1}^{k} \frac{\mathrm{cut}(C_l, X \setminus C_l)}{\mathrm{card}\, C_l}$$

$$\mathrm{Ncut}(C_1,\ldots,C_k) := \sum_{l=1}^{k} \frac{\mathrm{cut}(C_l, X \setminus C_l)}{\mathrm{cut}(C_l, X)}$$

$$\mathrm{M\,cut}(C_1,\ldots,C_k) := \sum_{l=1}^{k} \frac{\mathrm{cut}(C_l, X \setminus C_l)}{\mathrm{cut}(C_l, C_l)}$$



- cut(*A, B) is the sum of the edge weights between* $\forall p \in A$ *and* $\forall p \in B$
- $P \setminus C_l$ *is the complement of* $C_l \subset X$,
- card C_l *denotes the number of points* in C_l

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering

- The solution of the minimization problem of any of the previous objective functions is obtained from the matrix of the first *k* eigenvectors of a matrix derived from the affinity matrix (Laplacian matrix)
  - The eigenvectors for Ncut and Mcut are identical, and obtained from the symmetrical Laplacian

$$L = L_{sym} := D^{-1/2} A D^{-1/2}$$

  - *D* is a diagonal matrix whose *ii*-th entry is the sum of the *i*-th row of *A*

  - Another common choice is $L = L_{rw} := D^{-1}(D - A)$

- Distinct algorithms differ on the way of producing and using the eigenvectors and how to derive clusters from them.
  - Some use each eigenvector one at a time
  - Other, use top k eigenvectors simultaneously

- Closely related with spectral graph partitioning, in which the second eigenvector of a graph's Laplacian is used to define a semi-optimal cut; the second eigenvector solves a relaxation of an NP-hard discrete graph partitioning problem, giving an approximation to the optimal cut.

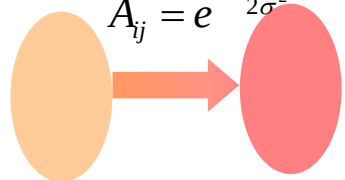
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

- Maps the feature space into a new space, *Y*, based on the eigenvectors of a matrix derived from an affinity matrix associated with the data set.
- The data partition is obtained by applying the K-means algorithm on the new space.

(NG. and Al. 2001)

$$A_{ij} = e^{\frac{\left(\|x_i - x_j\|\right)^2}{2\sigma^2}}$$



Original feature space

Eigen vector Feature space

A. Y. Ng and M. I. Jordan and Y. Weiss, On Spectral Clustering: Analysis and an algorithm, NIPS 2001

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

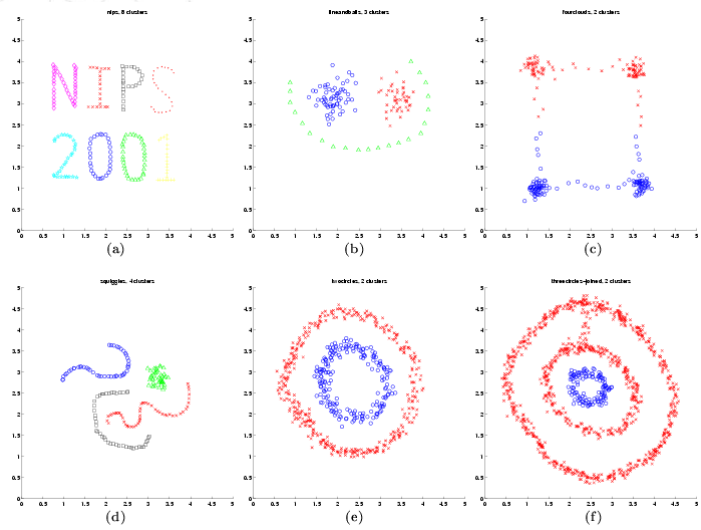
Algorithm:

Given a set of points $S = \{s_1, \ldots, s_n\}$ in $\mathbb{R}^l$ that we want to cluster into k subsets:

1. Form the affinity matrix $A \in \mathbb{R}^{n \times n}$ defined by $A_{ij} = \exp(-\|s_i - s_j\|^2/2\sigma^2)$ if $i \neq j$, and $A_{ii} = 0$.
2. Define D to be the diagonal matrix whose (i,i)-element is the sum of A's i-th row, and construct the matrix $L = D^{-1/2} A D^{-1/2}$.[1]
3. Find $x_1, x_2, \ldots, x_k$, the k largest eigenvectors of L (chosen to be orthogonal to each other in the case of repeated eigenvalues), and form the matrix $X = [x_1 x_2 \ldots x_k] \in \mathbb{R}^{n \times k}$ by stacking the eigenvectors in columns.
4. Form the matrix Y from X by renormalizing each of X's rows to have unit length (i.e. $Y_{ij} = X_{ij}/(\sum_j X_{ij}^2)^{1/2}$).
5. Treating each row of Y as a point in $\mathbb{R}^k$, cluster them into k clusters via K-means or any other algorithm (that attempts to minimize distortion).
6. Finally, assign the original point s_i to cluster j if and only if row i of the matrix Y was assigned to cluster j.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

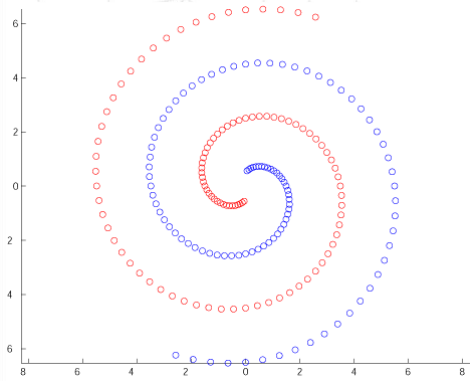
Spectral Clustering (Ng et al, 2001)



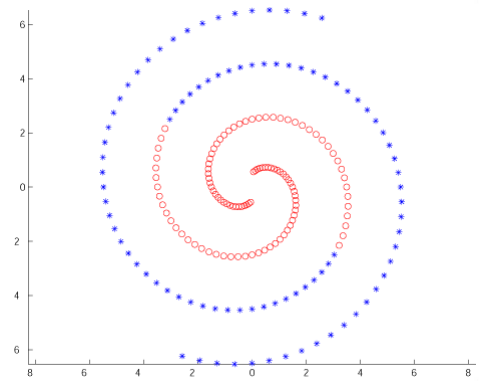
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

- Results strongly depend on parameter values: *k* e σ



K=2, σ=0.1

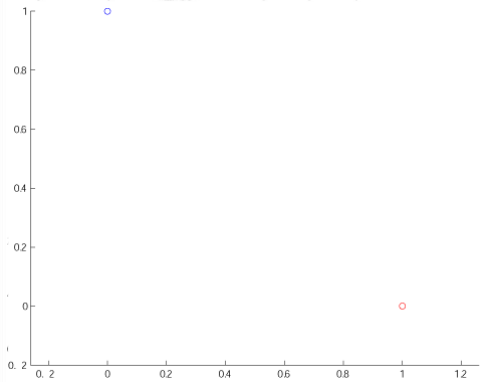


K=2, σ=0.4

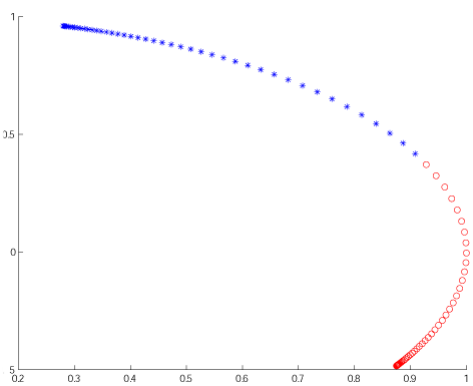
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

