Chapter DM:II (continued)

II. Cluster Analysis

- ❑ Cluster Analysis Basics
- ❑ Hierarchical Cluster Analysis
- ❑ Iterative Cluster Analysis
- ❑ Density-Based Cluster Analysis
- ❑ Cluster Evaluation
- ❑ Constrained Cluster Analysis

Iterative Cluster Analysis

Merging Principles

- Cluster analysis
  - hierarchical
    - agglomerative — *single link, group average*
    - divisive — *MinCut*
  - iterative
    - exemplar-based — *k-means, k-medoid*
    - exchange-based — *Kerninghan-Lin*
  - density-based
    - point-density-based — *DBSCAN*
    - attraction-based — *MajorClust*
  - meta-search-controlled
    - gradient-based — *simulated annealing*
    - competitive — *evolutionary strategies*
  - stochastic
    - Gaussian mixtures
    - ...

Iterative Cluster Analysis

Exemplar-Based Algorithm

Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance measure for two nodes in V.
e. Minimization criterion for cluster representatives, based on d.
k. Number of desired clusters.

Output: $r_1, \ldots, r_k$. Cluster representatives.

FOR $i = 1$ `to` k **DO** $r_i(t) = \textit{choose}(V)$ `// init representatives`

  FOREACH $v \in V$ **DO** `// find closest representative`

    $i = \underset{j:\, j \in \{1,\ldots,k\}}{\text{argmin}}\ d(r_j(t), v)$, $C_i = C_i \cup \{v\}$ `// cluster assignment`

  ENDDO

  FOR $i = 1$ `to` k **DO** $r_i(t) = \underset{v \in C_i \text{ or } v \in \mathbf{R}^p}{\text{argmin}}\ e(C_i)$ `// update representatives`

Iterative Cluster Analysis

Exemplar-Based Algorithm

Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance measure for two nodes in V.
e. Minimization criterion for cluster representatives, based on d.
k. Number of desired clusters.

Output: $r_1, \ldots, r_k$. Cluster representatives.

$t = 0$
FOR $i = 1$ to k **DO** $r_i(t) = \mathit{choose}(V)$ // init representatives
REPEAT
  $t = t + 1$
  FOR $i = 1$ to k **DO** $C_i = \emptyset$

  FOREACH $v \in V$ **DO** // find closest representative
    $i = \underset{j:\, j \in \{1,\ldots,k\}}{\text{argmin}}\ d(r_j(t), v)$, $C_i = C_i \cup \{v\}$ // cluster assignment
  ENDDO
  FOR $i = 1$ to k **DO** $r_i(t) = \underset{v \in C_i \text{ or } v \in \mathbf{R}^p}{\text{argmin}}\ e(C_i)$ // update representatives

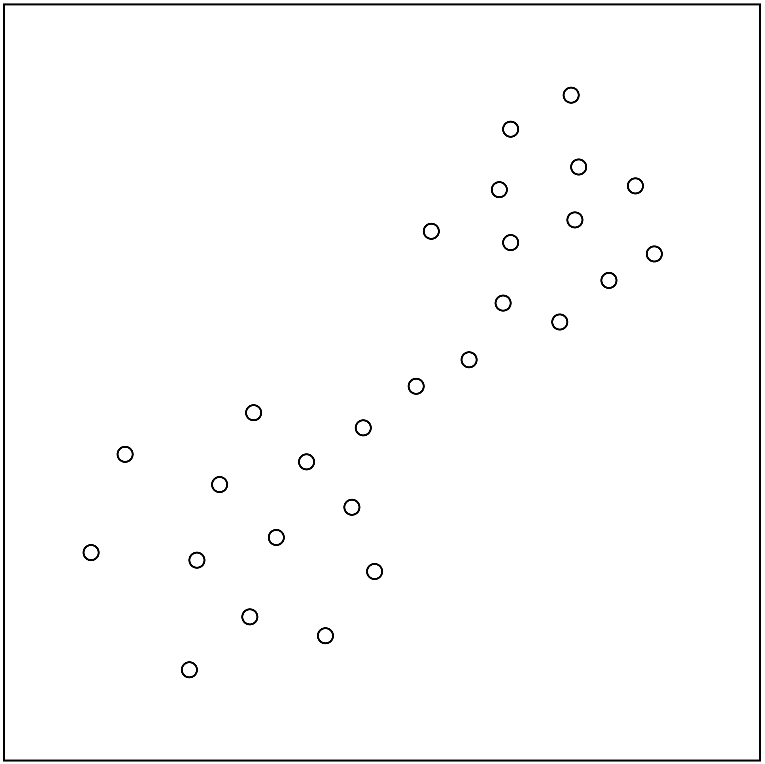
UNTIL $(\mathit{convergence}(r_1(t), \ldots, r_k(t))$ **OR** $t > t_{\text{max}})$
RETURN$(\{r_1(t), \ldots, r_k(t)\})$

Remarks:

- The cluster representatives are called centroids or, more general, medoids.
- The function *choose*(V) operationalizes a random sampling without replacement (in German: „zufälliges Ziehen ohne Zurücklegen“).
- If the data is from a metric space, then the Euclidean distance between two data points is usually chosen as distance function d. An alternative and more general approach is to choose the *shortest path* between two points in the graph G.
- If the data is from a metric space, then the sum of the squared distances to the cluster representatives (= variance criterion) is usually chosen as minimization criterion e: For points $v \in V$ from $\mathbf{R}^p$, the components of the optimum cluster representative (= vector of minimum variance) are given by the component-wise arithmetic mean of the points in the cluster.

