Clustering

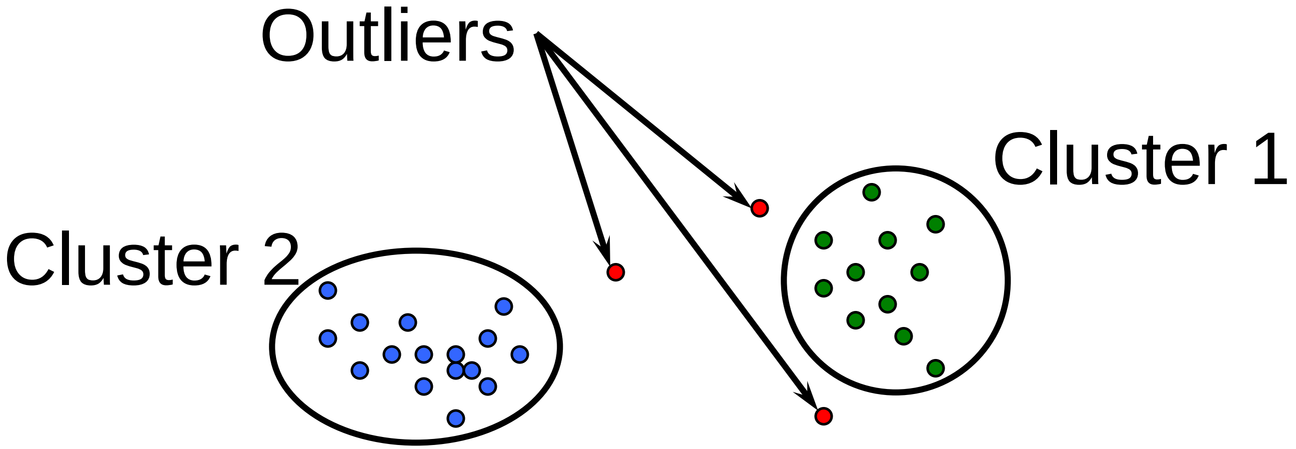
CS 145
Fall 2015
Wei Wang

Outline

- What is clustering
- Partitioning methods
- Hierarchical methods
- Density-based methods
- Grid-based methods
- Model-based clustering methods
- Outlier analysis

What Is Clustering?

- Group data into clusters
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
  - Unsupervised learning: no predefined classes



Application Examples

- A stand-alone tool: explore data distribution
- A preprocessing step for other algorithms
- Pattern recognition, spatial data analysis, image processing, market research, WWW, …
  - Cluster documents
  - Cluster web log data to discover groups of similar access patterns

What Is A Good Clustering?

- High intra-class similarity and low inter-class similarity
  - Depending on the similarity measure
- The ability to discover some or all of the hidden patterns

Requirements of Clustering

- Scalability
- Ability to deal with various types of attributes
- Discovery of clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters

Requirements of Clustering

- Able to deal with noise and outliers
- Insensitive to order of input records
- High dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

Data Matrix

- For memory-based clustering
  - Also called object-by-variable structure
- Represents n objects with p variables (attributes, measures)
  - A relational table

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

Dissimilarity Matrix

- For memory-based clustering
  - Also called object-by-object structure
  - Proximities of pairs of objects
  - d(i,j): dissimilarity between objects i and j
  - Nonnegative
  - Close to 0: similar

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

How Good Is A Clustering?

- Dissimilarity/similarity depends on distance function
  - Different applications have different functions
- Judgment of clustering quality is typically highly subjective

Types of Data in Clustering

- Interval-scaled variables
- Binary variables
- Nominal, ordinal, and ratio variables
- Variables of mixed types

Similarity and Dissimilarity Between Objects

- Distances are normally used measures
- Minkowski distance: a generalization

$$d(i,j)=\sqrt[q]{|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+\ldots+|x_{i_p}-x_{j_p}|^q} \quad (q>0)$$

- If q = 2, d is Euclidean distance
- If q = 1, d is Manhattan distance
- Weighted distance

$$d(i,j)=\sqrt[q]{w_1|x_{i_1}-x_{j_1}|^q+w_2|x_{i_2}-x_{j_2}|^q+\ldots+w_p|x_{i_p}-x_{j_p}|^q)} \quad (q>0)$$

