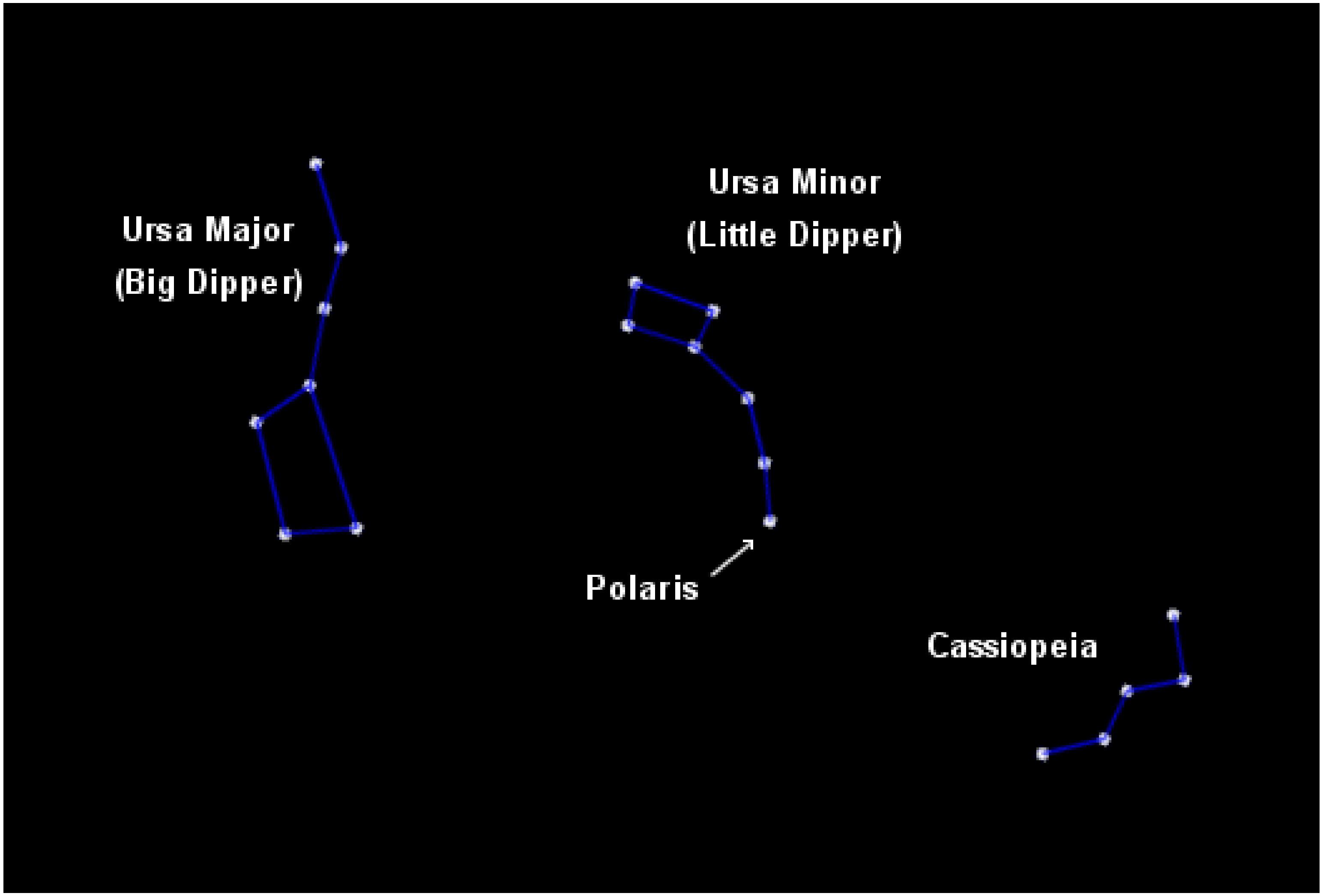
Cluster Analysis

Nazerfard, Ehsan
nazerfard@aut.ac.ir

Summer Sky



https://www.synapticsystems.com/sky/learnsky.html

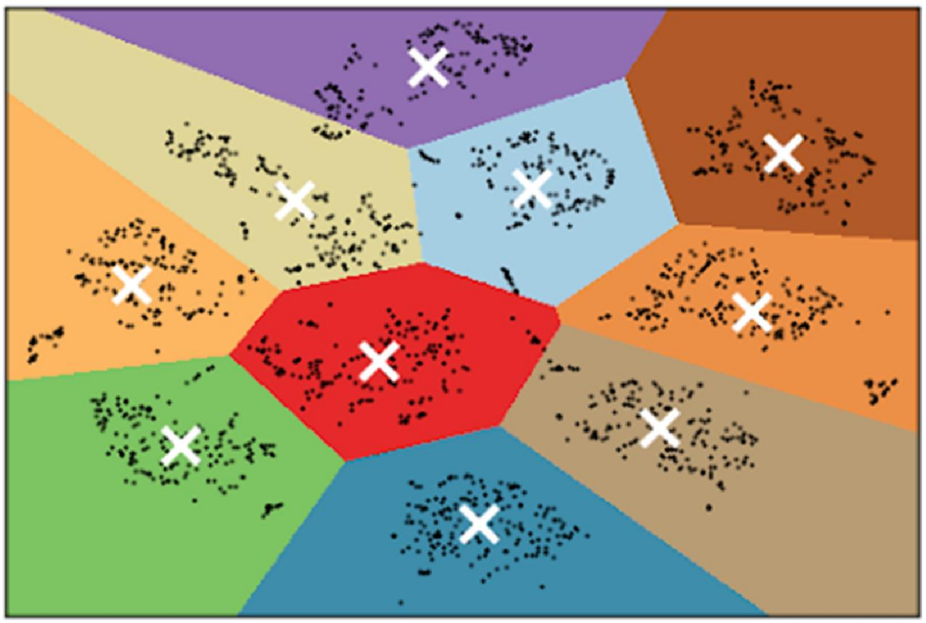
Major Clustering Approaches

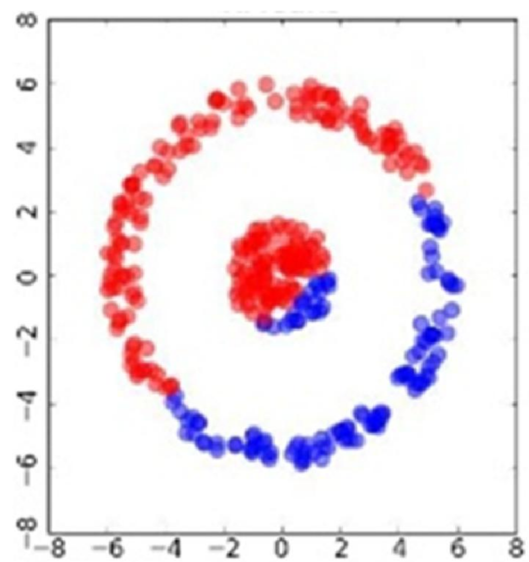
- ✓ Partitioning-based approach
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors.
  - Typical methods: k-means, k-medoids, CLARA, CLARANS
- Density-based approach
  - Based on connectivity and density functions
  - Typical methods: DBSCAN, OPTICS, DenClue
- Hierarchical approach
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Agnes, Diana, BIRCH, CURE, CHAMELEON
- Model-based approach
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
  - Typical methods: EM, SOM

Major Clustering Approaches (cont.)

- Grid-based approach
  - Based on a multiple-level granularity structure
  - Typical methods: STING, CLIQUE, WaveCluster
- Frequent Pattern-based approach
  - Based on the analysis of frequent patterns
  - Typical methods: p-Cluster
- Support Vector approach
  - Based on the idea of mapping data points into higher dimensional feature space via a kernel function.
  - Typical methods: SVC, Kernel K-means
- Graph Theoretic approach
  - Typical methods: Spectral Clustering
- …