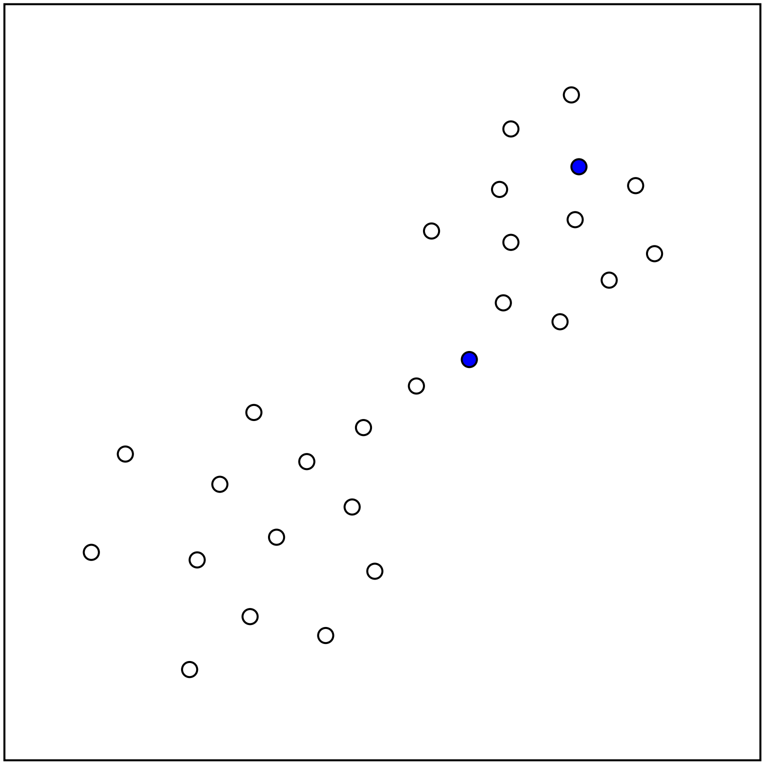
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



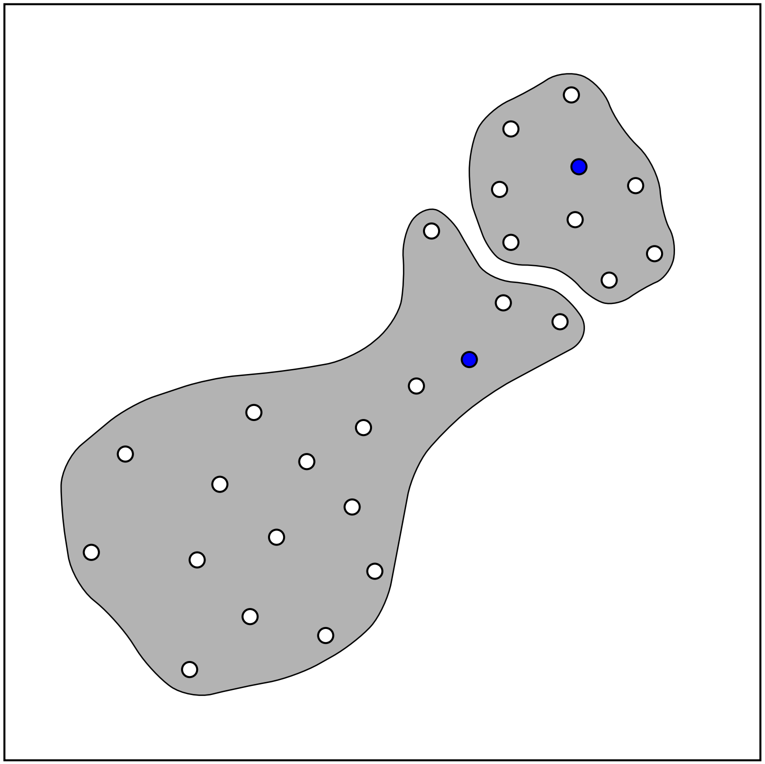
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



