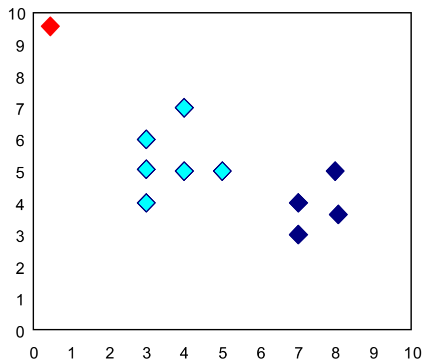
What Is the Problem of the K-Means Method?

- The k-means algorithm is sensitive to *outliers* !
  - An object with an extremely large value may substantially distort the distribution of the data
- K-Medoids: Instead of taking the **mean** value (i.e., *centroids*) of the object in a cluster as a reference point, *a medoids* can be used, which is the *most centrally-located object* in a cluster



The *K-Medoids* Clustering Method

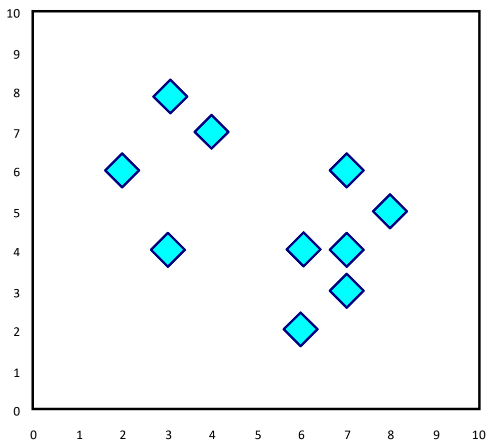
- Find *representative* objects, called medoids, in clusters
  - *PAM* (Partitioning Around Medoids, 1987)
  - *CLARA* (Kaufmann & Rousseeuw, 1990)
  - *CLARANS* (Ng & Han, 1994): Randomized sampling

PAM (Partitioning Around Medoids) (1987)

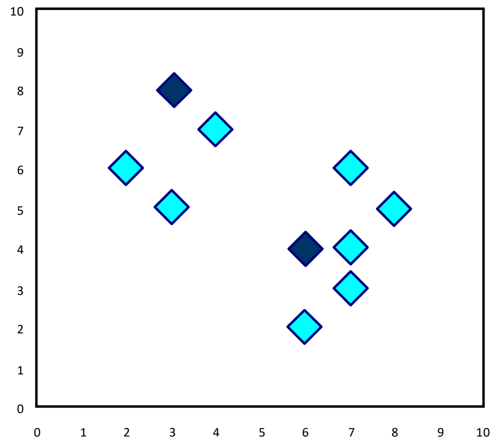
- PAM (Kaufman and Rousseeuw, 1987), built in Splus
- Use a real object to represent the cluster
  - Select ***k*** representative objects arbitrarily
  - For each pair of non-selected object ***h*** and selected object (i.e., seed) ***i***, calculate the total swapping cost TC_{ih}
  - For each pair of ***i*** and ***h***,
    - If $TC_{ih} < 0$, ***i*** is replaced by ***h***
    - Then, each non-selected object is assigned to the most similar representative object
  - Repeat steps 2-3 until there is no change

A Typical K-Medoids Algorithm (PAM)

