Data Mining - Clustering

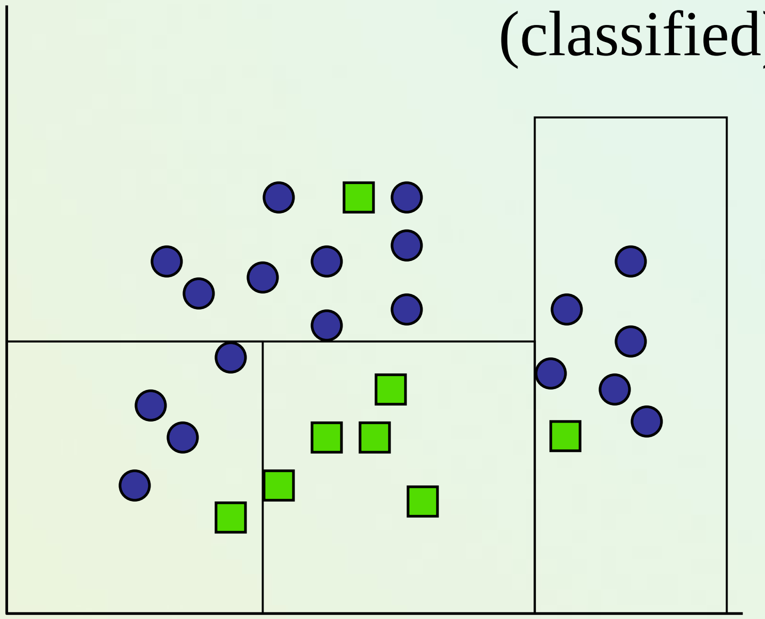
Lecturer: JERZY STEFANOWSKI
Institute of Computing Sciences
Poznan University of Technology
Poznan, Poland

Lecture 7
SE Master Course
2008/2009

Classification vs. Clustering

Classification: Supervised learning:

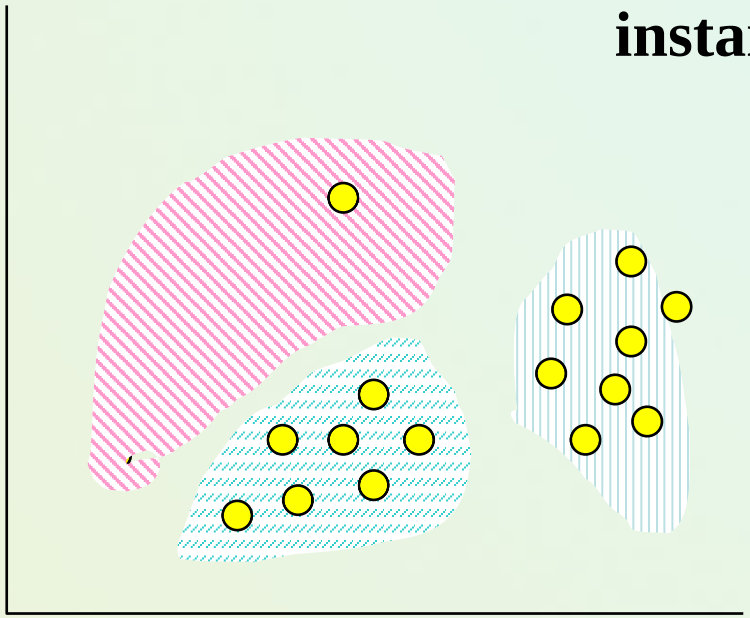
Learns a method for predicting the instance class from pre-labeled (classified) instances



Clustering

Unsupervised learning:

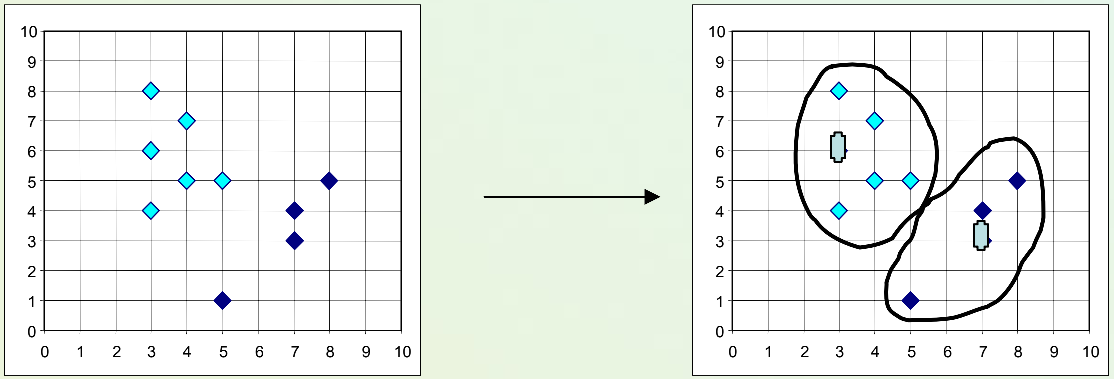
Finds "natural" grouping of instances given un-labeled data



Problem Statement

Given a set of records (instances, examples, objects, observations, …), organize them into clusters (groups, classes)

- Clustering: the process of grouping physical or abstract objects into classes of similar objects



What is a cluster?

1. A cluster is a subset of objects which are “similar”
2. A subset of objects such that the distance between any two objects in the cluster is less than the distance between any object in the cluster and any object not located inside it.
3. A connected region of a multidimensional space containing a relatively high density of objects.

What Is Clustering ?

- Clustering is a process of partitioning a set of data (or objects) into a set of meaningful sub-classes, called clusters.
  - Help users understand the natural grouping or structure in a data set.
- Clustering: unsupervised classification: no predefined classes.
- Used either as a stand-alone tool to get insight into data distribution or as a preprocessing step for other algorithms.
  - Moreover, data compression, outliers detection, understand human concept formation.

What Is Good Clustering?

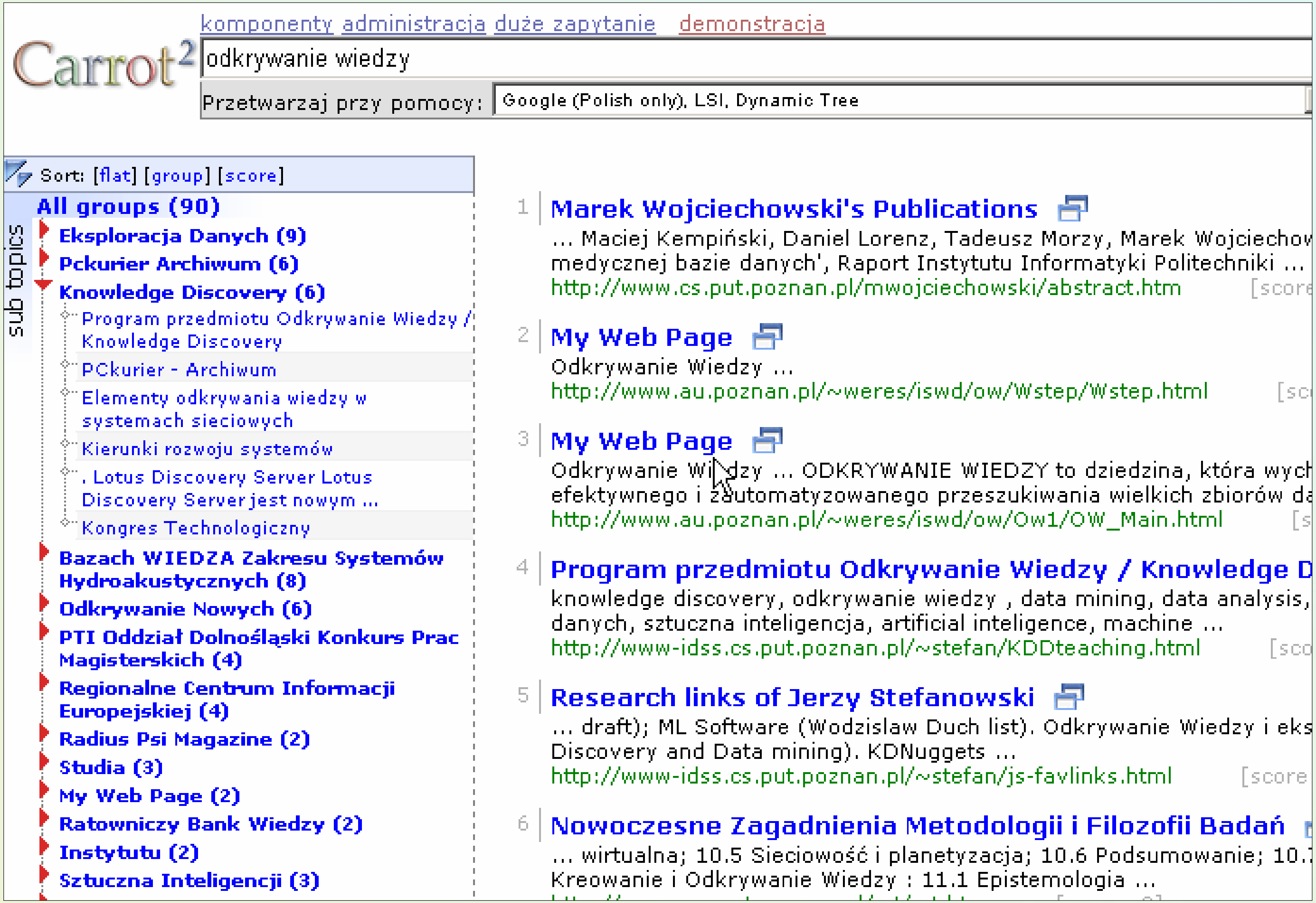
- A good clustering method will produce high quality clusters in which:
  - the intra-class (that is, intra-cluster) similarity is high.
  - the inter-class similarity is low.
- The quality of a clustering result also depends on both the similarity measure used by the method and its implementation.
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns.
- However, objective evaluation is problematic: usually done by human / expert inspection.

Applications of Clustering

Clustering has wide applications in

- Economic Science (especially market research).
- WWW:
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns
- Pattern Recognition.
- Spatial Data Analysis:
  - create thematic maps in GIS by clustering feature spaces
- Image Processing

Web Search Result Clustering



Clustering Methods

- Many different method and algorithms:
  - For numeric and/or symbolic data
  - Exclusive vs. overlapping
    - Crisp vs. soft computing paradigms
  - Hierarchical vs. flat (non-hierarchical)
  - Access to all data or incremental learning
  - Semi-supervised mode
- Algorithms also vary by:
  - Measures of similarity
  - Linkage methods
  - Computational efficiency

Measuring Dissimilarity or Similarity in Clustering

- Dissimilarity/Similarity metric: Similarity is expressed in terms of a distance function, which is typically metric:
  $d(i, j)$
- There are also used in “quality” functions, which estimate the “goodness” of a cluster.
- The definitions of distance functions are usually very different for interval-scaled, boolean, categorical, ordinal and ratio variables.
- Weights should be associated with different variables based on applications and data semantics.