- Results strongly depend on parameter values: *k* e σ



K=2, σ=0.1

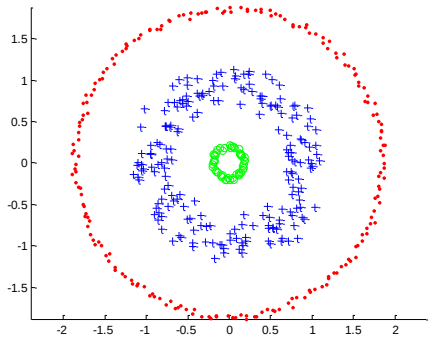


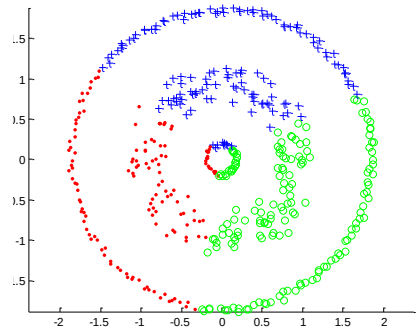
K=2, σ=0.4

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

- Results strongly depend on parameter values: *k* e σ



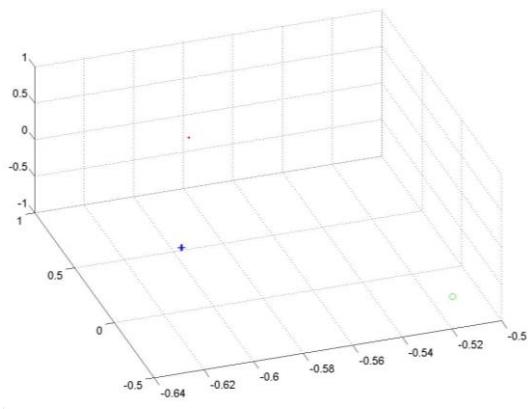


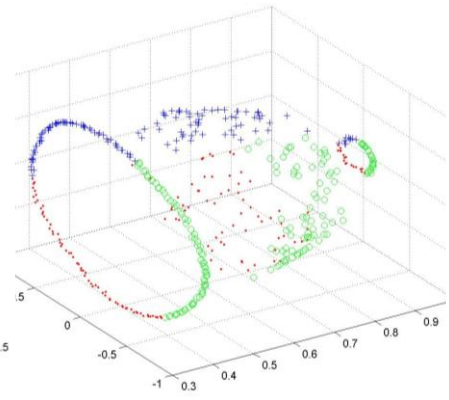
K=3, σ=0.1 K=3, σ=0.45

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering (Ng et al, 2001)

- Results strongly depend on parameter values: *k* e σ





K=3, σ=0.1 K=3, σ=0.45

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Spectral Clustering

Selection of parameter values:

MSE:

$$MSE_Y = \frac{1}{N}\sum_{i=1}^{k}\sum_{j=1}^{n_i}\sqrt{\left(y_i^j - m_i\right)^2}$$

Eigengap

$$\delta(A) = 1 - \frac{\lambda_2}{\lambda_1}$$

Rcut

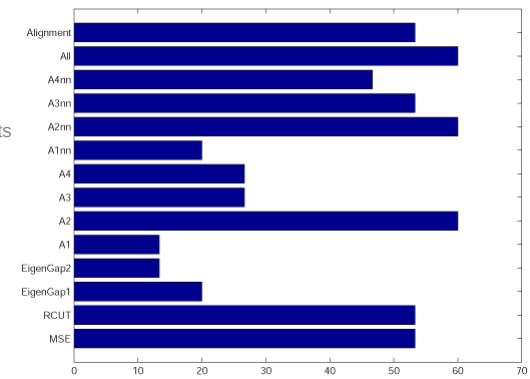
$$Rcut_K = \frac{\sum_{k=1}^{K}\sum_{l=1,l\neq k}^{K}\sum_{j\in S_k}\sum_{j\in S_l} A_{jk}}{\sum_i\sum_j A_{ij}}$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Selection of Parameters: Global Results on selecting σ and K

None of the studied methods is suitable for the automatic selection of the spectral clustering parameters

A majority voting decision did not significantly improve the results



Percentage of correct classification:
σ

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

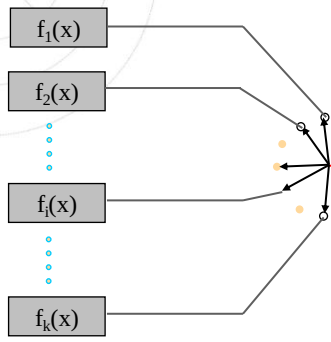
Spectral Clustering

- Strength
  - Detects arbitrary-shaped clusters.
  - By using an adequate similarity measure between patterns, can be applied to all types of data
- Weakness
  - Computationally heavy
  - Needs criteria to set the final number of clusters and scaling factor

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Model-Based Clustering: Finite Mixtures

- k random sources, with probability density functions $f_i(x)$, i=1,…,k



Choose at random

X → random variable

Conditional: $f(x|\text{source } i) = f_i(x)$

Joint: $f(x \text{ and source } i) = f_i(x)\,\alpha_i$

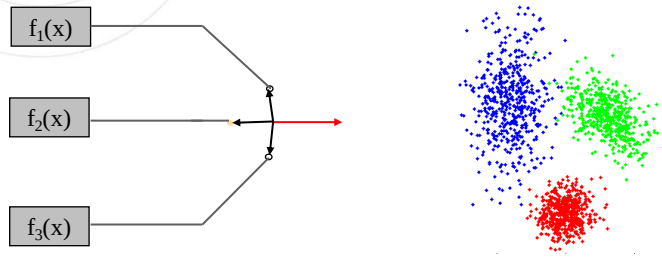
Unconditional: $f(x) = \sum_{\text{all sources}} f(x \text{ and source } i) = \sum_{i=1}^{k} \alpha_i f_i(x)$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Model-Based Clustering: Finite Mixtures

