Clustering: Basic Methods

Lecturer: Dr. Nguyen Ngoc Thao
Department of Computer Science, FIT, HCMUS

Slides adapted from Jiawei Han, Micheline Kamber and Jian Pei

Outline

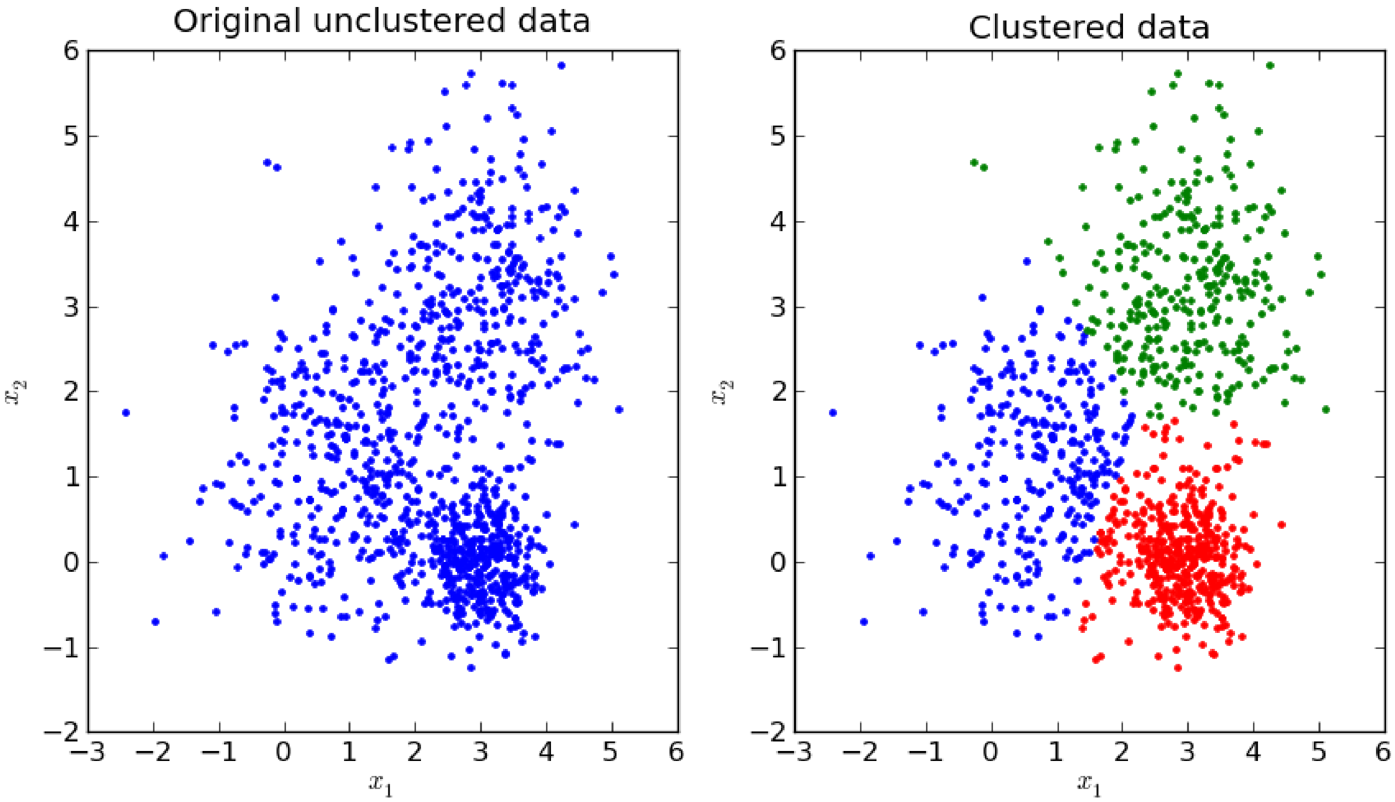
- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Evaluation of Clustering
- Summary

Outline

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

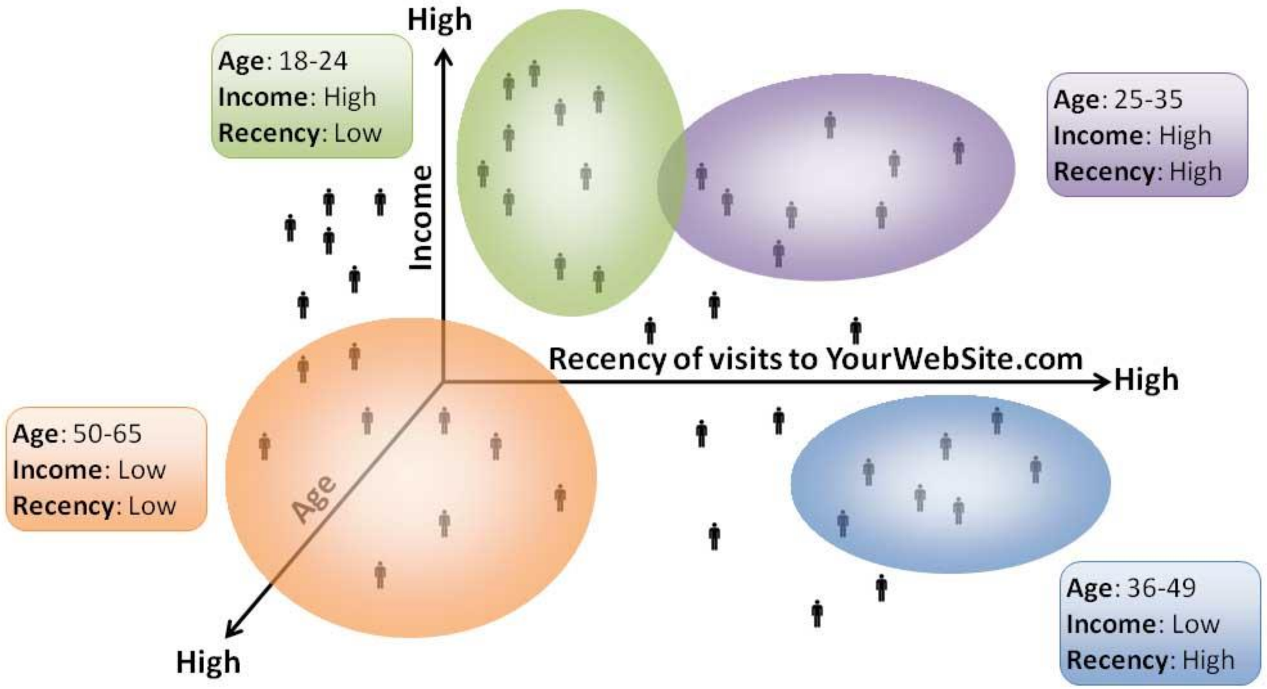
What is Cluster Analysis?

- 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



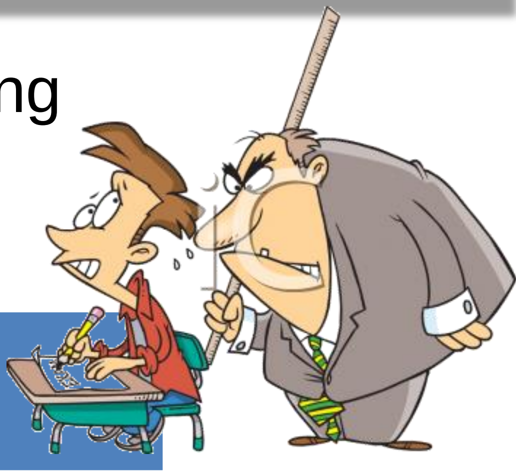
What is Cluster Analysis?

- Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Other terms: clustering, data segmentation, …



What is Cluster Analysis?

- Unsupervised learning vs. supervised learning



Unsupervised learning	Supervised learning
Also called Clustering	Also called Classification
No predefined classes	Predefined classes
Learning by observations	Learning by examples

- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Clustering for Data Understanding and 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: 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
- Economic Science: market resarch

Clustering as a Preprocessing Tool (Utility)

- 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

Quality: 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

Measure the Quality of Clustering

- 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

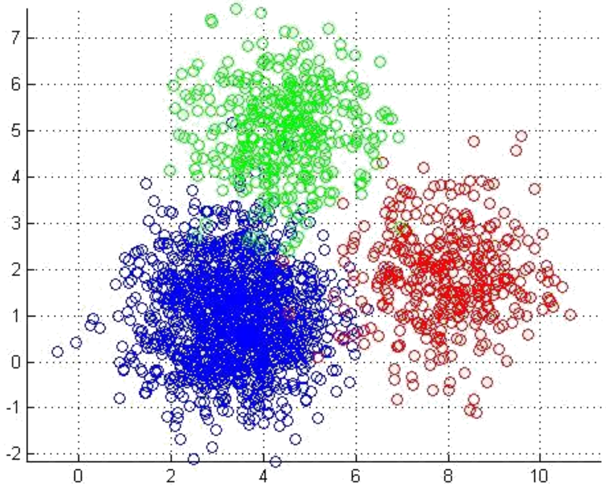
- Partitioning criteria
  - Single level vs. hierarchical partitioning (often, multi-level hierarchical partitioning is desirable)
- Separation of clusters
  - Exclusive (e.g., one customer belongs to only one region) vs. non-exclusive (e.g., one document may belong to more than one class)
- Similarity measure
  - Distance-based (e.g., Euclidean, 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