Distance Measures

To discuss whether a set of points is close enough to be considered a cluster, we need a distance measure - $D(x, y)$

The usual axioms for a distance measure D are:

- $D(x, x) = 0$
- $D(x, y) = D(y, x)$
- $D(x, y) \leq D(x, z) + D(z, y)$ the triangle inequality

Distance Measures (2)

Assume a k-dimensional Euclidean space, the distance between two points, $x=[x_1, x_2, ..., x_k]$ and $y=[y_1, y_2, ..., y_k]$ may be defined using one of the measures:

- Euclidean distance: ("L_2 norm") $$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$
- Manhattan distance: ("L_1 norm") $$\sum_{i=1}^{k}| x_i - y_i |$$
- Max of dimensions: ("L_∞ norm") $$\max_{i=1}^{k} | x_i - y_i |$$

Distance Measures (3)

- Minkowski distance: $$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

When there is no Euclidean space in which to place the points, clustering becomes more difficult: Web page accesses, DNA sequences, customer sequences, categorical attributes, documents, etc.

Standarization / Normalization

- If the values of attributes are in different units then it is likely that some of them will take vary large values, and hence the "distance" between two cases, on this variable, can be a big number.
- Other attributes may be small in values, or not vary much between cases, in which case the difference between the two cases will be small.
- The attributes with high variability / range will dominate the metric.
- Overcome this by standardization or normalization

$$z_i = \frac{x_i - \bar{x}_i}{s_{x_i}}$$

Main Categories of Clustering Methods

- Partitioning algorithms: Construct various partitions and then evaluate them by some criterion.
- Hierarchy algorithms: Create a hierarchical decomposition of the set of data (or objects) using some criterion.
- Density-based: based on connectivity and density functions
- Grid-based: based on a multiple-level granularity structure
- Model-based: A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other.

Partitioning Algorithms: Basic Concept

- Partitioning method: Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters
- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion.
  - Global optimal: exhaustively enumerate all partitions.
  - Heuristic methods: *k-means* and *k-medoids* algorithms.
  - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster.

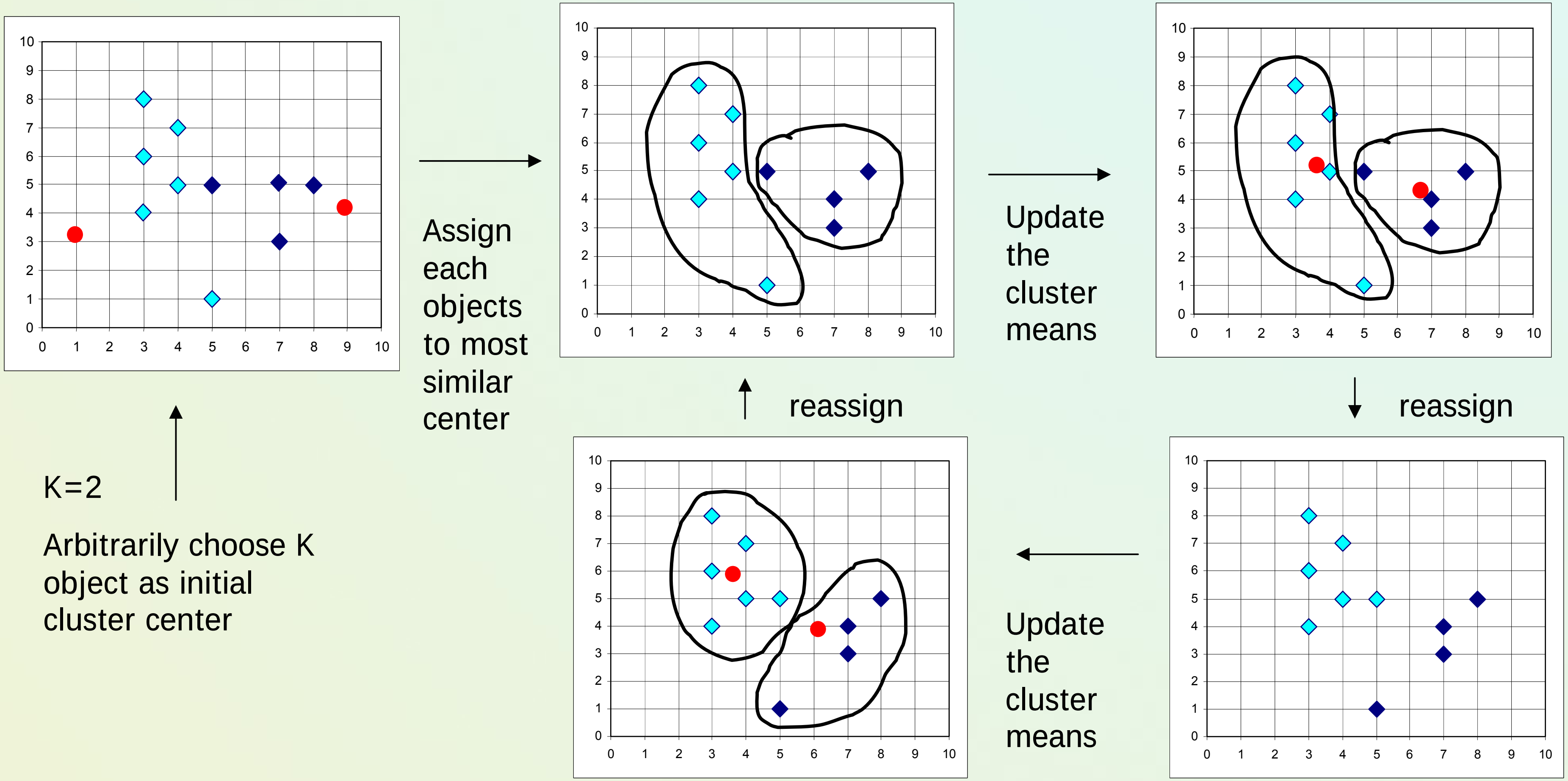
Simple Clustering: K-means

Basic version works with numeric data only

1) Pick a number (K) of cluster centers - *centroids* (at random)
2) Assign every item to its nearest cluster center (e.g. using Euclidean distance)
3) Move each cluster center to the mean of its assigned items
4) Repeat steps 2,3 until convergence (change in cluster assignments less than a threshold)

