

# Data Mining:

---

# Concepts and Techniques

## — Chapter 5 —

Jiawei Han

Department of Computer Science


University of Illinois at Urbana-Champaign

[www.cs.uiuc.edu/~hanj](http://www.cs.uiuc.edu/~hanj)

©2006 Jiawei Han and Micheline Kamber, All rights reserved

# Chapter 5. 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. Summary

# What is Cluster Analysis?

---

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- **Unsupervised learning**: no predefined classes
- Typical applications
  - As a **stand-alone tool** to get insight into data distribution
  - As a **preprocessing step** for other algorithms

# Clustering: Rich Applications and Multidisciplinary Efforts

---

- Pattern Recognition
- Spatial Data Analysis
  - Create thematic maps in GIS by clustering feature spaces
  - Detect spatial clusters or for other spatial mining tasks
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

# Quality: What Is Good Clustering?

---

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
- The quality of a clustering result 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

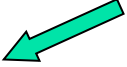
# Measure the Quality of Clustering

---

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

# Chapter 5. 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. Summary





# Type of data in clustering analysis

---

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

# Interval-valued variables

---

- Standardize data

- Calculate the mean absolute deviation:

$$s_f = \frac{1}{n} (|x_{1f} - m_f| + |x_{2f} - m_f| + \dots + |x_{nf} - m_f|)$$

where  $m_f = \frac{1}{n}(x_{1f} + x_{2f} + \dots + x_{nf})$ .

- Calculate the standardized measurement (*z-score*)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

- Using mean absolute deviation is more robust than using standard deviation

# Similarity and Dissimilarity Between Objects

- Distances are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include: *Minkowski distance*:

$$d(i, j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \dots + |x_{ip} - x_{jp}|^q)}$$

where  $i = (x_{i1}, x_{i2}, \dots, x_{ip})$  and  $j = (x_{j1}, x_{j2}, \dots, x_{jp})$  are two  $p$ -dimensional data objects, and  $q$  is a positive integer

- If  $q = 1$ ,  $d$  is Manhattan distance

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|$$

# Similarity and Dissimilarity Between Objects (Cont.)

---

- If  $q = 2$ ,  $d$  is Euclidean distance:

$$d(i, j) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + \dots + |x_{i_p} - x_{j_p}|^2)}$$

- Properties

- $d(i, j) \geq 0$
  - $d(i, i) = 0$
  - $d(i, j) = d(j, i)$
  - $d(i, j) \leq d(i, k) + d(k, j)$
- Also, one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

# Binary Variables

- A contingency table for binary data

		Object $j$		
		1	0	<i>sum</i>
Object $i$	1	$a$	$b$	$a+b$
	0	$c$	$d$	$c+d$
<i>sum</i>		$a+c$	$b+d$	$p$

- Distance measure for symmetric binary variables:

$$d(i, j) = \frac{b+c}{a+b+c+d}$$

- Distance measure for asymmetric binary variables:

$$d(i, j) = \frac{b+c}{a+b+c}$$

- Jaccard coefficient (*similarity* measure for *asymmetric* binary variables):

$$sim_{Jaccard}(i, j) = \frac{a}{a+b+c}$$

# Dissimilarity between Binary Variables

- Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- gender is a symmetric attribute
- the remaining attributes are asymmetric binary
- let the values Y and P be set to 1, and the value N be set to 0

$$d(\mathit{jack}, \mathit{mary}) = \frac{\mathbf{0} + \mathbf{1}}{\mathbf{2} + \mathbf{0} + \mathbf{1}} = \mathbf{0.33}$$

$$d(\mathit{jack}, \mathit{jim}) = \frac{\mathbf{1} + \mathbf{1}}{\mathbf{1} + \mathbf{1} + \mathbf{1}} = \mathbf{0.67}$$

$$d(\mathit{jim}, \mathit{mary}) = \frac{\mathbf{1} + \mathbf{2}}{\mathbf{1} + \mathbf{1} + \mathbf{2}} = \mathbf{0.75}$$

# Nominal Variables

---

- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
  - $m$ : # of matches,  $p$ : total # of variables

$$d(i, j) = \frac{p - m}{p}$$

- Method 2: use a large number of binary variables
  - creating a new binary variable for each of the  $M$  nominal states

# Ordinal Variables

---

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled
  - replace  $x_{if}$  by their rank  $r_{if} \in \{1, \dots, M_f\}$
  - map the range of each variable onto  $[0, 1]$  by replacing  $i$ -th object in the  $f$ -th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

- compute the dissimilarity using methods for interval-scaled variables



# Ratio-Scaled Variables

---

- Ratio-scaled variable: a positive measurement on a nonlinear scale, approximately at exponential scale, such as  $Ae^{Bt}$  or  $Ae^{-Bt}$
- Methods:
  - treat them like interval-scaled variables—*not a good choice!* (why?—the scale can be distorted)
  - apply logarithmic transformation
$$y_{if} = \log(x_{if})$$
  - treat them as continuous ordinal data treat their rank as interval-scaled

# Variables of Mixed Types

- A database may contain all the six types of variables
  - symmetric binary, asymmetric binary, nominal, ordinal, interval and ratio

- One may use a weighted formula to combine their effects

$$d(i, j) = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}}$$

- $f$  is binary or nominal:

$d_{ij}^{(f)} = 0$  if  $x_{if} = x_{jf}$ , or  $d_{ij}^{(f)} = 1$  otherwise

- $f$  is interval-based: use the normalized distance
- $f$  is ordinal or ratio-scaled

- compute ranks  $r_{if}$  and

- and treat  $z_{if}$  as interval-scaled

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

# Vector Objects

- Vector objects: keywords in documents, gene features in micro-arrays, etc.
- Broad applications: information retrieval, biologic taxonomy, etc.