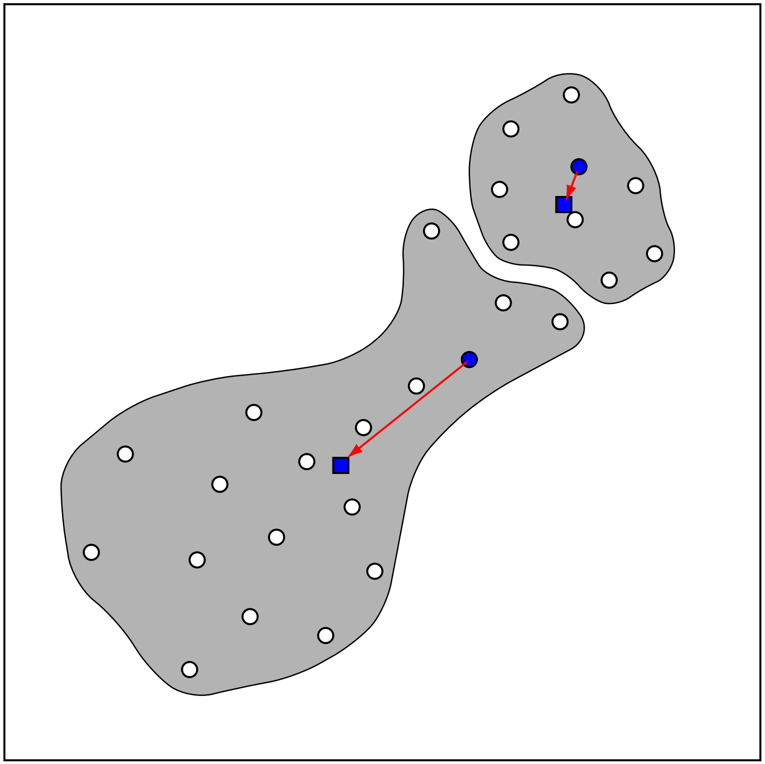
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



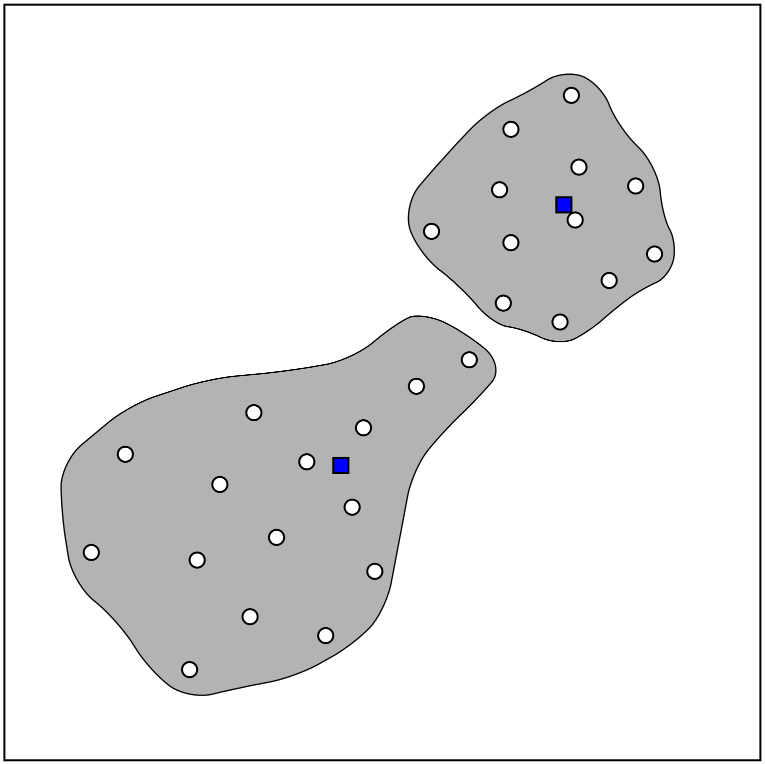
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



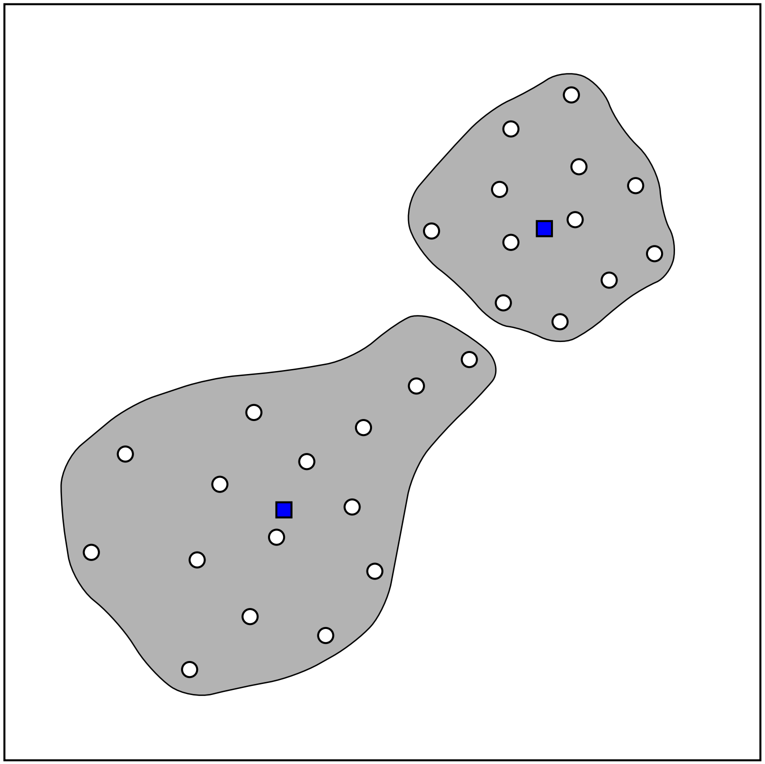
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