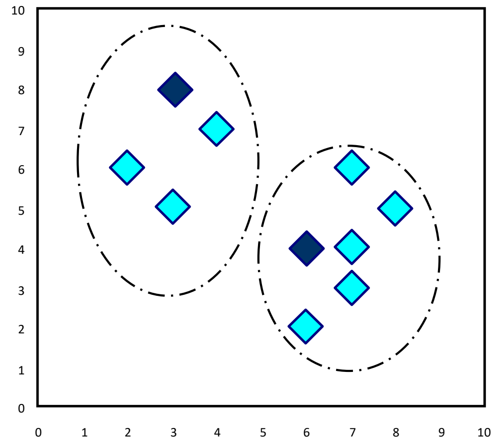
Total Cost = 20



Arbitrary choose k objects as initial medoids



Assign each remaining object to nearest medoids



K=2

Randomly select a nonmedoid object,O_{ramdom}

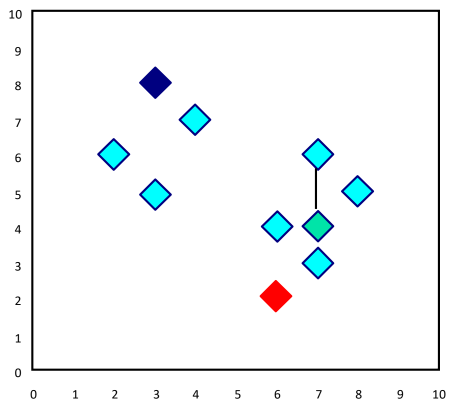
Total Cost = 26

Do loop

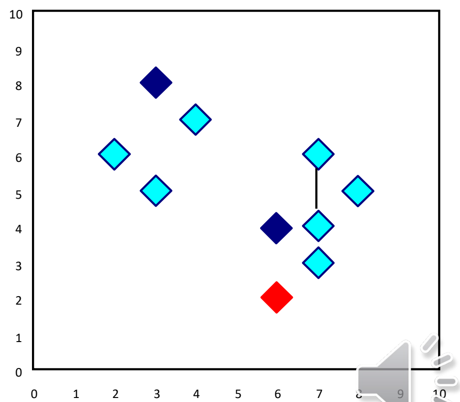
Until no change

Swapping O and O_{ramdom}

If quality is improved.



Compute total cost of swapping

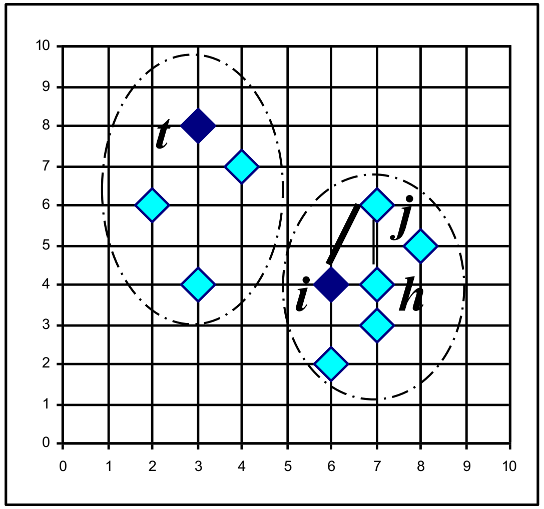


PAM Clustering: Total swapping cost $TC_{ih}=\sum_j C_{jih}$

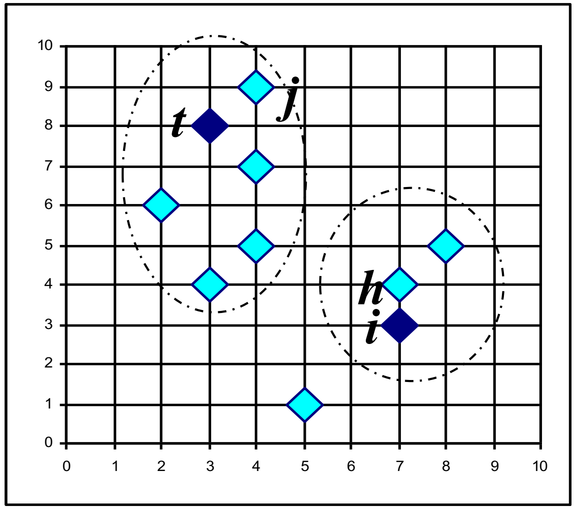
NewC - OldC

- i: original seed
- h: new seed
- t: other seed
- j: non-seed

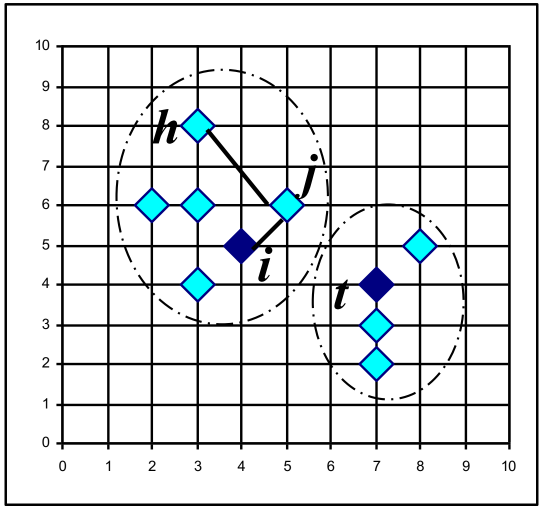
A: j belonged to i and now belongs to h
B: j belonged to t and again belongs to t
C: j belonged to i and now belongs to t
D: j belonged to t and now belongs to h