- Cosine measure 
$$s(\vec{X}, \vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{|\vec{X}| |\vec{Y}|},$$

$\vec{X}^t$  is a transposition of vector  $\vec{X}$ ,  $|\vec{X}|$  is the Euclidean normal of vector  $\vec{X}$ ,

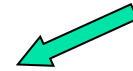
- A variant: Tanimoto coefficient

$$s(\vec{X}, \vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{\vec{X}^t \cdot \vec{X} + \vec{Y}^t \cdot \vec{Y} - \vec{X}^t \cdot \vec{Y}},$$

# Chapter 5. 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. Summary



# Major Clustering Approaches

---

- Partitioning approach:
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: **k-means**, **k-medoids**, **CLARANS**
- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: **Diana**, **Agnes**
- Density-based approach:
  - Based on connectivity and density functions
  - Typical methods: **DBSCAN**

# Typical Alternatives to Calculate the Distance between Clusters

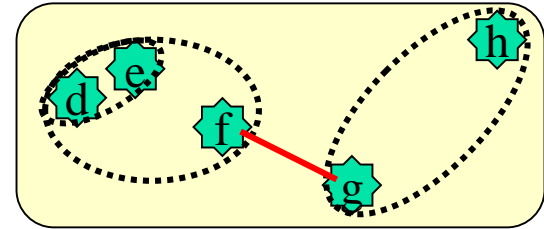
---

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

# Distance between clusters

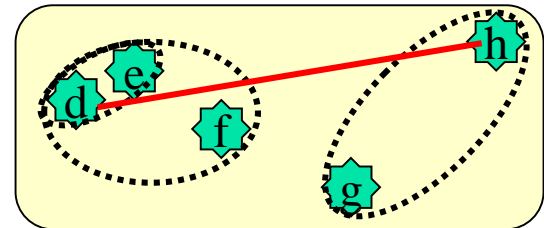
- $CD(X,Y)=\min_{x \in X, y \in Y} D(x,y)$

*Single-link method*



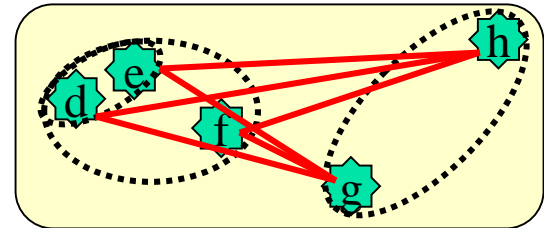
- $CD(X,Y)=\max_{x \in X, y \in Y} D(x,y)$

*Complete-link method*



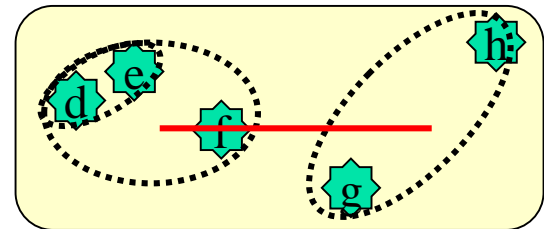
- $CD(X,Y)=\text{avg}_{x \in X, y \in Y} D(x,y)$

*Average-link method*



- $CD(X,Y)=D(\text{avg}(X), \text{avg}(Y))$

*Centroid method*



# 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

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

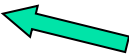
- Diameter: square root of average mean squared distance between all pairs of points in the cluster

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



# Chapter 5. 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. Summary

# Partitioning Algorithms: Basic Concept

- Partitioning method: Construct a partition of a database  $D$  of  $n$  objects into a set of  $k$  clusters, s.t., min sum of squared distance

$$\sum_{m=1}^k \sum_{t_{mi} \in K_m} (C_m - t_{mi})^2$$

- 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

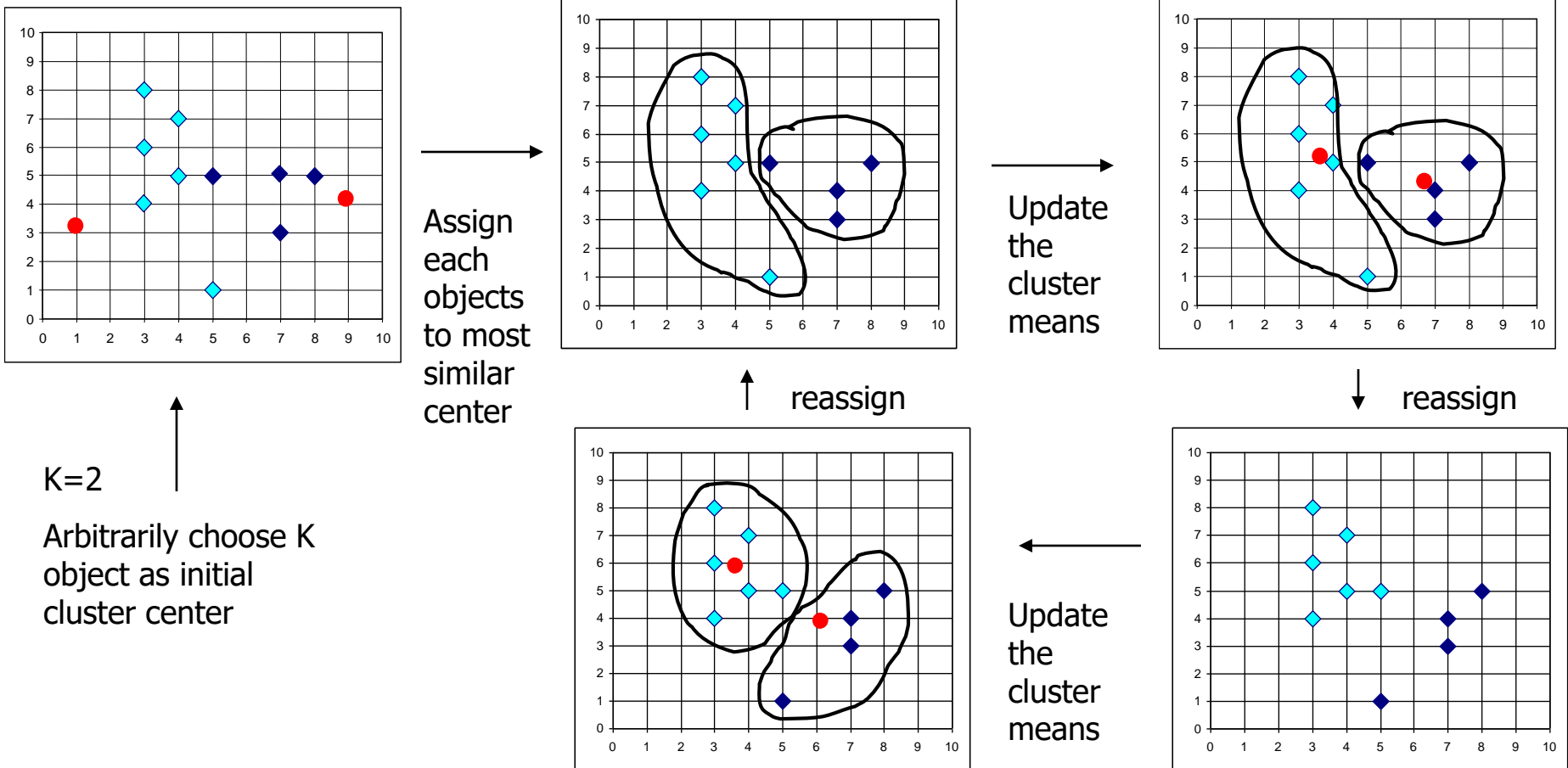
# The *K-Means* Clustering Method

---

- 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 partition (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 no more new assignment