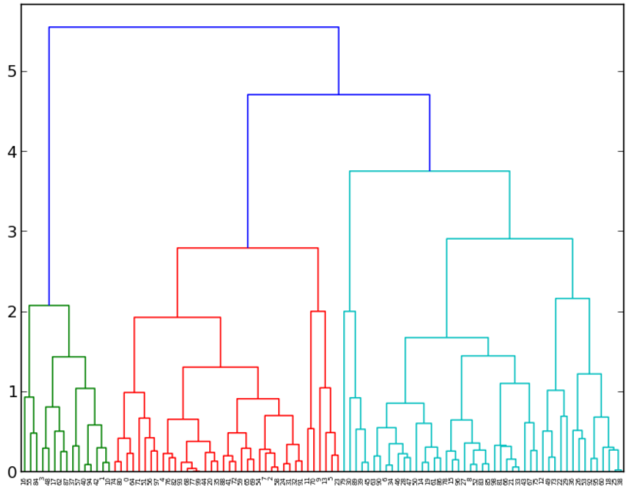
- Scalability
  - Clustering all the data instead of only on samples
- Ability to deal with different types of attributes
  - Numerical, binary, categorical, ordinal, linked, and mixture of these
- Constraint-based clustering
  - User may give inputs on constraints
  - Use domain knowledge to determine input parameters
- Interpretability and usability
- Others
  - Discovery of clusters with arbitrary shape, ability to deal with noisy data, incremental clustering and insensitivity to input order, high dimensionality, etc.

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

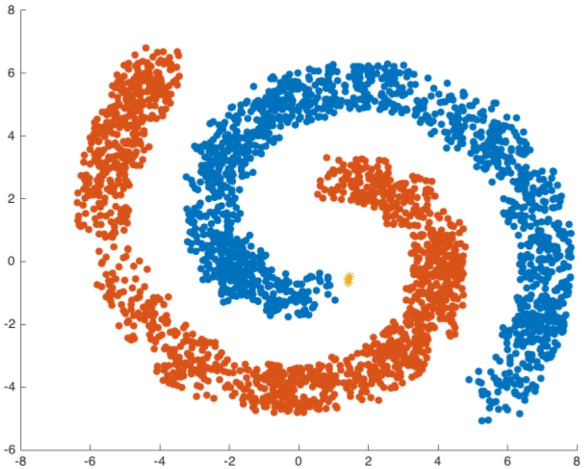


- Hierarchical approach
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, CAMELEON

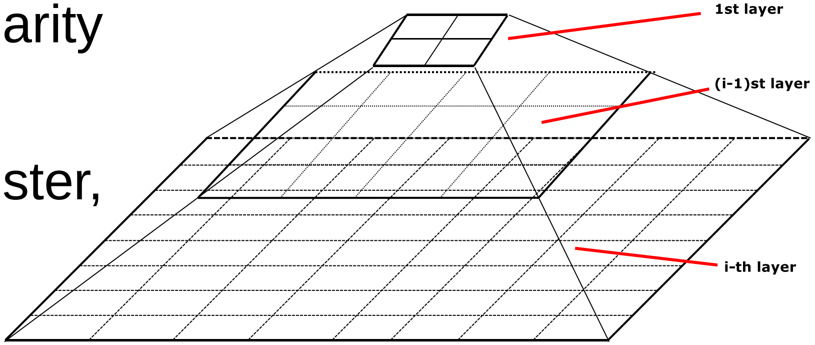


Major Clustering Approaches

- Density-based approach
  - Based on connectivity and density functions
  - Typical methods: DBSCAN, OPTICS, DenClue



- Grid-based approach
  - based on a multiple-level granularity structure
  - Typical methods: 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
  - Typical methods: EM, SOM, COBWEB
- Frequent pattern-based:
  - Based on the analysis of frequent patterns
  - Typical methods: p-Cluster
- User-guided or constraint-based:
  - Considering user-specified or application-specific constraints
  - Typical methods: COD (obstacles), constrained clustering
- Link-based clustering:
  - Objects are often linked together in various ways
  - Massive links can be used to cluster objects: SimRank, LinkClus

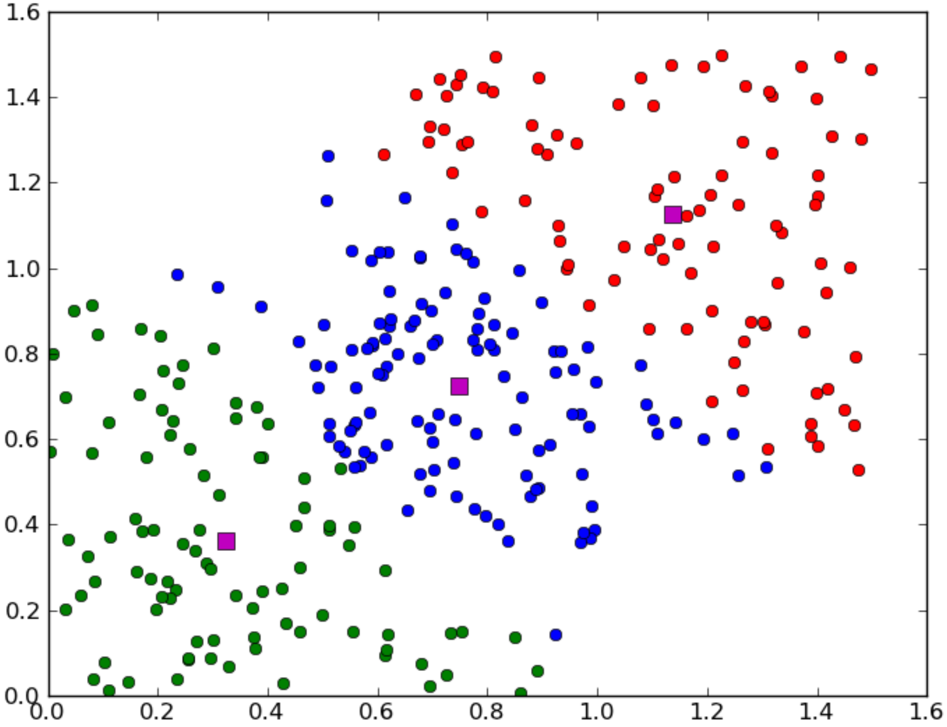
Outline

- Cluster Analysis: Basic Concepts
- **Partitioning Methods**
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Evaluation of Clustering
- Summary

Partitioning Algorithms: Basic Concepts

- 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)

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



Partitioning Algorithms: Basic Concepts

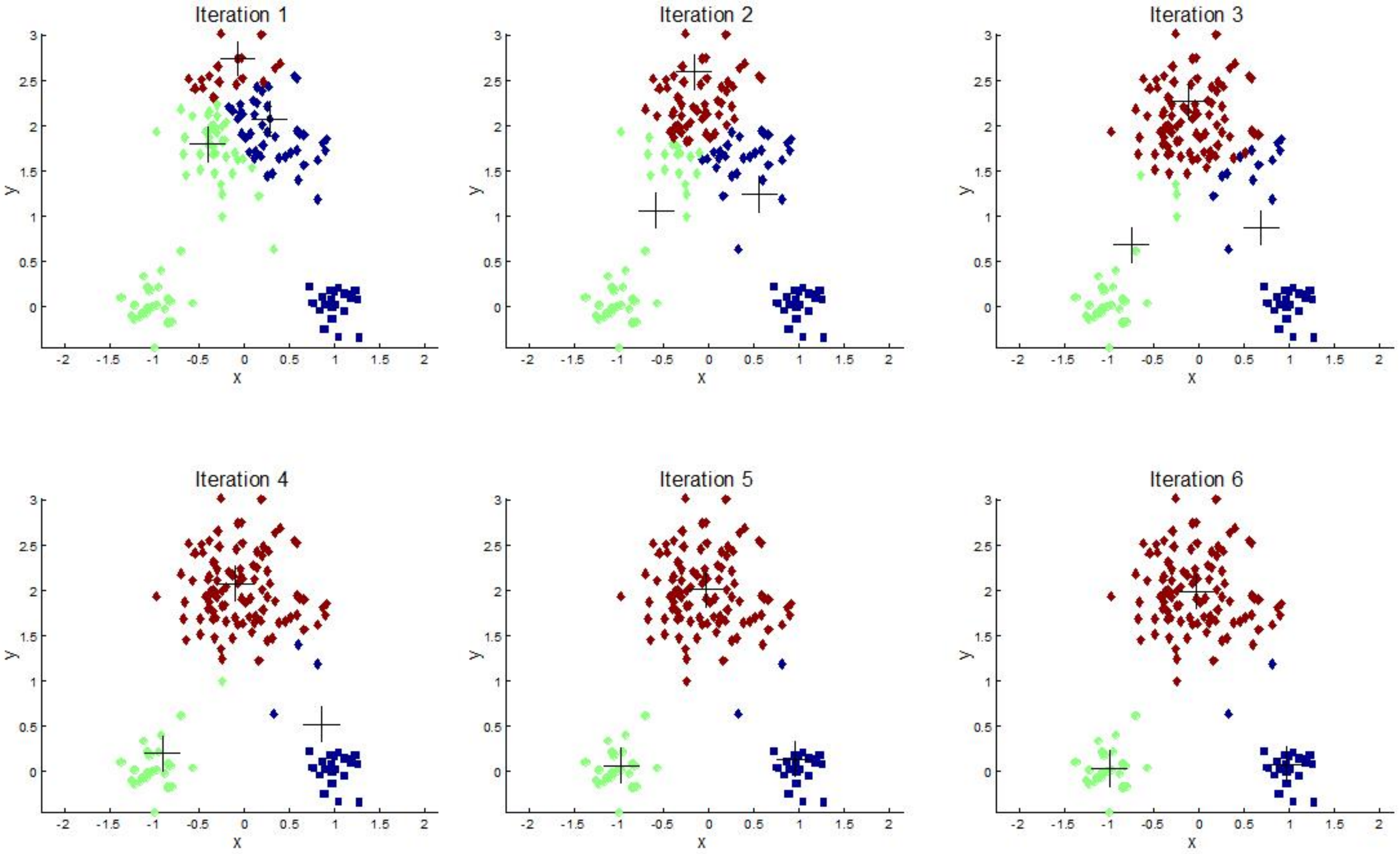
- Given k, find a partition of k clusters that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: k-means and k-medoids algorithms
  - k-means (MacQueen’67, Lloyd’57/’82): Each cluster is represented by the center of the cluster
  - k-medoids or PAM (Partition around medoids) (Kaufman & Rousseeuw’87): Each cluster is represented by one of the objects in the cluster