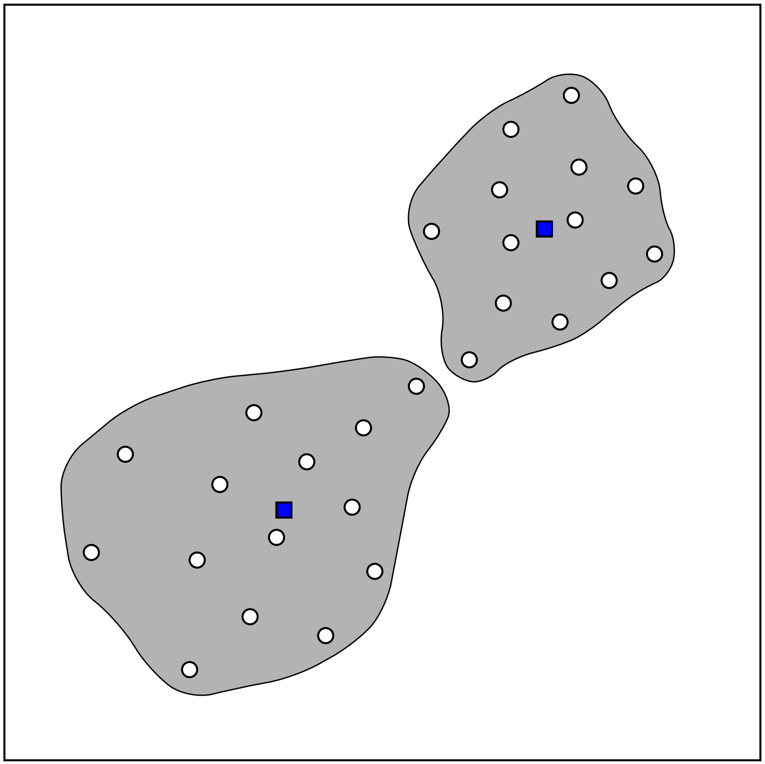
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



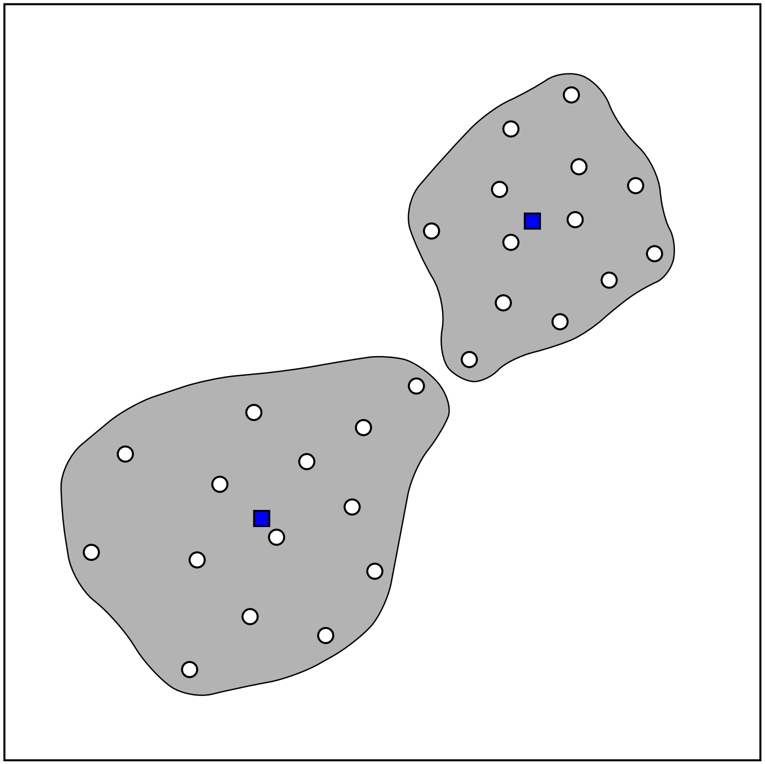
Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



Iterative Cluster Analysis

k-Means with Minimization Criterion e = Variance



Iterative Cluster Analysis

Minimization Criteria of Exemplar-Based Algorithms [algorithm]