# The *K-Means* Clustering Method

## ■ Example



# Comments on the *K-Means* Method

---

- Strength: *Relatively efficient*:  $O(tkn)$ , where  $n$  is # objects,  $k$  is # clusters, and  $t$  is # iterations. Normally,  $k, t \ll n$ .
  - Comparing: PAM:  $O(k(n-k)^2)$ , CLARA:  $O(ks^2 + k(n-k))$
- Comment: Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- Weakness
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify  $k$ , the *number* of clusters, in advance
  - Unable to handle noisy data and *outliers*
  - Not suitable to discover clusters with *non-convex shapes*

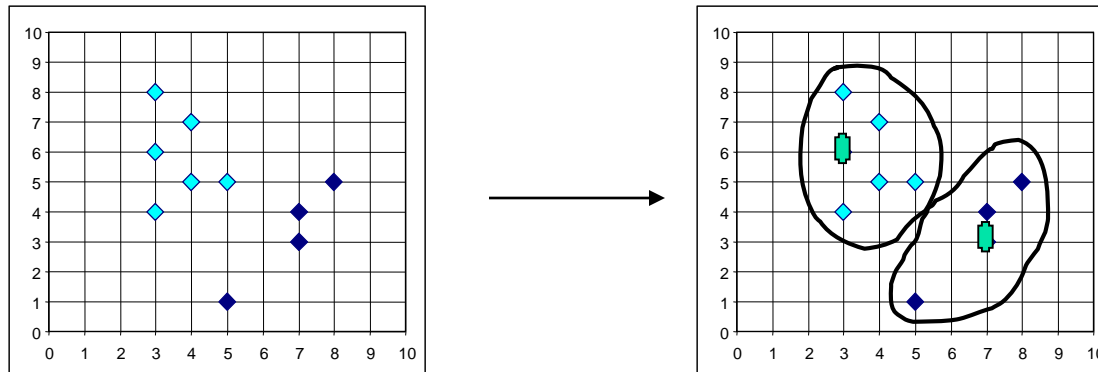
# Variations of the *K-Means* Method

---

- A few 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* (Huang'98)
  - 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

# What Is the 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.



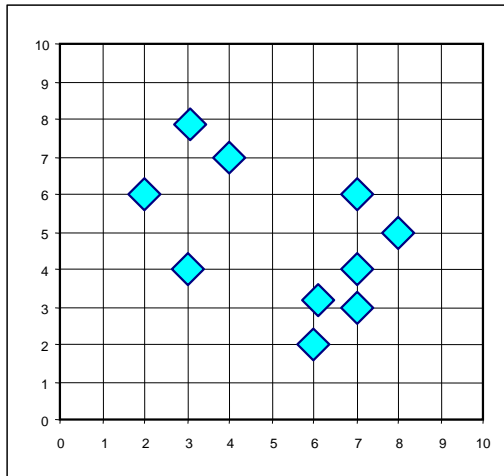
# The *K-Medoids* Clustering Method

---

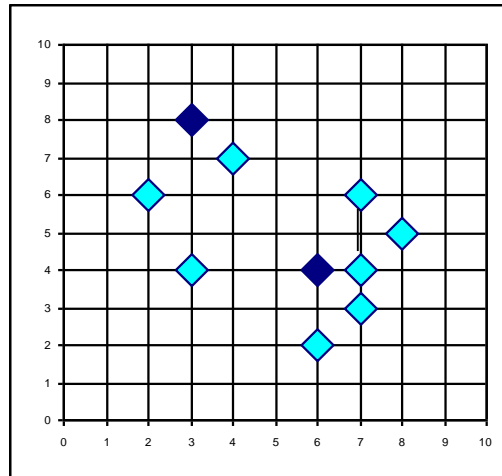
- Find *representative* objects, called medoids, in clusters
- *PAM* (Partitioning Around Medoids, 1987)
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
  - *PAM* works effectively 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)



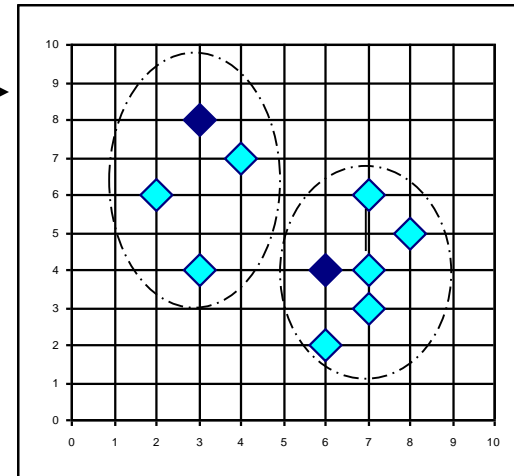
# A Typical K-Medoids Algorithm (PAM)



Arbitrary  
choose  $k$   
object as  
initial  
medoids



Assign  
each  
remainin  
g object  
to  
nearest  
medoids



Total Cost = 20

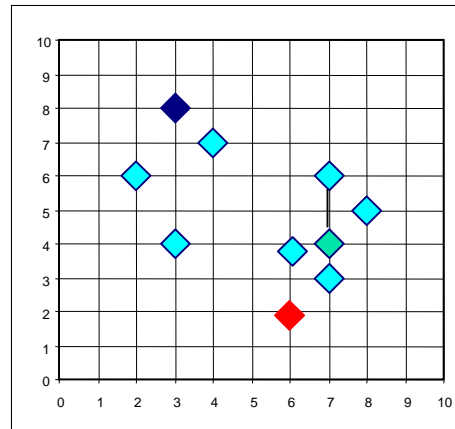
$K=2$

Total Cost = 26

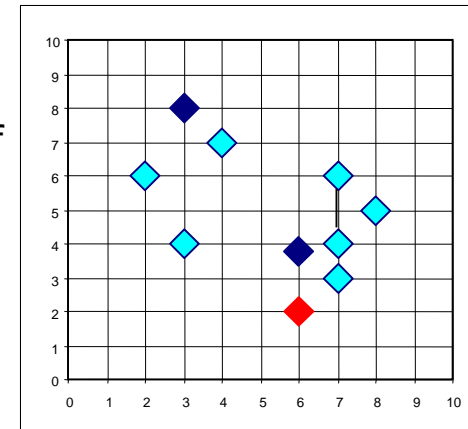
Randomly select a  
nonmedoid object,  $O_{\text{random}}$

**Do loop**  
**Until no**  
**change**

Swapping  $O$   
and  $O_{\text{random}}$   
If quality is  
improved.