k-means: Algorithm

- Input
  - k: the number of clusters
  - D: a data set containing n objects
- Output: A set of k clusters
- Method

```
(1) arbitrarily choose $k$ objects from $D$ as the initial cluster centers;
(2) repeat
(3)     (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;
(4)     update the cluster means, that is calculate the mean value of
            the objects for each cluster;
(5) until no change;
```

k-means: Grouping points into clusters



k-means: An Example

- Given five points whose XY-coordinates are A(1,1), B(1,2), C(4,1), D(4,2), E(3,3). Apply k-means (k = 2) to cluster these points.
- Assume that the initial centers are C_1 = A(1,1) and C_2 = B(1,2)
- The distance metric is Euclidean distance
- First iteration
  - Cluster 1 = {A, C}
  - Cluster 2 = {B, D, E}
  - C_1 = (2.5, 1) and C_2 = (2.67,2.33)

Points	Distance to C_1	Distance to C_2
A(1,1)	0	1
B(1,2)	1	0
C(4,1)	3	3.16
D(4,2)	3.16	3
E(3,3)	2.83	2.24

- Second iteration
  - Cluster 1 = {A, C}
  - Cluster 2 = {B, D, E}
  - C_1 = (2.5, 1) and C_2 = (2.67,2.33)

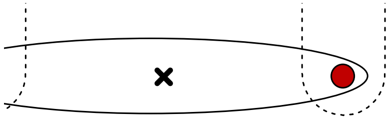
Points	Distance to C_1	Distance to C_2
A(1,1)	1.50	2.13
B(1,2)	1.80	1.70
C(4,1)	1.50	1.88
D(4,2)	1.80	1.37
E(3,3)	2.06	0.75

Comments on k-Means

- 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: $O(ks^2 + k(n-k))$
- 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 k-means

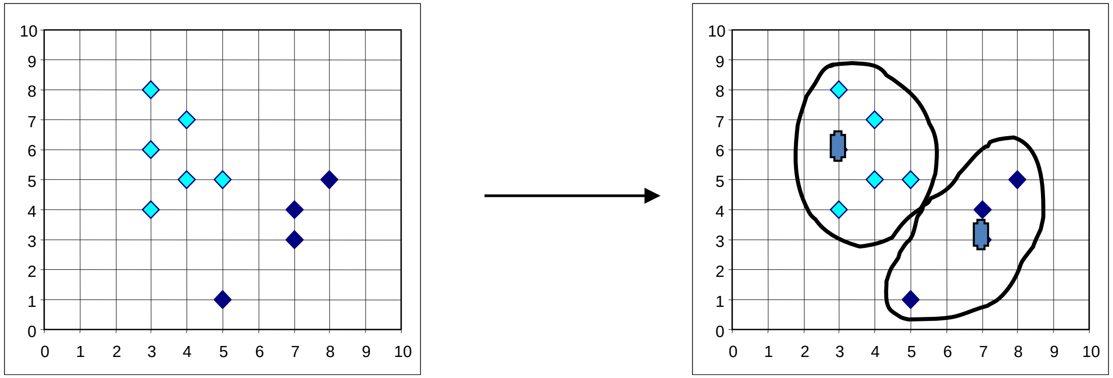
- Most of the variants of the k-means which differ in
  - Selection of the initial k means
  - Dissimilarity calculations
  - Strategies to calculate cluster means



- 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

What Is the Problem of k-means?

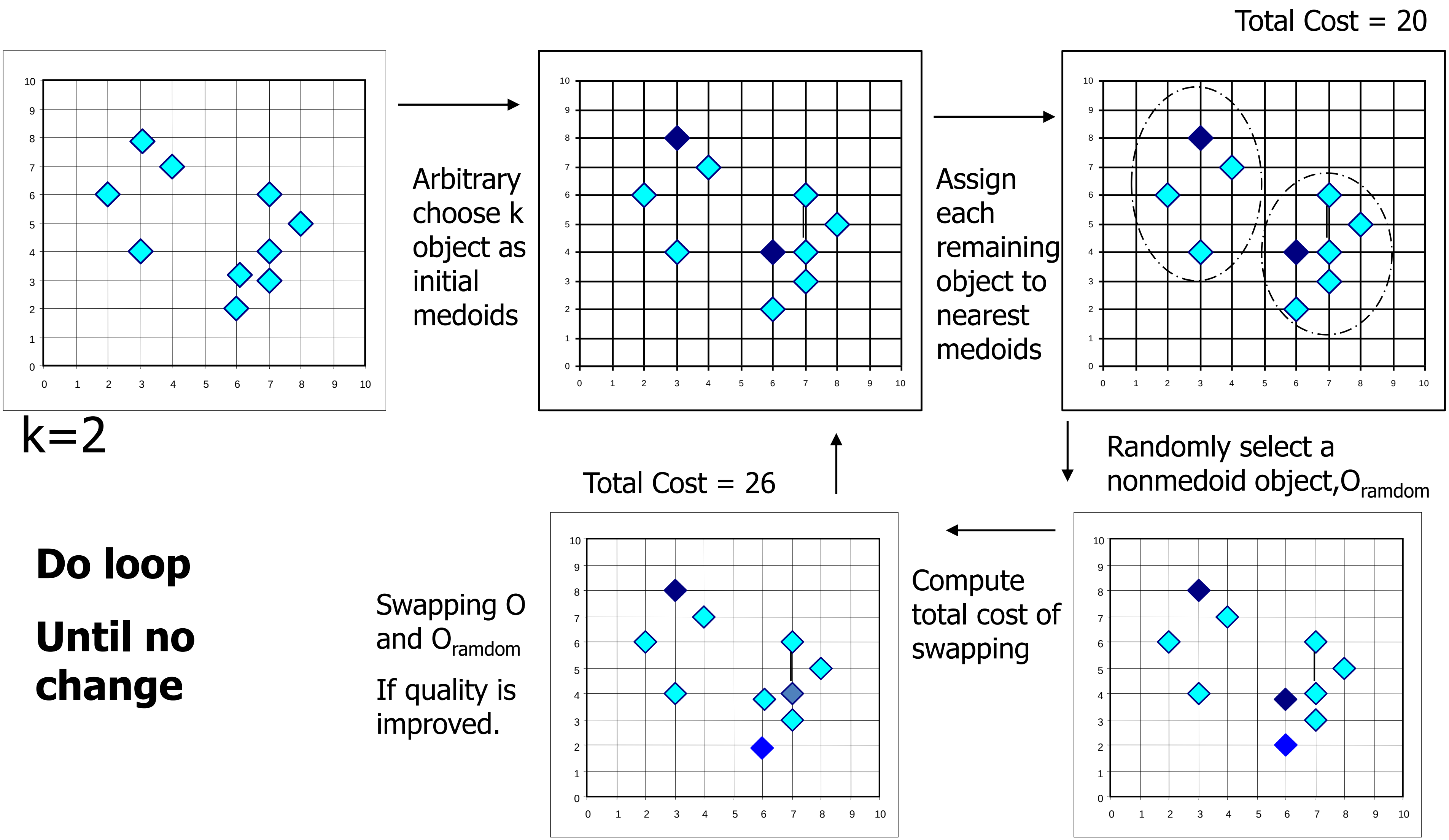
- The k-means algorithm is sensitive to outliers !
- Since an object with an extremely large value may substantially distort the distribution of the data
- k-medoids: Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster



k-medoids Clustering

- 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

PAM: A Typical k-medoids Algorithm



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

k-medoids: An Example

- Cluster the following data set of ten objects into two clusters i.e. k = 2.
- Consider a data set of ten objects as follows:
- The distance metric is Manhattan distance

Point	X	Y
X_1	2	6
X_2	3	4
X_3	3	8
X_4	4	7
X_5	6	2
X_6	6	4
X_7	7	3
X_8	7	4
X_9	8	5
X_{10}	7	6

k-medoids: An Example

- Let us assume X_2 and X_8 are selected as medoids, so the centers are C_1 = (3, 4) and C_2 = (7, 4)
- Cluster 1 = {X_1, X_3, X_4}
- Cluster 2 = {X_5, X_6, X_7, X_9, X_{10}}
- Total cost = 20

Point	X	Y	Distance to C_1	Distance to C_2
X_1	2	6	3	7
X_3	3	8	4	8
X_4	4	7	4	6
X_5	6	2	5	3
X_6	6	4	3	1
X_7	7	3	5	1
X_9	8	5	6	2
X_{10}	7	6	6	2

k-medoids: An Example

- Select one of the nonmedoids O′, Let us assume O′ = (7,3), i.e. X_7. So now the medoids are C_1(3,4) and O′(7,3)
- Total cost = 22. So cost of swapping medoid from c_2 to O′ is

  S = current total cost – past total cost = 22 – 20 = 2 > 0

  ⇒ moving to O′ would be a bad idea

We try other nonmedoids and found that our first choice was the best. So the configuration does not change and algorithm terminates here (i.e. there is no change in the medoids).

