POSTECH
POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Week 11-1

Clustering 1: Basics, Partitioning, Hierarchical Methods

Big Data

Prof. Hwanjo Yu
POSTECH

Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Summary

What is cluster analysis?

- Cluster: A collection of data objects
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis (or clustering, data segmentation, …)
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes (i.e. learning by observations vs. learning by examples: supervised)
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Clustering applications

- Biology
  - Taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- Information retrieval
  - Document clustering
- Land use
  - Identification of areas of similar land use in an earth observation database
- Marketing
  - Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- City-planning
  - Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies
  - Observed earth quake epicenters should be clustered along continent faults
- Climate
  - Understanding earth climate, find patterns of atmospheric and ocean
- …

Clustering as a preprocessing

- Summarization:
  - Preprocessing for regression, PCA, classification, and association analysis
- Compression:
  - Image processing: vector quantization
- Finding K-nearest Neighbors
  - Localizing search to one or a small number of clusters
- Outlier detection
  - Outliers are often viewed as those “far away” from any cluster

What is good clustering?

- A good clustering method will produce high quality clusters
  - high intra-class similarity: cohesive within clusters
  - low inter-class similarity: distinctive between clusters
- The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns

How to measure the clustering quality?

- Dissimilarity/Similarity metric
  - Similarity is expressed in terms of a distance function, typically metric: d(i, j)
  - The definitions of distance functions are usually rather different for interval-scaled, boolean, categorical, ordinal ratio, and vector variables
  - Weights should be associated with different variables based on applications and data semantics
- Quality of clustering:
  - There is usually a separate “quality” function that measures the “goodness” of a cluster.
  - It is hard to define “similar enough” or “good enough”
    - The answer is typically highly subjective

Considerations for cluster analysis

- Partitioning criteria
  - Single level

    vs. hierarchical partitioning (often, multi-level hierarchical partitioning is desirable)
- Separation of clusters
  - Exclusive (e.g. one customer belongs to only one region)

    vs. non-exclusive (e.g. one document may belong to more than one class)
- Similarity measure
  - Distance-based (e.g. Euclidian, road network, vector)

    vs. connectivity-based (e.g. density or contiguity)
- Clustering space
  - Full space (often when low dimensional)

    vs. subspaces (often in high-dimensional clustering)

Requirements and challenges

- Scalability
- Ability to deal with different types of attributes
  - Numerical, binary, categorical, ordinal, linked, and mixture of these
- Constraint-based clustering
  - User may give inputs on constraints
  - Use domain knowledge to determine input parameters
- Interpretability and usability
- Discovery of clusters with arbitrary shape
- Ability to deal with noisy data
- Incremental clustering and insensitivity to input order
- High dimensionality

Major clustering approaches

- Partitioning approach:
  - Construct various partitions and then evaluate them by some criterion, e.g. minimizing the sum of square errors
  - k-means, k-medoids, CLARANS
- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Agglomerative clustering (AGNES), Diana, BIRCH, CAMELEON
- Density-based approach:
  - Based on connectivity and density functions
  - DBSCAN, OPTICS, DenClue
- Grid-based approach:
  - based on a multiple-level granularity structure
  - STING, WaveCluster, CLIQUE

Major clustering approaches

- Model-based:
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
  - EM, SOM, COBWEB
- Frequent pattern-based:
  - Based on the analysis of frequent patterns
  - p-Cluster
- User-guided or constraint-based:
  - Clustering by considering user-specified or application-specific constraints
  - COD (obstacles), constrained clustering
- Link-based clustering:
  - Objects are often linked together in various ways
  - SimRank, LinkClus

Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Summary

Partitioning algorithms

- Partitioning method: Partitioning a database ***D*** of ***n*** objects into a set of ***k*** clusters, such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)