When K-means clustering fails





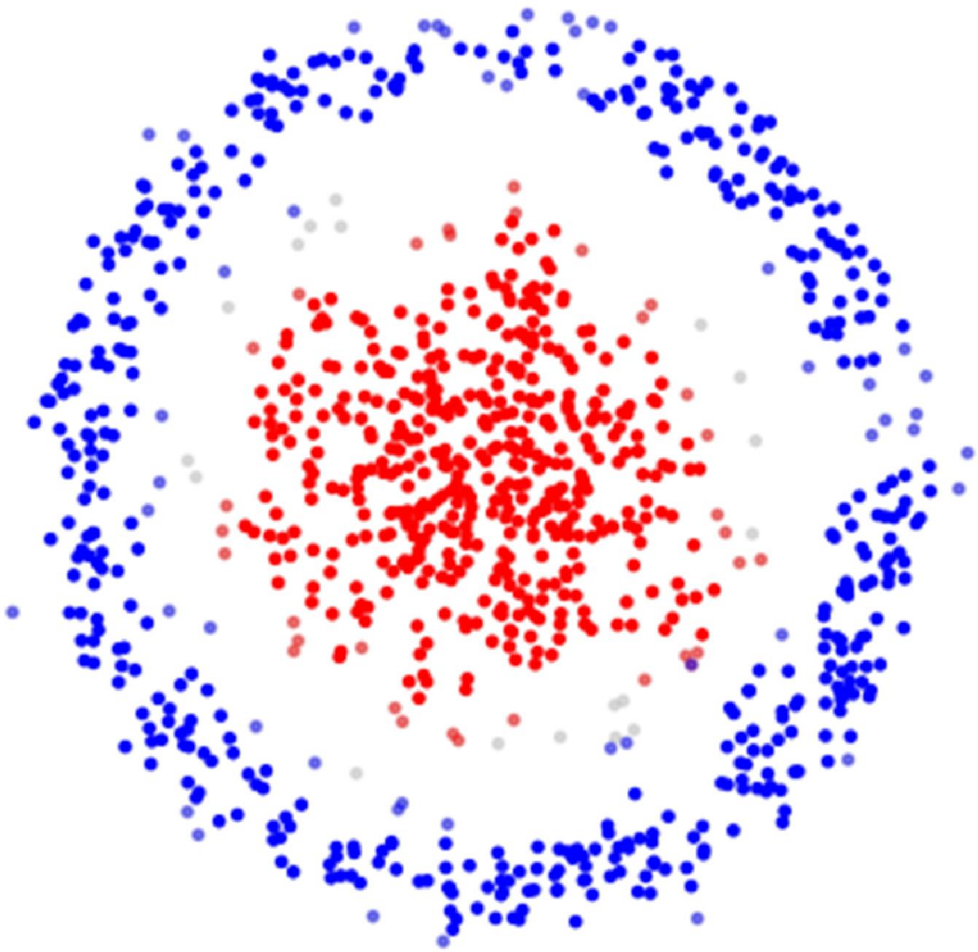
Density-based Approach

- It is based on connectivity and density functions

- Example: DBSCAN

DBSCAN

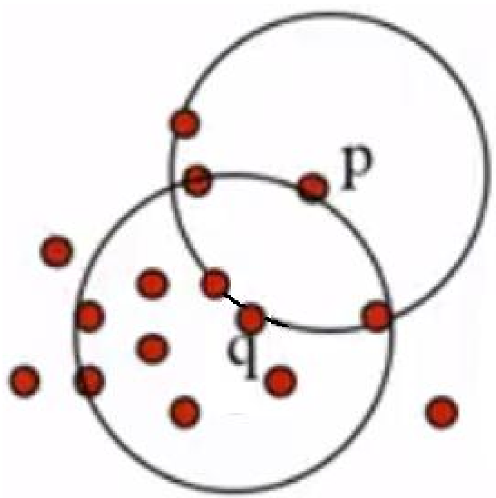
- Density-Based Spatial Clustering of Applications with Noise



Density Definition

- Cluster: maximal set of density-connected points
- Parameters:
  - ϵ: max radius of the neighborhood
  - minPts: min # of points in a ϵ-neighborhood of a point
    - The ϵ-neighborhood of a point q:

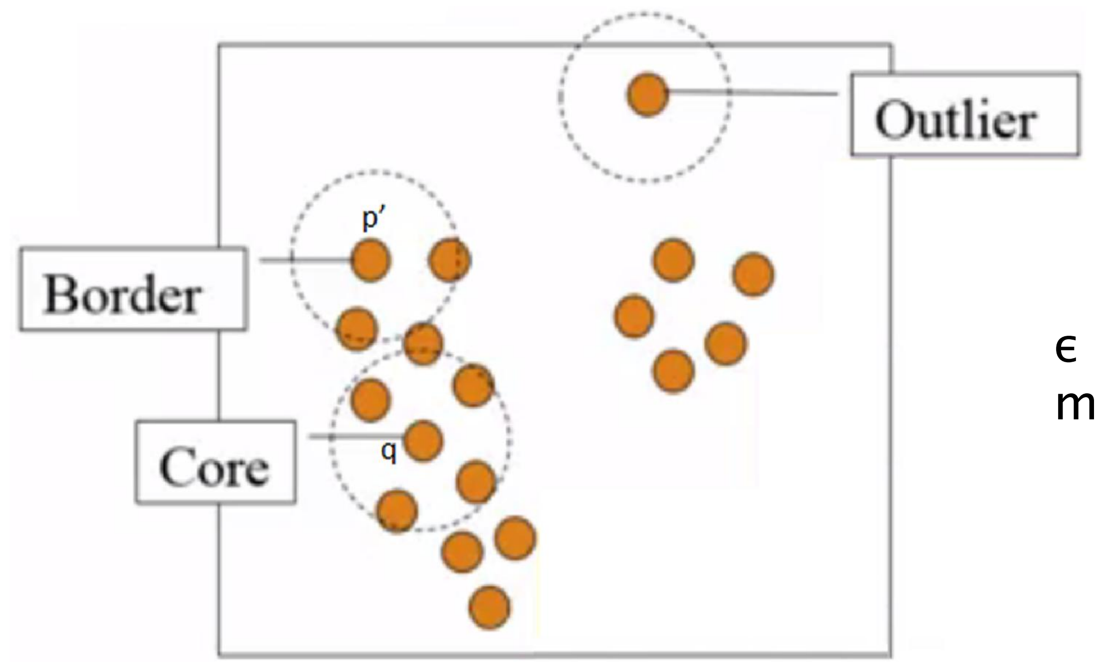
$$N_\epsilon(q) = \{p \text{ in } D \mid d(p,q) \le \epsilon\}$$



ϵ = 1 unit
minPts = 5

Core, Border, and Outlier

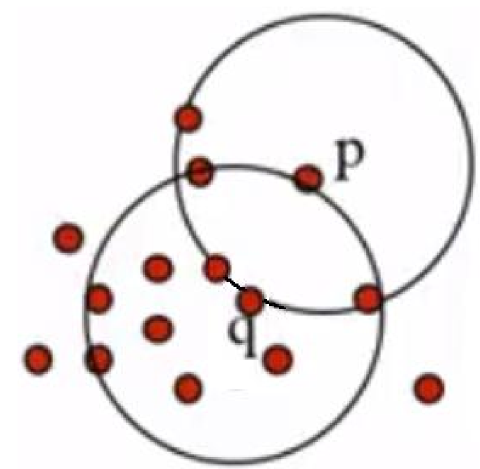
- Core point (q): dense neighborhood
- Border point (p’): in cluster, but neighborhood is not dense (reachable by the cluster)
- Outlier/noise: not in a cluster



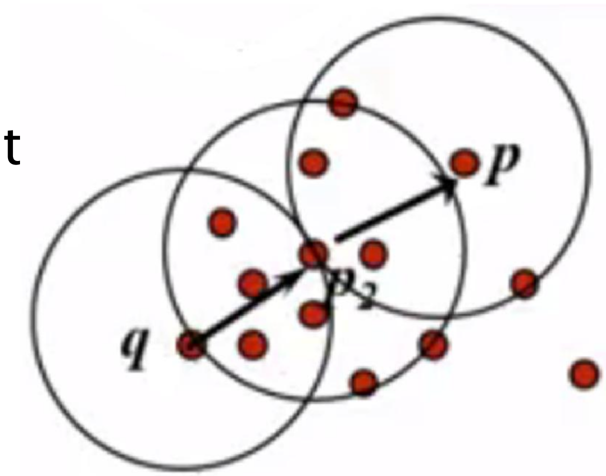
ϵ = 1 unit
minPts = 5

Density Reachability

- Directly density-reachable (ϵ, minPts):
  - A point p is density directly-reachable from a point q if:
    - p in $N_\epsilon(q)$
    - Core point condition: $|N_\epsilon(q)| \geq$ minPts



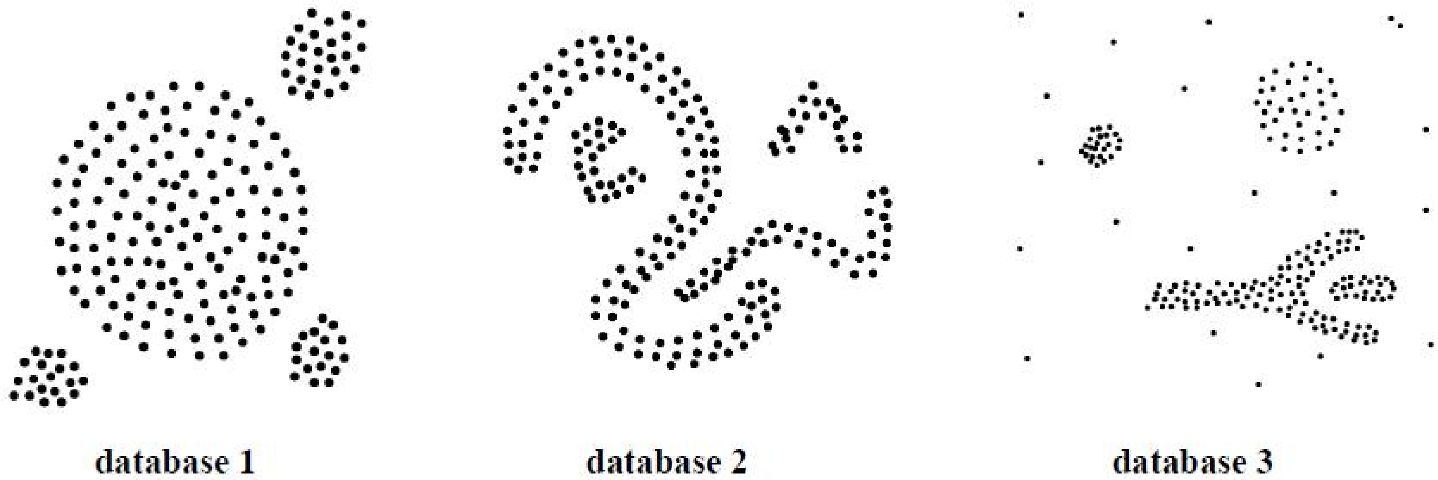
- Density-reachable (ϵ, minPts):
  - A point p is directly-reachable from a point q if there is a chain of points $p_1, \ldots, p_n$, $p_1=q$, $p_n=p$ such that p_{i+1} is directly density-reachable from p_i.