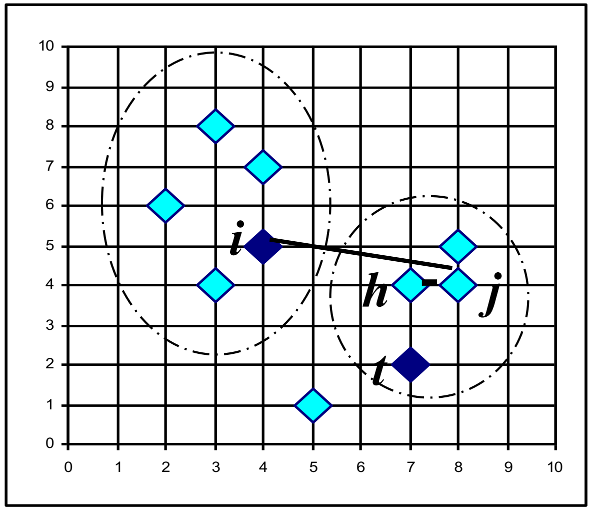
$$C_{jih} = d(j, h) - d(j, i)$$



$$C_{jih} = 0$$



$$C_{jih} = d(j, t) - d(j, i)$$



$$C_{jih} = d(j, h) - d(j, t)$$

What Is the Problem with PAM?

- PAM is more robust than k-means in the presence of noise and outliers
  - because a medoid is less influenced by outliers or other extreme values than a mean (i.e., centroid)
- PAM works efficiently for small data sets but does not **scale well** for large data sets.
  - $O(i*k*(n-k)^2)$ where *n* is # of data, *k* is # of clusters, *i* is # of iterations

→ Sampling based method,

CLARA (Clustering LARge Applications)

CLARA (Clustering Large Applications) (1990)