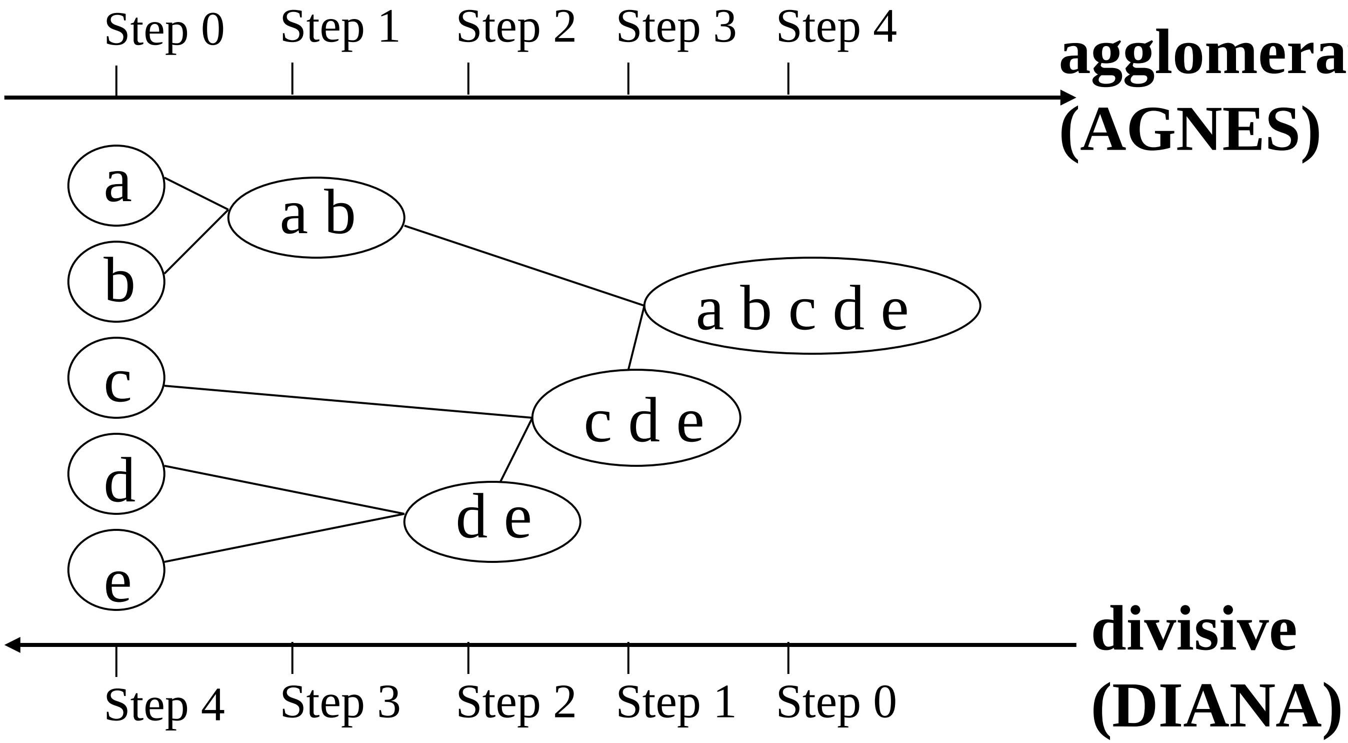
Point	X	Y	Distance to C_1	Distance to O’
X_1	2	6	3	8
X_3	3	8	4	9
X_4	4	7	4	7
X_5	6	2	5	2
X_6	6	4	3	2
X_7	7	3	5	1
X_9	8	5	6	3
X_{10}	7	6	6	3

Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- **Hierarchical Methods**
- Density-Based Methods
- Grid-Based Methods
- Evaluation of Clustering
- Summary

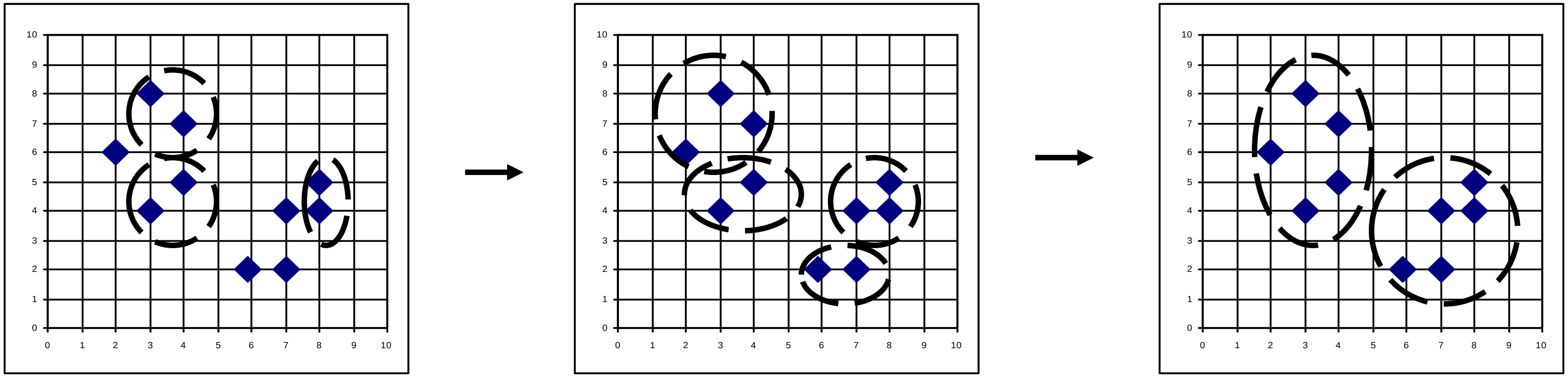
Hierarchical Clustering

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



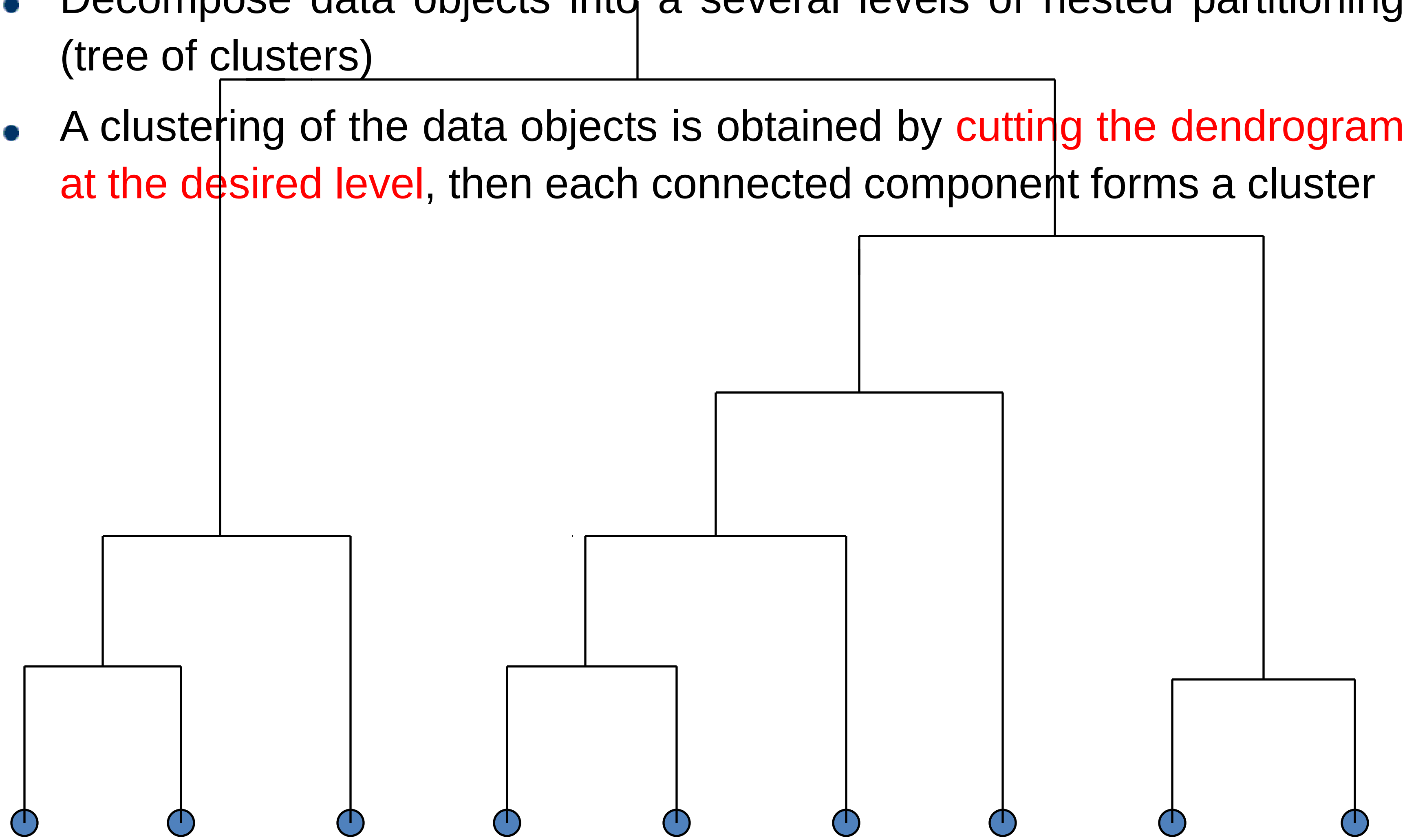
AGNES (Agglomerative Nesting)

- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical 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



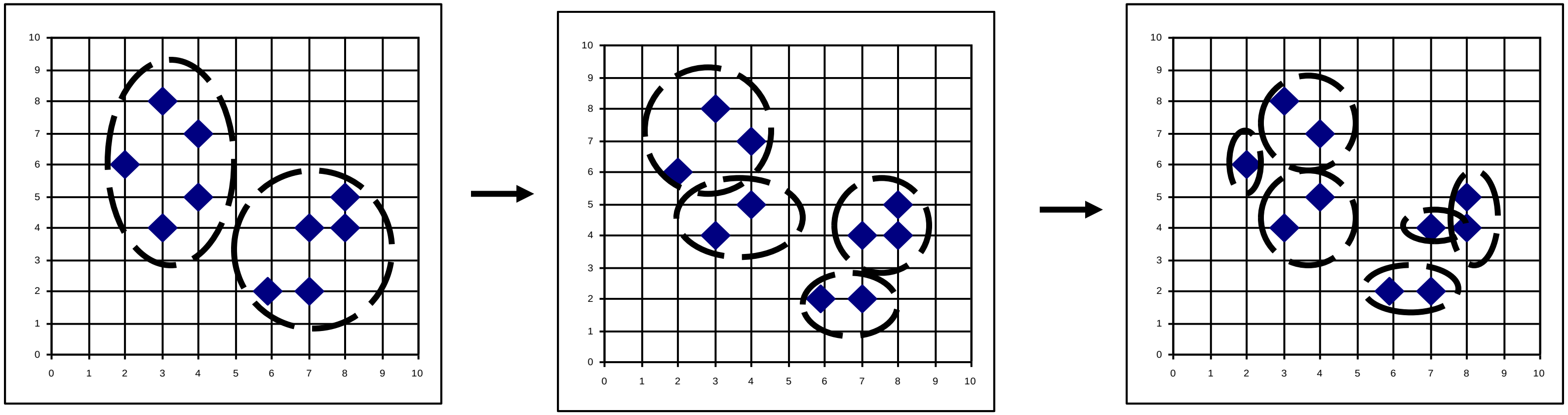
Dendrogram: Shows How Clusters are Merged

- Decompose data objects into a several levels of nested partitioning (tree of clusters)
- 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



Distance between Clusters

- Single link: smallest distance between an element in one cluster and an element in the other

$$\text{dist}(C_i, C_j) = \min_{p \in C_i, p' \in C_j} \|p - p'\|$$

- Complete link: largest distance between an element in one cluster and an element in the other

$$\text{dist}(C_i, C_j) = \max_{p \in C_i, p' \in C_j} \|p - p'\|$$

- Average: average distance between an element in one cluster and an element in the other

$$\text{dist}(C_i, C_j) = \frac{1}{n_i n_j} \sum_{p \in C_i, p' \in C_j} |p - p'|$$

  - n_i and n_j are the number of objects in C_i and C_j, respectively

Distance between Clusters

- Average: distance between the means of two clusters

$$\mathrm{dist}(C_i, C_j) = |m_i - m_j|$$

  - m_i and m_j are the means of objects in C_i and C_j, respectively
- Centroid: distance between the centroids of two clusters

$$\mathrm{dist}(C_i, C_j) = \mathrm{dist}(c_i, c_j)$$

  - c_i and c_j are the centroids of objects in C_i and C_j, respectively
- Medoid: distance between the medoids of two clusters

$$\mathrm{dist}(C_i, C_j) = \mathrm{dist}(M_i, M_j)$$

  - M_i and M_j are the medoids of objects in C_i and C_j, respectively
  - Medoid: a chosen, centrally located object in the cluster

Centroid, Radius and Diameter of a Cluster

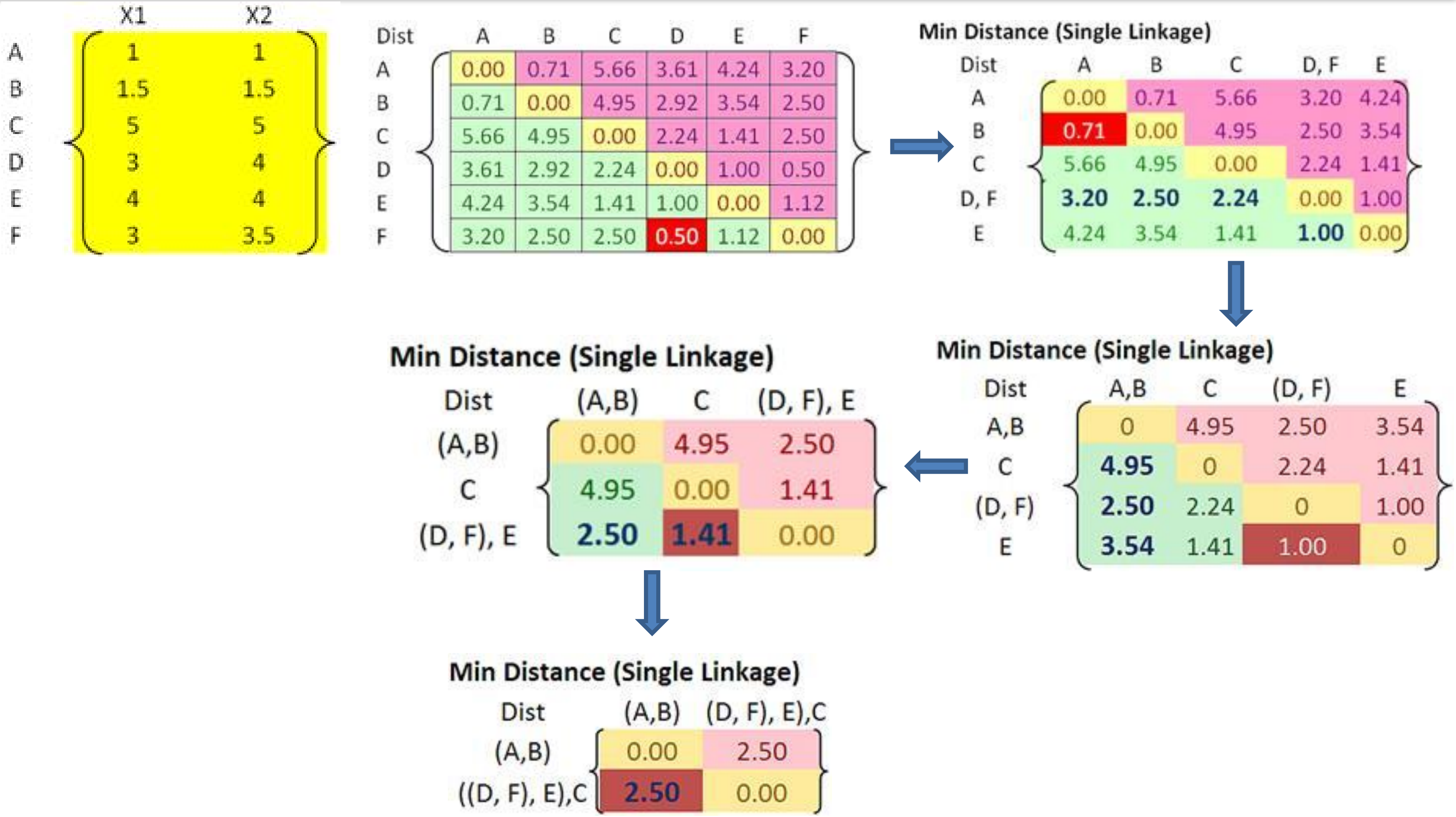
- These are for numerical data sets
- Centroid: the “middle” of a cluster $C_m = \frac{\sum_{i=1}^{N}(t_{ip})}{N}$
- Radius: square root of average distance from any point of the cluster to its centroid $R_m = \sqrt{\frac{\sum_{i=1}^{N}(t_{ip} - c_m)^2}{N}}$
- Diameter: square root of average mean squared distance between all pairs of points in the cluster