- each component models one cluster
- clustering = mixture fitting

$$f(x|\Theta) = \sum_{i=1}^{k} \alpha_i \; f(x|\theta_i)$$



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gaussian Mixture Decomposition

- Mixture Model:

$$f(x|\Theta) = \sum_{i=1}^{k} \alpha_i \; f(x|\theta_i)$$

Component densities

Mixing probabilities: $\alpha_i \geq 0$ and $\sum_{i=1}^{k} \alpha_i = 1$

: $f(x \mid \theta_i) \longrightarrow$ Gaussian

- Arbitrary covariances: $f(x \mid \theta_i) = N(x \mid \boldsymbol{\mu}_i, \mathbf{C}_i)$

$$\Theta = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_k, \mathbf{C}_1, \mathbf{C}_2, ..., \mathbf{C}_k, \alpha_1, \alpha_2, ...\alpha_{k-1}\}$$

- Common covariance: $f(x \mid \theta_i) = N(x \mid \boldsymbol{\mu}_i, \mathbf{C})$

$$\Theta = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_k, \mathbf{C}, \alpha_1, \alpha_2, ...\alpha_{k-1}\}$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Mixture Model Fitting

- n independent observations $\mathbf{x} = \{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$
- Mixture density model: $f(x|\Theta) = \sum_{i=1}^{k} \alpha_i \; f(x|\theta_i)$
- Estimate Θ that maximizes (log)likelihood (ML estimate of Θ):

$$\hat{\Theta} = \arg\max_{\Theta} L(x, \Theta)$$

$$L(x, \Theta) = \log \prod_{j=1}^{n} f(x^{(j)} \mid \Theta) = \sum_{j=1}^{n} \log \sum_{i=1}^{k} \alpha_i f(\mathbf{x}^{(j)} \mid \theta_i)$$

mixture

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gaussian Mixture Model Fitting

- Problem: the likelihood function is unbounded as $\det(\mathbf{C}_i) \to 0$
  - There is no global maximum
  - Unusual goal: a "good" local maximum
- Example: a 2-component Gaussian mixture

$$f(x \mid \mu_1, \mu_2, \sigma^2, \alpha) = \frac{\alpha}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} + \frac{1-\alpha}{\sqrt{2\pi}} e^{-\frac{(x-\mu_2)^2}{2}}$$

some data points $\{x_1, x_2, ..., x_n\}$

$$\mu_1 = x_1$$

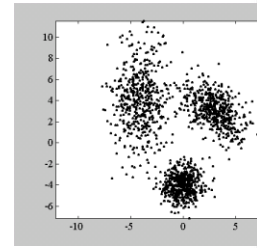
$$L(x, \Theta) = \log\left(\frac{\alpha}{\sqrt{2\pi\sigma^2}} + \frac{1-\alpha}{\sqrt{2\pi}} e^{-\frac{(x_1-\mu_2)^2}{2}}\right) + \sum_{j=2}^{n} \log(...)$$

$$\to \infty, \quad \text{as} \quad \sigma^2 \to 0$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Mixture Model Fitting

- ML estimate has no closed-form solution
- Standard alternative: expectation-maximization (EM) algorithm:
  - Missing data problem:
    - Observed data: $\mathbf{x} = \{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

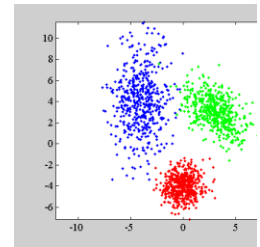
Mixture Model Fitting

- ML estimate has no closed-form solution
- Standard alternative: expectation-maximization (EM) algorithm:
  - Missing data problem:
    - Observed data: $\mathbf{x} = \{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$
    - Missing data: $\mathbf{z} = \{\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, ..., \mathbf{z}^{(n)}\}$

Missing labels ("colors")

$$\mathbf{z}^{(j)} = [z_1^{(j)}, \; z_2^{(j)}, \; ..., \; z_k^{(j)},] = [0 \; ... \; 0 \; 1 \; 0 \; ... \; 0]^T$$