The DBSCAN Algorithm

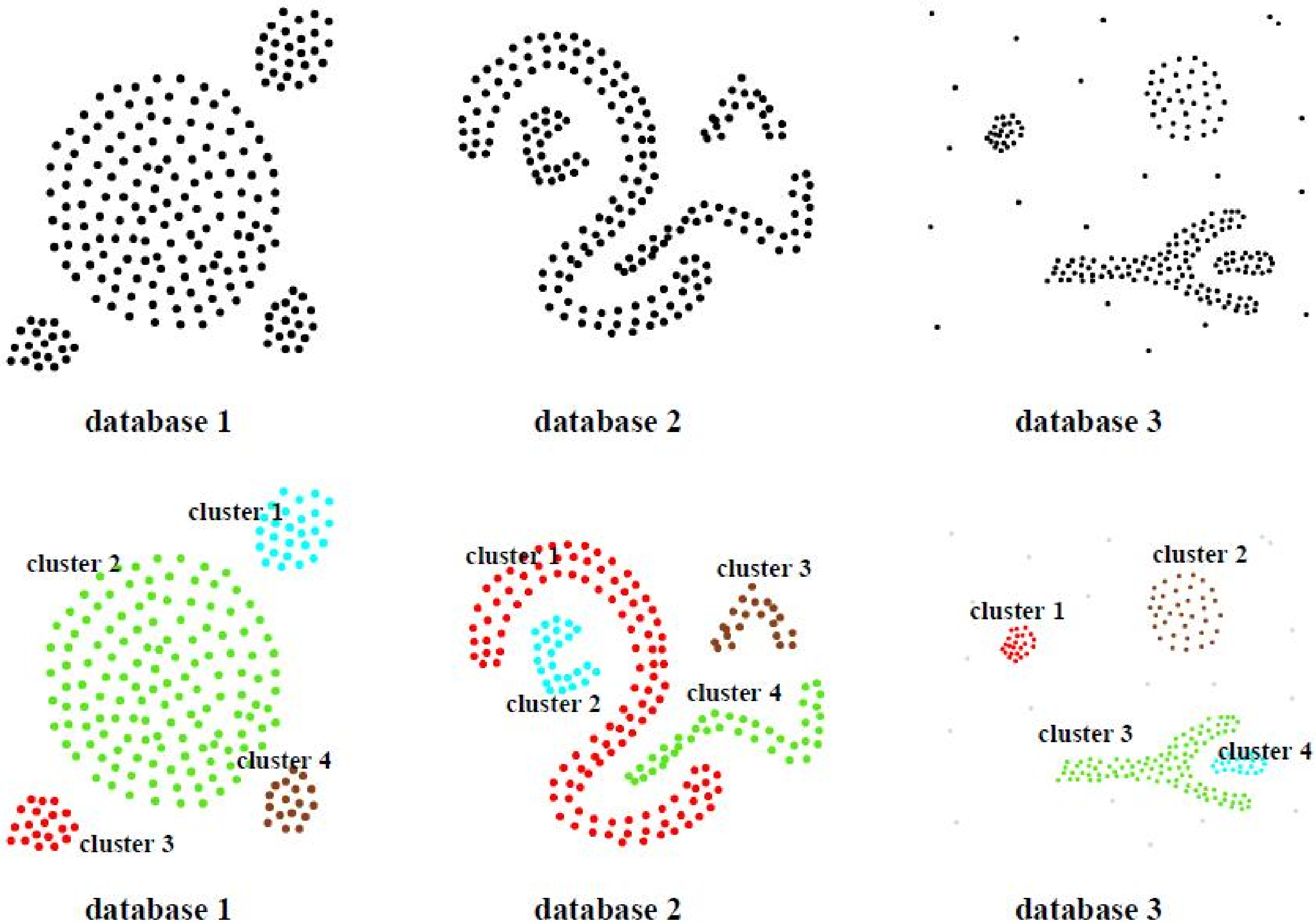
- The DBSCAN algorithm (ϵ, minPts):
  - Start with an arbitrary point p from the database
  - Retrieve all density-reachable points from p
    - If p is a core point, a cluster is formed
    - If p is a border point, no points are density reachable from p. Thus DBSCAN visits the next points of the database
  - Continue the process until all of the points have been processed.

Example – Original Points



M. Ester et al., A density-based algorithm for discovering clusters in large spatial databases with noise, 1996

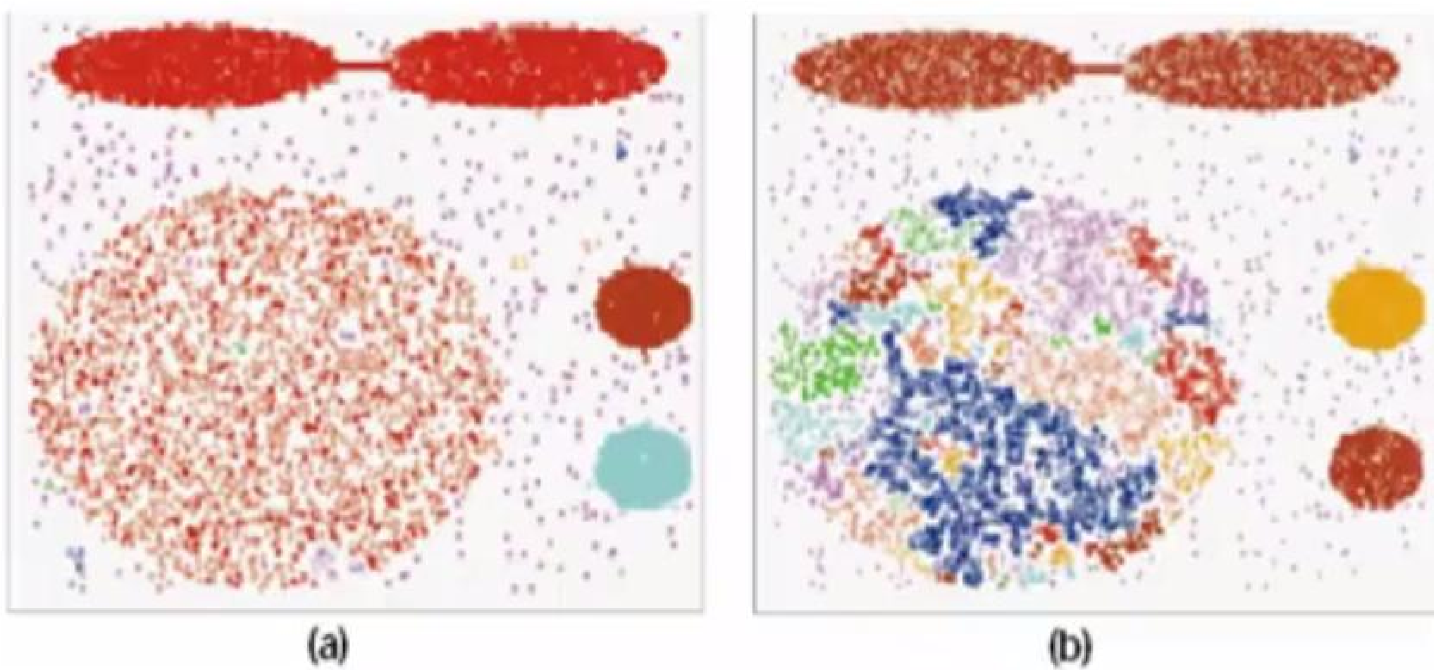
Discovered Clusters



M. Ester et al., A density-based algorithm for discovering clusters in large spatial databases with noise, 1996

Sensitivity to Parameters

- minPts = 4
  - (a) $\epsilon = 0.5$
  - (b) $\epsilon = 0.4$



J. Sander et al., The Algorithm GDBSCAN and Its Applications, 1998

Further Reading

- The OPTICS Clustering Algorithm
  - M. Ankerst, MM. Breunig, H-P. Kriegel, J. Sander, OPTICS: Ordering Points To Identify the Clustering Structure, ACM SIGMOD international conference on Management of data, pp. 49-60, 1999.

References

- M. Ester, H-P. Kriegel , J. Sander, X. Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise", 1996.

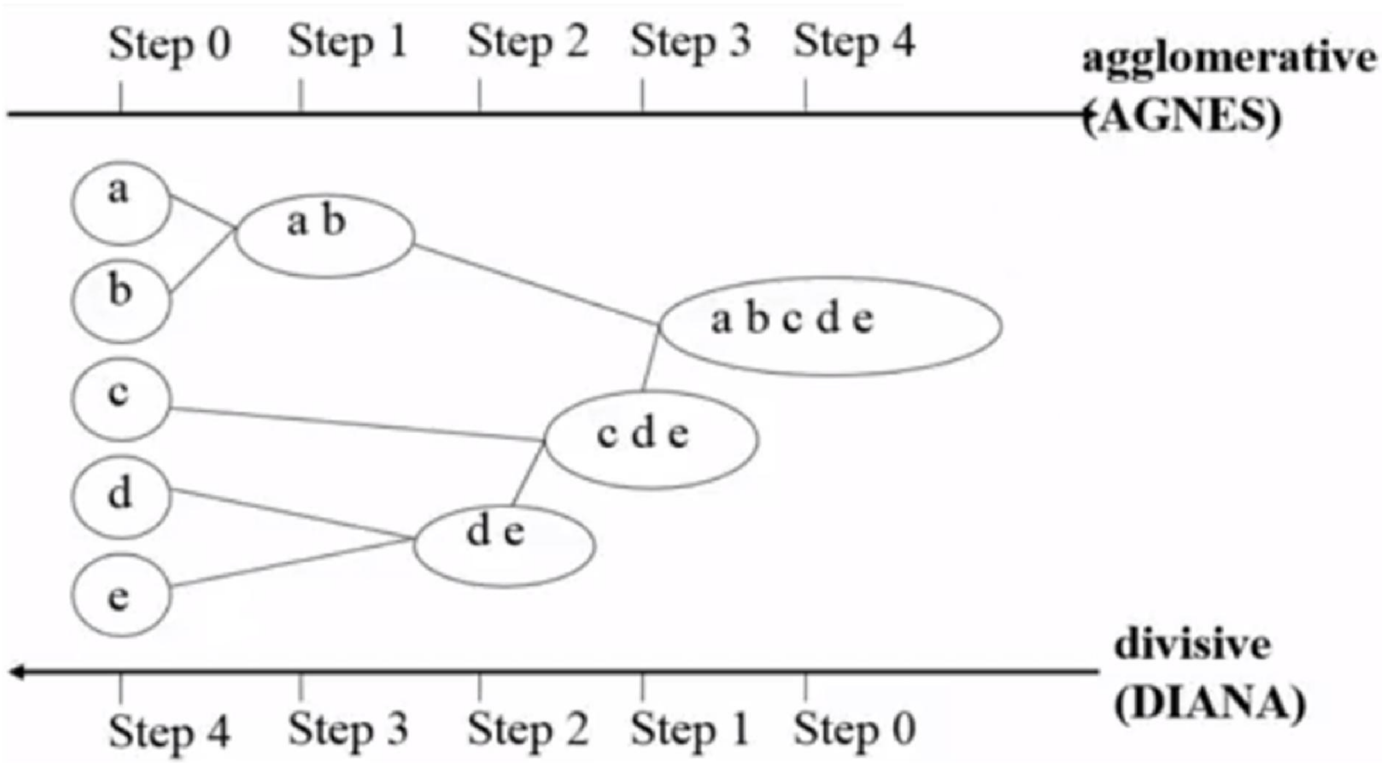
  [Highest impact paper award, 2014]
- J. Sander, M. Ester, H-P. Kriegel, X. Xu. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications, 1998.