$e(C_i) = \sum_{v \in C_i} (v - r_i)^2$	$r_i = \bar{v}_{C_i}$	centroid computation via variance minimization (k-means)
$e(C_i) = \sum_{v \in C_i} \lvert v - r_i \rvert$	$r_i \in C_i$	medoid computation (k-medoid)
$e(C_i) = \max_{v \in C_i} \lvert v - r_i \rvert$	$r_i \in C_i$	k-center
$e(C_i) = \sum_{v \in V} (\mu_i(v))^2 \cdot (v - r_i)^2$	$r_i = \frac{\sum_{v \in V} (\mu_i(v))^2 \cdot v}{\sum_{v \in V} (\mu_i(v))^2}$	Fuzzy k-means

Remarks:

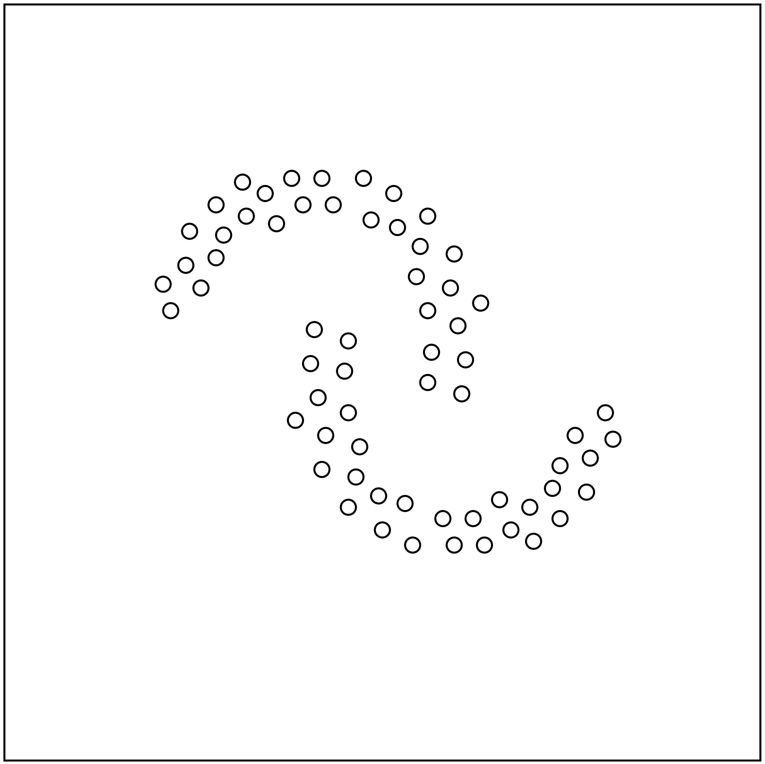
- $\bar{v}_{C_i}$ denotes the arithmetic mean of the points $v \in C_i$.
- To simplify notation the cluster representative is denoted with r_i instead of with $r_i(t)$.
- The sum of the squared distances to a cluster representative r_i becomes minimum, if r_i is the arithmetic mean of the points in C_i. Hence, the computation of the centroid in k-means corresponds to a local—i.e., cluster-specific—minimization of the variance.
- The *medoid* or central element of a cluster denotes a point $r_i \in C_i$ that minimizes the sum of the distances from r_i to all other points in C_i. An advantage of medoids compared to centroids is their robustness with respect to outliers and, as a consequence, an improved convergence behavior (= less iterations).
- k-medoid and k-center can employ nearly arbitrary distance or similarity measures.
- k-means and Fuzzy k-means presume interval-based measurement scales for all features.
- Within Fuzzy k-means, $\mu_i(v)$ denotes the membership value of the point $v \in V$ with respect to cluster C_i.

Remarks: (continued)

- k-means can be operationalized straightforwardly as Kohonen self-organizing map, SOM, a particular kind of neural network:
  - The SOM network is comprised of an input layer with p nodes, which correspond one-to-one to the features, and a so-called "competitive layer" with k nodes.
  - Based on the network's current edge weights the training algorithm determines for a feature vector the so-called "winning neuron", whose edge weights are raised according to a learning rate η.