Compute  
total cost of  
swapping

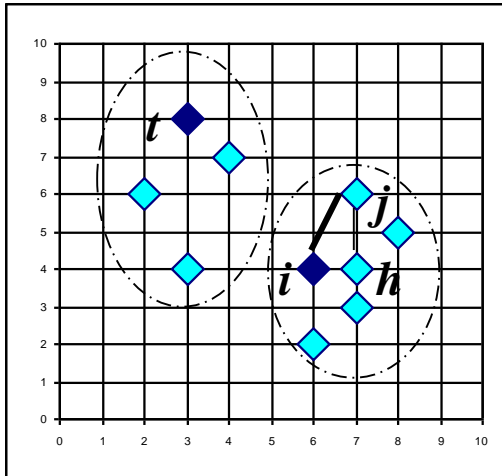


# PAM (Partitioning Around Medoids) (1987)

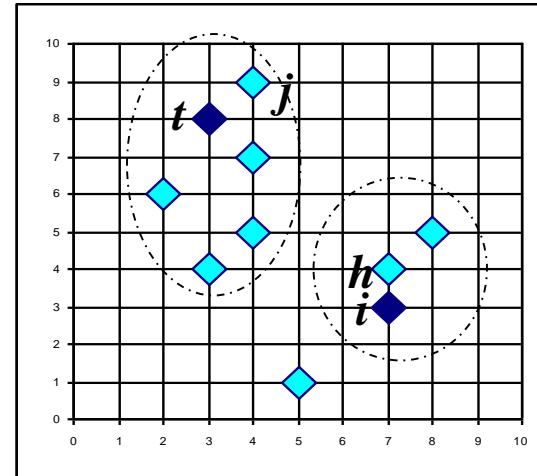
---

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

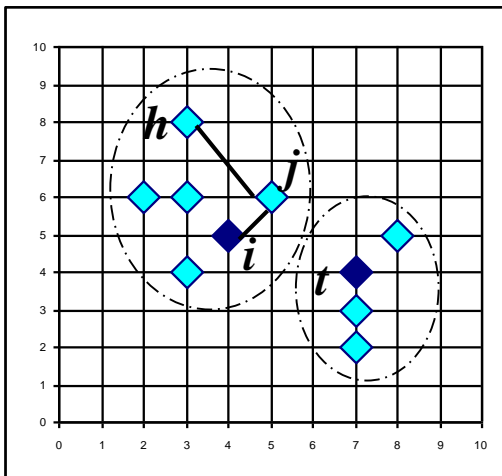
# PAM Clustering: Total swapping cost $TC_{ih} = \sum_j C_{jih}$



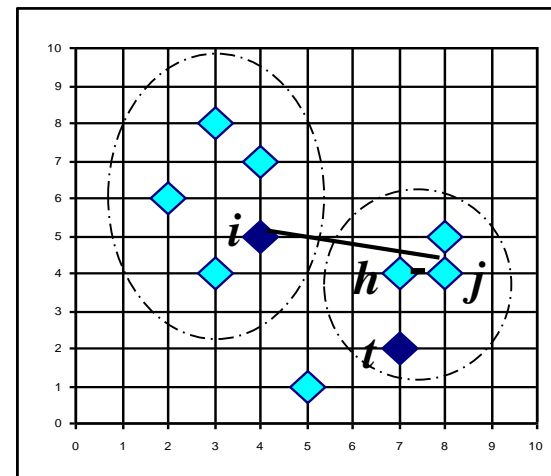
$$C_{jih} = d(j, h) - d(j, i)$$



$$C_{jih} = 0$$



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



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

# 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
- Pam works efficiently for small data sets but does not **scale well** for large data sets.
  - $O(k(n-k)^2)$  for each iteration

where  $n$  is # of data,  $k$  is # of clusters

→ 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


# *CLARANS* ("Randomized" CLARA) (1994)

---

- *CLARANS* (A Clustering Algorithm based on Randomized Search) (Ng and Han'94)
- *CLARANS* draws sample of neighbors dynamically
- The clustering process can be presented as searching a graph where every node is a potential solution, that is, a set of  $k$  medoids
- If the local optimum is found, *CLARANS* starts with new randomly selected node in search for a new local optimum
- It is more efficient and scalable than both *PAM* and *CLARA*
- Focusing techniques and spatial access structures may further improve its performance (Ester et al.'95)