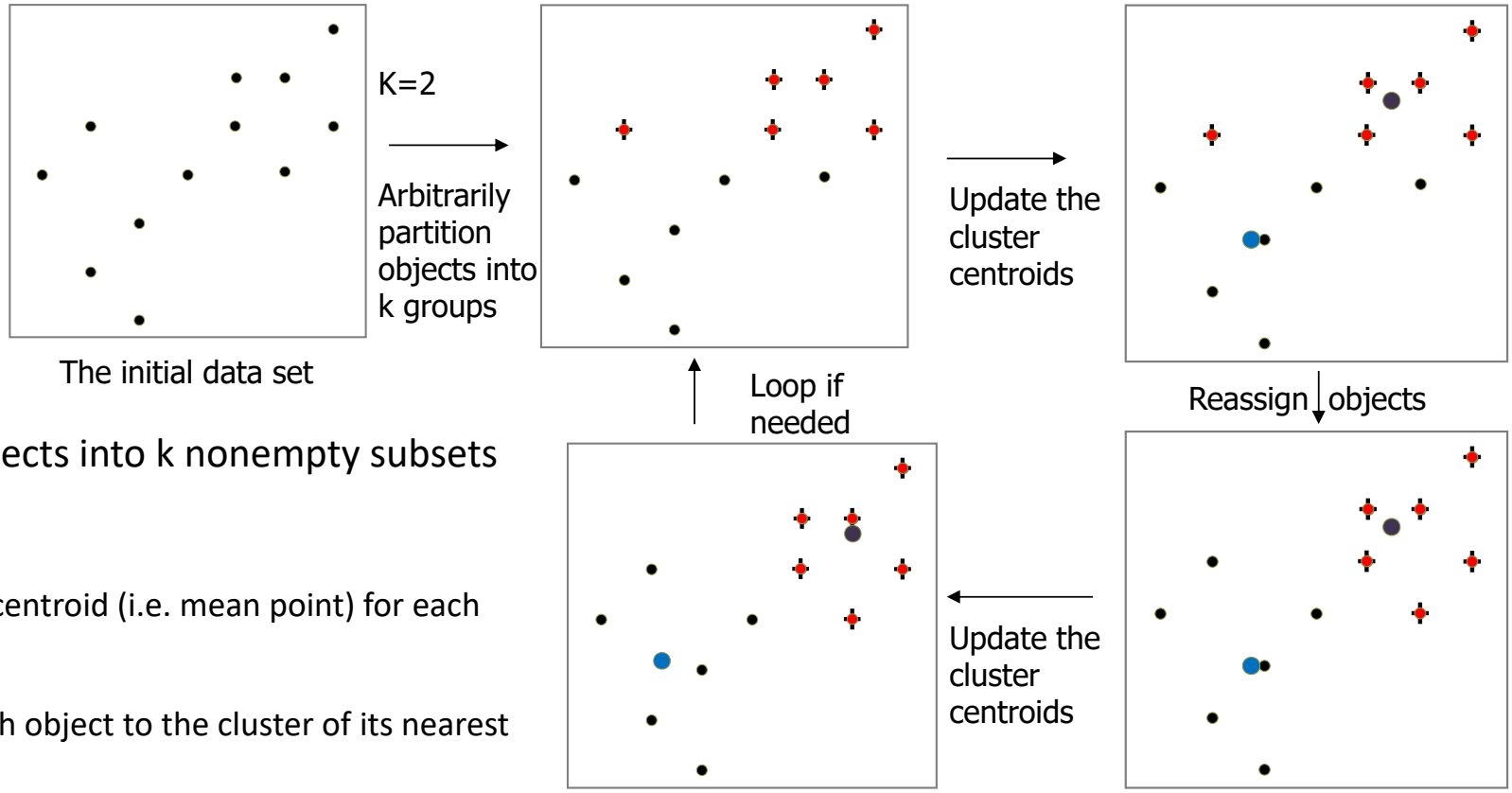
$$E = \sum_{i=1}^{k} \sum_{p \in C_i} (p - c_i)^2$$

- Given *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, Lloyd'57/'82): 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

The k-means clustering method

- Given k, the k-means algorithm is implemented in four steps:
  - Partition objects into k nonempty subsets
  - Compute seed points as the centroids of the clusters of the current partitioning (the centroid is the center, i.e. mean point, of the cluster)
  - Assign each object to the cluster with the nearest seed point
  - Go back to Step 2, stop when the assignment does not change

An example of k-means clustering



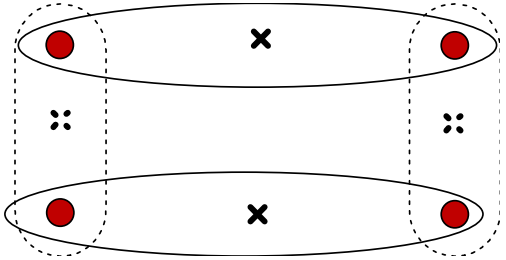
- Partition objects into k nonempty subsets
- Repeat
  - Compute centroid (i.e. mean point) for each partition
  - Assign each object to the cluster of its nearest centroid
- Until no change