Properties of Minkowski Distance

- Nonnegative: $d(i,j) \geq 0$
- The distance of an object to itself is 0
  - $d(i,i) = 0$
- Symmetric: $d(i,j) = d(j,i)$
- Triangular inequality
  - $d(i,j) \leq d(i,k) + d(k,j)$

Categories of Clustering Approaches (1)

- Partitioning algorithms
  - Partition the objects into k clusters
  - Iteratively reallocate objects to improve the clustering
- Hierarchy algorithms
  - Agglomerative: each object is a cluster, merge clusters to form larger ones
  - Divisive: all objects are in a cluster, split it up into smaller clusters

Categories of Clustering Approaches (2)

- Density-based methods
  - Based on connectivity and density functions
  - Filter out noise, find clusters of arbitrary shape
- Grid-based methods
  - Quantize the object space into a grid structure
- Model-based
  - Use a model to find the best fit of data

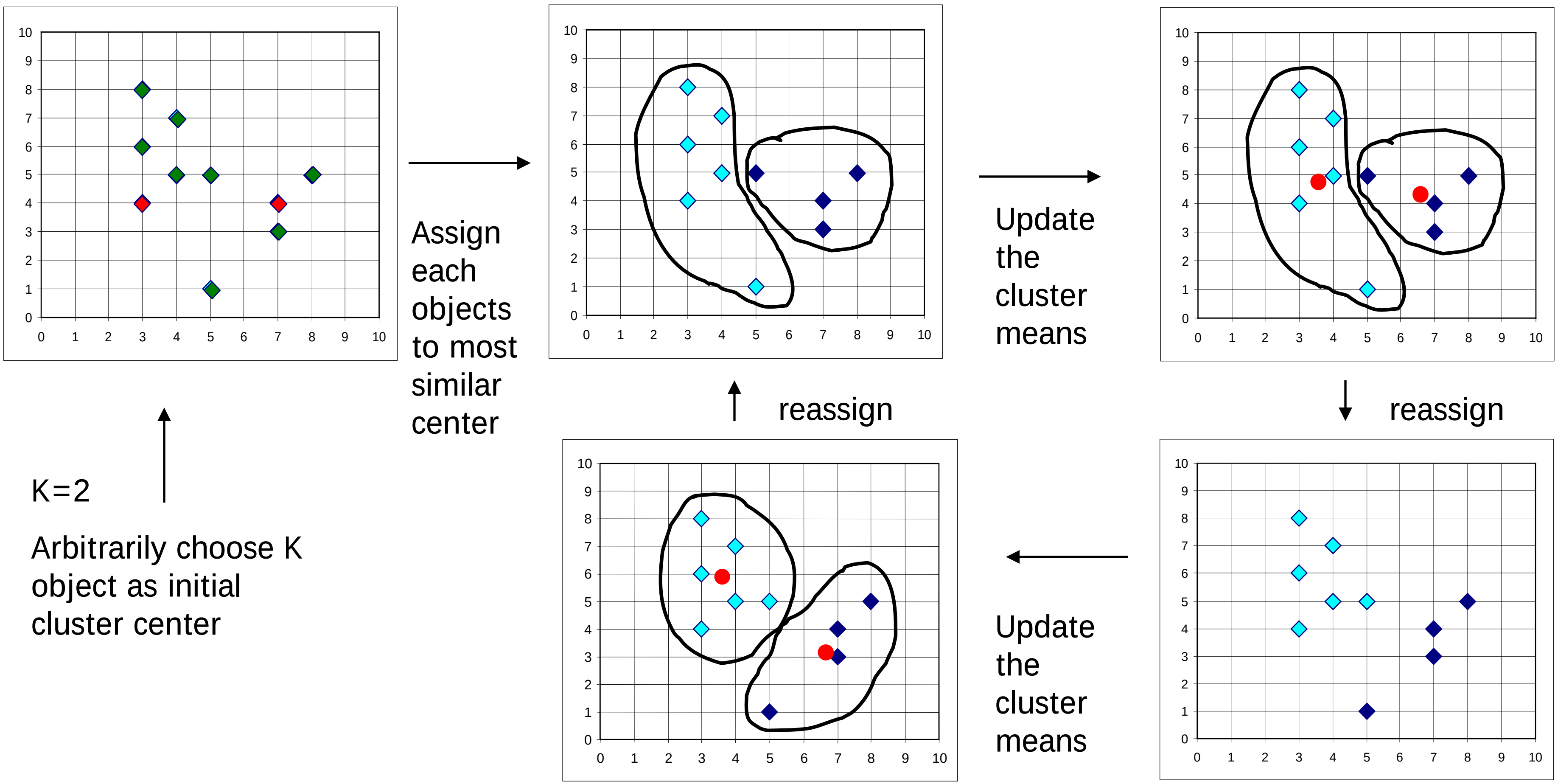
Partitioning Algorithms: Basic Concepts

- Partition n objects into k clusters
  - Optimize the chosen partitioning criterion
- Global optimal: examine all partitions
  - $(k^n-(k-1)^n-\ldots-1)$ possible partitions, too expensive!
- Heuristic methods: k-means and k-medoids
  - K-means: a cluster is represented by the center
  - K-medoids or PAM (partition around medoids): each cluster is represented by one of the objects in the cluster

K-means

- Arbitrarily choose k objects as the initial cluster centers
- Until no change, do
  - (Re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster
  - Update the cluster means, i.e., calculate the mean value of the objects for each cluster

K-Means: Example



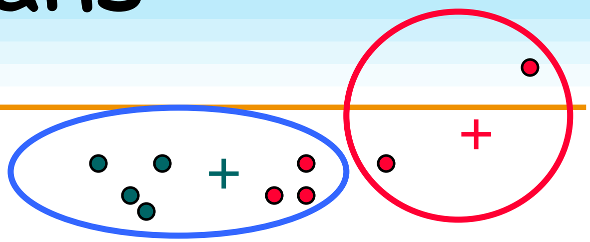
Pros and Cons of K-means

- Relatively efficient: O(tkn)
  - n: # objects, k: # clusters, t: # iterations; k, t << n.
- Often terminate at a local optimum
- Applicable only when mean is defined
  - What about categorical data?
- Need to specify the number of clusters
- Unable to handle noisy data and outliers
- unsuitable to discover non-convex clusters

Variations of the K-means

- Aspects of variations
  - Selection of the initial k means
  - Dissimilarity calculations
  - Strategies to calculate cluster means
- Handling categorical data: k-modes
  - Use mode instead of mean
    - Mode: the most frequent item(s)
  - A mixture of categorical and numerical data: k-prototype method

A Problem of K-means



- Sensitive to outliers
  - Outlier: objects with extremely large values
    - May substantially distort the distribution of the data
- K-medoids: the most centrally located object in a cluster

PAM: A K-medoids Method

- PAM: partitioning around Medoids
- Arbitrarily choose k objects as the initial medoids
- Until no change, do
  - (Re)assign each object to the cluster to which the nearest medoid
  - Randomly select a non-medoid object o’, compute the total cost, S, of swapping medoid o with o’
  - If S < 0 then swap o with o’ to form the new set of k medoids

Swapping Cost

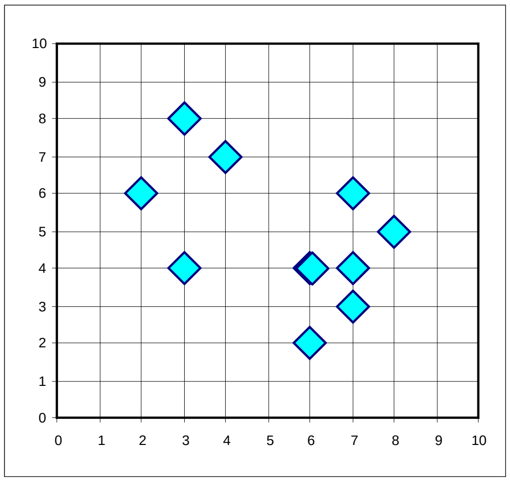
- Measure whether o' is better than o as a medoid
- Use the squared-error criterion