Hierarchical Approach

- Create a hierarchical decomposition of the set of data (or objects) using some criterion

- Two categories:
  - Agglomerative (bottom-up)
  - Divisive (top-down)

Dendrogram



AGNES (Agglomerative Nesting)
DIANA (Divisive Analysis)

Agglomerative Clustering

- The algorithm:
  - Start with a collection C on n singleton clusters
    - Each cluster contains one data point: $c_i = \{\boldsymbol{x_i}\}$
  - Repeat
    - Find a pair of clusters that are closest: $\min D(c_i, c_j)$
    - Merge the clusters c_i and c_j into a new cluster c_{i+j}
    - Remove c_i and c_j from the collection C, then add c_{i+j}
  - Until only one cluster left

Example

- Cluster distance measure: single link

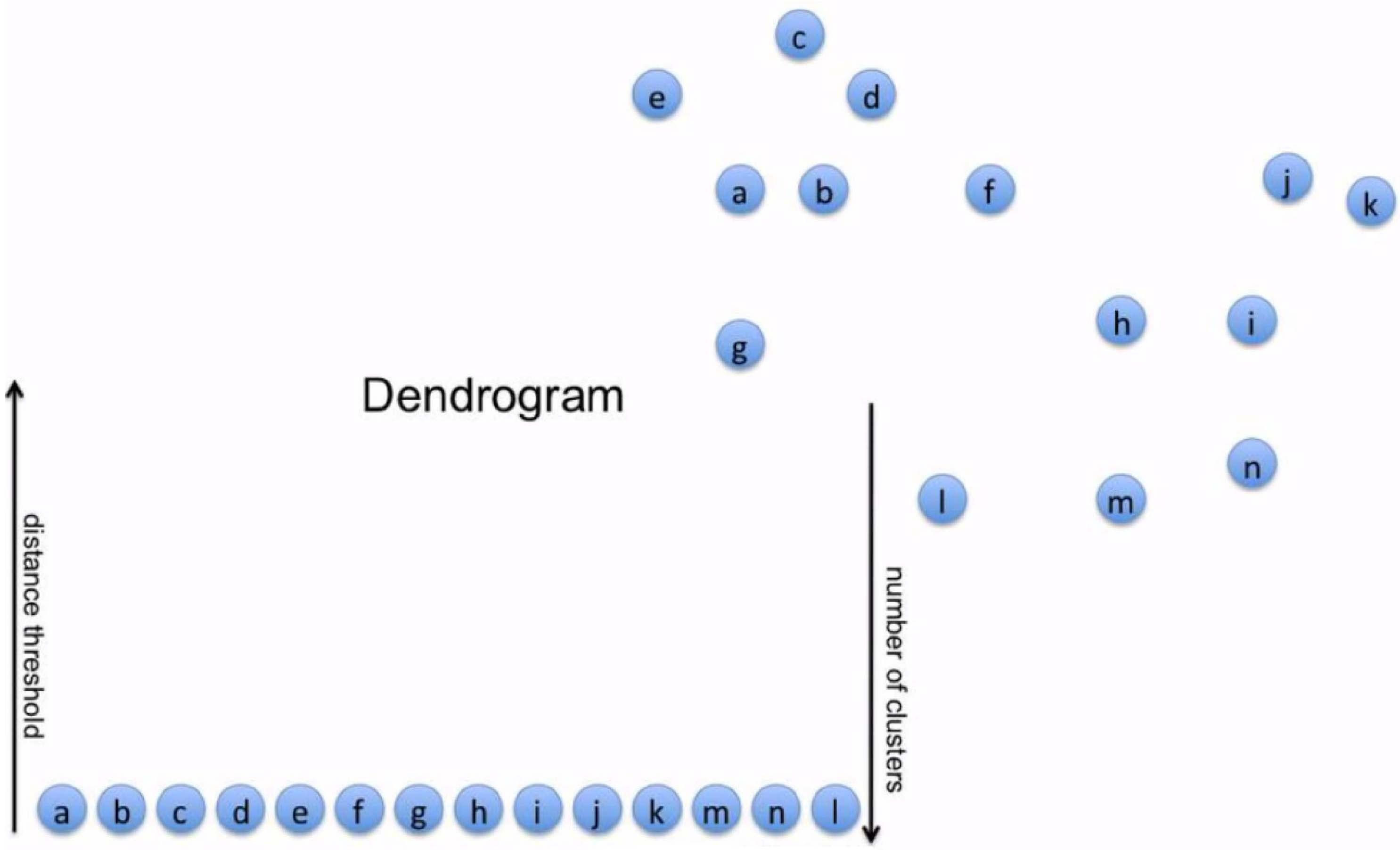


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Example (cont.)

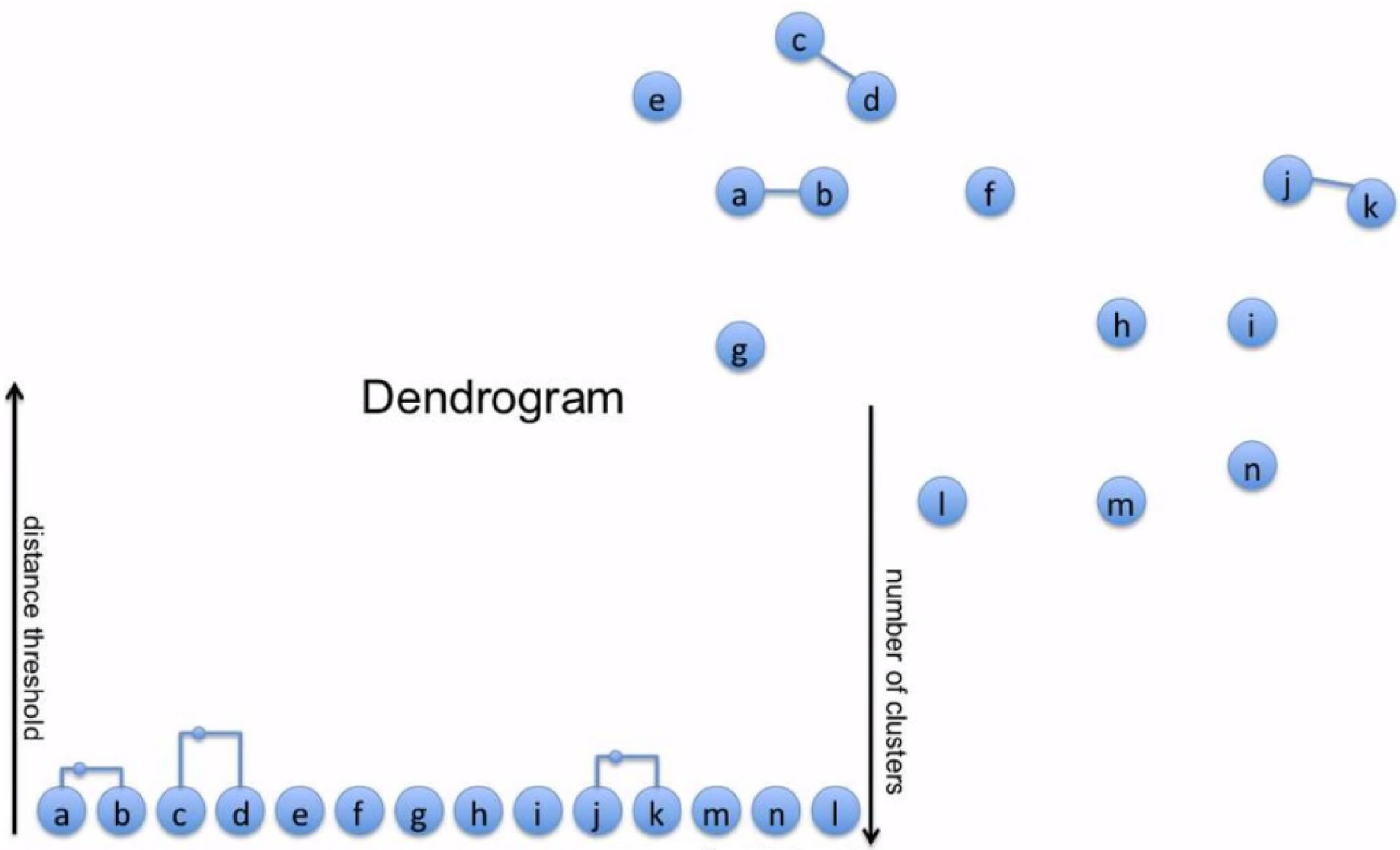


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Example (cont.)