"1" at position i $\Leftrightarrow$ $x^{(j)}$ generated by component i



- Complete log-likelihood function:

$$L_c(\mathbf{x}, \mathbf{z}, \Theta) = \sum_{j=1}^{n} \sum_{i=1}^{k} z_i^{(j)} \log\left(\alpha_i \, f_i(\mathbf{x}^{(j)} \mid \theta_i)\right)$$

$$\log f(x^{(j)}, z^{(j)} \mid \Theta)$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

The EM Algorithm

- The E-step: compute the expected value of $L_c(\mathbf{x}, \mathbf{z}, \Theta)$

$$E[L_c(\mathbf{x}, \mathbf{z}, \Theta) \mid x, \hat{\Theta}^{(t)}] \equiv Q(\Theta, \hat{\Theta}^{(t)})$$

- The M-step: update parameter estimates

$$\hat{\Theta}^{(t+1)} = \arg\max_{\Theta} Q(\Theta, \hat{\Theta}^{(t)})$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

EM-Algorithm for Mixture of Gaussian

- Iterative procedure: $\hat{\Theta}^{(0)}, \hat{\Theta}^{(1)}, \ldots, \hat{\Theta}^{(t)}, \hat{\Theta}^{(t+1)}, \ldots$
- The E-step: $w_i^{(j,t)} \equiv \dfrac{\hat{\alpha}_i \; f(x^{(j)} \mid \hat{\theta}_i^{(t)})}{\sum_{n=1}^{k} \hat{\alpha}_n \; f(x^{(j)} \mid \hat{\theta}_n^{(t)})}$

$w_i^{(j,t)} \rightarrow$ Estimate, at iteration t, of the probability that $x^{(j)}$ was produced by component i

- The M-step:

$$\hat{\alpha}_i^{(t+1)} = \frac{1}{n} \sum_{j=1}^{n} w_i^{(j,t)}$$

$$\hat{\mu}_i^{(t+1)} = \frac{\sum_{j=1}^{n} w_i^{(j,t)} \, x^{(j)}}{\sum_{j=1}^{n} w_i^{(j,t)}}$$

$$\hat{C}_i^{(t+1)} = \frac{\sum_{j=1}^{n} w_i^{(j,t)} \, (x^{(j)} - \hat{\mu}_i^{(t+1)}) (x^{(j)} - \hat{\mu}_i^{(t+1)})^T}{\sum_{j=1}^{n} w_i^{(j,t)}}$$

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Mixture Gaussian Decomposition: Model Selection

- How many components?
  - The maximized likelihood never decreases when k increases
  - $$\hat{k} = \arg\min \left\{ \mathcal{C}(\hat{\Theta}_{(k)}),\ k = k_{min}, k_{min}+1, \ldots, k_{max} \right\}$$
  - Usually: $$\mathcal{C}(\hat{\Theta}_{(k)}) = -L(\mathbf{x}, \hat{\Theta}_{(k)}) + P(\hat{\Theta}_{(k)})$$
  - Criteria in this cathegory:
    - Minimum description length (MDL), Rissanen and Ristad, 1992.
    - Akaike's information criterion (AIC), Whindham and Cutler, 1992.
    - Schwarz´s Bayesian inference criterion (BIC), Fraley and Raftery, 1998.
  - Resampling-based techniques
    - Bootstrap for clustering, Jain and Moreau, 1987.
    - Bootstrap for Gaussian mixtures, McLachlan, 1987.
    - Cross validation, Smyth, 1998.

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Mixture Gaussian Decomposition: Model Selection

- Given $\Theta(k)$, shortest code-length for x (Shannon's):
$$L(\mathbf{x} | \Theta_{(k)}) = -\log f(\mathbf{x} | \Theta_{(k)})$$
- MDL criterion: parameter code length
  - Total code-length (two part code):
$$L(\mathbf{x}, \Theta_{(k)}) = -\log f(\mathbf{x} | \Theta_{(k)}) + L(\Theta_{(k)})$$
Parameter code-length
  - MDL criterion:
$$\hat{\Theta}_{(k)} = \arg\min_{\Theta_{(k)}} \left\{ -\log f(\mathbf{x} | \Theta_{(k)}) + L(\Theta_{(k)}) \right\}$$
$$L(\text{each component of } \Theta_{(k)}) = \frac{1}{2}\log(n')$$

$n' \longrightarrow$ Amount of data from which the parameter is estimated

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Mixture Gaussian Decomposition: Model Selection

- Classical MDL: n' = n

$$\hat{\Theta}_{(k)} = \arg\min_{\Theta_{(k)}} \left\{ -\log f(\mathbf{x} \mid \Theta_{(k)}) + \frac{k}{2}\log(n) \right\}$$