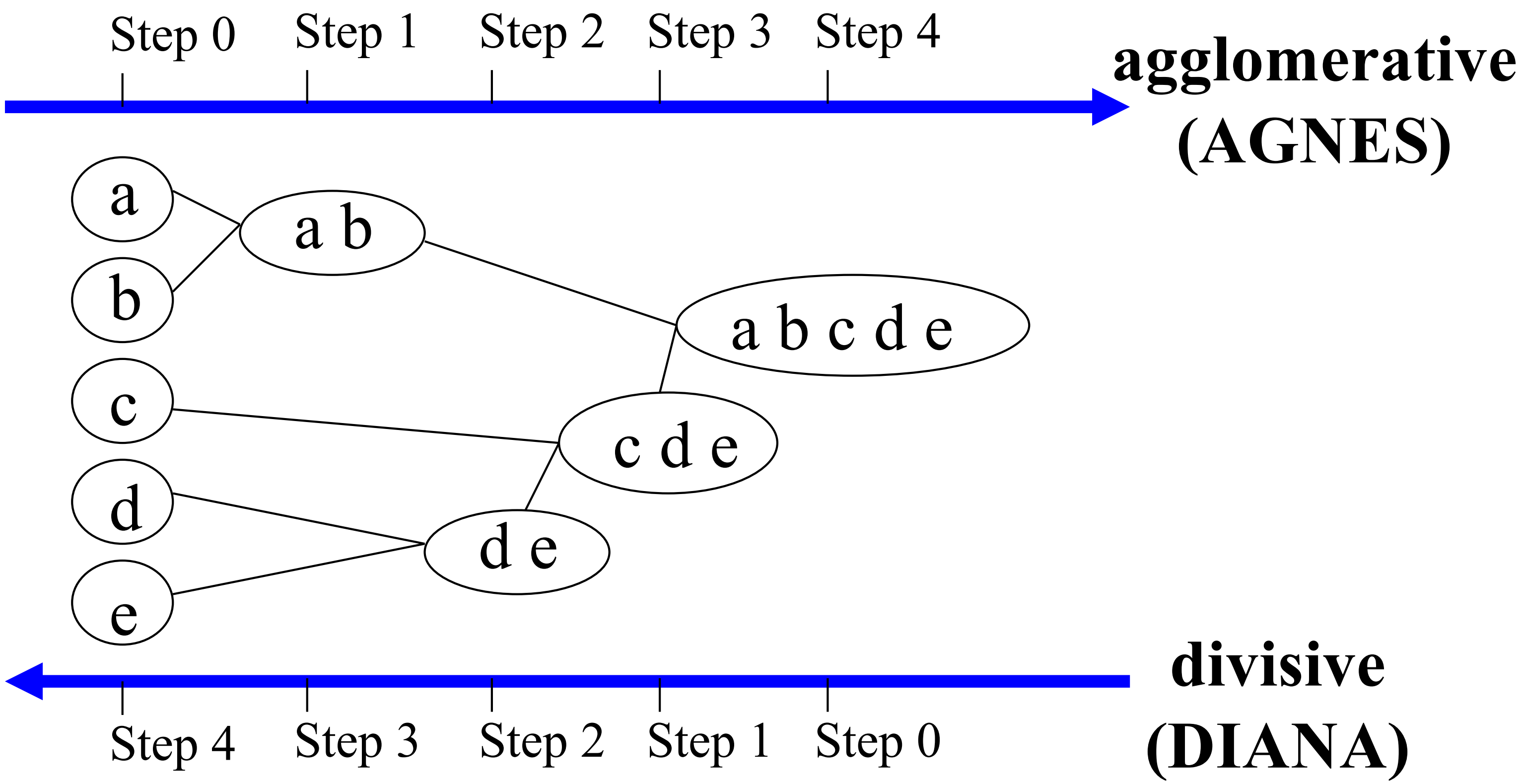
- *CLARA* (Kaufmann and Rousseeuw in 1990)
  - Built in statistical analysis packages, such as S+
- It draws *multiple samples* of the data set, applies *PAM* on each sample, and gives the best clustering as the output
- Strength: deals with larger data sets than *PAM*
- Weakness:
  - Efficiency *depends on the sample size*
  - A good clustering based on samples will not necessarily represent a good clustering of the whole data set *if the sample is biased*

Chapter 7. Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Outlier Analysis
8. Summary

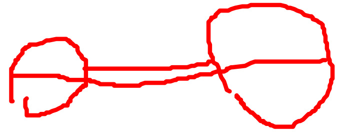
Hierarchical Clustering

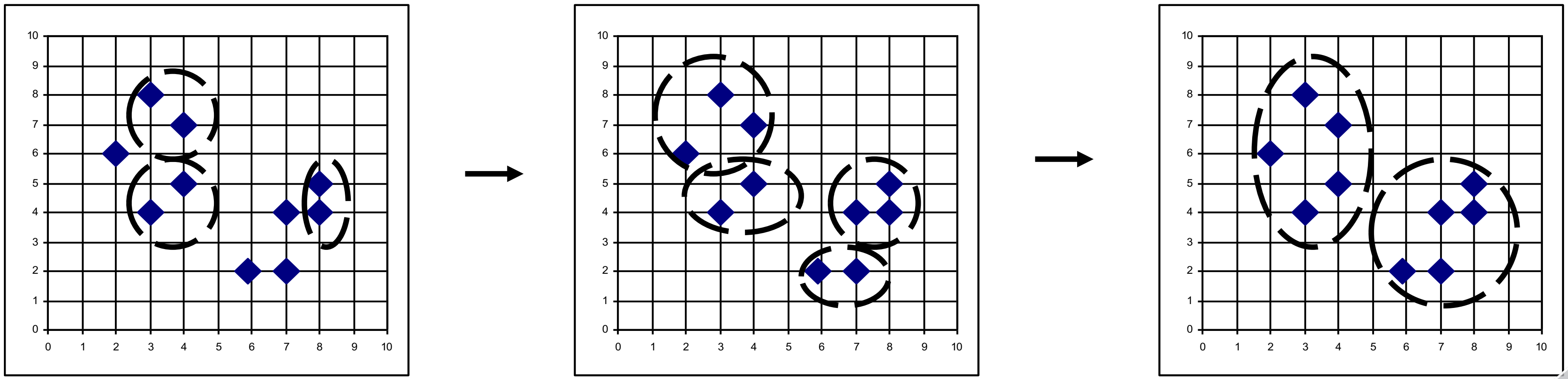
- Use a distance matrix as clustering criteria
- Does not require the number of clusters ***k*** as an input, but needs a termination condition