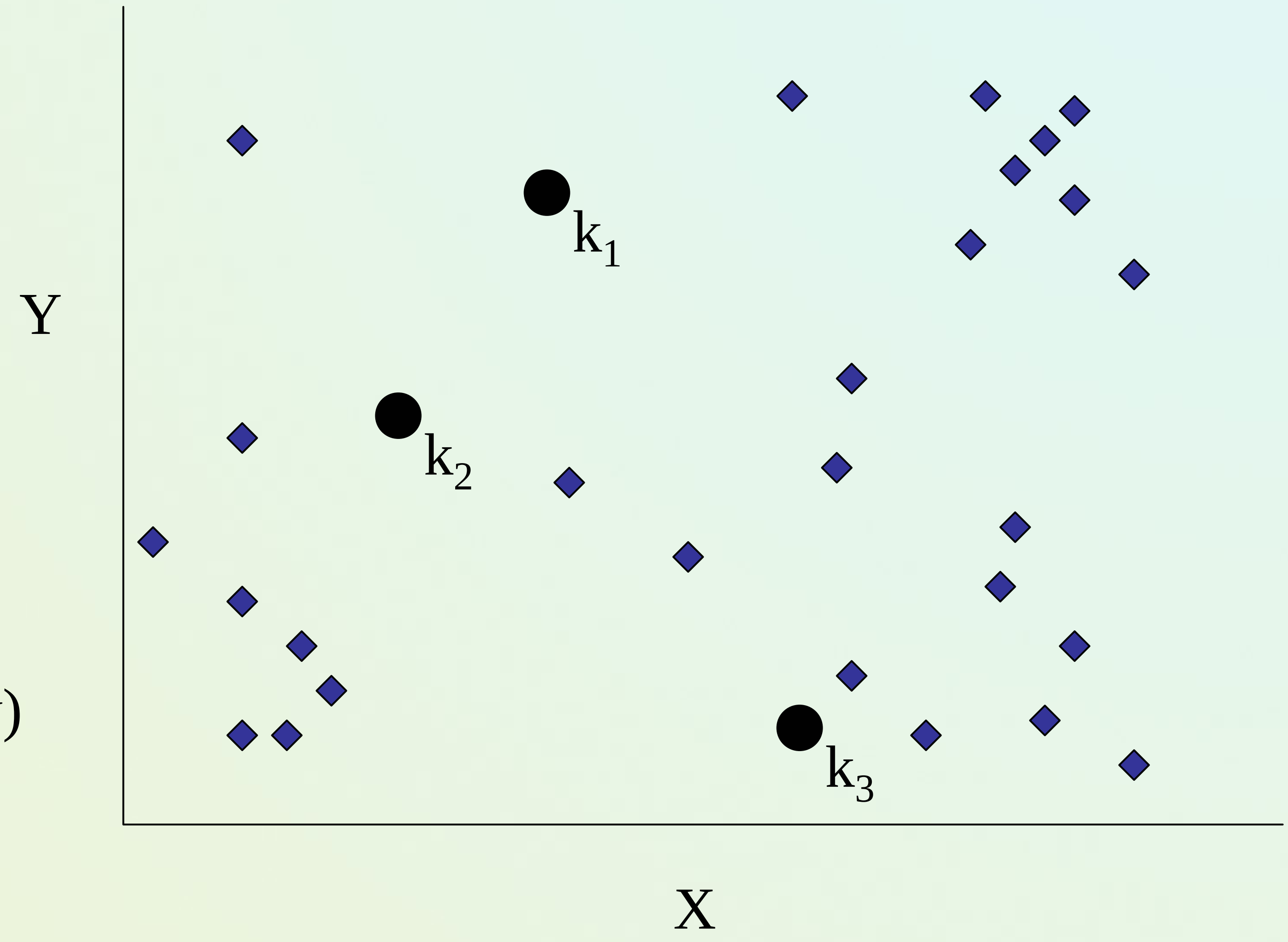
Illustrating *K-Means*

- Example



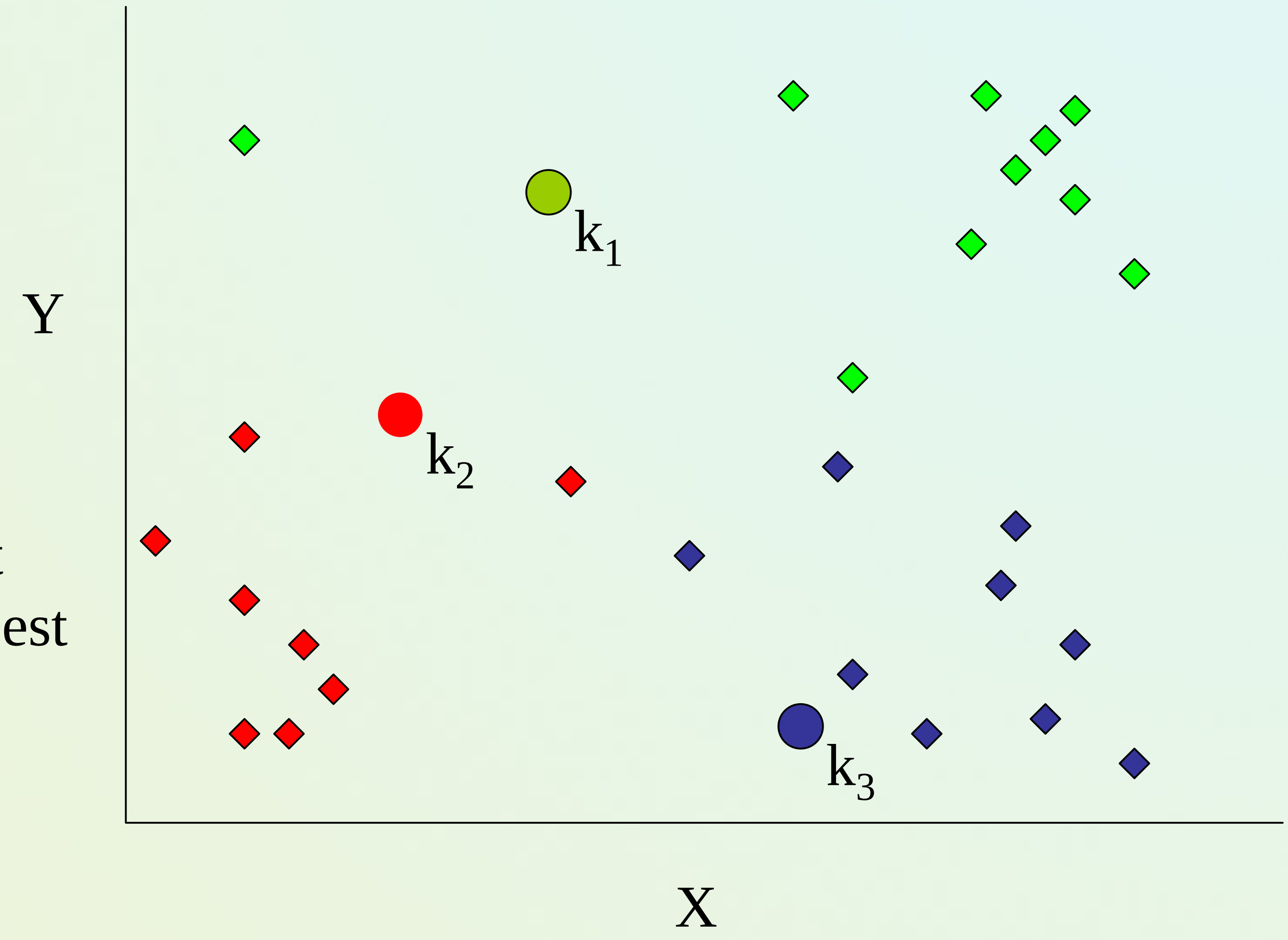
K-means example, step 1

Pick 3 initial cluster centers (randomly)