Iterative Cluster Analysis

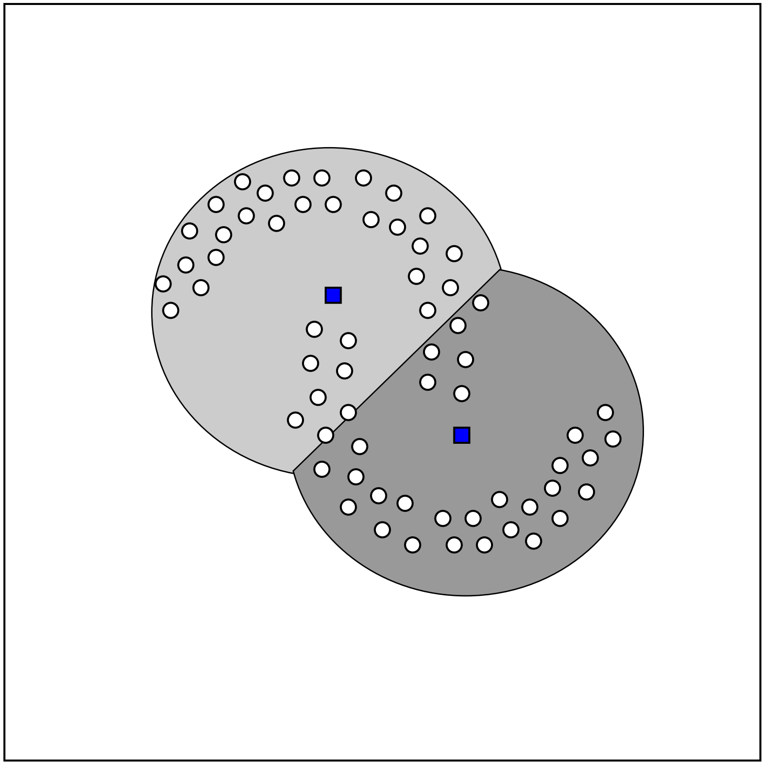
k-Means versus Single Link



Exemplar-based algorithms fail to detect nested clusters.

Iterative Cluster Analysis

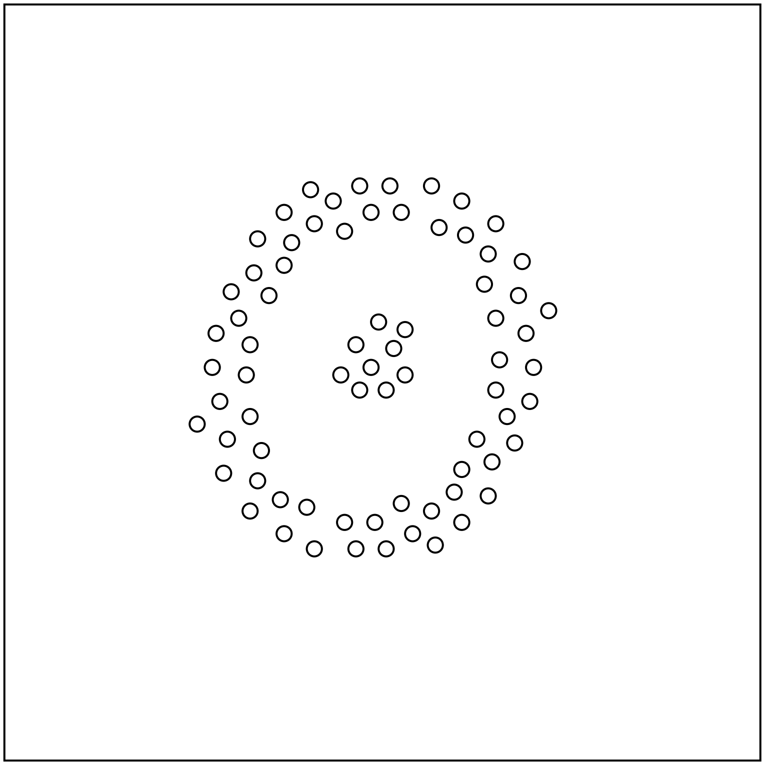
k-Means versus Single Link



Exemplar-based algorithms fail to detect nested clusters.

Iterative Cluster Analysis

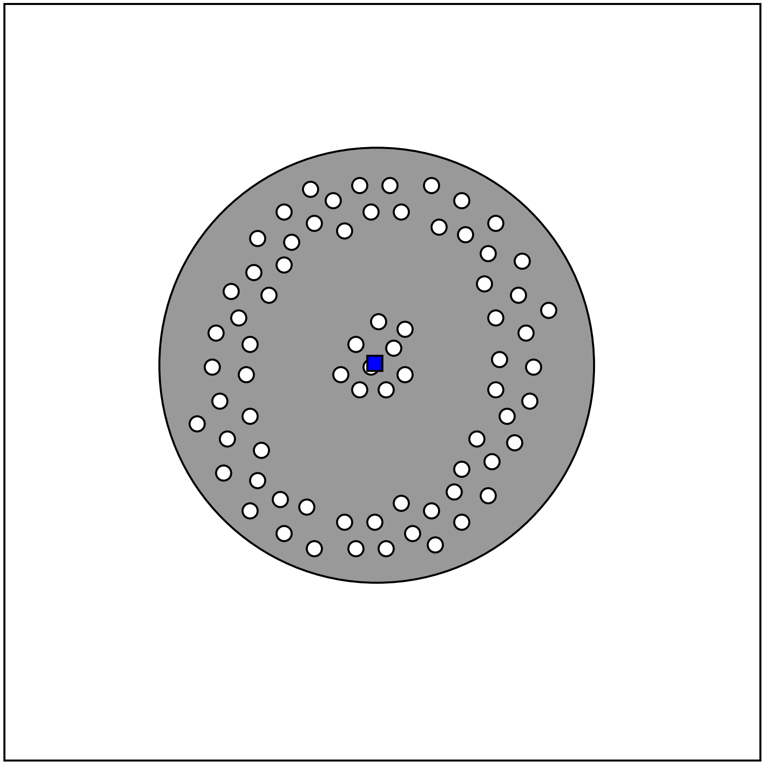
k-Means versus Single Link



Exemplar-based algorithms fail to detect nested clusters.