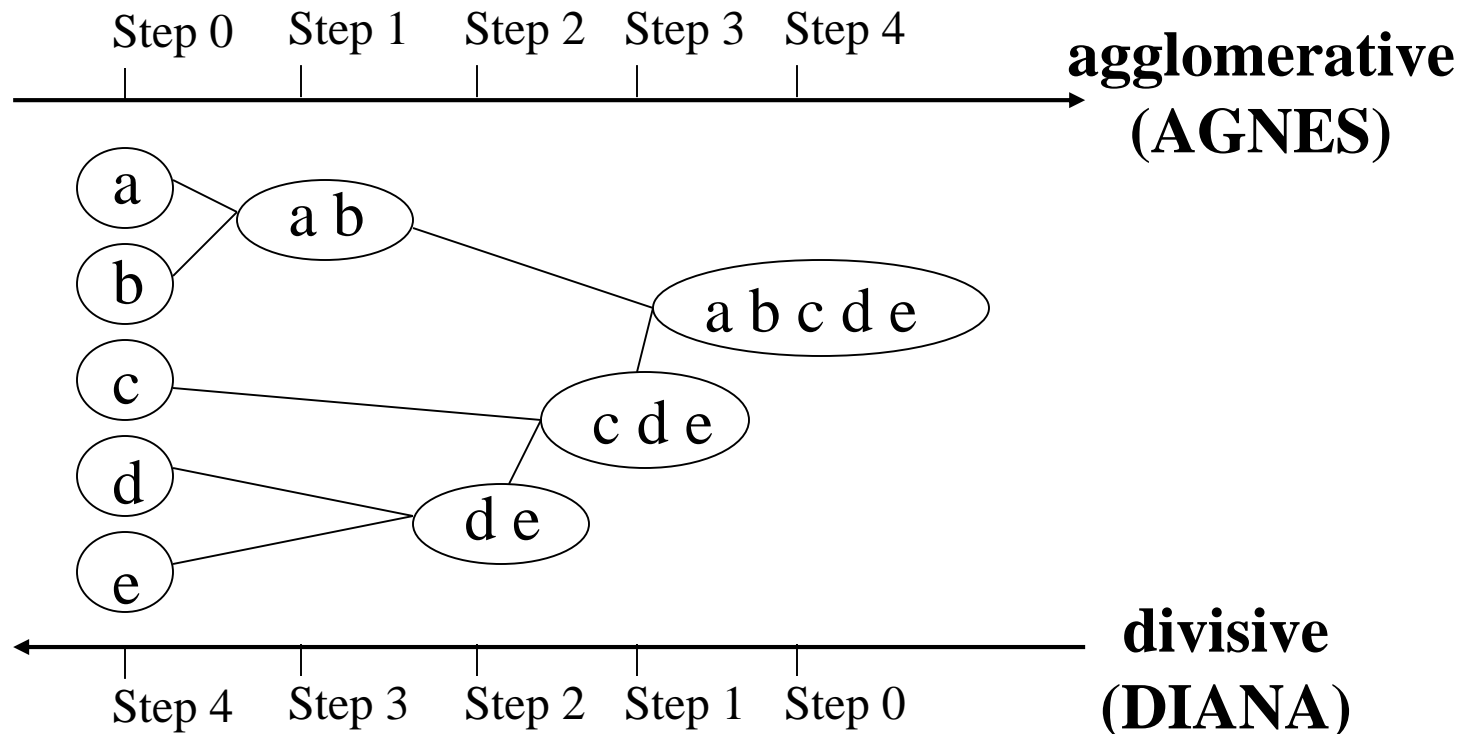
# Chapter 5. Cluster Analysis

---

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. Summary

# Hierarchical Clustering

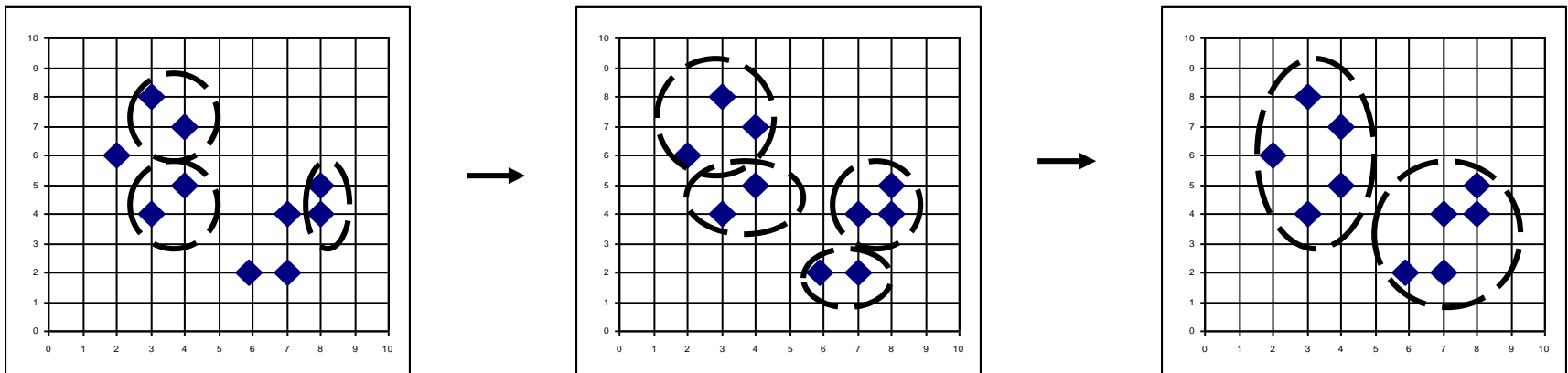
- Use distance matrix as clustering criteria. This method does not require the number of clusters  $k$  as an input, but needs a termination condition