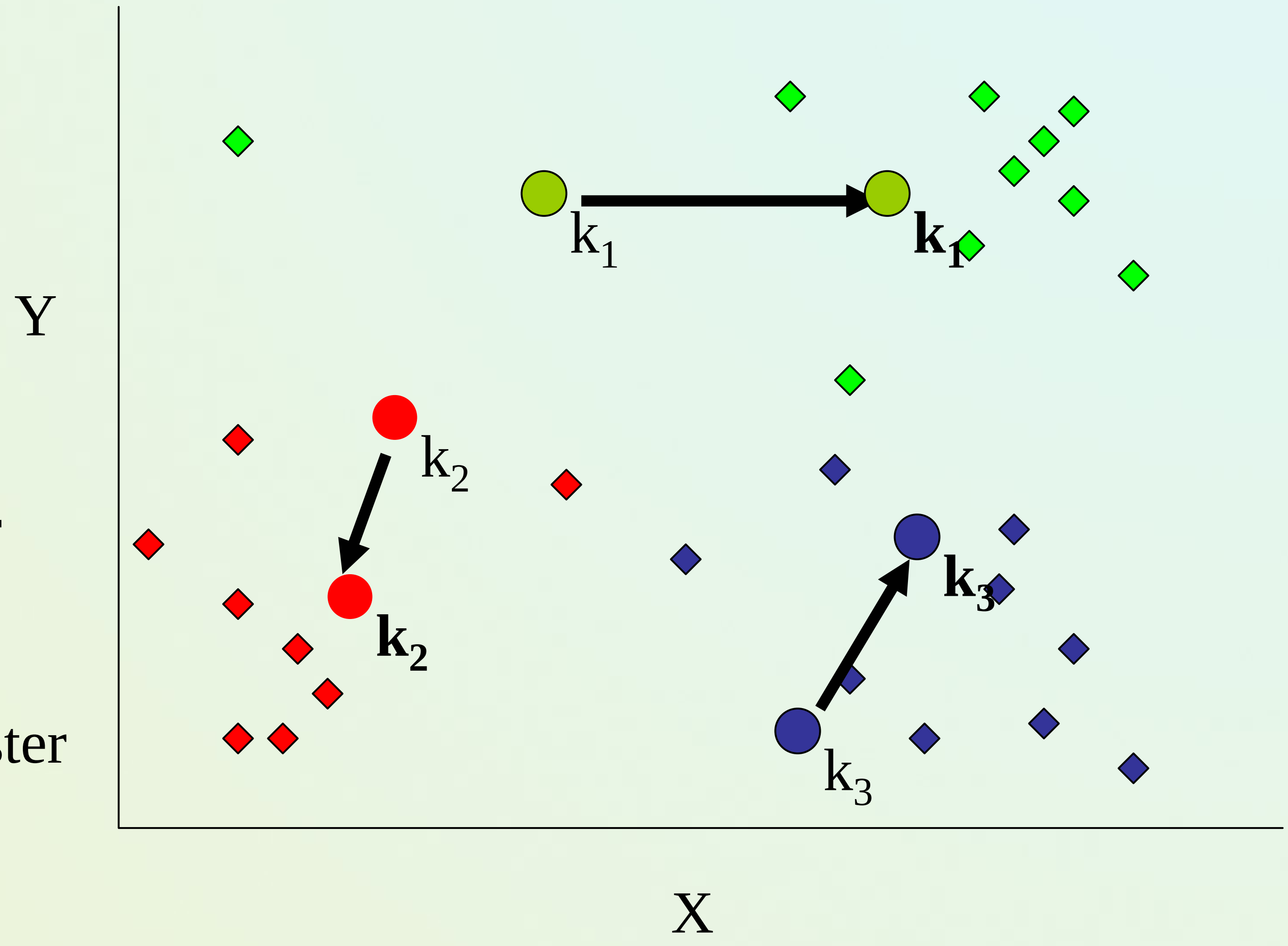


K-means example, step 2

Assign each point to the closest cluster center



K-means example, step 3

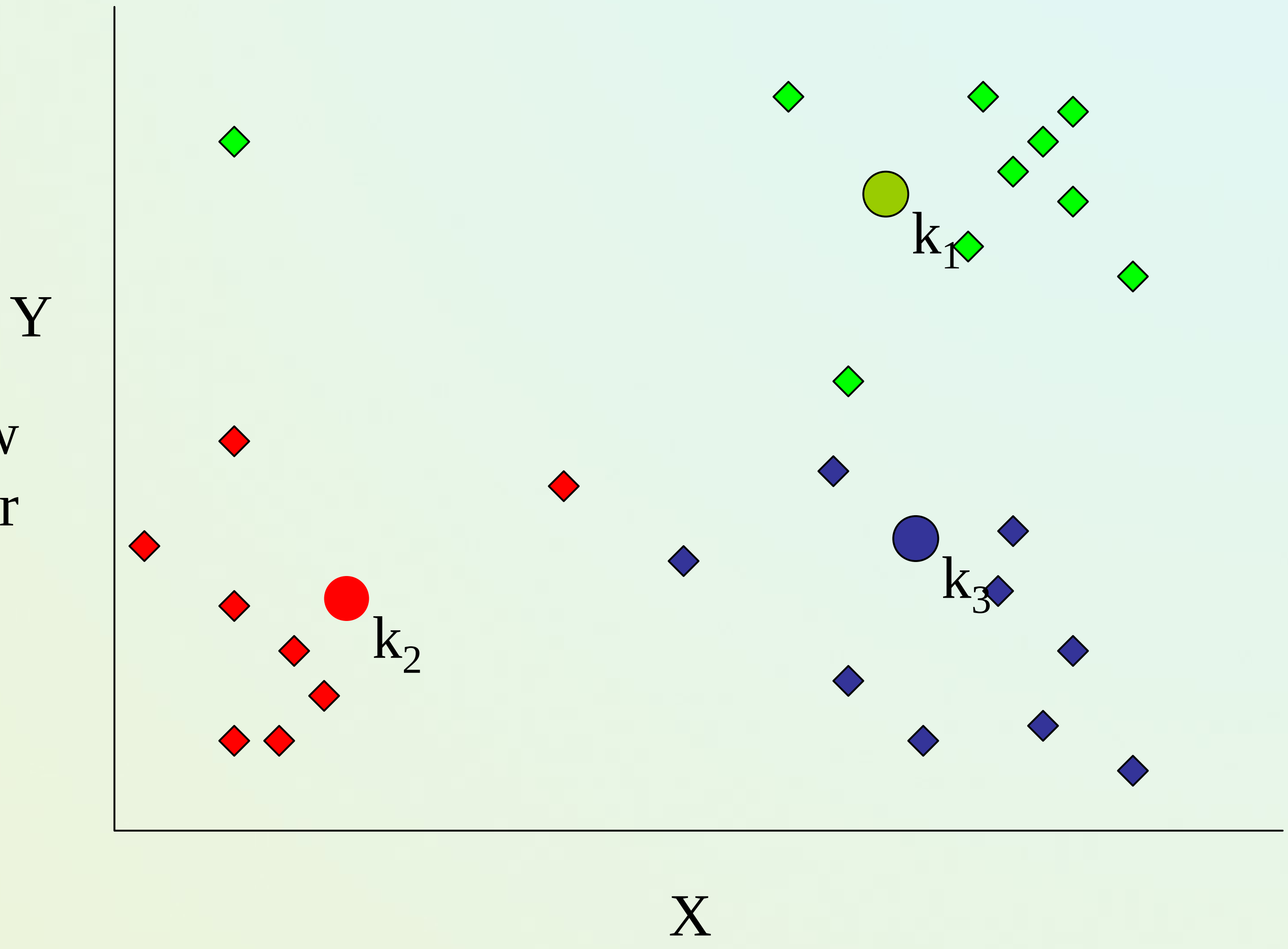


Move
each cluster
center
to the mean
of each cluster

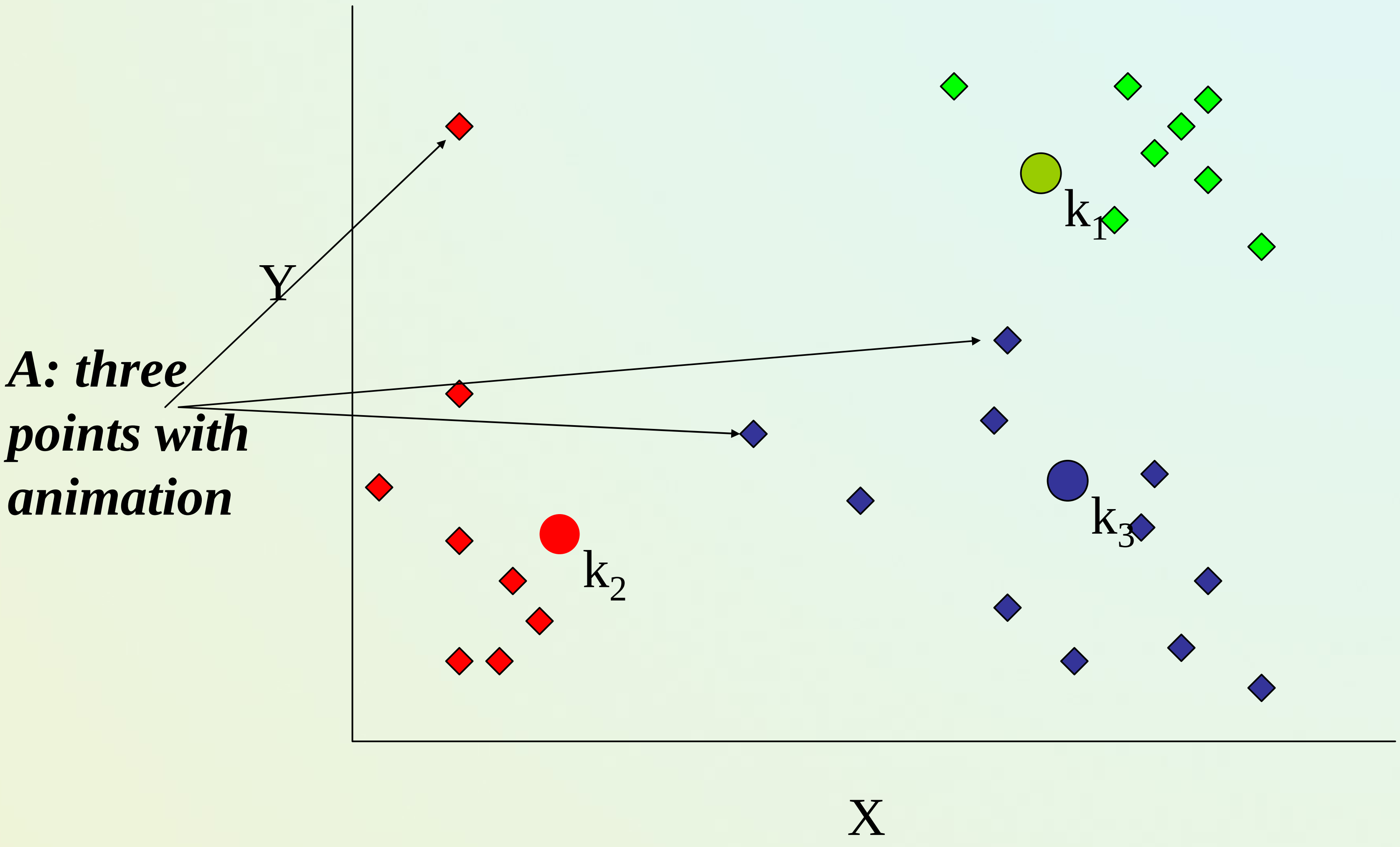
K-means example, step 4

Reassign points closest to a different new cluster center

Q: Which points are reassigned?

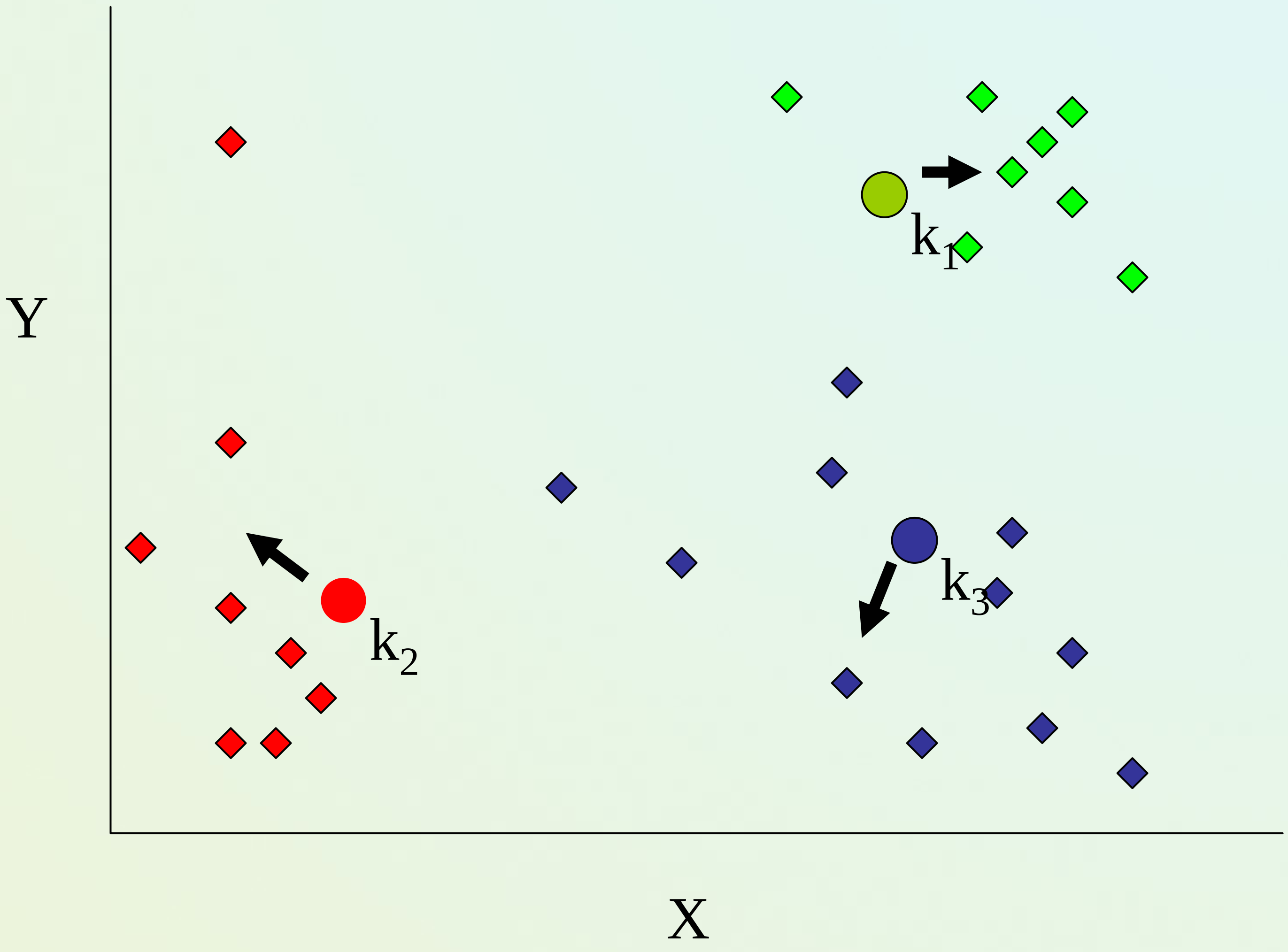


K-means example, step 4 …

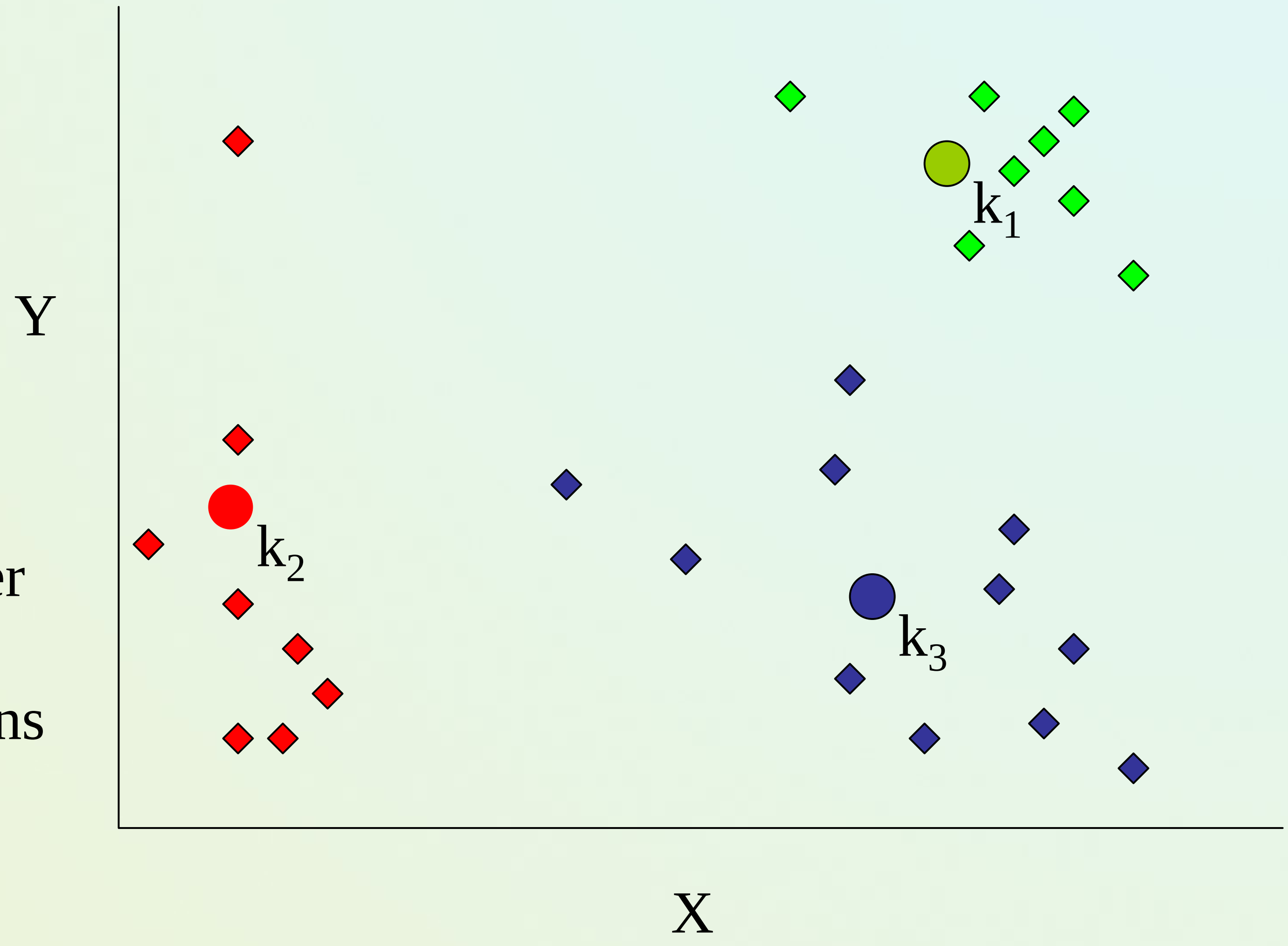


K-means example, step 4b

re-compute cluster means



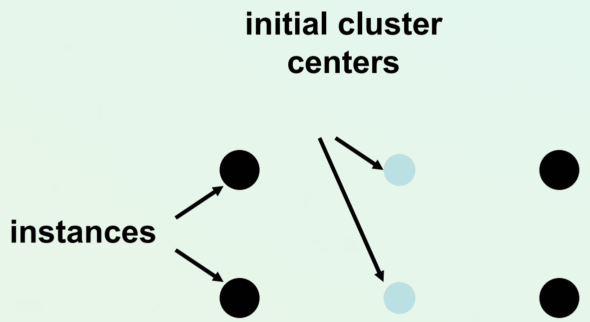
K-means example, step 5



move cluster centers to cluster means

Discussion

- Result can vary significantly depending on initial choice of seeds
- Can get trapped in local minimum
  - Example:



- To increase chance of finding global optimum: restart with different random seeds

K-means clustering summary

Advantages

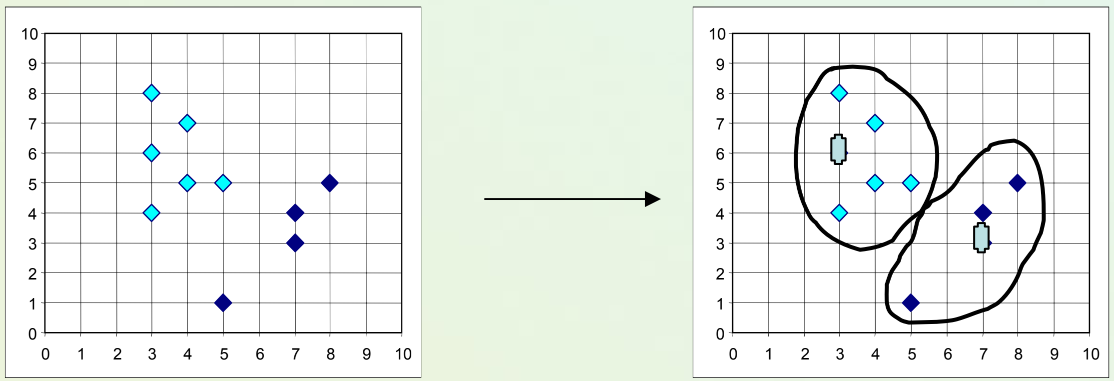
- Simple, understandable
- items automatically assigned to clusters

Disadvantages

- Must pick number of clusters before hand
- Often terminates at a *local optimum.*
- All items forced into a cluster
- Too sensitive to outliers

What is the problem of k-Means Method?

- The k-means algorithm is sensitive to outliers !
  - Since an object with an extremely large value may substantially distort the distribution of the data.
- There are other limitations – still a need for reducing costs of calculating distances to centroids.
- **K-Medoids**: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most centrally located** object in a cluster.

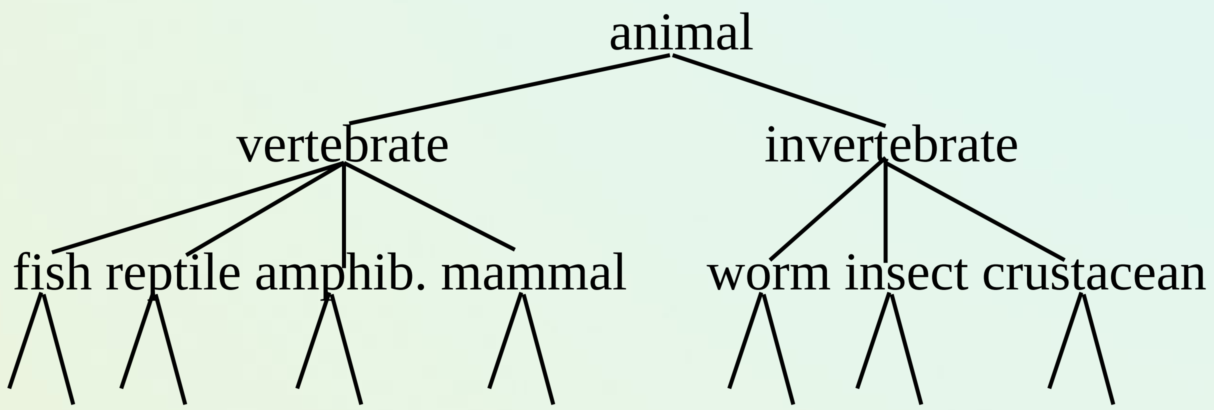


The *K-Medoids* Clustering Method

- Find *representative* objects, called medoids, in clusters
  - To achieve this goal, only the definition of distance from any two objects is needed.
- *PAM* (Partitioning Around Medoids, 1987)
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering.
  - *PAM* works effectively for small data sets, but does not scale well for large data sets.
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling.
- Focusing + spatial data structure (Ester et al., 1995).

Hierarchical Clustering

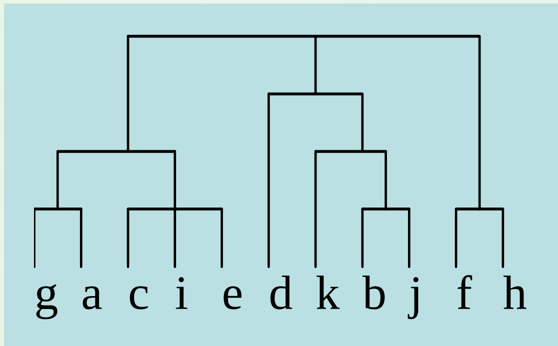
- Build a tree-based hierarchical taxonomy (*dendrogram*) from a set of unlabeled examples.



- Recursive application of a standard clustering algorithm can produce a hierarchical clustering.

*Hierarchical clustering

- Bottom up (aglomerative)
  - Start with single-instance clusters
  - At each step, join the two closest clusters
  - Design decision: distance between clusters
    - e.g. two closest instances in clusters vs. distance between means
- Top down (divisive approach / deglomerative)
  - Start with one universal cluster
  - Find two clusters
  - Proceed recursively on each subset
  - Can be very fast
- Both methods produce a *dendrogram*



HAC Algorithm (aglomerative)

Start with all instances in their own cluster.
Until there is only one cluster:
- Among the current clusters, determine the two clusters, c_i and c_j, that are most similar.
- Replace c_i and c_j with a single cluster $c_i \cup c_j$

Distance between Clusters

Single linkage
minimum distance: $$d_{\min}(C_i, C_j) = \min_{p \in C_i, p' \in C_j} \|p - p'\|$$

Complete linkage
maximum distance: $$d_{\max}(C_i, C_j) = \max_{p \in C_i, p' \in C_j} \|p - p'\|$$

mean distance: $$d_{mean}(C_i, C_j) = \|m_i - m_j\|$$

average distance: $$d_{ave}(C_i, C_j) = 1/(n_i n_j) \sum_{p \in C_i} \sum_{p' \in C_j} \|p - p'\|$$

m_i is the mean for cluster C_i n_i is the number of points in C_i

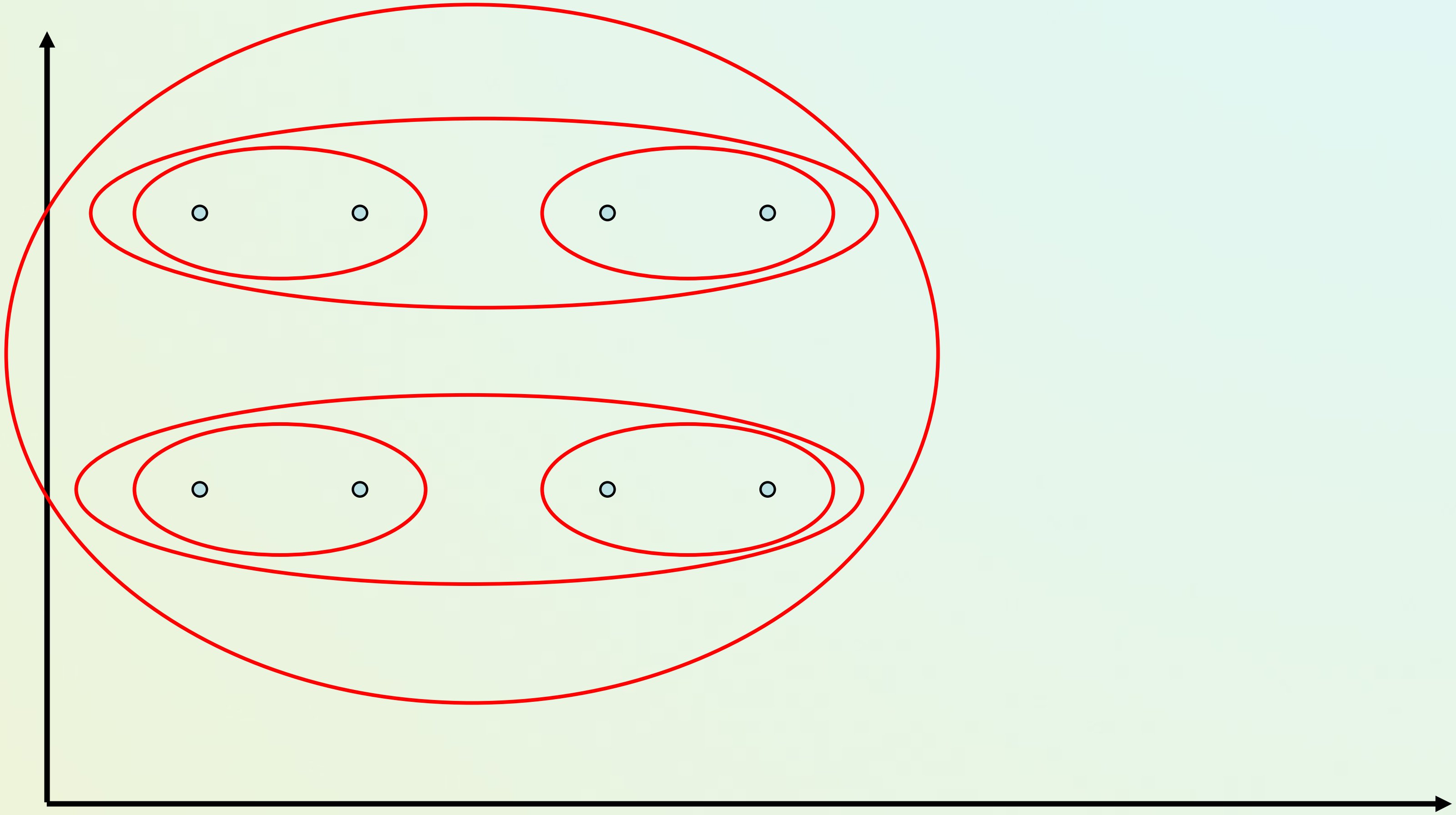
Single Link Agglomerative Clustering

- Use minium similarity of pairs:

$$sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

- Can result in "straggly" (long and thin) clusters due to *chaining effect*.
    - Appropriate in some domains, such as clustering islands.