AGNES (Agglomerative Nesting)

- Introduced in Kaufmann and Rousseeuw (1990)
  - Implemented in statistical analysis packages, Splus
- Use the single-link method and the dissimilarity matrix
- Merge nodes that have the least dissimilarity
- Go on in a non-descending fashion
- Eventually all nodes belong to the same cluster

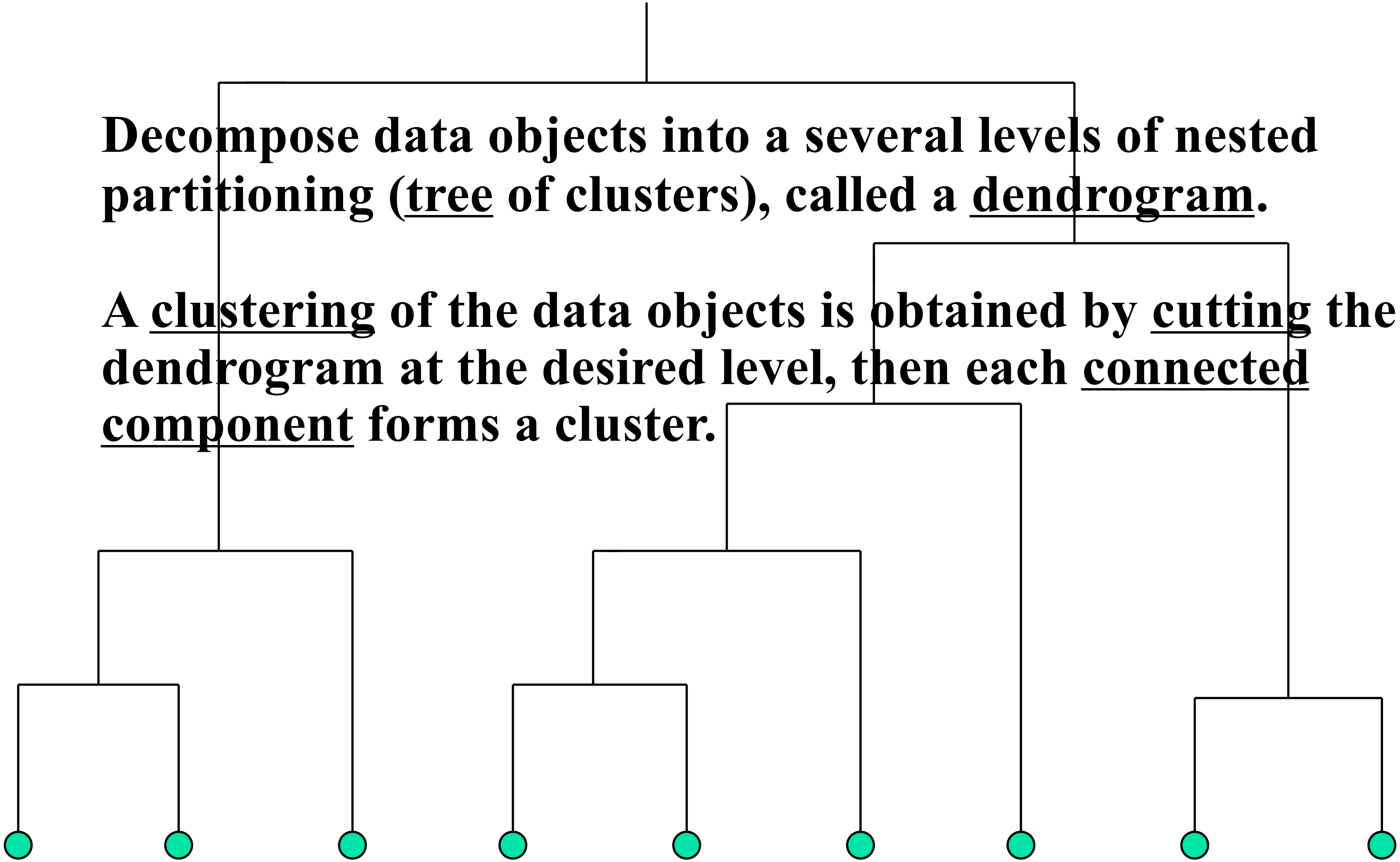




Dendrogram: How the Clusters are Merged

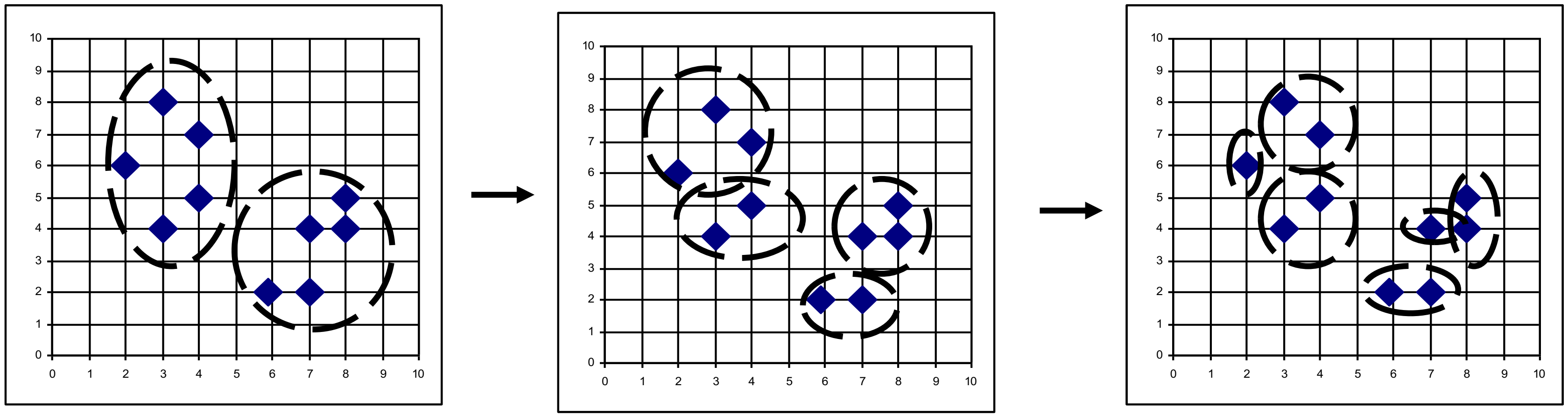
Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram.

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)
- Implemented in statistical analysis packages, Splus
- Inverse order of AGNES