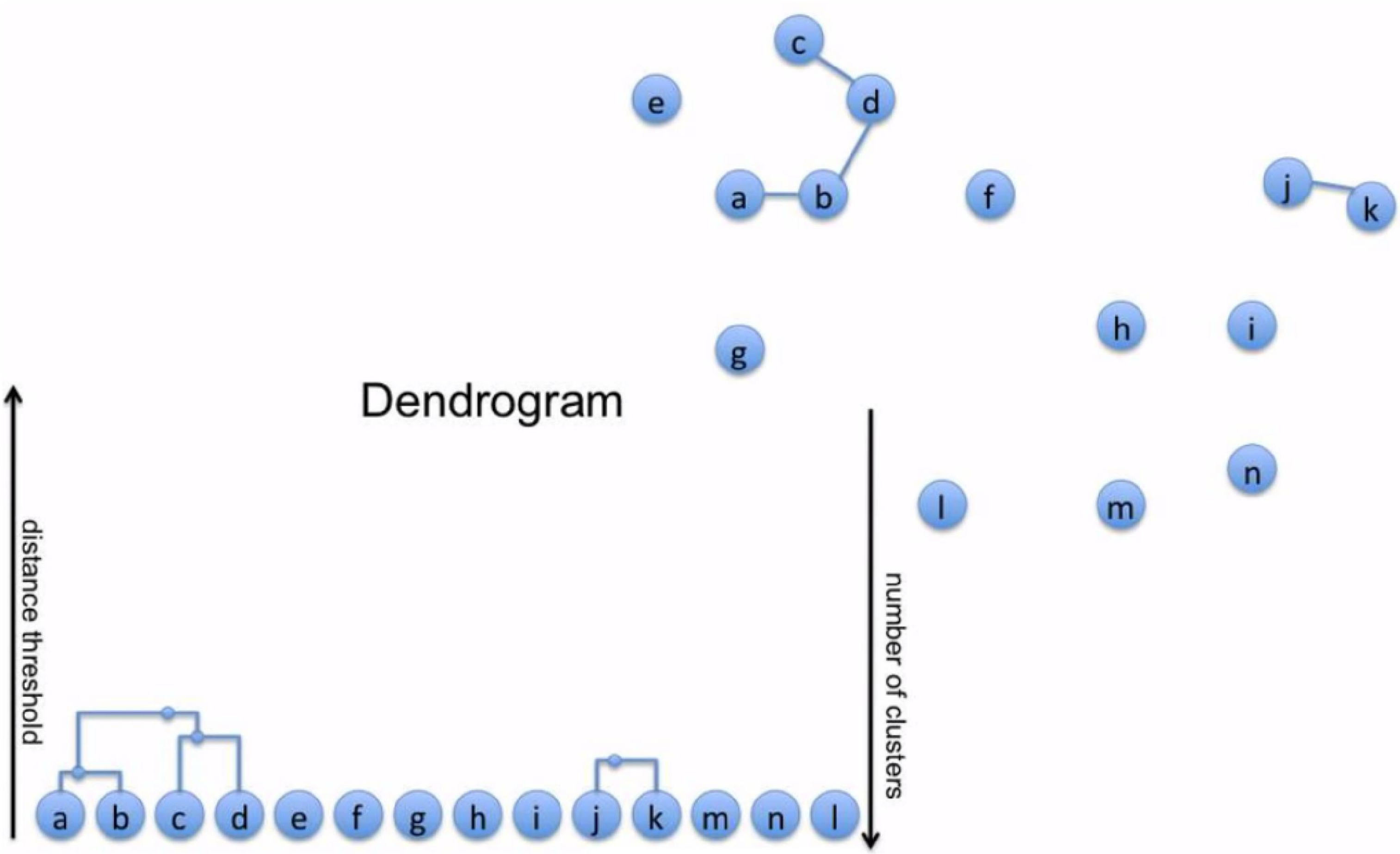


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Example (cont.)

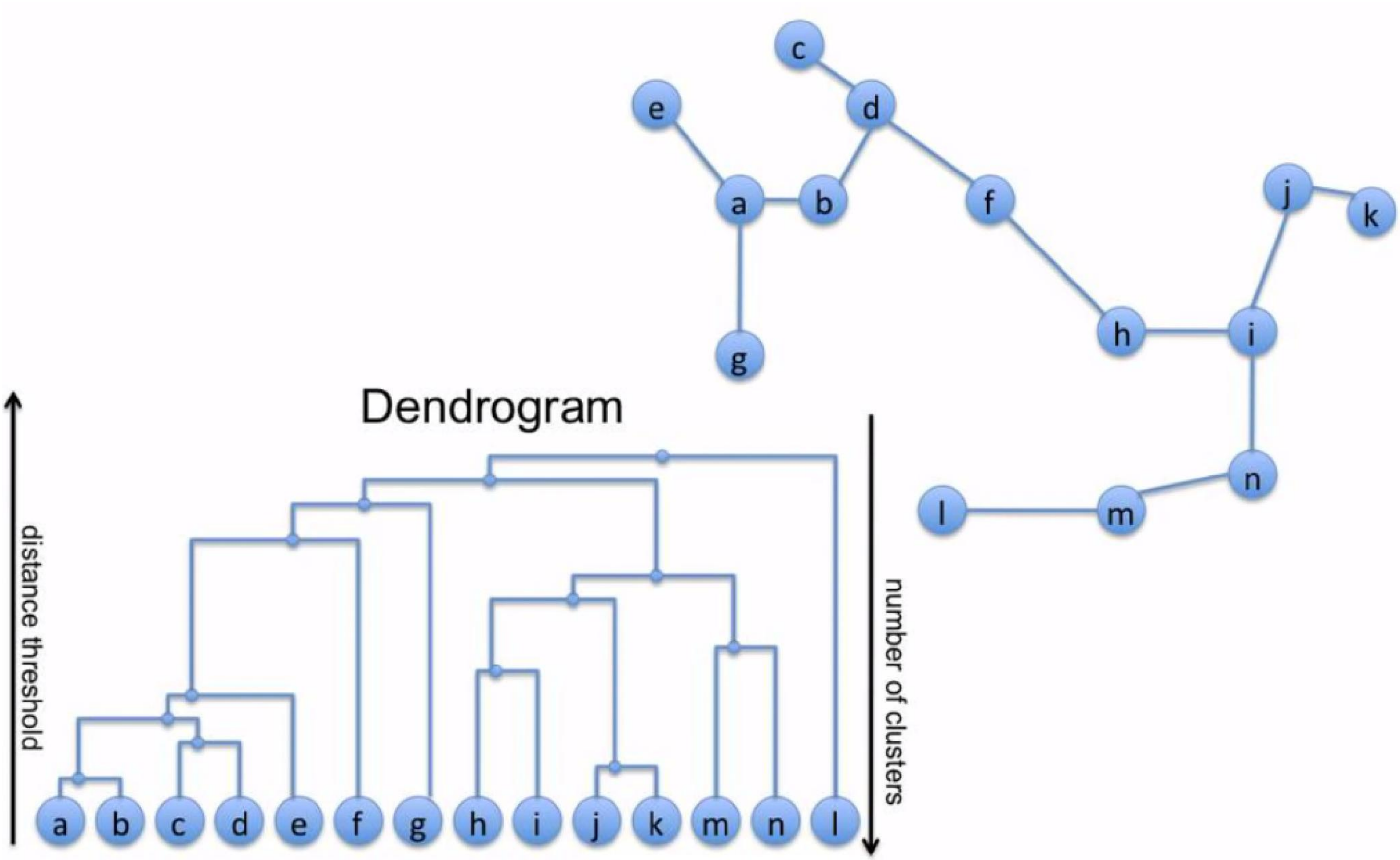


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Example (cont.)