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

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

AGNES: An Example



	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5

Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	**3.20**	**2.50**	**2.24**	0.00	1.00
E	4.24	3.54	1.41	**1.00**	0.00

Min Distance (Single Linkage)

Dist	A,B	C	(D, F)	E
A,B	0	4.95	2.50	3.54
C	**4.95**	0	2.24	1.41
(D, F)	**2.50**	2.24	0	1.00
E	**3.54**	1.41	1.00	0

Min Distance (Single Linkage)

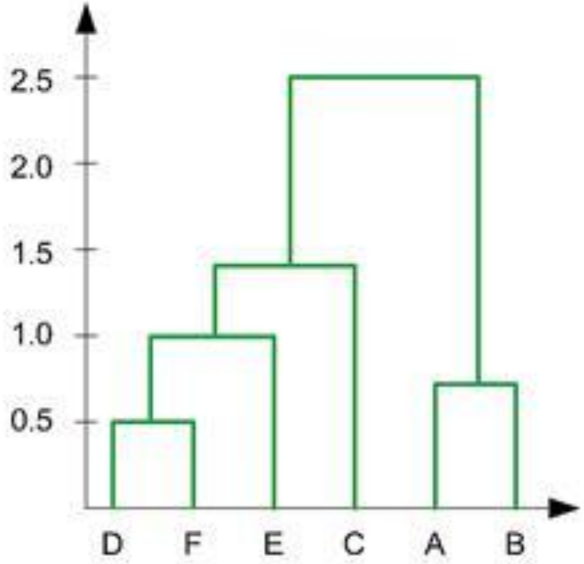
Dist	(A,B)	C	(D, F), E
(A,B)	0.00	4.95	2.50
C	4.95	0.00	1.41
(D, F), E	**2.50**	**1.41**	0.00

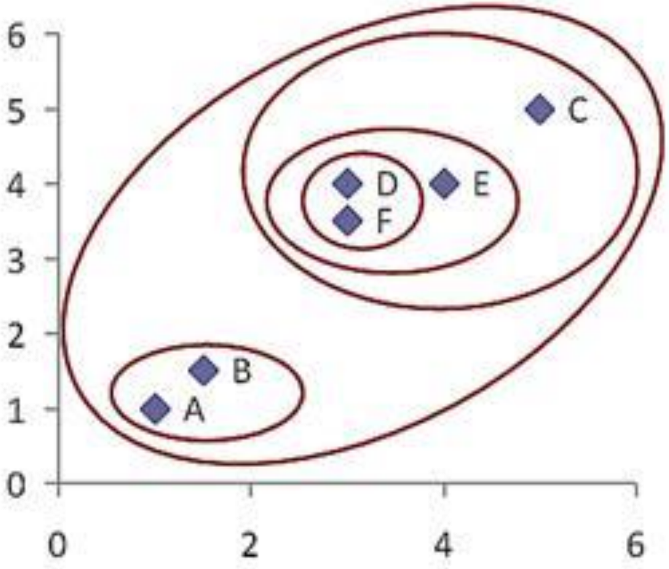
Min Distance (Single Linkage)

Dist	(A,B)	(D, F), E),C
(A,B)	0.00	2.50
((D, F), E),C	**2.50**	0.00

AGNES: An Example

$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \end{array} \begin{array}{c} X1 \quad\quad X2 \\ \left\{ \begin{array}{cc} 1 & 1 \\ 1.5 & 1.5 \\ 5 & 5 \\ 3 & 4 \\ 4 & 4 \\ 3 & 3.5 \end{array} \right\} \end{array}$$





Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- **Density-Based Methods**
- Grid-Based Methods
- Evaluation of Clustering
- Summary

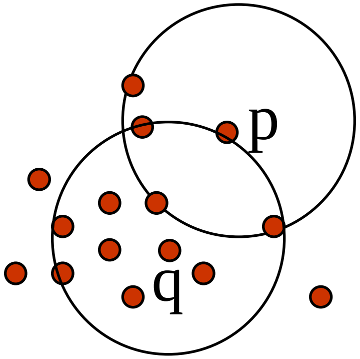
Density-Based Clustering Methods

- 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
- $N_{Eps}(p)$: $\{q$ belongs to $D \mid \mathrm{dist}(p, q) \leq Eps\}$
- 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:

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

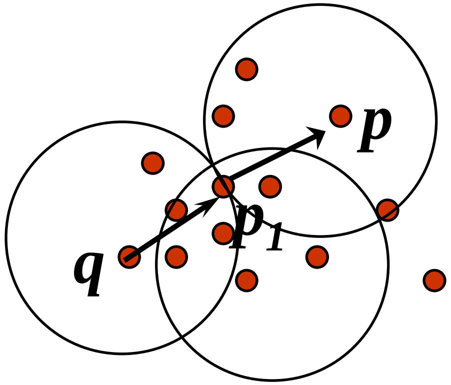


MinPts = 5

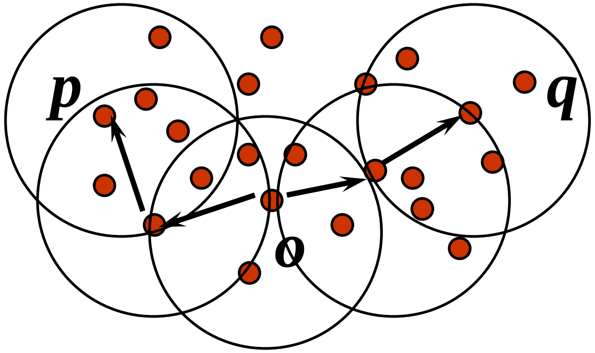
Eps = 1 cm

Density-Reachable and Density-Connected

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

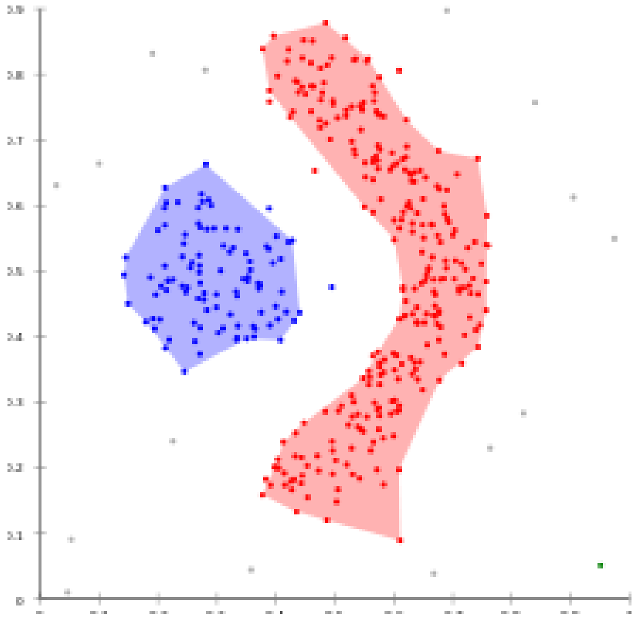


- A point p is density-connected to a point q w.r.t. Eps, MinPts if there is a point o such that both, p and q are density-reachable from o w.r.t. Eps and MinPts



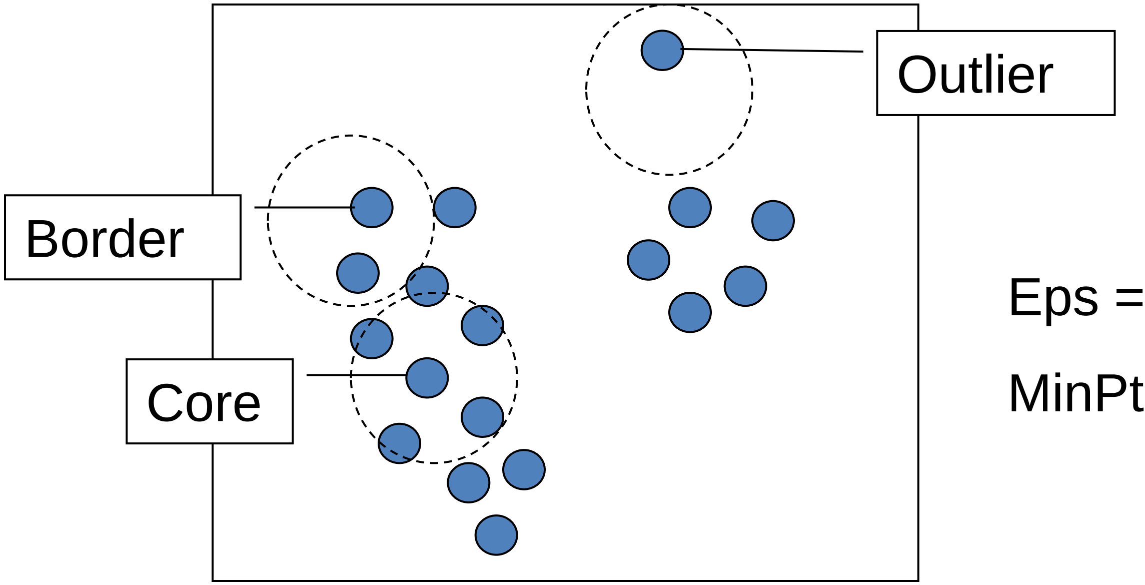
DBSCAN Clustering Algorithm

- DBSCAN: Density-Based Spatial Clustering
- 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 Method: 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