- Eventually each node forms a cluster on its own

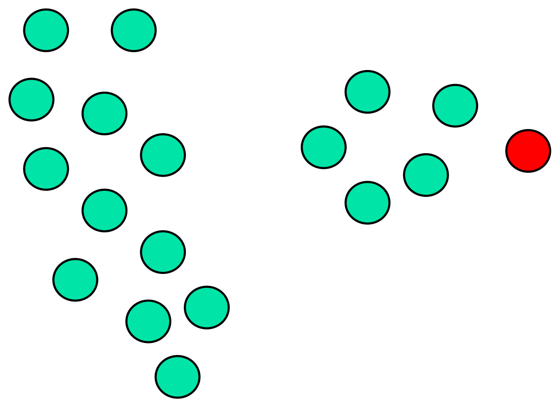


DIANA (Divisive Analysis)

- Outline
  - Initially, there is one large cluster consisting of all *n* objects
  - At each subsequent step, the largest available cluster is split into two clusters
    - Until finally all clusters comprise of a single object.
    - Thus, the hierarchy is built in *n*-1 steps.
- Complexity in the first step
  - Agglomerative method: $\frac{n(n-1)}{2}$ possible combinations
  - Divisive method: $2^{n-1}-1$ possible combinations
    - Considerably larger than an agglomerative method

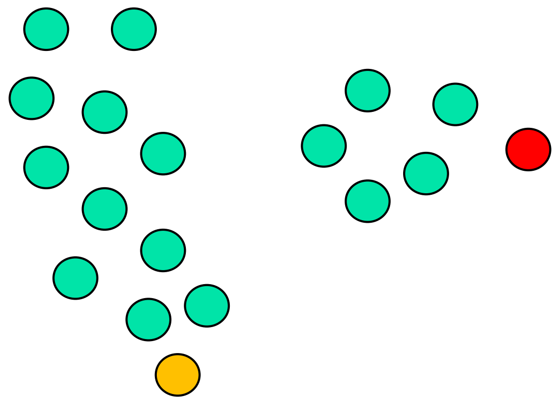
DIANA (Divisive Analysis)