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} d(p, o_i)^2$$

  - Compute $E_{o'}-E_o$
  - Negative: swapping brings benefit

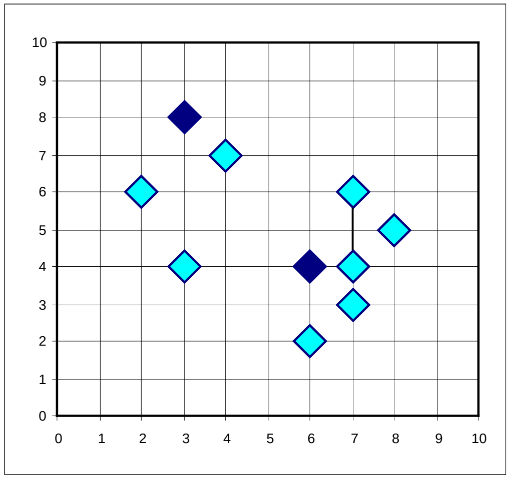
PAM: Example

Total Cost = 20

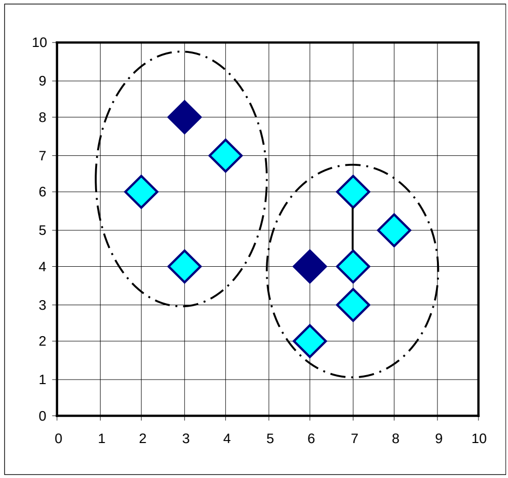


K=2

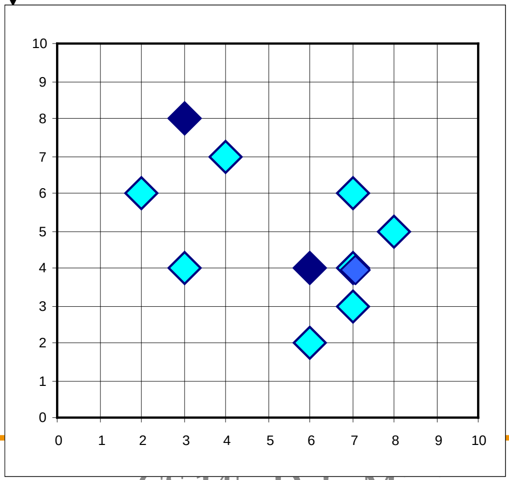
Arbitrary choose k object as initial medoids



Assign each remaining object to nearest medoids

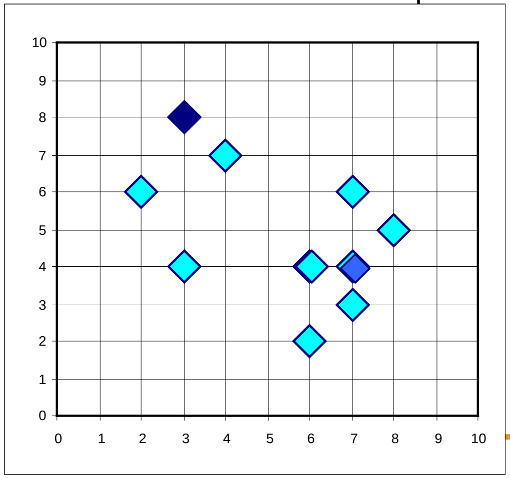


Randomly select a nonmedoid object,O_{ramdom}



Compute total cost of swapping

Total Cost = 26



Swapping O and O_{ramdom}

If quality is improved.