# AGNES (Agglomerative Nesting)

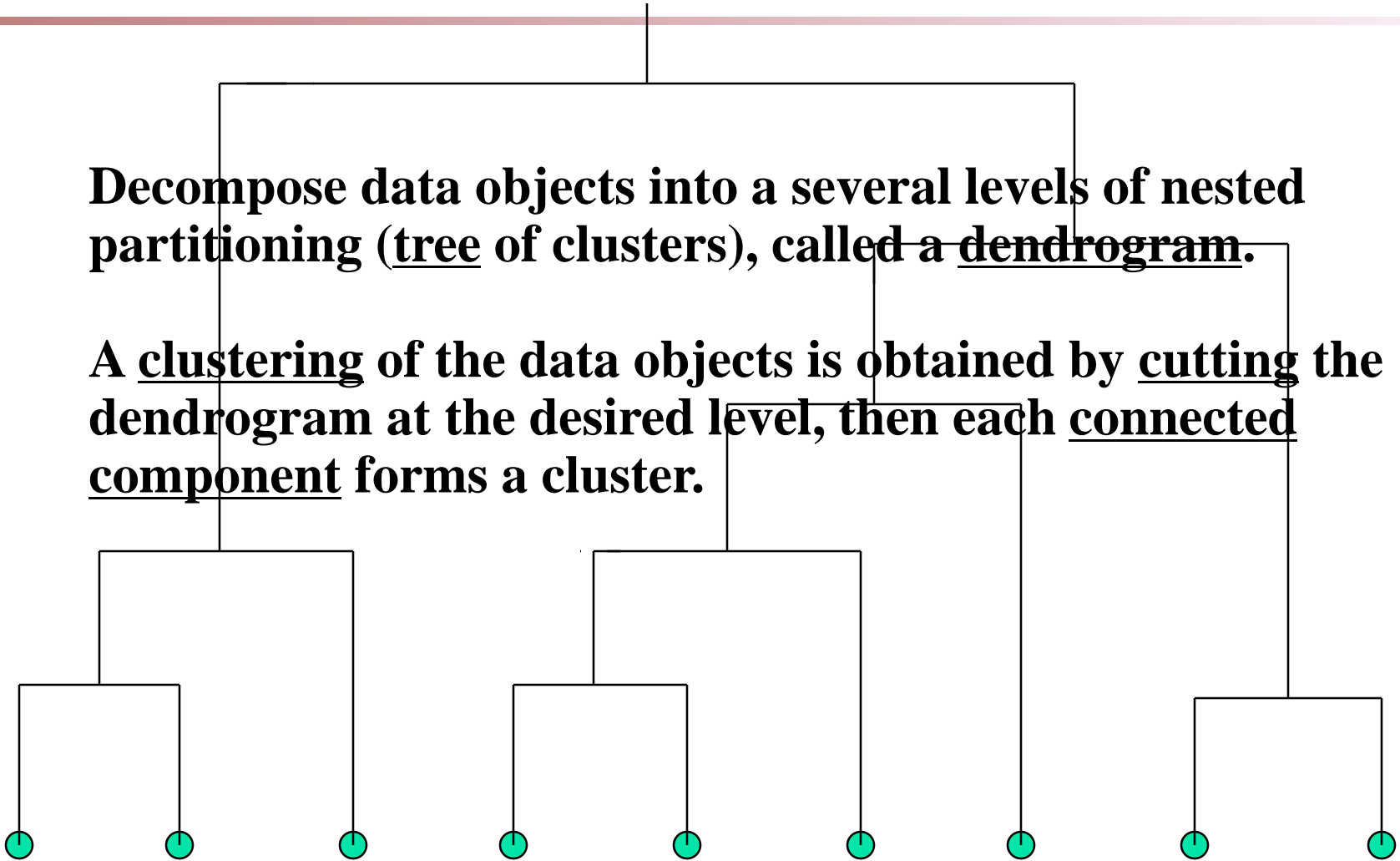
- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical analysis packages, e.g., Splus
- Use the Single-Link method and the dissimilarity matrix.
- Merge nodes that have the least dissimilarity
- Go on in a non-descending fashion
- Eventually all nodes belong to the same cluster



# *Dendrogram: Shows How the Clusters are Merged*

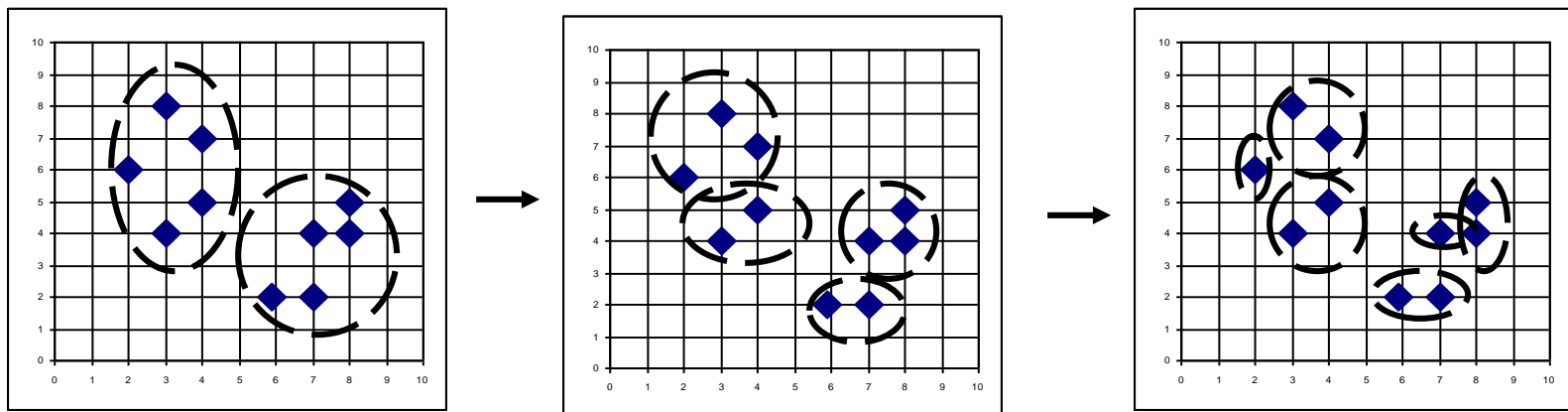
**Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram.**

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



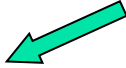
# DIANA (Divisive Analysis)

- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical analysis packages, e.g., Splus
- Inverse order of AGNES
- Eventually each node forms a cluster on its own



# Chapter 5. 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. Summary

# Density-Based Clustering Methods

---

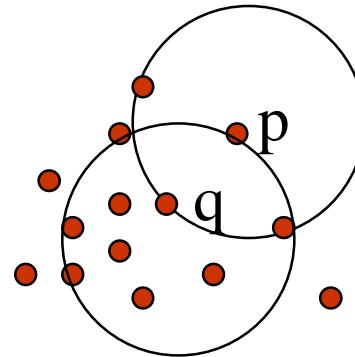
- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan
  - Need density parameters as termination condition
- Several interesting studies:
  - DBSCAN: Ester, et al. (KDD'96)
  - OPTICS: Ankerst, et al (SIGMOD'99).
  - DENCLUE: Hinneburg & D. Keim (KDD'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98) (more grid-based)

# Density-Based Clustering: Basic Concepts

- Two parameters:
  - *Eps*: Maximum radius of the neighbourhood
  - *MinPts*: Minimum number of points in an Eps-neighbourhood of that point
- $N_{Eps}(p)$ :  $\{q \text{ belongs to } D \mid \text{dist}(p,q) \leq Eps\}$
- **Directly density-reachable**: A point  $p$  is directly density-reachable from a point  $q$  w.r.t.  $Eps$ ,  $MinPts$  if