Iterative Cluster Analysis

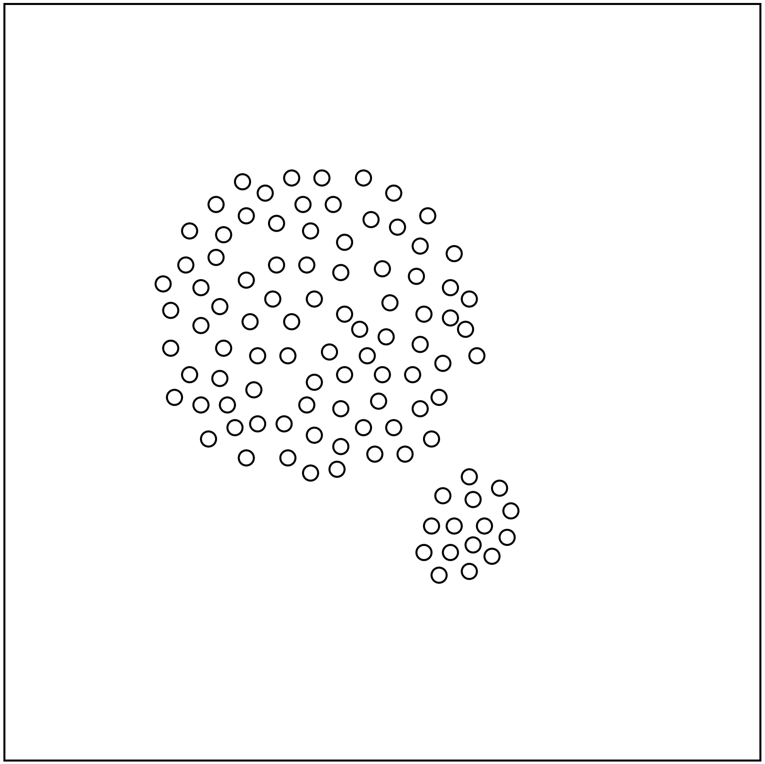
k-Means versus Single Link



Exemplar-based algorithms fail to detect nested clusters.

Iterative Cluster Analysis

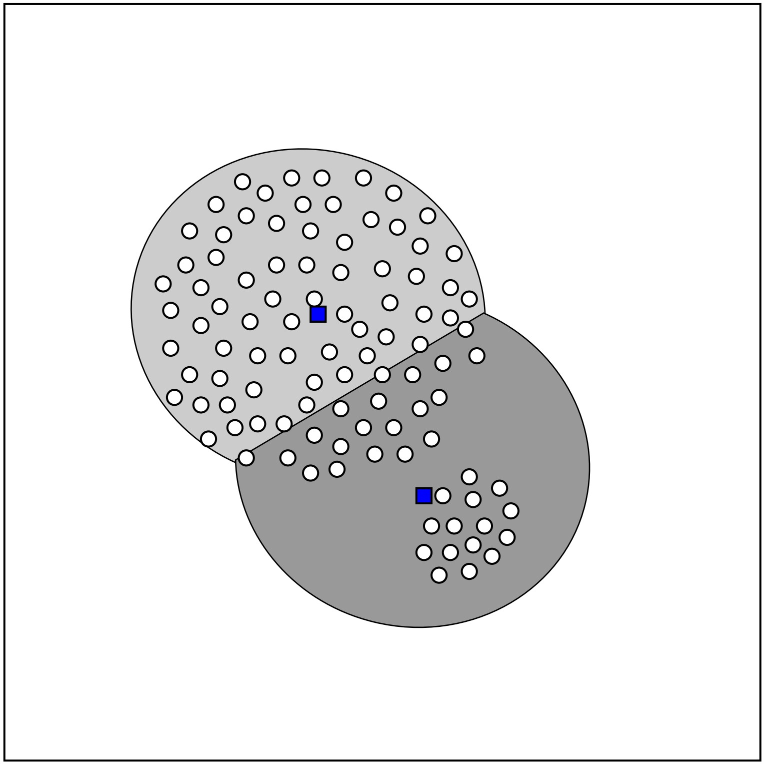
k-Means versus Single Link



Exemplar-based algorithms fail to detect clusters with large difference in size.

Iterative Cluster Analysis

k-Means versus Single Link



Exemplar-based algorithms fail to detect clusters with large difference in size.

Iterative Cluster Analysis

Exclusive versus Non-Exclusive Algorithms

Let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be a partitioning of a set V with $\bigcup\limits_{i=1\ldots k} C_i = V$.