Do loop

Until no change

Pros and Cons of PAM

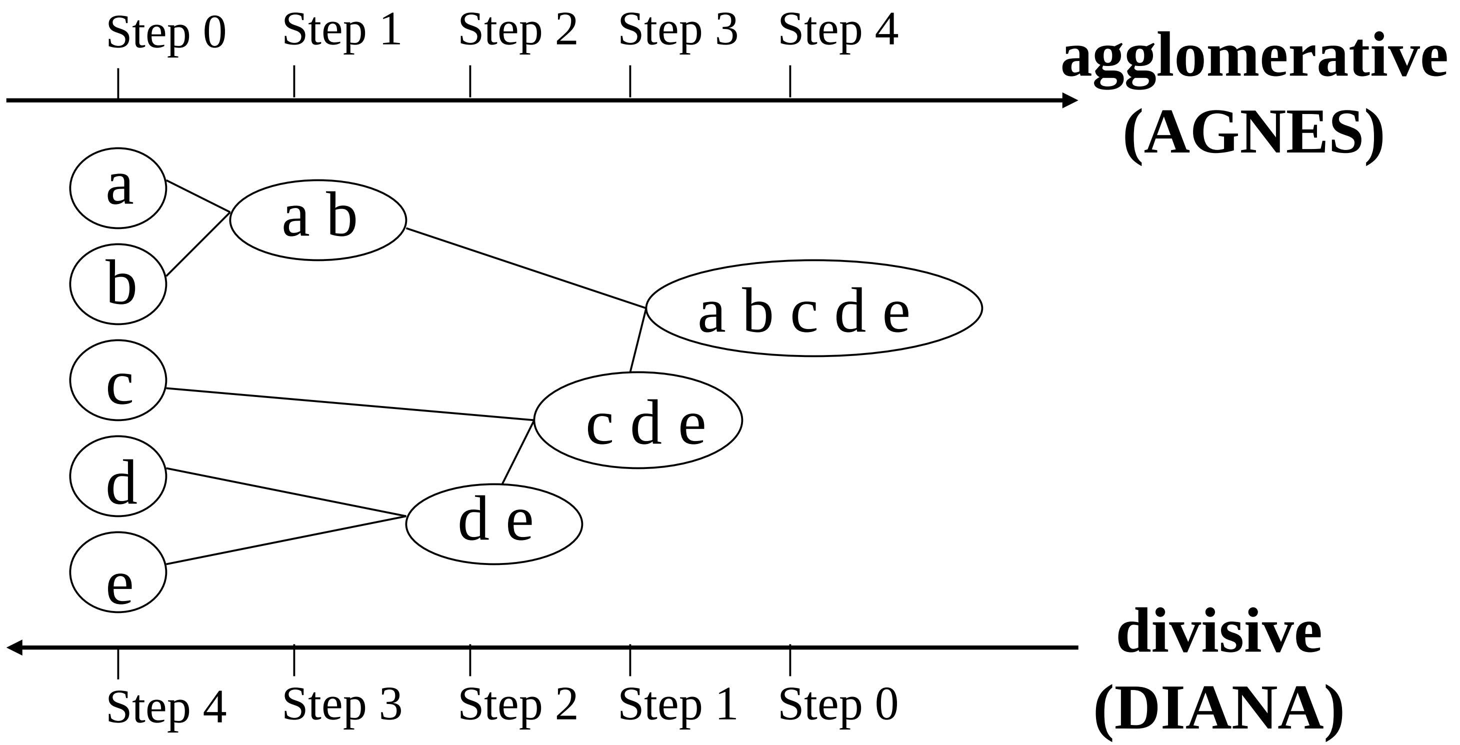
- PAM is more robust than k-means in the presence of noise and outliers
  - Medoids are less influenced by outliers
- PAM is efficiently for small data sets but does not scale well for large data sets
- Sampling based method: CLARA

CLARA (Clustering LARge Applications)

- CLARA (Kaufmann and Rousseeuw in 1990)
  - Built in statistical analysis packages, such as S+
- Draw multiple samples of the data set, apply PAM on each sample, give the best clustering
- Perform better than PAM in larger data sets
- Efficiency depends on the sample size
  - A good clustering on samples may not be a good clustering of the whole data set