Comments on the k-means method

- Strength: Efficient: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, t << n.
  - Comparing: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
- Comment: Often terminates at a local optimal.
- Weakness
  - Applicable only to objects in a continuous n-dimensional space
    - Using the k-modes method for categorical data
    - In comparison, k-medoids can be applied to a wide range of data
  - Need to specify k, the number of clusters, in advance (there are ways to automatically determine the best k (see Hastie et al., 2009)
  - Sensitive to noisy data and outliers
  - Not suitable to discover clusters with non-convex shapes

Variations of the k-means method

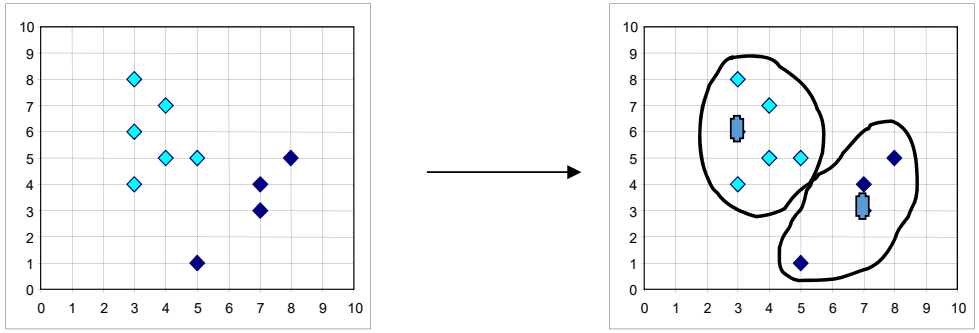
- Most of the variants of the k-means which differ in
  - Selection of the initial k means
  - Dissimilarity calculations
  - Strategies to calculate cluster means



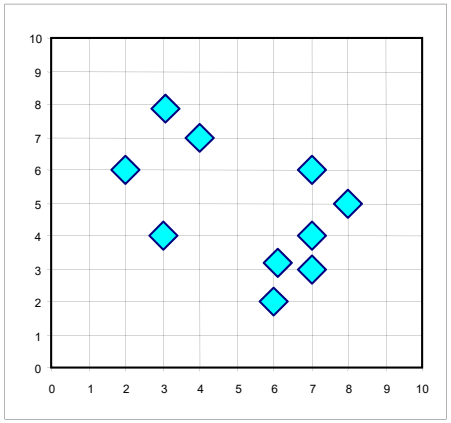
- Handling categorical data: k-modes
  - Replacing means of clusters with modes
  - Using new dissimilarity measures to deal with categorical objects
  - Using a frequency-based method to update modes of clusters
  - A mixture of categorical and numerical data: k-prototype method

Problem of the 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
- 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