- exclusive algorithms: $\forall i, j \in \{1, \ldots, k\} : i \neq j$ implies $C_i \cap C_j = \emptyset$
- non-exclusive algorithms allow for multiple cluster membership

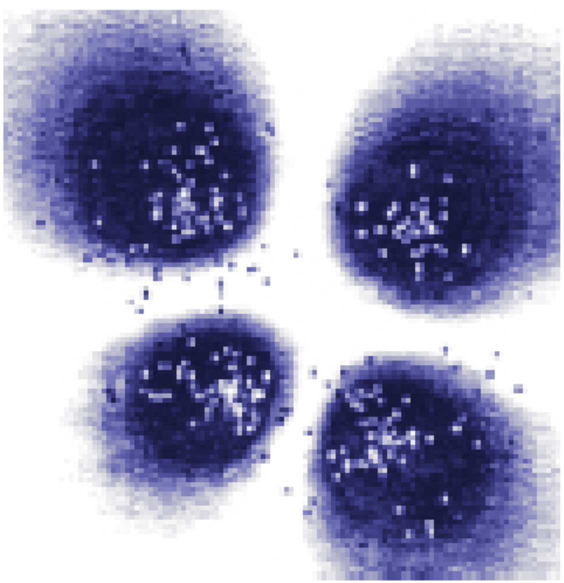
Iterative Cluster Analysis

Exclusive versus Non-Exclusive Algorithms

Let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be a partitioning of a set V with $\bigcup_{i=1\ldots k} C_i = V$.

- exclusive algorithms: $\forall i, j \in \{1, \ldots, k\} : i \neq j$ implies $C_i \cap C_j = \emptyset$
- non-exclusive algorithms allow for multiple cluster membership

- Fuzzy cluster analysis quantifies cluster membership of the $v \in V$ by means of a membership function $\mu_i(v),\ i \in \{1, \ldots, k\}$. [minimization criterion]

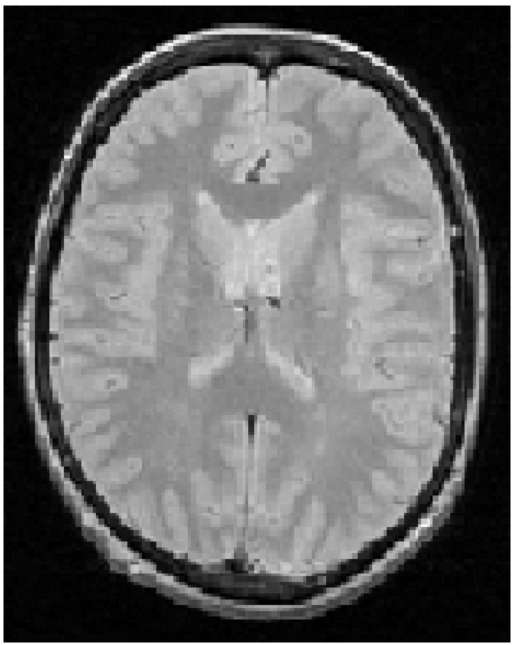


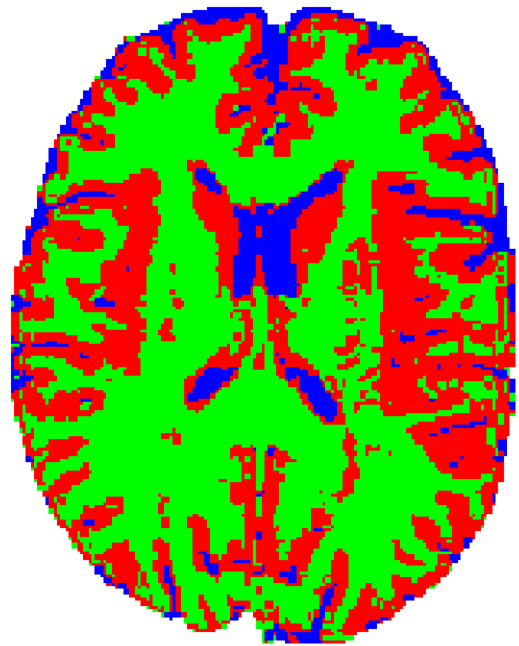
[Höppner/Klawonn/Kruse 1997]

Iterative Cluster Analysis

Exclusive versus Non-Exclusive Algorithms

Application of Fuzzy cluster analysis to represent and envision cerebral tissue:





[Pham/Prince/Dagher/Xn 1996]

Remarks:

- The domain of the linguistic variable of the Fuzzy model is comprised of k elements, which correspond to the clusters $C_1, \ldots, C_k$.
- Usually a normalization constraint for the membership function is stated: $\sum_{i=1 \ldots k} \mu_i(v) = 1$
- A drawback of Fuzzy k-means variants that neglect normalization is that points with small membership function values for a cluster are treated as outliers, instead of moving the cluster towards these points. Hence it is useful to apply the iteration procedure with a normalization constraint—at least within an initialization phase.
- A categorization by a Fuzzy cluster analysis is beneficial if no clear class structure is given or if various feature vectors belong to several classes at the same time.
- A disadvantage of Fuzzy cluster analysis is the fact that the concept of cluster representatives does not exist.