Single Link Example



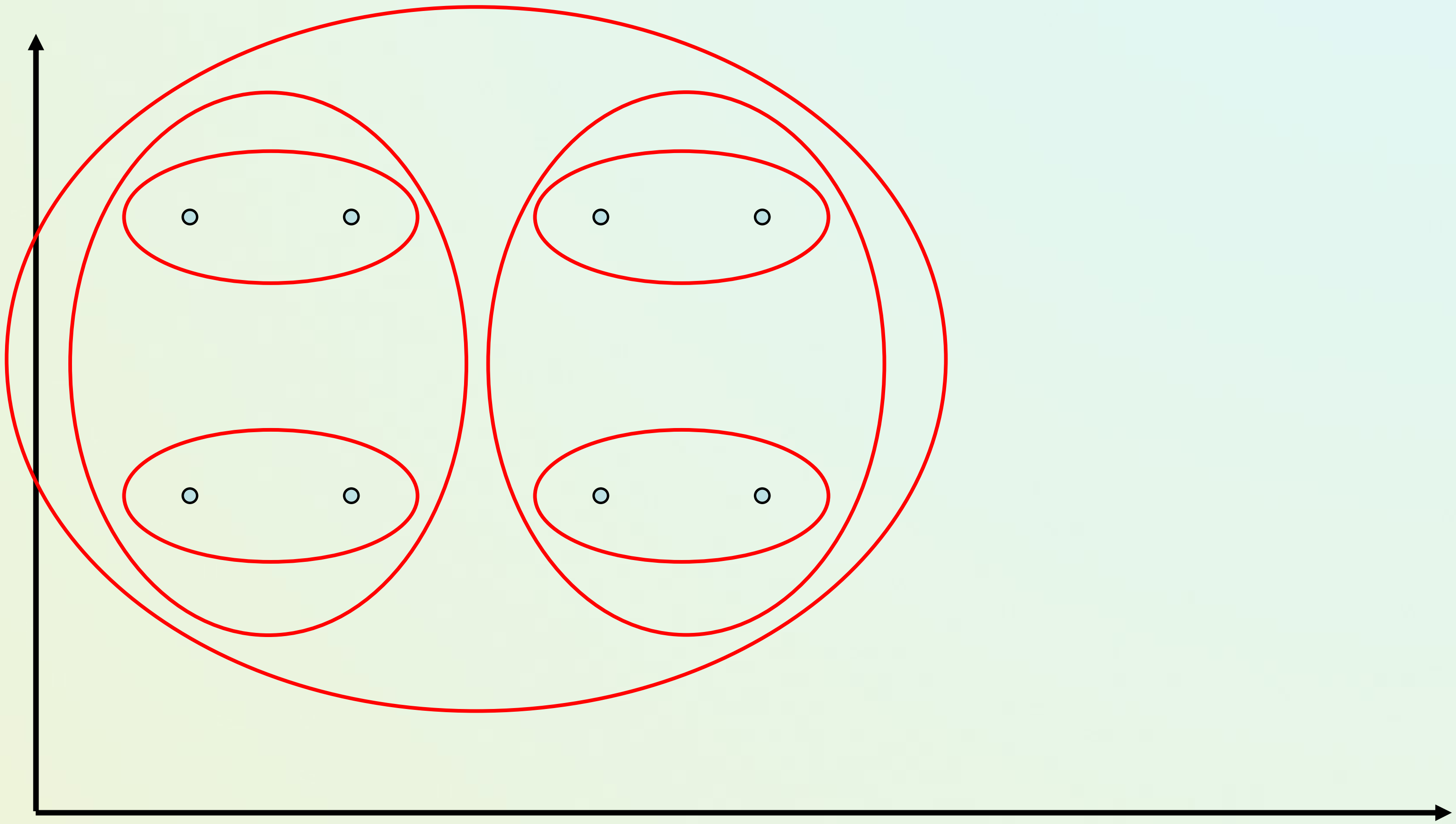
Complete Link Agglomerative Clustering

- Use maximum similarity of pairs:

$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

- Makes more “tight,” spherical clusters that are typically preferable.

Complete Link Example



Single vs. Complete Linkage

- A.Jain et al.: Data Clustering. A Review.

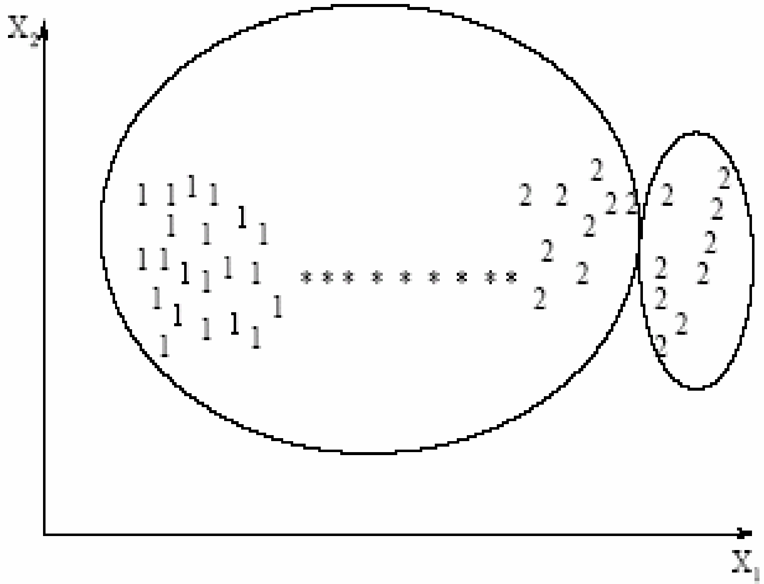


Figure 12. A single-link clustering of a pattern set containing two classes (1 and 2) connected by a chain of noisy patterns (*).

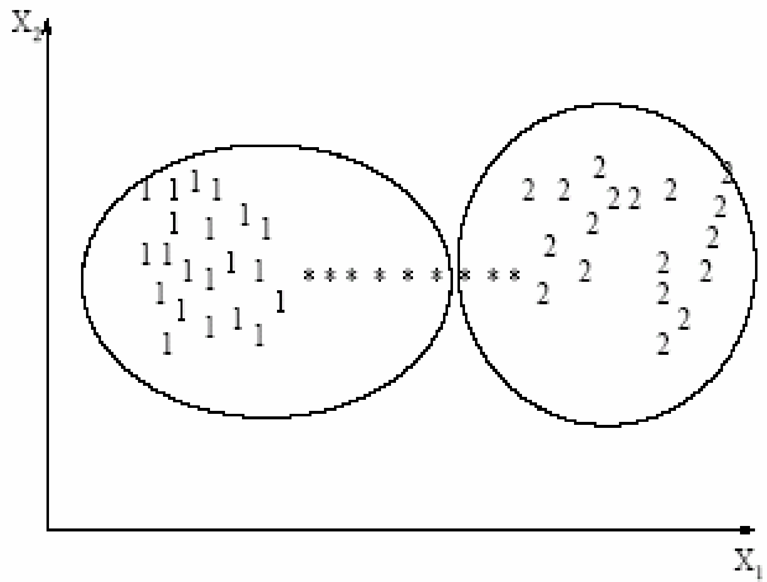
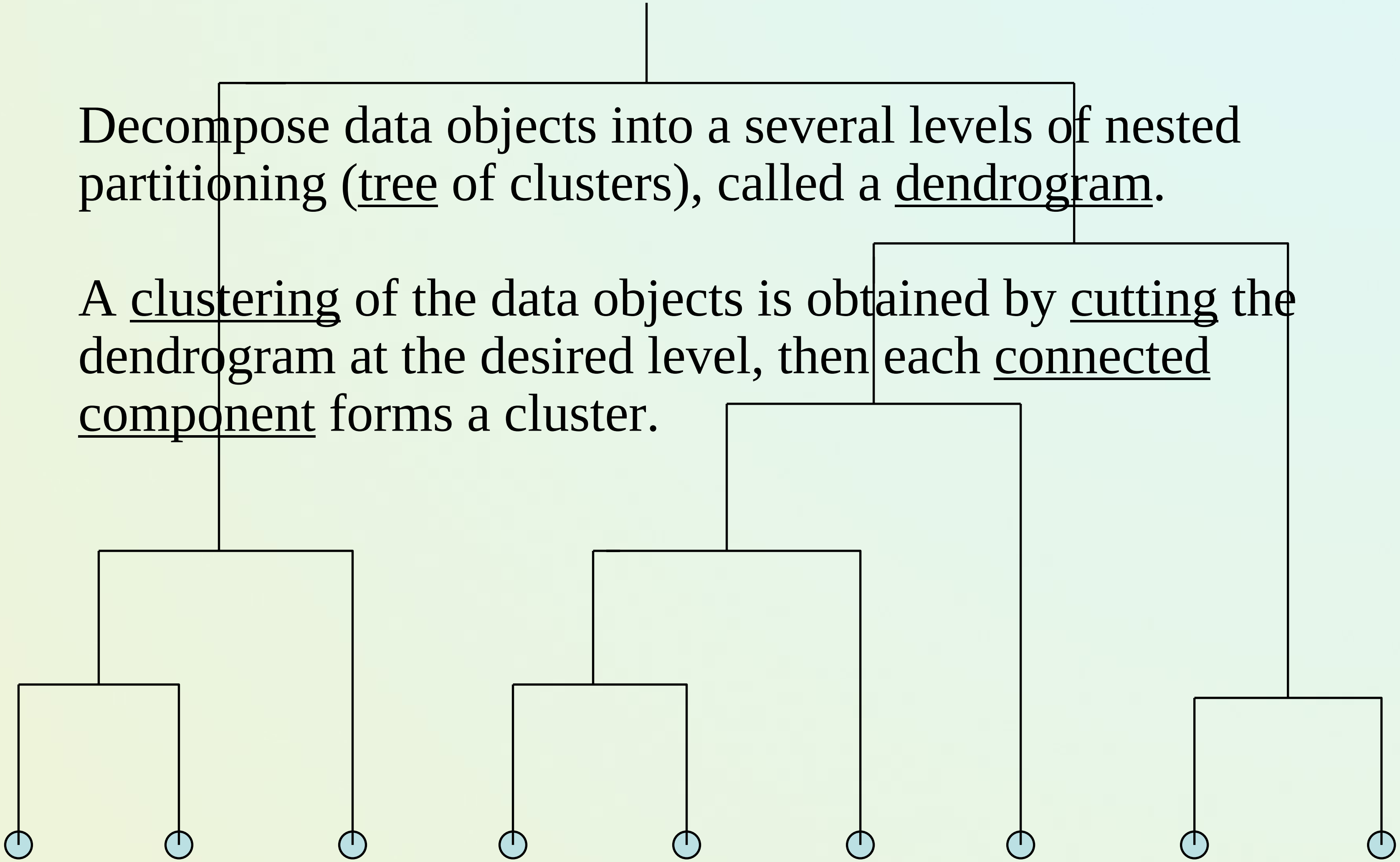


Figure 13. A complete-link clustering of a pattern set containing two classes (1 and 2) connected by a chain of noisy patterns (*).