- To avoid considering all possibilities, the algorithm proceeds as follows.
  1. Find the object, which has the highest average dissimilarity to all other objects. This object initiates a new cluster– a sort of a *splinter group*.
  2. For each object *i* outside the *splinter group*, compute
     $D_i = \left[average\ d(i,j)\ j \notin R_{splinter\ group}\right] - \left[average\ d(i,j)\ j \in R_{splinter\ group}\right]$
  3. Find an object *h* for which the difference D_h is the largest. If D_h is positive, then *h* is, on the average close to the splinter group. Put h into the splinter group.



DIANA (Divisive Analysis)

- To avoid considering all possibilities, the algorithm proceeds as follows.
  1. Repeat *Steps* 2 and 3 until all differences D_h are negative. The data set is then split into two clusters.
  2. Select the cluster with the largest diameter. The diameter of a cluster is the largest dissimilarity between any two of its objects. Then divide this cluster, following steps 1-4.
  3. Repeat *Step* 5 until all clusters contain only a single object.



Advacned Hierarchical Clustering Methods

- Major weakness of agglomerative clustering methods
  - do not scale well: time complexity of at least $O(n^2)$, where n is the number of total objects
- Integration of hierarchical with distance-based clustering
  - BIRCH (1996): uses CF-tree and incrementally adjusts the quality of sub-clusters
  - ROCK (1999): clustering categorical data by neighbor and link analysis
  - CHAMELEON (1999): hierarchical clustering using dynamic modeling

BIRCH (1996)

- Birch: Balanced Iterative Reducing and Clustering using Hierarchies (Zhang, Ramakrishnan & Livny, SIGMOD'96)
- Incrementally construct a CF (Clustering Feature) tree (cf. B-tree), a hierarchical data structure for multiphase clustering
  - Phase 1: scan DB to build an initial in-memory CF tree (a multi-level compression of the data that tries to preserve the inherent clustering structure of the data)
  - Phase 2: use an arbitrary clustering algorithm to cluster the leaf nodes of the CF-tree
- *Scales linearly*: finds a good clustering with a single scan and improves the quality with a few additional scans
- *Weakness:* handles only numeric data, and sensitive to the order of the data records

Clustering Feature Vector in BIRCH

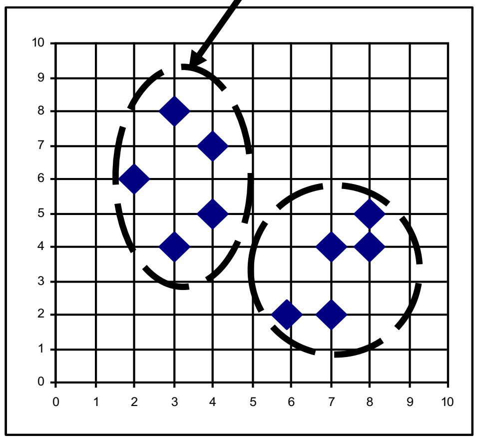
Clustering Feature: $CF = (N, \overrightarrow{LS}, SS)$

N: **Number of data points**

LS: $\sum_{i=1}^{N}=\overrightarrow{X_i}$

SS: $\sum_{i=1}^{N}=\overrightarrow{X_i}^2$

CF = (5, (16,30),(54,190))



(3,4)
(2,6)
(4,5)
(4,7)
(3,8)

CF-Tree in BIRCH

- Clustering feature:
  - Summary of the statistics for a given cluster: the 0-th, 1st and 2nd moments of the cluster from the statistical point of view
  - Registers crucial measurements for computing cluster and utilizes storage efficiently
- A CF tree is a height-balanced tree that stores the clustering features for a hierarchical clustering
  - A non-leaf node in a tree has descendants or "children"
  - A non-leaf node stores the sum of the CFs of their children
- A CF tree has two parameters
  - Branching factor: specify the maximum number of children
  - threshold: max diameter of a cluster stored at the leaf node

The CF Tree Structure