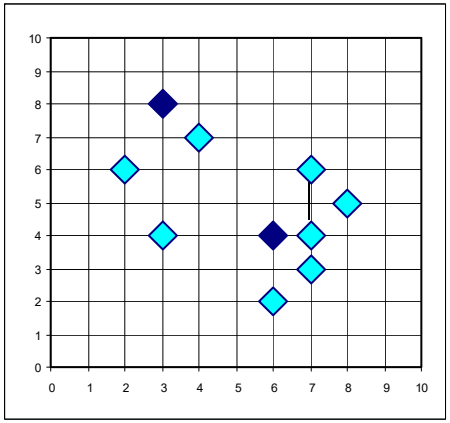


PAM: A typical k-medoids algorithm



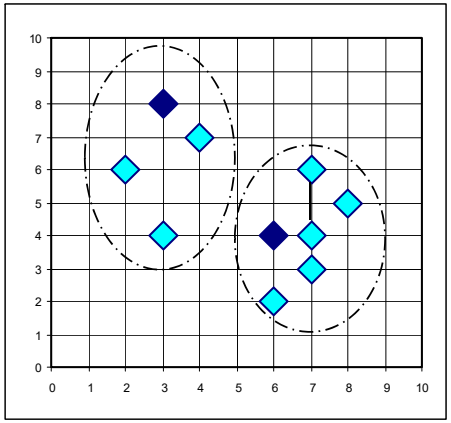
K=2

Arbitrary choose k object as initial medoids

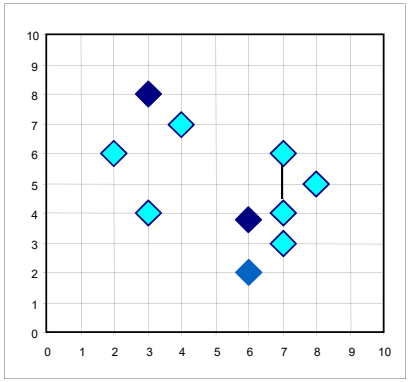


Assign each remaining object to nearest medoids

Total Cost = 20

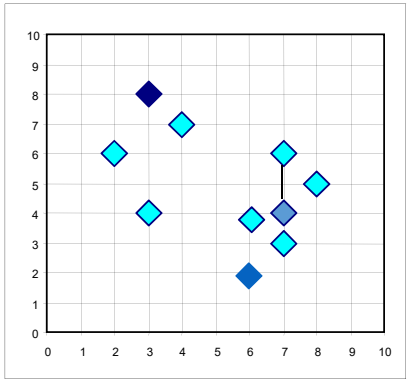


Randomly select a nonmedoid object,O_{ramdom}



Compute total cost of swapping

Total Cost = 26



Swapping O and O_{ramdom}

If quality is improved.

Do loop

Until no change

The k-medoid clustering method

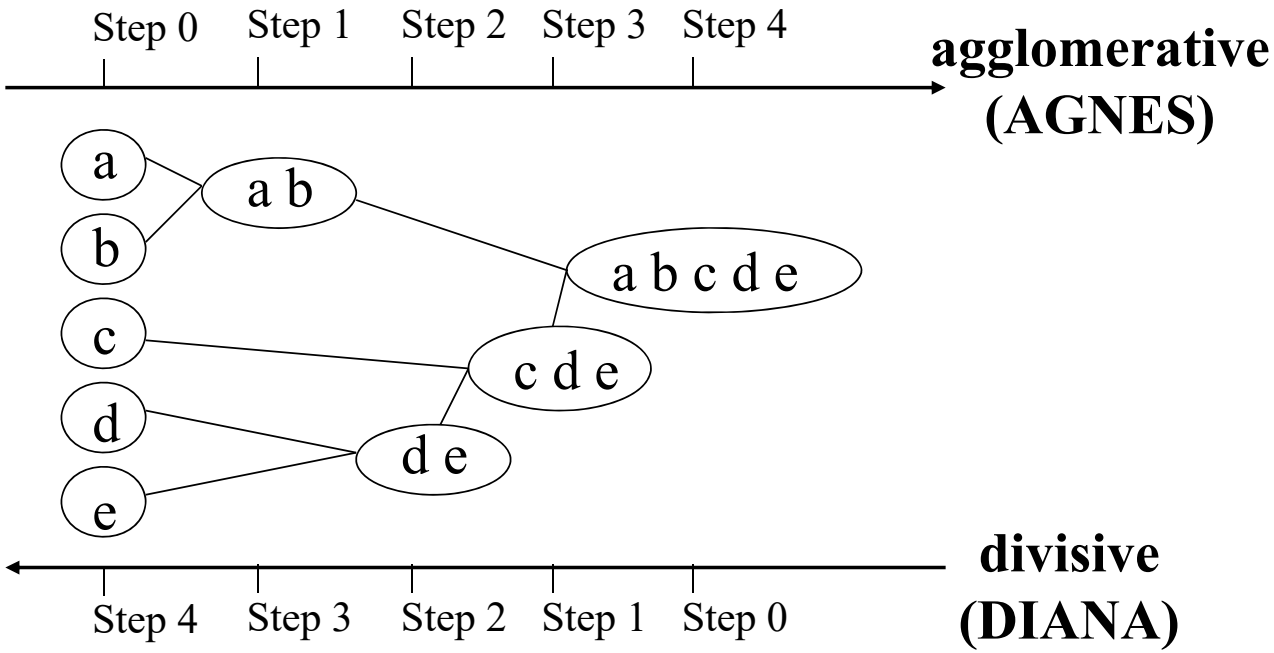
- K-Medoids Clustering: Find representative objects (medoids) in clusters
  - PAM (Partitioning Around Medoids, Kaufmann & Rousseeuw 1987)
    - Starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
    - PAM works effectively for small data sets, but does not scale well for large data sets (due to the computational complexity)
- Efficiency improvement on PAM
  - CLARA (Kaufmann & Rousseeuw, 1990): PAM on samples
  - CLARANS (Ng & Han, 1994): Randomized re-sampling

Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Summary

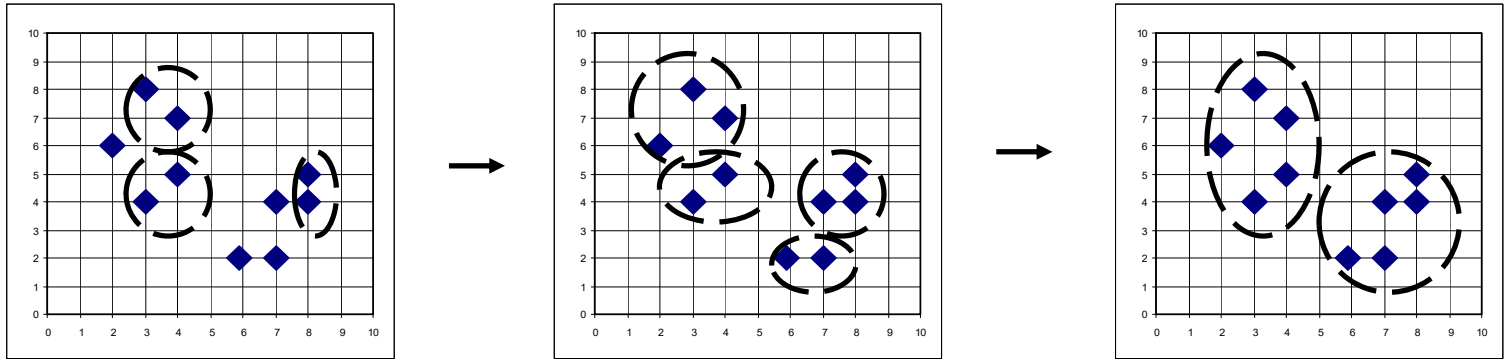
Hierarchical clustering

- Does not require the number of clusters k as an input, but needs a termination condition



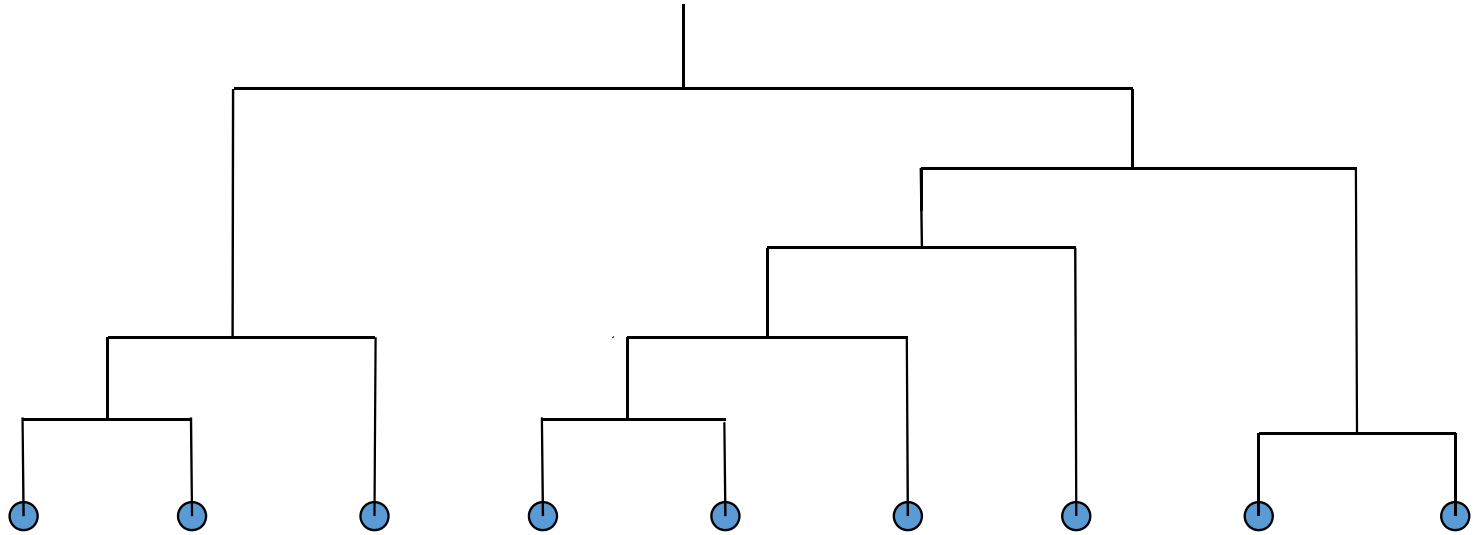
AGNES (Agglomerative Nesting)