Dendrogram: Shows How the Clusters are Merged

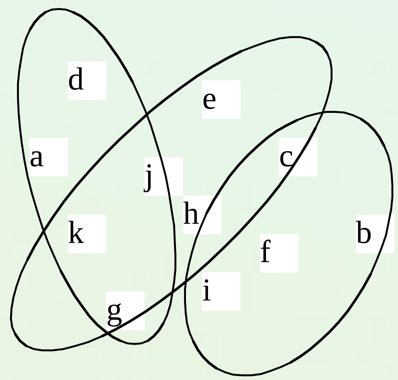
Decompose data objects into a several levels of nested partitioning (<u>tree</u> of clusters), called a <u>dendrogram</u>.

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



Soft Clustering

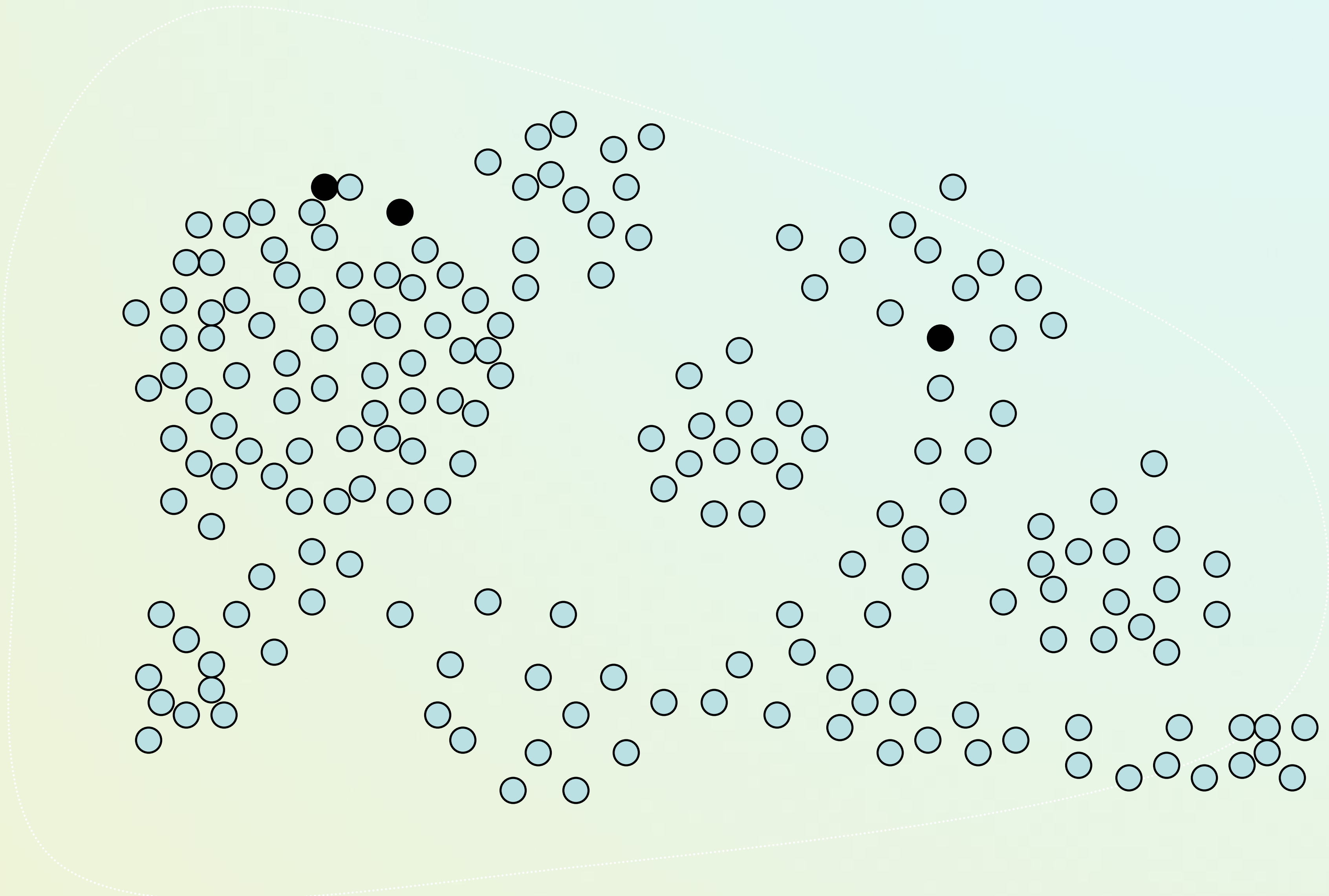
- Clustering typically assumes that each instance is given a “hard” assignment to exactly one cluster.
- Does not allow uncertainty in class membership or for an instance to belong to more than one cluster.
- *Soft clustering* gives probabilities that an instance belongs to each of a set of clusters.
- Each instance is assigned a probability distribution across a set of discovered categories (probabilities of all categories must sum to 1).



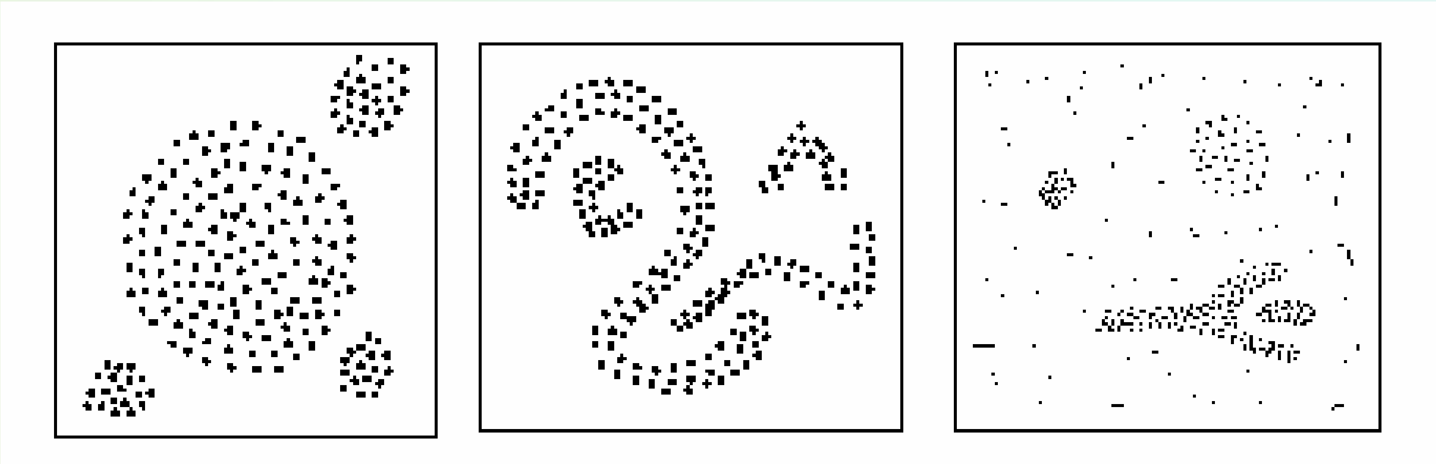
Expectation Maximization (EM Algorithm)

- Probabilistic method for soft clustering.
- Direct method that assumes k clusters: $\{c_1, c_2, \ldots c_k\}$
- Soft version of k-means.
- Assumes a probabilistic model of categories that allows computing $P(c_i \mid E)$ for each category, c_i, for a given example, E.
- For text, typically assume a naïve-Bayes category model.
  - Parameters $\theta = \{P(c_i), P(w_j \mid c_i): i \in \{1,\ldots k\}, j \in \{1,\ldots,|V|\}\}$

Handling Complex Shaped Clusters

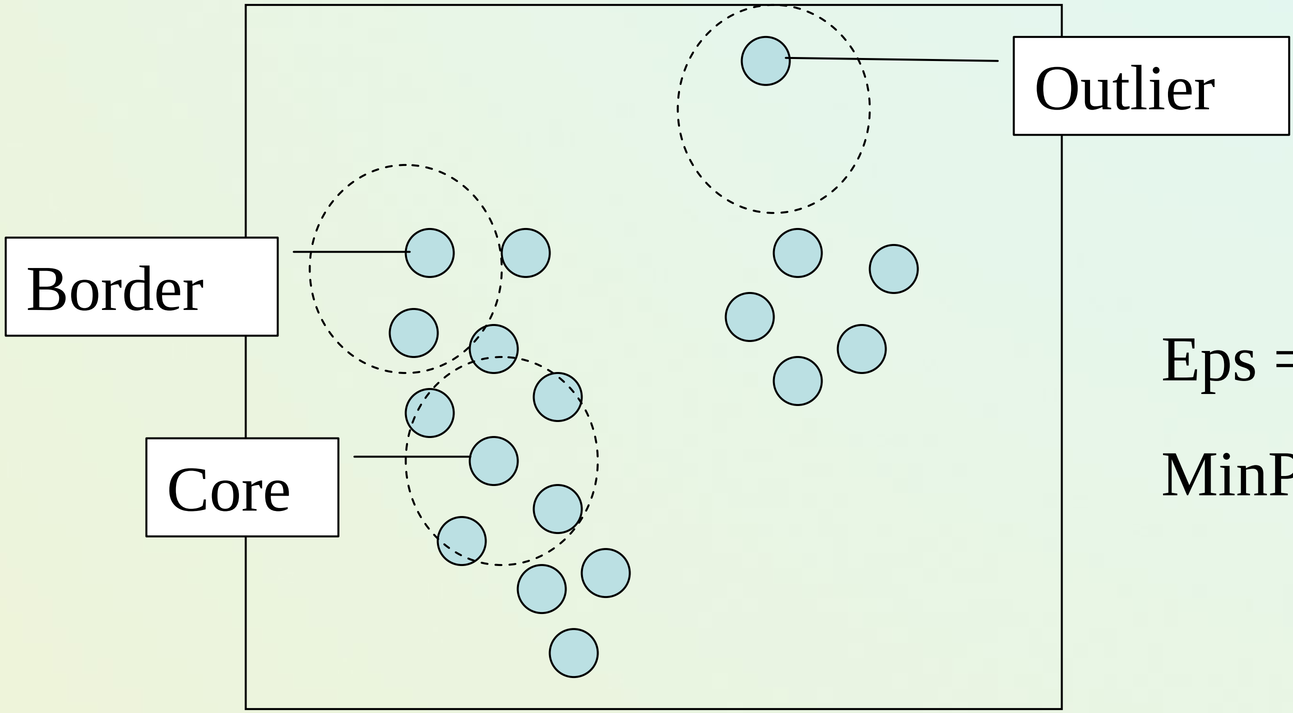


Density-Based Clustering



- Clustering based on density (local cluster criterion), such as density-connected points
- Each cluster has a considerable higher density of points than outside of the cluster