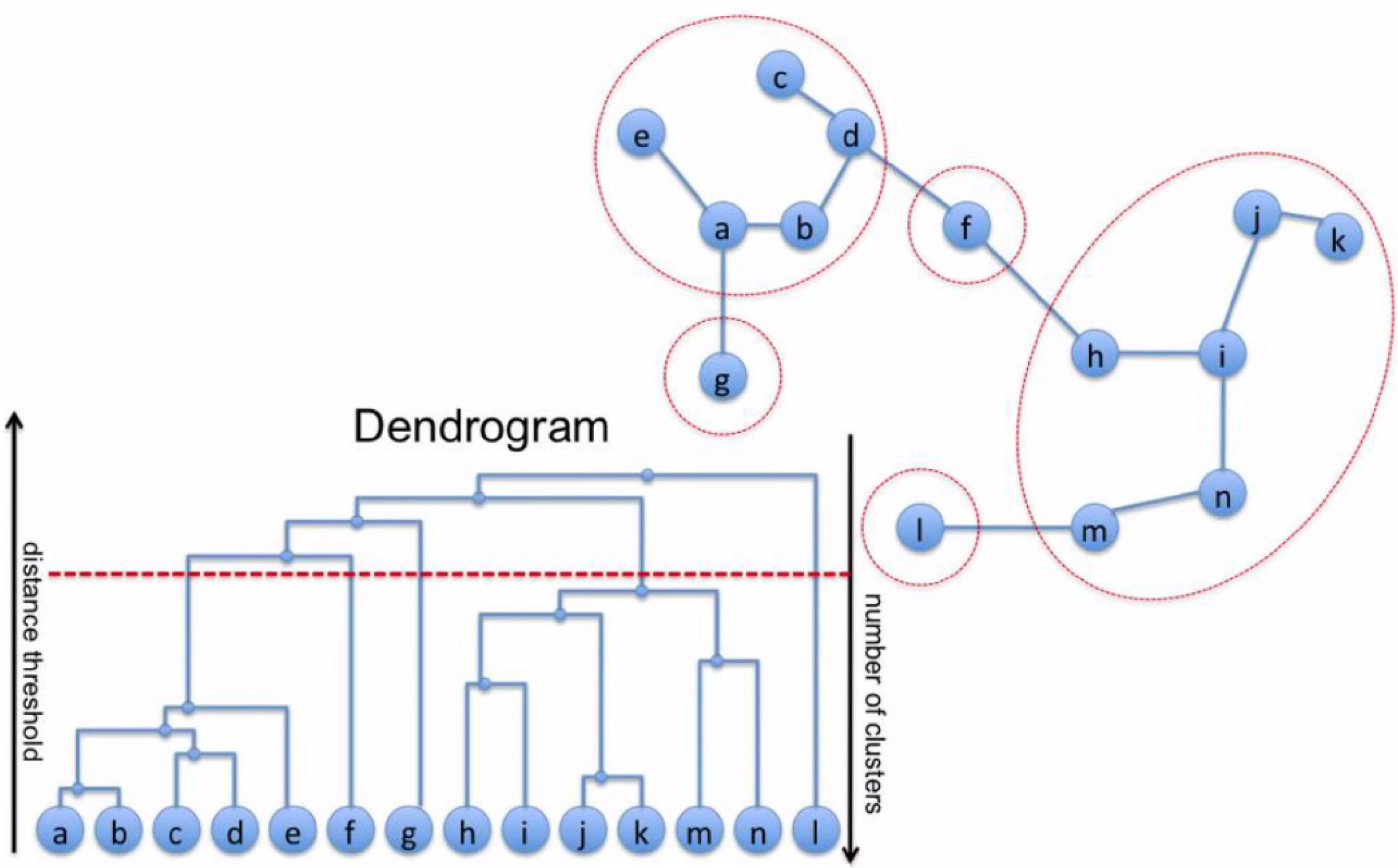


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Cluster Distance Measure

- Single link: $D(C_1, C_2) = \min(\boldsymbol{x_1}, \boldsymbol{x_2})$
  - Distance between the closest elements in clusters
- Complete link: $D(C_1, C_2) = \mathrm{m}ax(\boldsymbol{x_1}, \boldsymbol{x_2})$
  - Distance between the farthest elements in clusters
- Average link: average all pairwise distances
  - Less affected by outliers
- Centroids: distance between centroids (mean) of clusters
- Ward's method: sum of deviations from the centroid
  - The smaller, the better for merging.

Further Reading

- The Birch Clustering Algorithm
  - T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering method for very large databases, Proceedings of the ACM SIGMOD international conference on Management of data, SIGMOD, pp 103-114, 1996.

- The CHAMELEON Clustering Algorithm
  - G. Karypis, E-H. Han, V. Kumar, CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling, IEEE Computer 32(8): 68-75, 1999.

Model-based Approach

- A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other

- Example: EM / GM Clustering
  - EM: Expectation-Maximization
  - Gaussian Mixtures

Mixture Models in 1-D

- Case 1
  - Assume the data points come from two Gaussians with unknown parameters
  - We know each point comes from which Gaussian

Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Mixture Models in 1-D

- Case 1
  - Assume the data points come from two Gaussians with unknown parameters
  - We know each point comes from which Gaussian

$$\mu_b = \frac{x_1 + x_2 + \ldots + x_{n_b}}{n_b}$$

$$\sigma_b^2 = \frac{(x_1 - \mu_b)^2 + \ldots + (x_n - \mu_b)^2}{n_b}$$

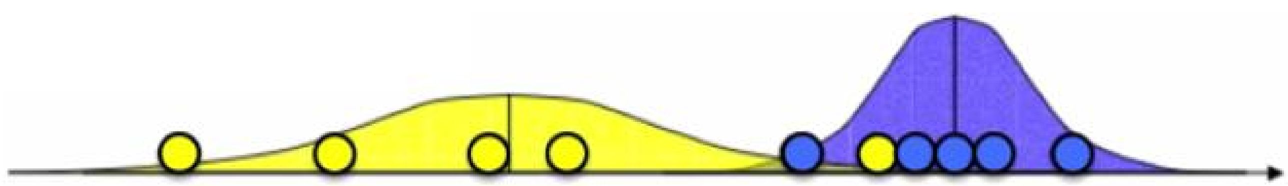


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Mixture Models of 1-D

- Case 2
  - We don’t know the sources
  - We know the parameters of the Gaussians

Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Mixture Models of 1-D

- Case 2
  - We don’t know the sources
  - We know the parameters of the Gaussians

$$P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_i)^2}{2\sigma_b^2}\right\}$$

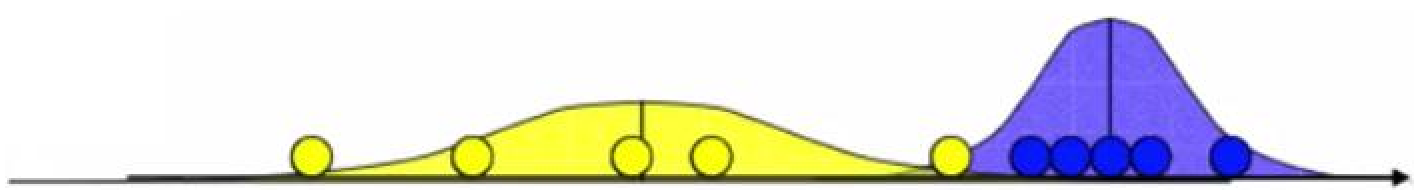


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

Expectation-Maximization

- Chicken and egg problem!
  - We need parameters to guess source of points
  - We need to know source to estimate parameters
- EM algorithm
  - Start with two randomly placed Gaussians
  - REPEAT
    - Use the Gaussians to determine which point comes from which Gaussian (Case 2)
    - Adjust parameters to fit points assigned to them (Case 1)
  - UNTIL Convergence

Expectation-Maximization (cont.)

- Chicken and egg problem!
  - We need parameters to guess source of points
  - We need to know source to estimate parameters
- EM algorithm
  - Start with two randomly placed Gaussians
  - REPEAT
    - Use the Gaussians to determine which point comes from which Gaussian (E-step)
    - Adjust parameters to fit points assigned to them (M-step)
  - UNTIL Convergence

EM Visualization

- Initialization

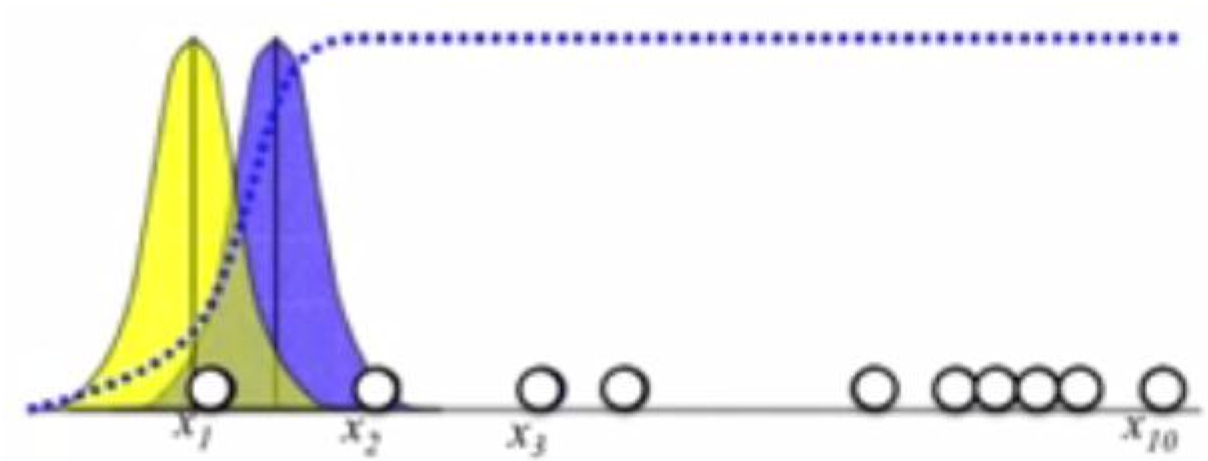


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

EM Visualization (cont.)