- Introduced in Kaufmann and Rousseeuw (1990)
- Use the **single-link** method and the dissimilarity matrix
- Merge nodes that have the least dissimilarity
- Eventually all nodes belong to the same cluster



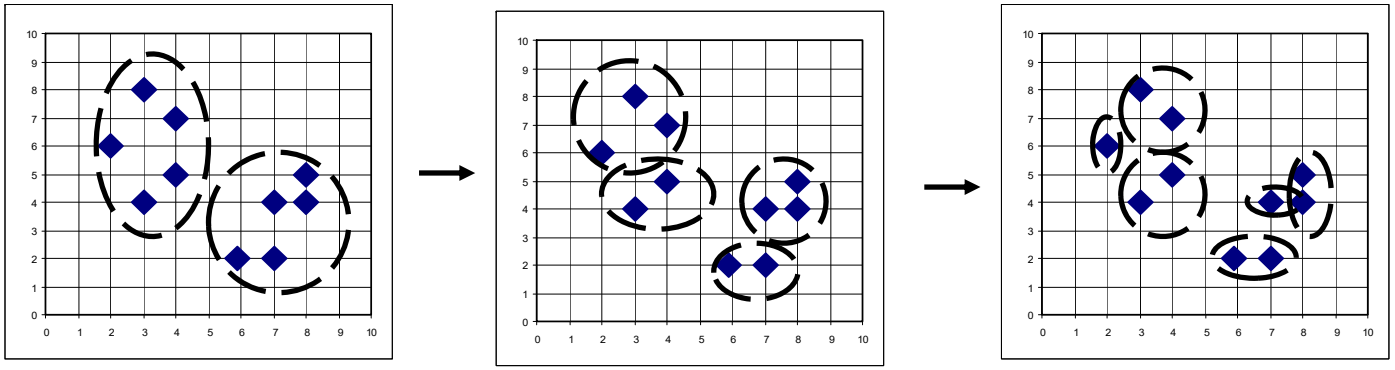
Dendrogram: Shows how clusters are merged

- Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram
- A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster



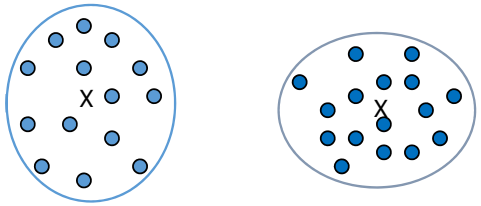
DIANA (Divisive Analysis)

- Introduced in Kaufmann and Rousseeuw (1990)
- Inverse order of AGNES
- Eventually each node forms a cluster on its own



Distance between clusters

- **Single link**: smallest distance between an element in one cluster and an element in the other, i.e. dist(K_i, K_j) = min(t_{ip}, t_{jq})



- **Complete link**: largest distance between an element in one cluster and an element in the other, i.e. dist(K_i, K_j) = max(t_{ip}, t_{jq})
- **Average**: avg distance between an element in one cluster and an element in the other, i.e. dist(K_i, K_j) = avg(t_{ip}, t_{jq})
- **Centroid**: distance between the centroids of two clusters, i.e. dist(K_i, K_j) = dist(C_i, C_j)
- **Medoid**: distance between the medoids of two clusters, i.e. dist(K_i, K_j) = dist(M_i, M_j)
  - Medoid: a chosen, centrally located object in the cluster

Centroid, radius and diameter of a cluster (for numerical data sets)

- Centroid: the "middle" of a cluster

$$C_m = \frac{\sum_{i=1}^{N}(t_{ip})}{N}$$

- Radius: square root of average distance from any point of the cluster to its centroid

$$R_m = \sqrt{\frac{\sum_{i=1}^{N}(t_{ip} - c_m)^2}{N}}$$

- Diameter: square root of average mean squared distance between all pairs of points in the cluster

$$D_m = \sqrt{\frac{\sum_{i=1}^{N}\ \sum_{j=1}^{N}(t_{ip} - t_{jq})^2}{N(N-1)}}$$

Extensions to hierarchical clustering

- Major weakness of agglomerative clustering methods
  - Can never undo what was done previously
  - Do not scale well: time complexity of at least O(n^2), where n is the number of total objects
- Integration of hierarchical & distance-based clustering
  - BIRCH (1996): uses CF-tree and incrementally adjusts the quality of sub-clusters
  - CHAMELEON (1999): hierarchical clustering using dynamic modeling