- $p$  belongs to  $N_{Eps}(q)$
- core point condition:

$$|N_{Eps}(q)| \geq MinPts$$



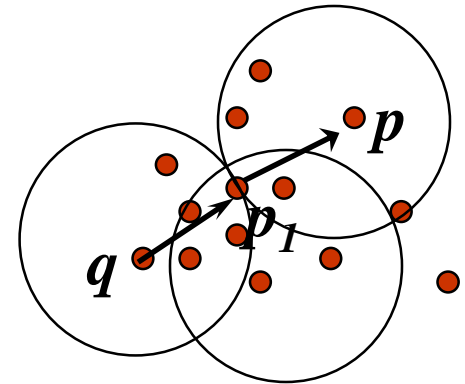
MinPts = 5

Eps = 1 cm

# Density-Reachable and Density-Connected

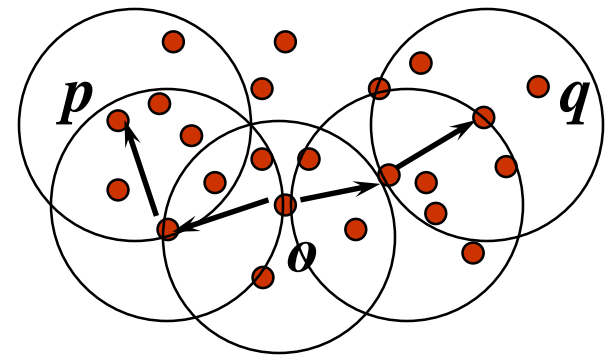
- Density-reachable:

- A point  $p$  is **density-reachable** from a point  $q$  w.r.t.  $Eps, MinPts$  if there is a chain of points  $p_1, \dots, p_n, p_1 = q, p_n = p$  such that  $p_{i+1}$  is directly density-reachable from  $p_i$



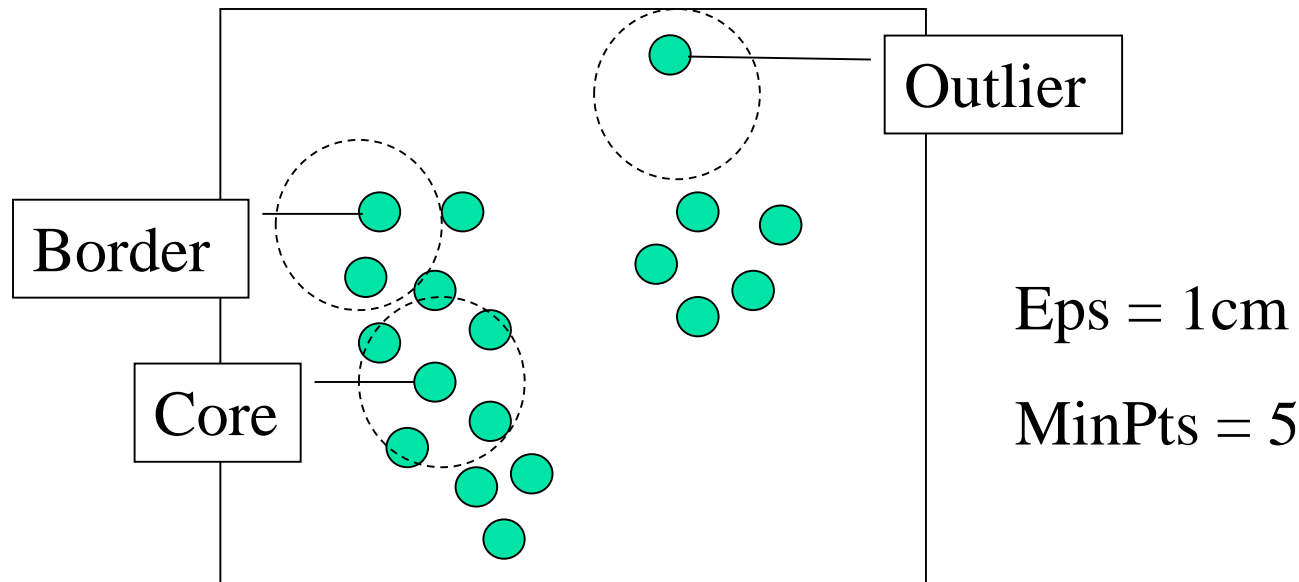
- Density-connected

- A point  $p$  is **density-connected** to a point  $q$  w.r.t.  $Eps, MinPts$  if there is a point  $o$  such that both,  $p$  and  $q$  are density-reachable from  $o$  w.r.t.  $Eps$  and  $MinPts$



# DBSCAN: Density Based Spatial Clustering of Applications with Noise

- Relies on a *density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
- Discovers clusters of arbitrary shape in spatial databases with noise





# DBSCAN: The Algorithm

---

- Arbitrary select a point  $p$
- Retrieve all points density-reachable from  $p$  w.r.t.  $Eps$  and  $MinPts$ .
- If  $p$  is a core point, a cluster is formed.
- If  $p$  is a border point, no points are density-reachable from  $p$  and DBSCAN visits the next point of the database.
- Continue the process until all of the points have been processed.

# DBSCAN: Sensitive to Parameters

Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.