DBSCAN: General Ideas



Eps = 1cm

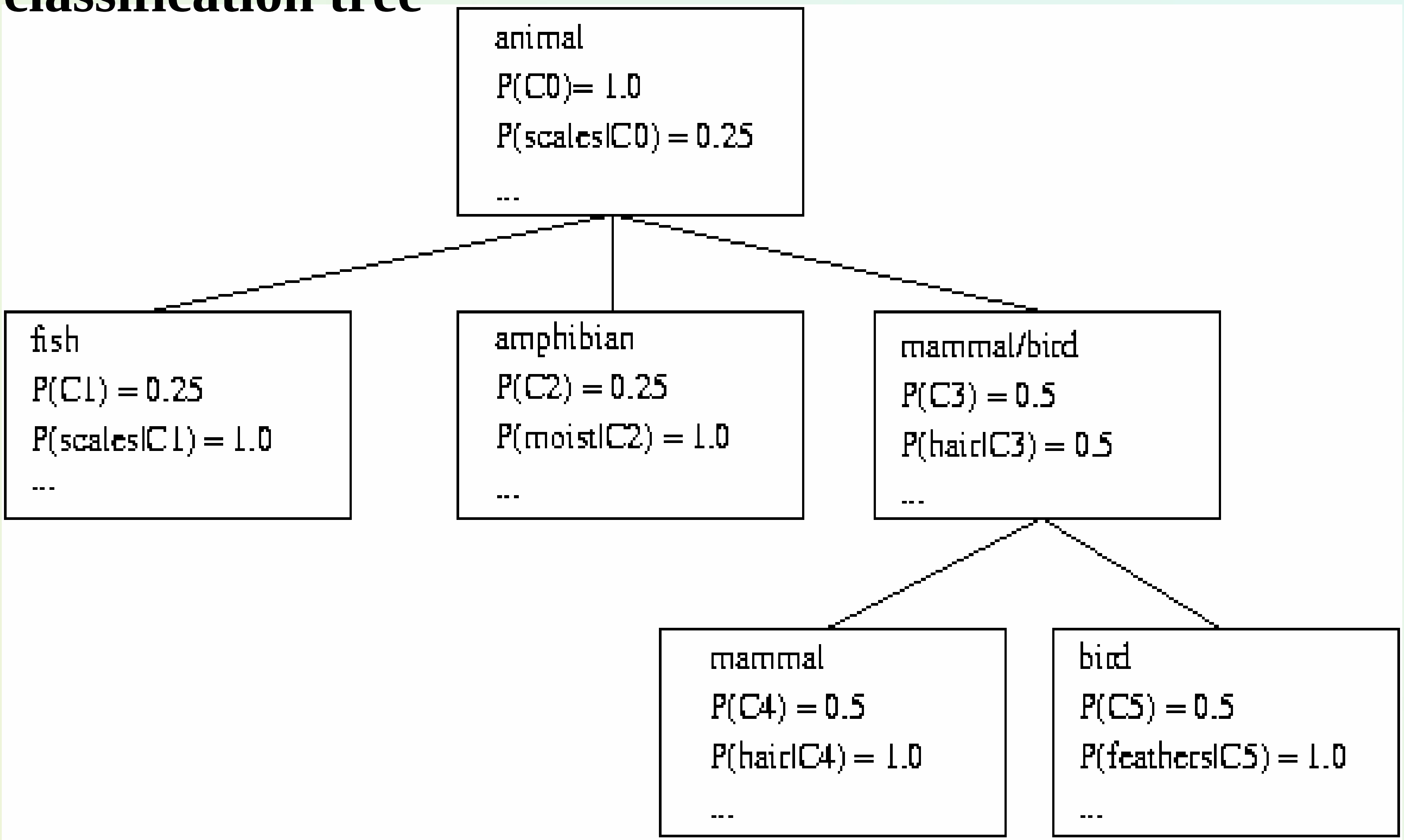
MinPts = 5

Model-Based Clustering Methods

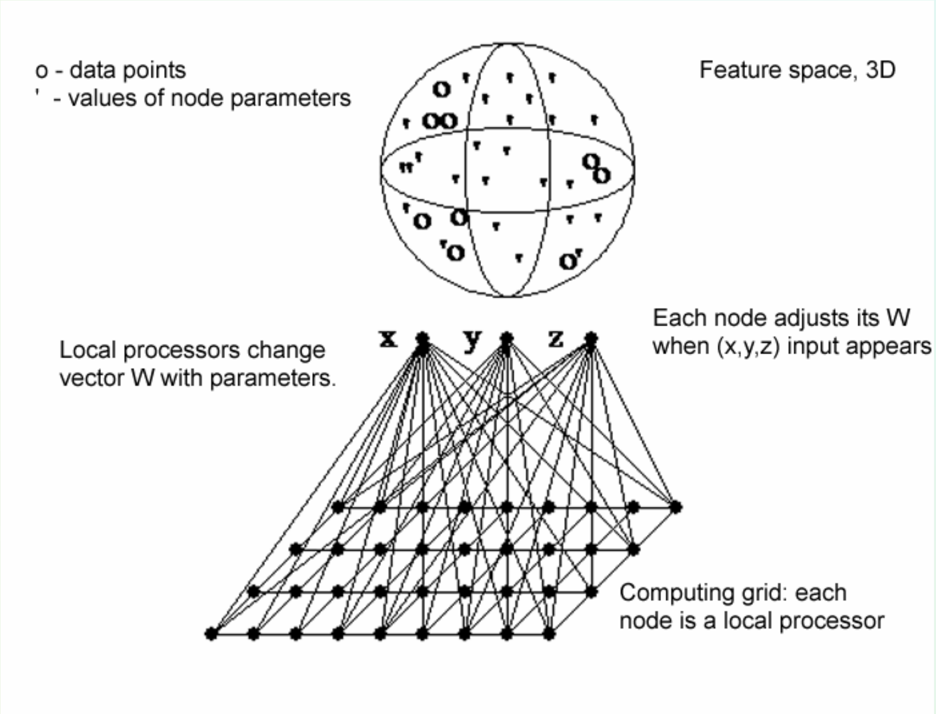
- Attempt to optimize the fit between the data and some mathematical model
- Statistical and AI approach
  - Conceptual clustering
    - A form of clustering in machine learning
    - Produces a classification scheme for a set of unlabeled objects
    - Finds characteristic description for each concept (class)
  - COBWEB (Fisher'87)
    - A popular a simple method of incremental conceptual learning
    - Creates a hierarchical clustering in the form of a classification tree
    - Each node refers to a concept and contains a probabilistic description of that concept

COBWEB Clustering Method

A classification tree



Self-Organizing Maps - more



Data: vectors $\mathbf{X}^T = (X_1, \dots X_d)$ from d-dimensional space.

Grid of nodes, with local processor (called neuron) in each node.

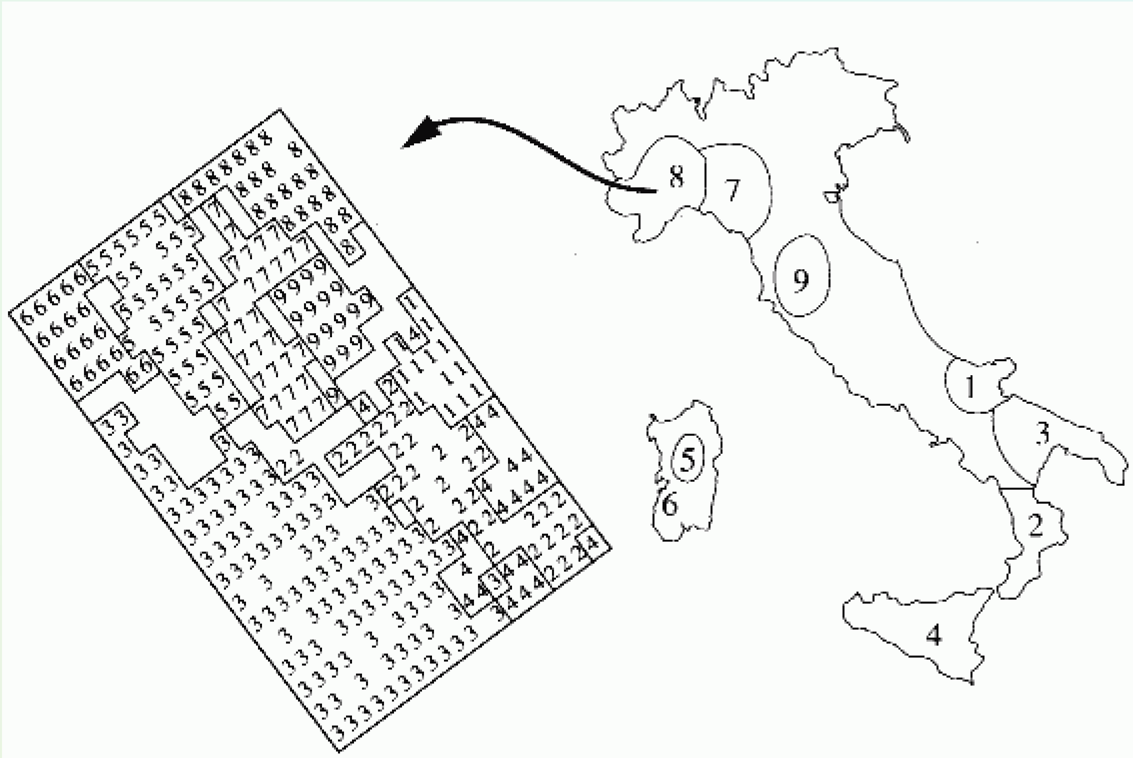
Local processor # j has d adaptive parameters $\mathbf{W}^{(j)}$.

Goal: change $\mathbf{W}^{(j)}$ parameters to recover data clusters in $\mathbf{X}$ space.

An example of analysing olive oil in Italy

An example of SOM application:

- 572 samples of olive oil were collected from 9 Italian provinces. Content of 8 fats was determine for each oil.
- SOM 20 x 20 network,
- Maps 8D => 2D.
- Classification accuracy was around 95-97%.



Note that topographical relations are preserved, region 3 is most diverse.

Clustering Evaluation

- Manual inspection
- Benchmarking on existing labels
  - Comparing clusters with ground-truth categories
- Cluster quality measures
  - distance measures
  - high similarity within a cluster, low across clusters

Evaluating variability of clusters

- Homogenuous clusters!
- Intuition $\rightarrow$ „zmienność wewnątrzskupieniowa" intra-class variability *wc*(C) i „zmienność międzyskupieniowa" inter-class distances *bc*(C)
  - May be defined in many ways
  - Take average of clusters $\mathbf{r}_k$ (centroids)
  - Then

$$wc(C) = \sum_{k=1}^{K} \sum_{\mathbf{x} \in C_k} d(\mathbf{x}, \mathbf{r}_k)^2$$

$$\mathbf{r}_k = \frac{1}{n_k} \sum_{\mathbf{x} \in C_k} \mathbf{x}$$

$$bc(C) = \sum_{1 \leq j < k \leq K} d(\mathbf{r}_j, \mathbf{r}_k)^2$$

Measure of Clustering Accuracy

- Accuracy
  - Measured by manually labeled data
    - We manually assign tuples into clusters according to their properties (e.g., professors in different research areas)
  - Accuracy of clustering: Percentage of pairs of tuples in the same cluster that share common label
    - This measure favors many small clusters
    - We let each approach generate the same number of clusters

Clustering Summary

- unsupervised
- many approaches
  - K-means – simple, sometimes useful
    - K-medoids is less sensitive to outliers
  - Hierarchical clustering – works for symbolic attributes
- Evaluation is a problem