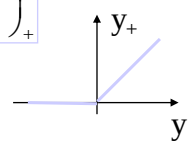
- Mixtures MDL (MMDL) (Figueiredo, 2002)

$$\hat{\Theta}_{(k)} = \arg\min_{\Theta_{(k)}} \left\{ -\log f(\mathbf{x} \mid \Theta_{(k)}) + \frac{k(N_p+1)}{2}\log(n) + \frac{N_p}{2}\sum_{m=1}^{k}\log(\alpha_m) \right\}$$

- Using EM and redefining the M-Step

$$\beta_i = \left(\sum_{j=1}^{n} w_i^{(j,t)} - \frac{N_p}{2} \right)_+$$

$$\hat{\alpha}_i^{(t+1)} = \frac{\beta_i}{\sum_{m=1}^{k}\beta_m}$$



This M-step may annihilate components

Np is the number of parameters of each component.

Gaussian, arbitrary covariances
Np = d + d(d+1)/2

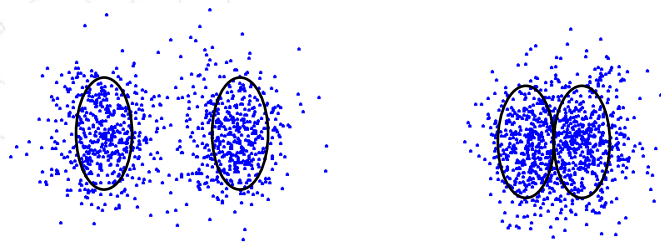
Gaussian, common covariance:
Np = d

M. Figueiredo and A. K. Jain, Unsupervised Learning of Finite Mixture Models, IEEE TPAMI, 2002

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gaussian Mixture Decomposition – EM MMDL

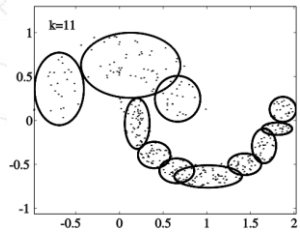
Examples

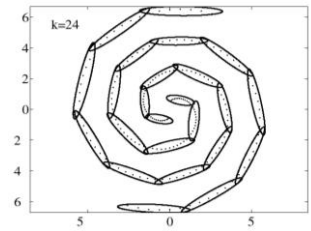


INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gaussian Mixture Decomposition – EM MMDL

Examples





INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gaussian Mixture Decomposition

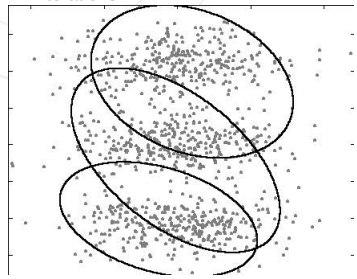
- Strength
  - Model-based approach
  - Good for Gaussian data
  - Handles touching clusters
- Weakness
  - Unable to detect arbitrary shaped clusters
  - Dependence on initialization

INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

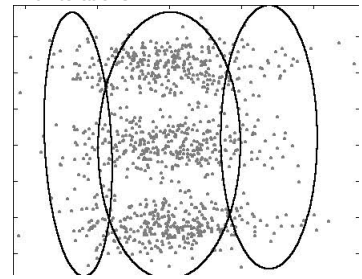
Gaussian Mixture Decomposition

It is a local (greedy) algorithm (likelihood never decreases)
=> Initialization dependent

74 iterations



270 iterations



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

References

A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Prentice Hall, 1988.

A.K. Jain and M. N. Murty and P.J. Flynn, Data Clustering: A Review, ACM Computing Surveys, vol 31. No 3,pp 264-323, 1999.

Data Mining: Concepts and Techniques, J. Han and M. Kamber, Morgan Kaufmann Pusblishers, 2001.

A. Y. Ng and M. I. Jordan and Y. Weiss, On Spectral Clustering: Analysis and an algorithm, in Advances in Neural Information Processing Systems 14, T. G. Dietterich and S. Becker and Z. Ghahramani, MIT Press, 2002,

M. Figueiredo and A. K. Jain, Unsupervised Learning of Finite Mixture Models, IEEE TPAMI, 2002

A. L. N. Fred, J. M. N. Leitão, A New Cluster Isolation Criterion Based on Dissimilarity Increments, IEEE Trans. On Pattern Analysis and Machine Intelligence, vol 25, NO. 8, pp 2003.