- Initialization

$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

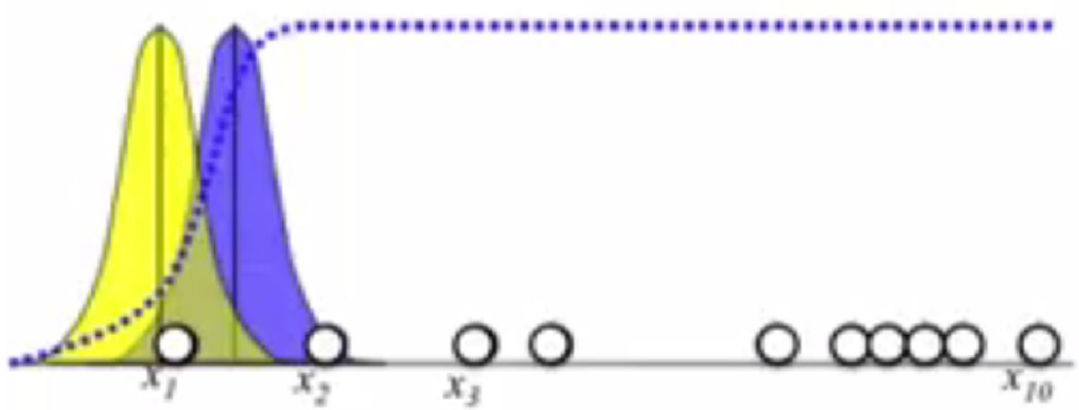


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

EM Visualization (E-Step)

$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

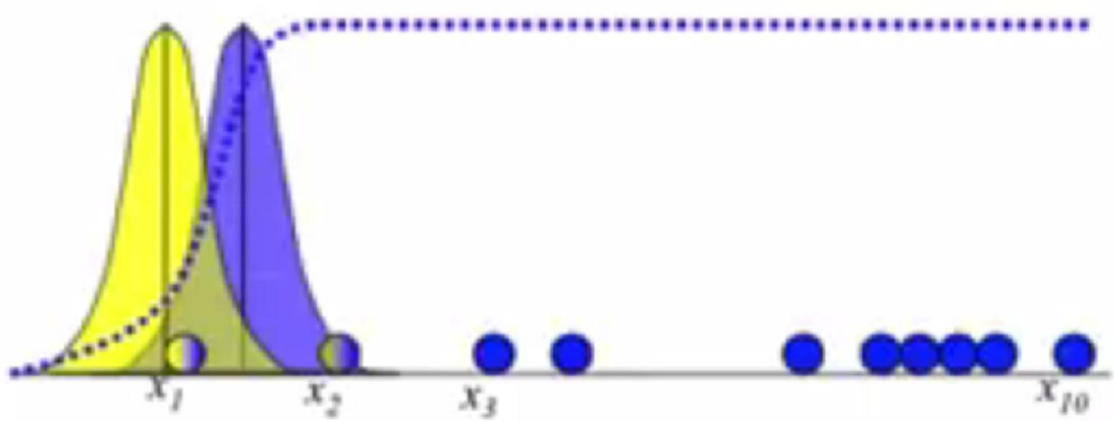
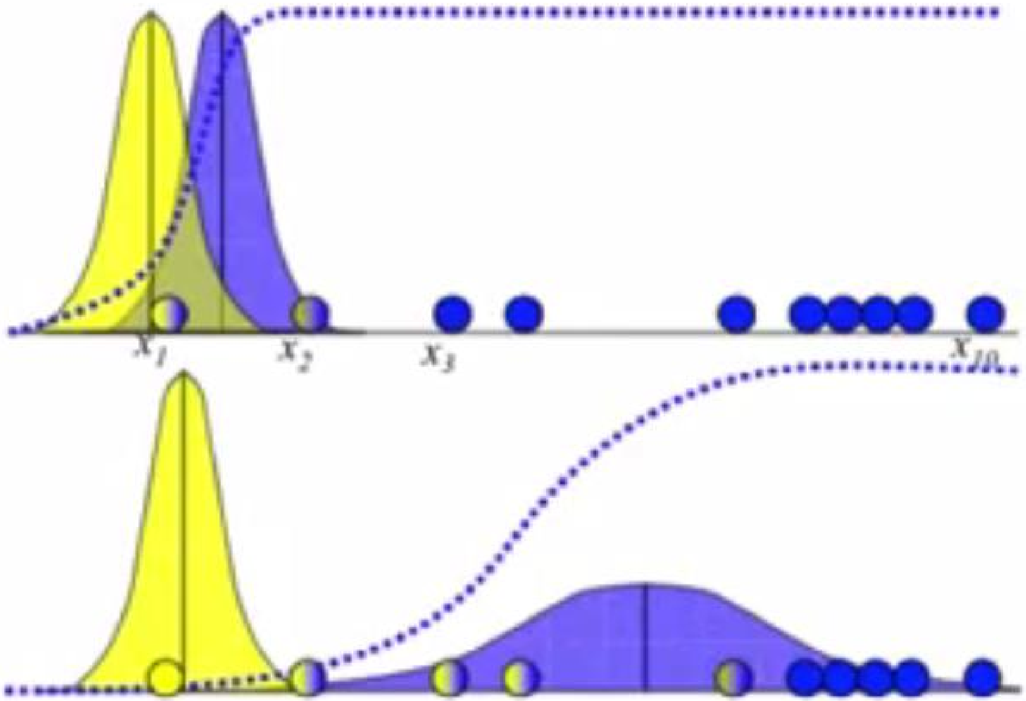


Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

EM Visualization (M-Step)



$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \ldots + b_n x_n}{b_1 + b_2 + \ldots + b_n}$$

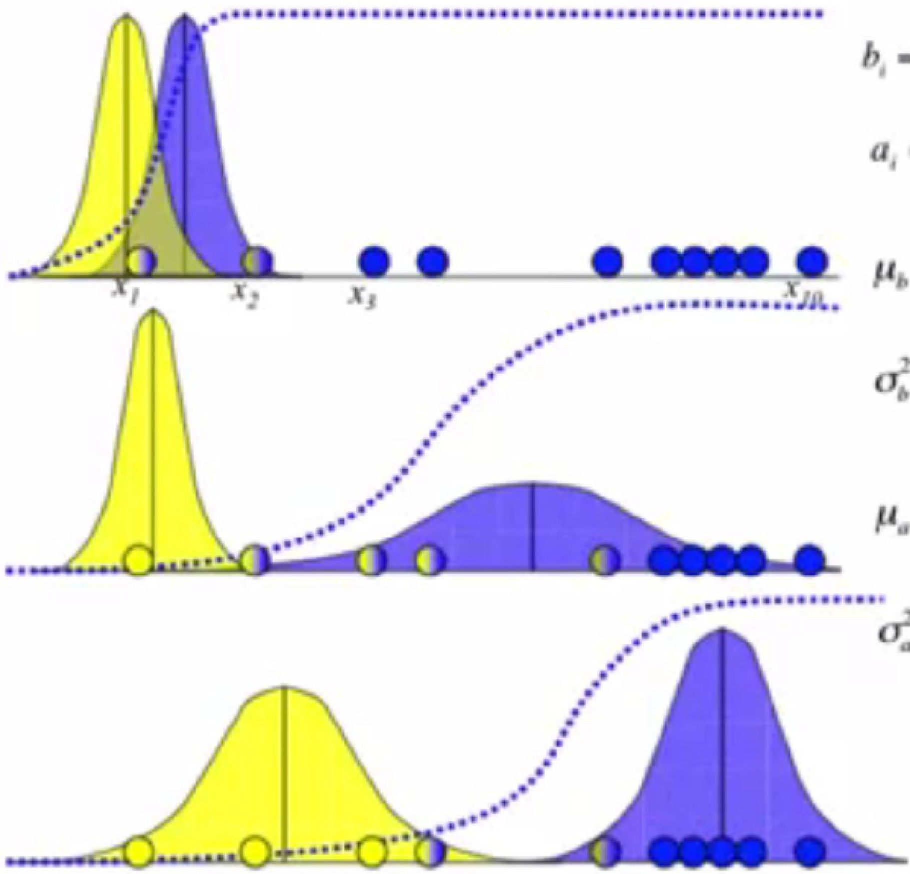
$$\sigma_b^2 = \frac{b_1(x_1 - \mu_b)^2 + \ldots + b_n(x_n - \mu_b)^2}{b_1 + b_2 + \ldots + b_n}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \ldots + a_n x_n}{a_1 + a_2 + \ldots + a_n}$$

$$\sigma_a^2 = \frac{a_1(x_1 - \mu_a)^2 + \ldots + a_n(x_n - \mu_a)^2}{a_1 + a_2 + \ldots + a_n}$$

Figure © V. Lavrenko, Lecture on Clustering, University of of Edinburg

EM Visualization (E-Step)



$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \ldots + b_n x_n}{b_1 + b_2 + \ldots + b_n}$$

$$\sigma_b^2 = \frac{b_1(x_1 - \mu_b)^2 + \ldots + b_n(x_n - \mu_b)^2}{b_1 + b_2 + \ldots + b_n}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \ldots + a_n x_n}{a_1 + a_2 + \ldots + a_n}$$

$$\sigma_a^2 = \frac{a_1(x_1 - \mu_a)^2 + \ldots + a_n(x_n - \mu_a)^2}{a_1 + a_2 + \ldots + a_n}$$

EM Clustering

- Soft computing clustering
  - Clusters overlap
- K-means clustering is a special case of (hard) EM.
- Sensitive to the initialization
- How to pick K?

Further Reading

- The SOM and Spectral Clustering Algorithms
- Grid-based Clustering
- Subspace Clustering
- Collaborative Clustering
- …

Measuring Clustering Quality

- Some issues when dealing with cluster validation:
  - Clustering tendency
  - Number of clusters required
  - How well it fits the data, without reference to any external information (labels).
  - Comparing the cluster analysis with any external information.
  - We compare two different clusters to determine which one is better.

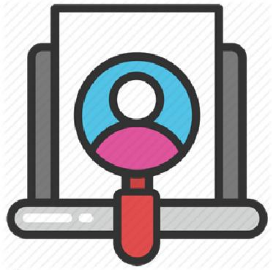
Measuring Clustering Quality (cont.)

- Unsupervised measure (internal)
  - Cluster cohesion & cluster separation
    - SSE, Silhouette coefficient

- Supervised measure (external)
  - Entropy, Purity, Rand index
    - We can also use the clustering results to solve another problem, such as classification.

- Relative approach
  - Combination of supervised and unsupervised approaches.

Measuring Clustering Quality (cont.)