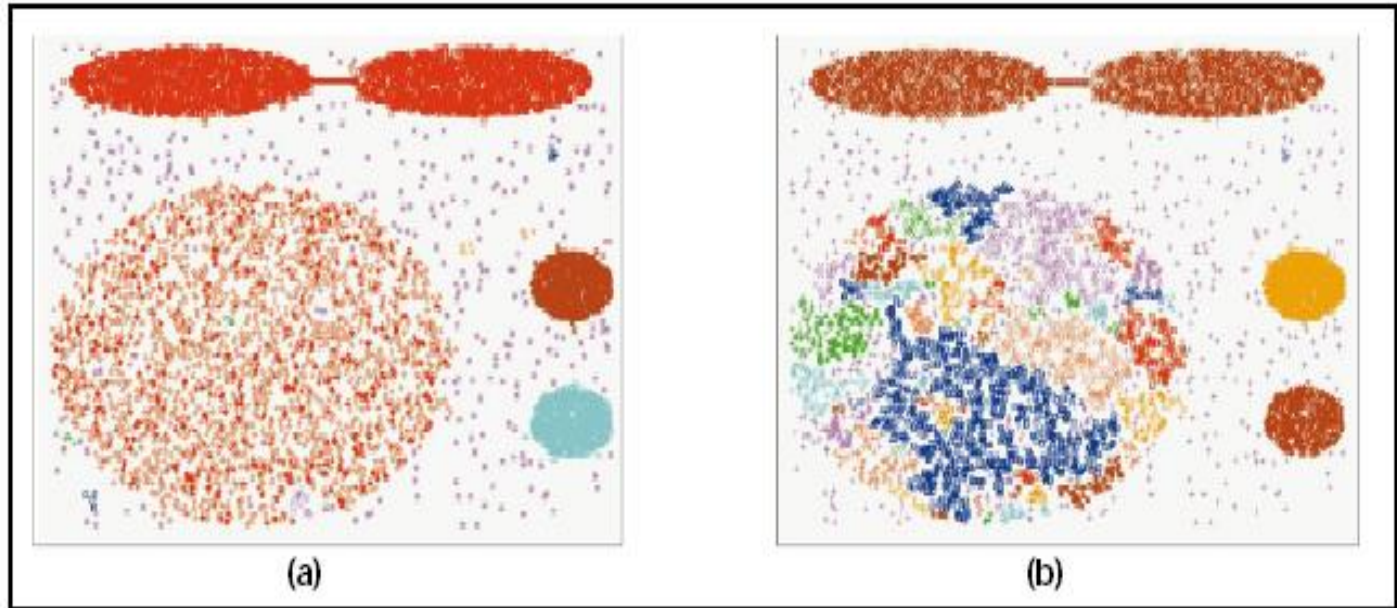
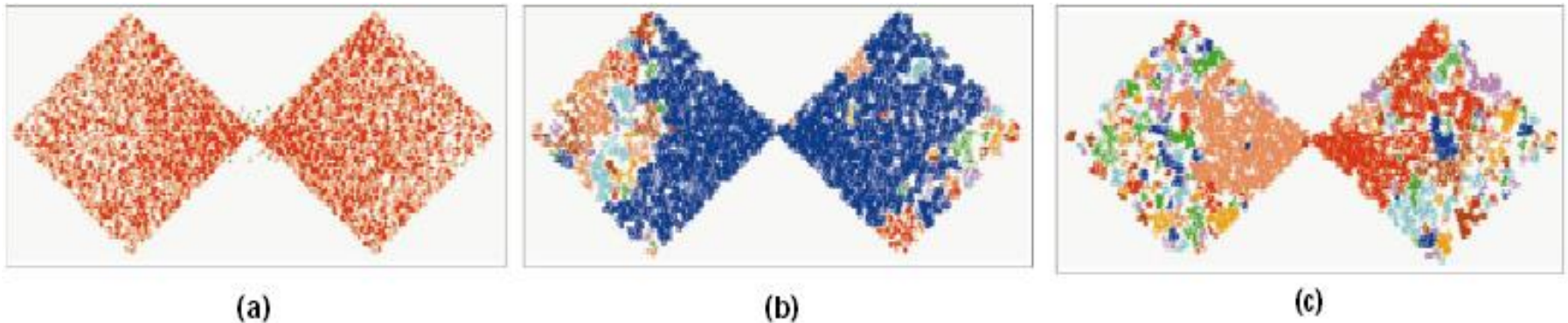



Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.



# Chapter 5. 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. Summary 

# Summary

---

- **Cluster analysis** groups objects based on their **similarity** and has wide applications
- Measure of similarity can be computed for **various types of data**
- Clustering algorithms can be **categorized** into partitioning methods, hierarchical methods, and density-based methods
- There are still lots of research issues on cluster analysis

# References (1)

---

- R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. SIGMOD'98
- M. R. Anderberg. Cluster Analysis for Applications. Academic Press, 1973.
- M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure, SIGMOD'99.
- P. Arabie, L. J. Hubert, and G. De Soete. Clustering and Classification. World Scientific, 1996
- Beil F., Ester M., Xu X.: "[Frequent Term-Based Text Clustering](#)", KDD'02
- M. M. Breunig, H.-P. Kriegel, R. Ng, J. Sander. LOF: Identifying Density-Based Local Outliers. SIGMOD 2000.
- M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases. KDD'96.
- M. Ester, H.-P. Kriegel, and X. Xu. Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification. SSD'95.
- D. Fisher. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2:139-172, 1987.
- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. VLDB'98.

# References (2)

- V. Ganti, J. Gehrke, R. Ramakrishan. CACTUS Clustering Categorical Data Using Summaries. *KDD'99*.
- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. In Proc. VLDB'98.
- S. Guha, R. Rastogi, and K. Shim. Cure: An efficient clustering algorithm for large databases. SIGMOD'98.
- S. Guha, R. Rastogi, and K. Shim. [ROCK: A robust clustering algorithm for categorical attributes](#). In *ICDE'99*, pp. 512-521, Sydney, Australia, March 1999.
- A. Hinneburg, D.I A. Keim: An Efficient Approach to Clustering in Large Multimedia Databases with Noise. KDD'98.
- A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Printice Hall, 1988.
- G. Karypis, E.-H. Han, and V. Kumar. [CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling](#). *COMPUTER*, 32(8): 68-75, 1999.
- L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.
- E. Knorr and R. Ng. Algorithms for mining distance-based outliers in large datasets. VLDB'98.
- G. J. McLachlan and K.E. Bkassford. Mixture Models: Inference and Applications to Clustering. John Wiley and Sons, 1988.
- P. Michaud. Clustering techniques. Future Generation Computer systems, 13, 1997.
- R. Ng and J. Han. Efficient and effective clustering method for spatial data mining. VLDB'94.

# References (3)

---

- *L. Parsons, E. Haque and H. Liu, [Subspace Clustering for High Dimensional Data: A Review](#), SIGKDD Explorations, 6(1), June 2004*
- E. Schikuta. Grid clustering: An efficient hierarchical clustering method for very large data sets. Proc. 1996 Int. Conf. on Pattern Recognition,.
- G. Sheikholeslami, S. Chatterjee, and A. Zhang. WaveCluster: A multi-resolution clustering approach for very large spatial databases. VLDB'98.
- A. K. H. Tung, J. Han, L. V. S. Lakshmanan, and R. T. Ng. [Constraint-Based Clustering in Large Databases](#), *ICDT'01*.
- A. K. H. Tung, J. Hou, and J. Han. [Spatial Clustering in the Presence of Obstacles](#), *ICDE'01*
- H. Wang, W. Wang, J. Yang, and P.S. Yu. [Clustering by pattern similarity in large data sets](#), *SIGMOD'02*.
- W. Wang, Yang, R. Muntz, STING: A Statistical Information grid Approach to Spatial Data Mining, VLDB'97.
- T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH : an efficient data clustering method for very large databases. SIGMOD'96.