Hierarchical Clustering

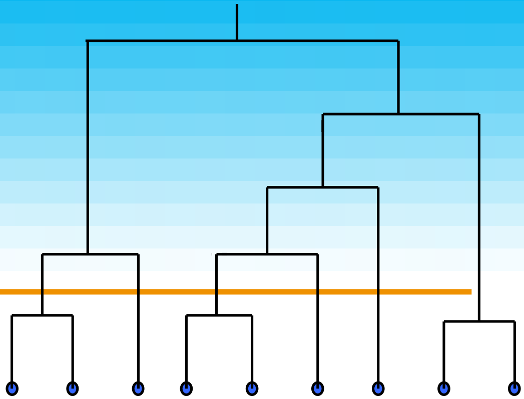
- Group data objects into a tree of clusters



AGNES (Agglomerative Nesting)

- Initially, each object is a cluster
- Step-by-step cluster merging, until all objects form a cluster
  - Single-link approach
  - Each cluster is represented by all of the objects in the cluster
  - The similarity between two clusters is measured by the similarity of the closest pair of data points belonging to different clusters

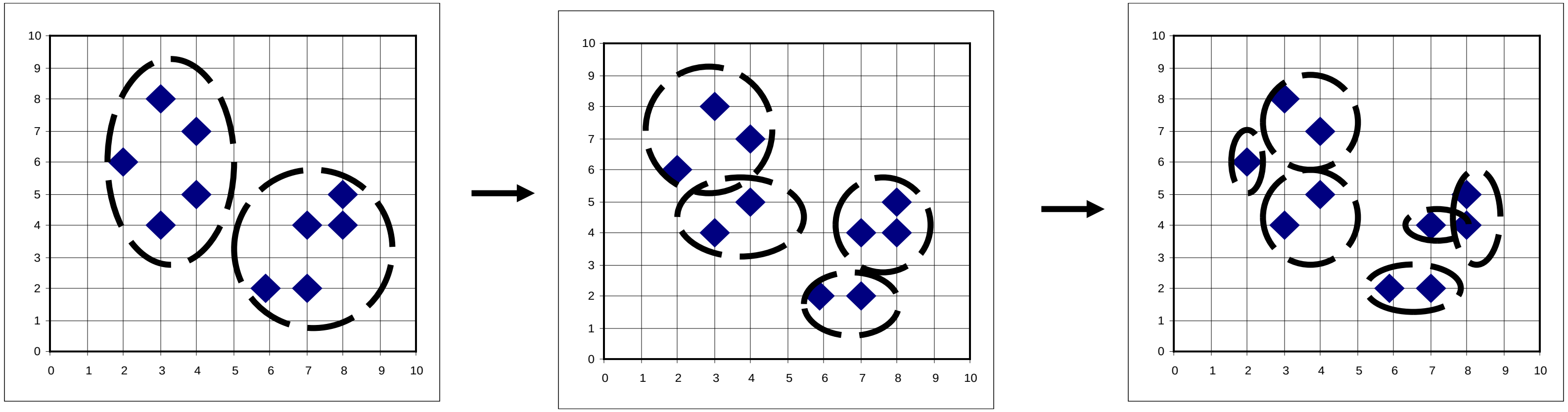
Dendrogram



- Show how to merge clusters hierarchically
- Decompose data objects into a multi-level nested partitioning (a tree of clusters)
- A clustering of the data objects: cutting the dendrogram at the desired level
  - Each connected component forms a cluster

DIANA (DIvisive ANAlysis)

- Initially, all objects are in one cluster
- Step-by-step splitting clusters until each cluster contains only one object



Distance Measures

- Minimum distance $d_{\min}(C_i, C_j) = \min_{p \in C_i, q \in C_j} d(p,q)$
- Maximum distance $d_{\max}(C_i, C_j) = \max_{p \in C_i, q \in C_j} d(p,q)$
- Mean distance $d_{mean}(C_i, C_j) = d(m_i, m_j)$
- Average distance $d_{avg}(C_i, C_j) = \frac{1}{n_i n_j} \sum_{p \in C_i} \sum_{q \in C_j} d(p,q)$

m: mean for a cluster
C: a cluster
n: the number of objects in a cluster