- Clustering quality:
  - User inspection → 
  - Sum of squared error → $$SSE = \sum_{j=1}^{k} \sum_{x \in C_j} \left\| \boldsymbol{x} - \boldsymbol{\mu}_j \right\|^2$$
  - Purity
  - Entropy → $$E_{total}(\Omega) = \sum_{w \in \Omega} \frac{m_w}{m} \times E(w)$$
  - Silhouette Index → …
  - …

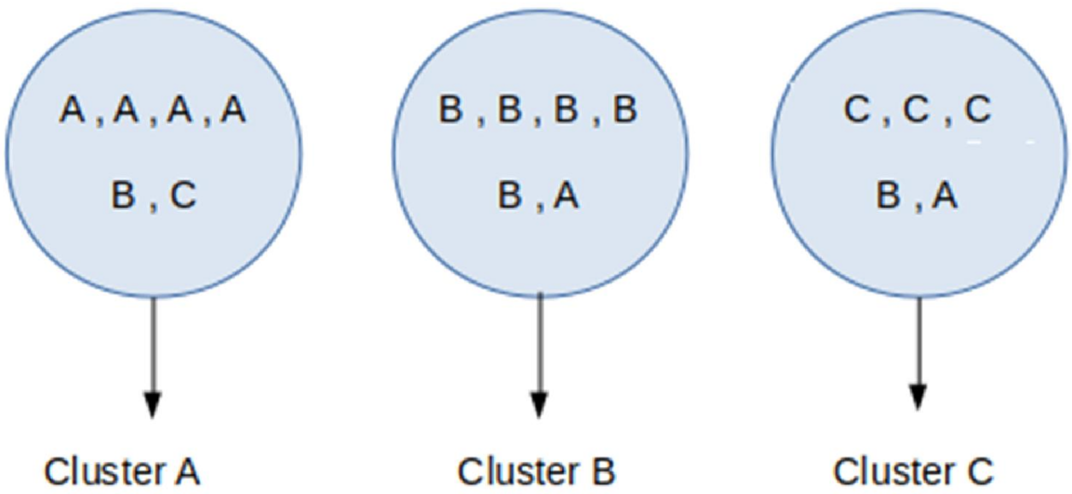
Purity

- We assign a label to each cluster based on the most frequent class in it.

- The purity becomes the number of correctly matched class and cluster labels divided by the number of total data points.

Purity (cont.)

$$Purity = \frac{cluster\ A + cluster\ B + cluster\ c}{total} = \frac{4+5+3}{17} = 0.71$$



Entropy

- Total clustering entropy:

$$E_{total}(\Omega) = \sum_{w_i \in \Omega} \frac{m_{w_i}}{m} \times E(w_i) \, ,$$

where $\Omega = \{w_1, w_2, \dots, w_k\}$,
m_{w_i} is the number of points in cluster w_i,
m is the total number of points

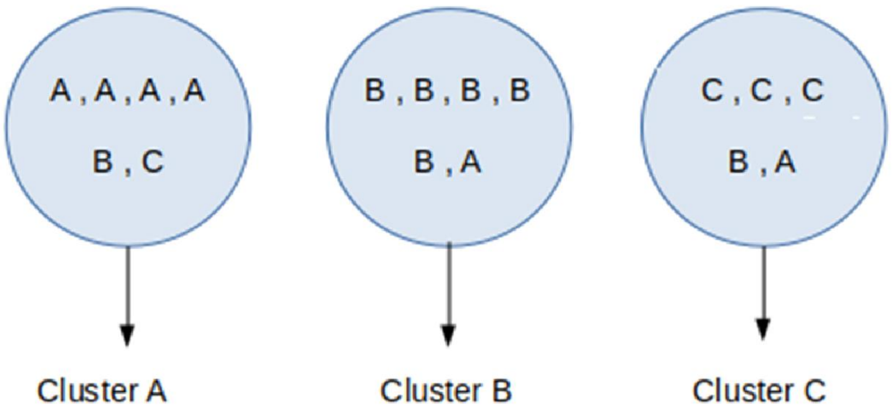
- Entropy of a single cluster w_i:

$$E(w_i) = -\sum_{c \in C} \frac{|w_{i,c}|}{m_{w_i}} \times \log_2 \frac{|w_{i,c}|}{m_{w_i}} \; ,$$

where c is a class in the set C of all classes,
$|w_{i,c}|$ is the number of points classified as c in cluster w_i

Entropy (cont.)

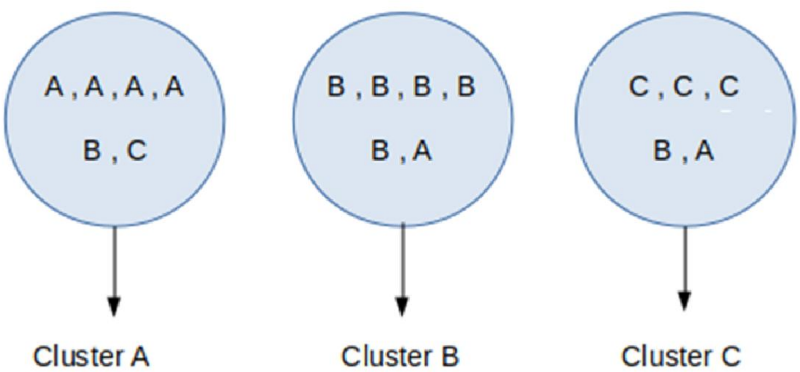
- $E(w_A) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{1}{6}\log_2\frac{1}{6} - \frac{1}{6}\log_2\frac{1}{6} = 1.25$
- $E(w_B) = -\frac{5}{6}\log_2\frac{5}{6} - \frac{1}{6}\log_2\frac{1}{6} = 0.65$
- $E(w_C) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{1}{5}\log_2\frac{1}{5} - \frac{1}{5}\log_2\frac{1}{5} = 1.37$



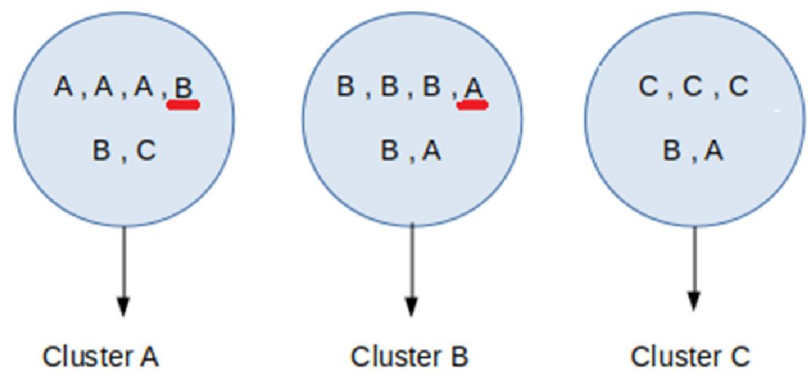
- $E_{total}(\Omega) = \sum_{w\in\Omega}\frac{m_{w_i}}{m} \times E(w_i) = \frac{6}{17} \times E(w_A) + \frac{6}{17} \times E(W_B) + \frac{5}{17} \times E(w_c) = 1.07$

Entropy (cont.)

$$E_{total}(\Omega) = 1.07$$



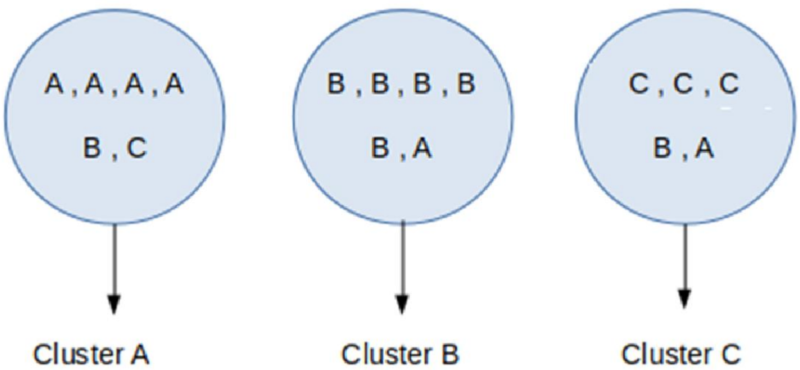
$$E_{total}(\Omega) = 1.43$$



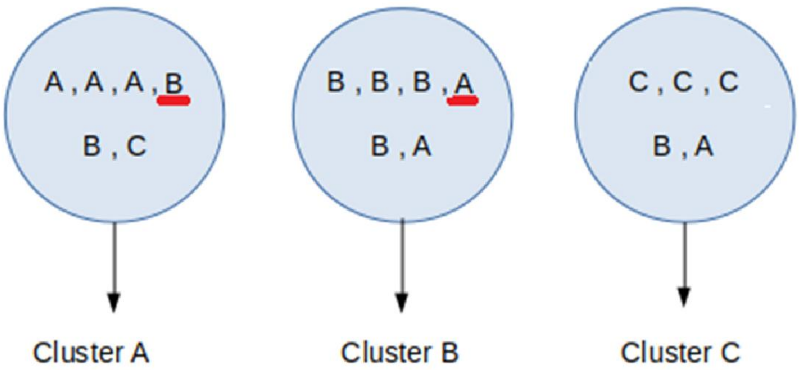
Entropy (cont.)

- The smaller values of overall entropy indicates a better clustering.

$$\boxed{E_{total}(\Omega) = 1.07}$$



$$E_{total}(\Omega) = 1.43$$



Further Reading

- Silhouette index
- Normalized Mutual Information (NMI)
- Rand index
- …

References

- Jiawei Han, Micheline Kamber and Jian Pei, Data Mining: Concepts and Techniques, 3rd edition, 2006.
- M. Ankerst, MM. Breunig, H-P. Kriegel, J. Sander, OPTICS: Ordering Points To Identify the Clustering Structure, ACM SIGMOD international conference on Management of data, pp. 49-60, 1999.
- M. Ester, H-P. Kriegel , J. Sander, X. Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise", 1996.
- J. Sander, M. Ester, H-P. Kriegel, X. Xu. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications, 1998.
- T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering method for very large databases, Proceedings of the ACM SIGMOD international conference on Management of data, SIGMOD, pp 103-114, 1996.
- G. Karypis, E-H. Han, V. Kumar, CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling, IEEE Computer 32(8): 68-75, 1999.