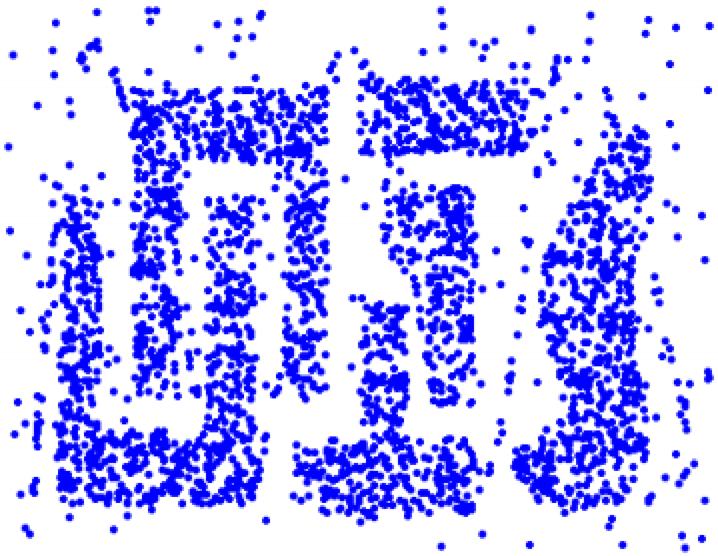
Eps = 1cm

MinPts = 5

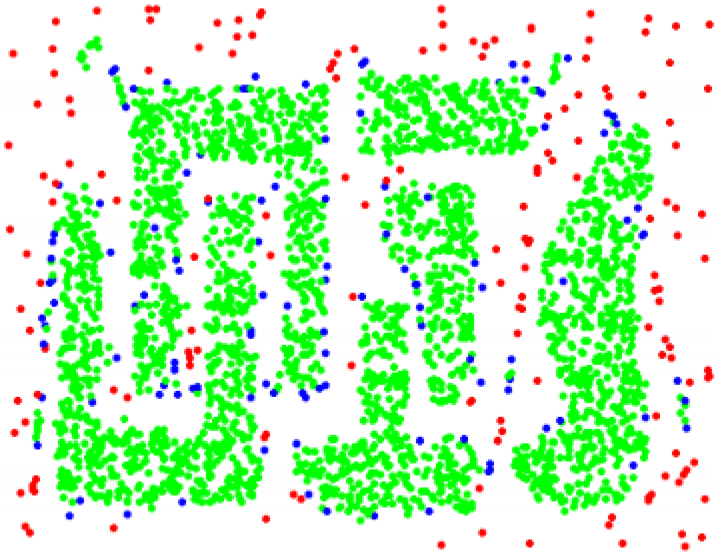
Core Point, Border Point and Noise Point

- A point is a core point if it has more than a specified number of points (MinPts) within Eps—These are points that are at the interior of a cluster.
- A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point.
- A noise point is any point that is not a core point nor a border point.

Core Point, Border Point and Noise Point



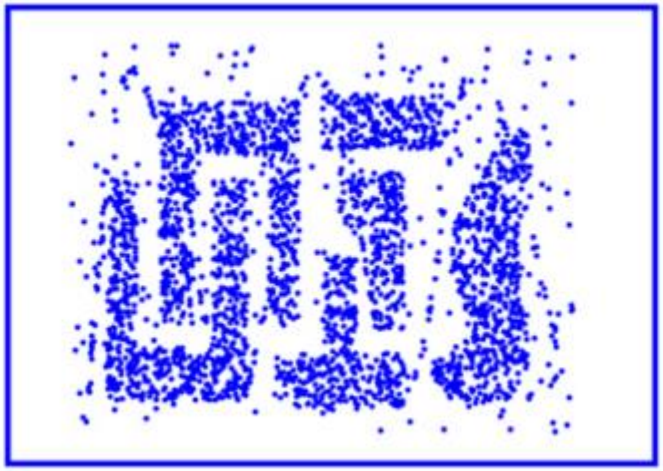
Original Points

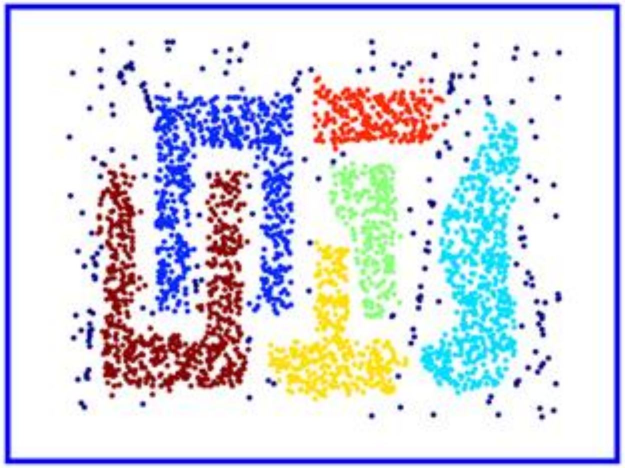


Point types: core, border and outliers

ε = 10, MinPts = 4

DBSCAN: An Example



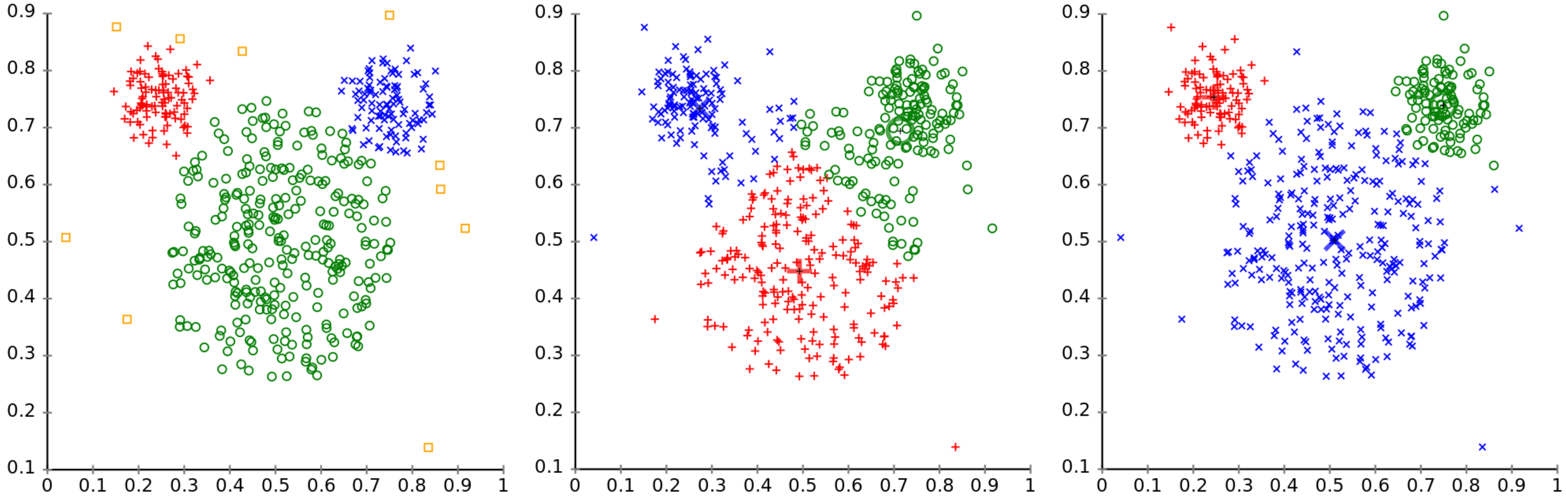


Different cluster analysis results on "mouse" data set:

Original Data

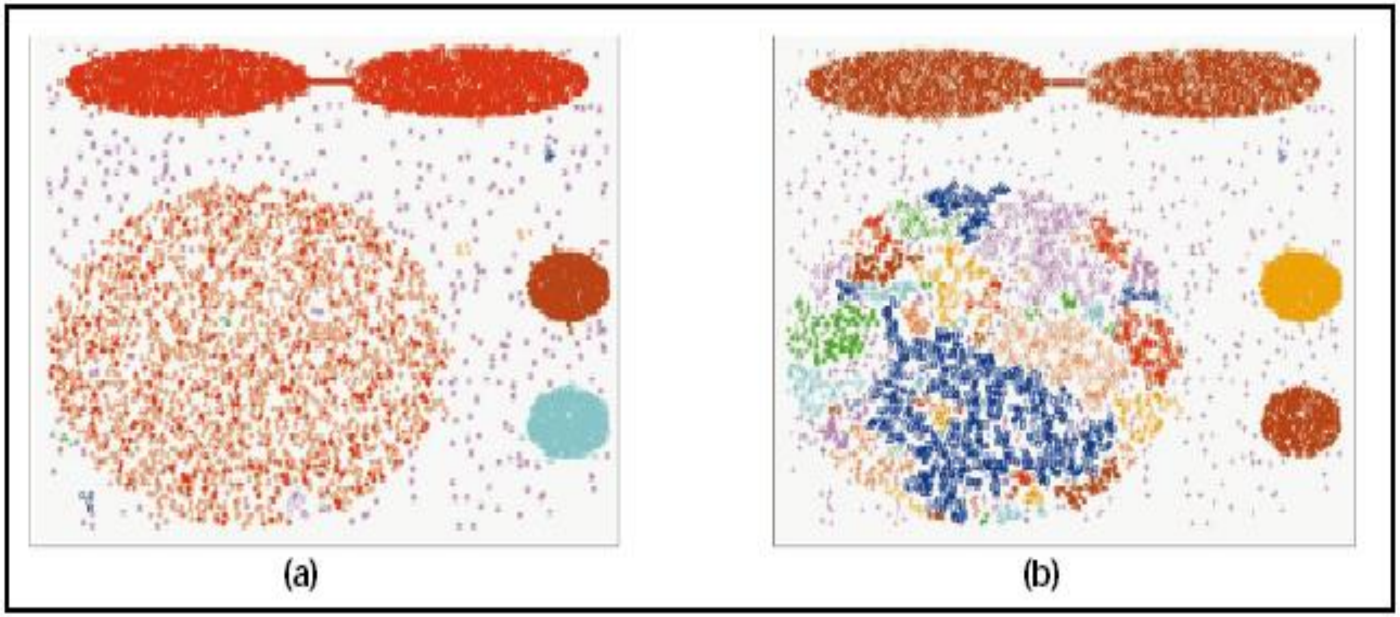
k-Means Clustering

EM Clustering



DBSCAN: Sensitive to Parameters

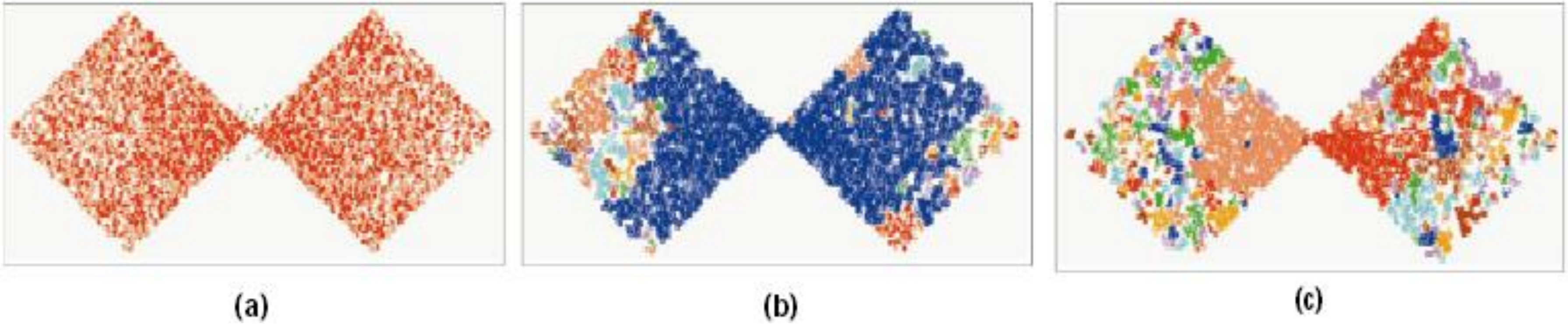
Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.



(a)

(b)

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



(a)

(b)

(c)

OPTICS Clustering Algorithm

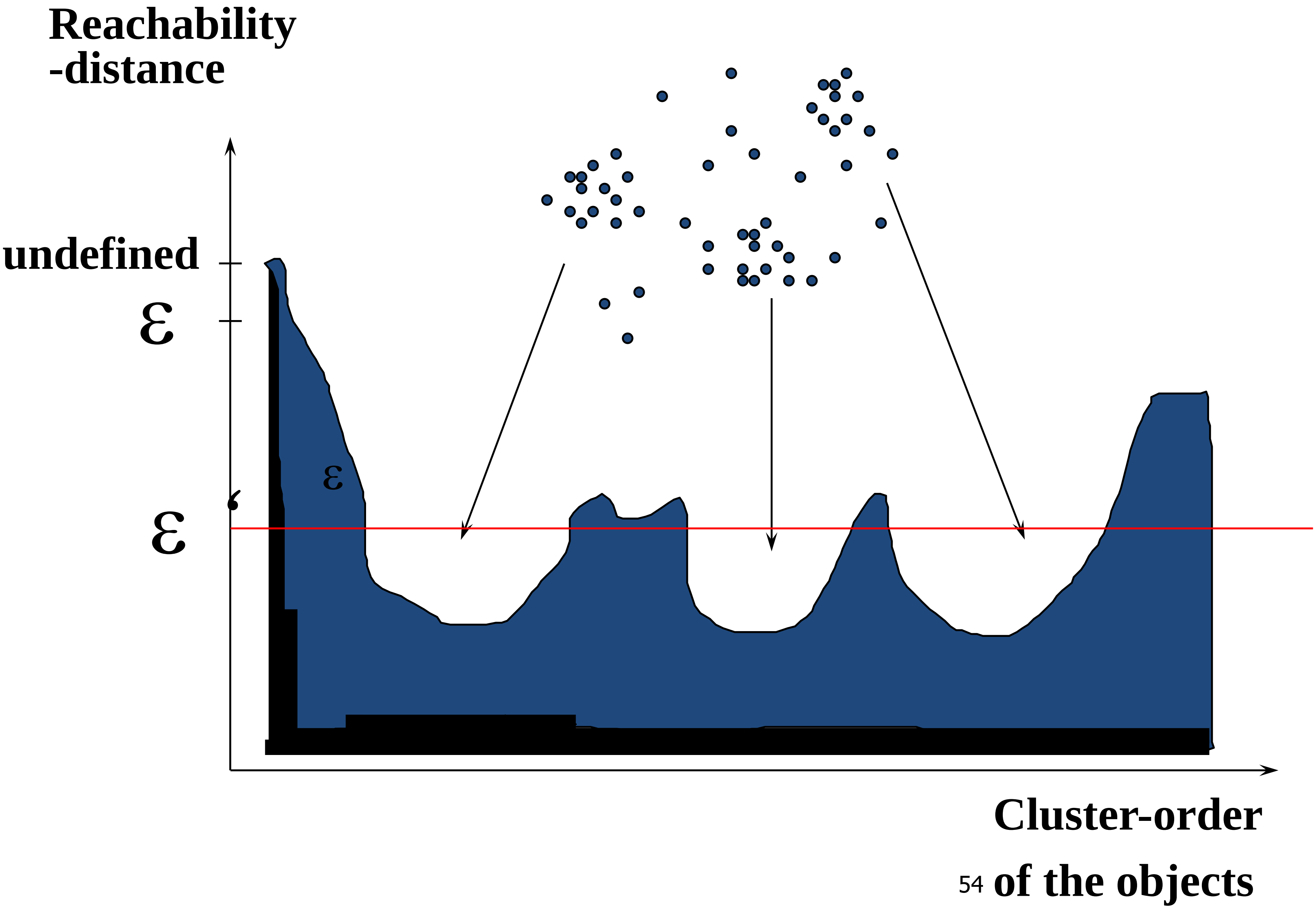
- OPTICS: Ordering Points To Identify The Clustering Structure
  - Ankerst, Breunig, Kriegel, and Sander (SIGMOD’99)
  - Produces a special order of the database wrt its density-based clustering structure
  - This cluster-ordering contains info equiv to the density-based clusterings corresponding to a broad range of parameter settings
  - Good for both automatic and interactive cluster analysis, including finding intrinsic clustering structure
  - Can be represented graphically or using visualization techniques

OPTICS Clustering Algorithm

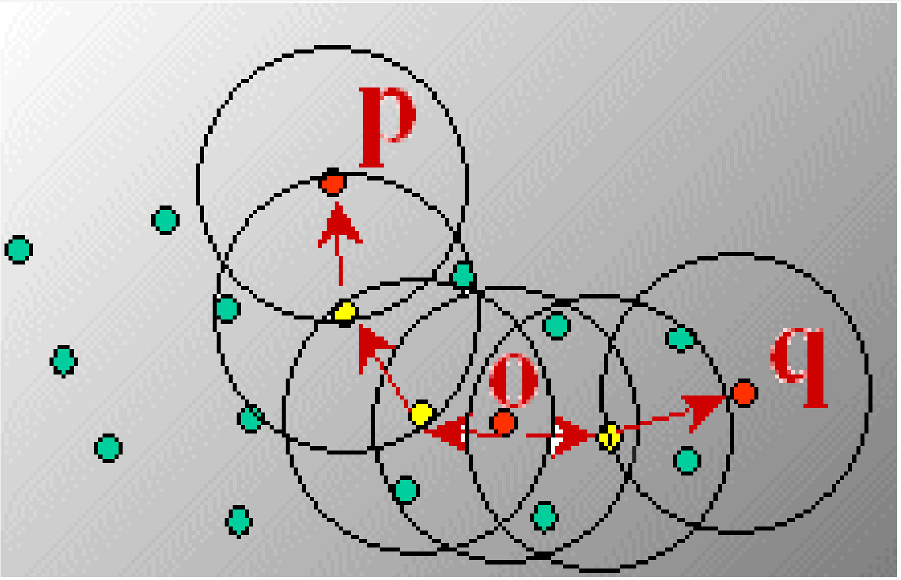
- OPTICS begins with an arbitrary object from the input database as the current object, p
- It retrieves the Eps-neighborhood of p, determines the core-distance, and sets the reachability-distance to undefined. The current object, p, is then written to output
  - The core-distance of an object p is the smallest value Eps such that the Eps-neighborhood of p has at least MinPts objects
  - The reachability-distance to object p from q is the minimum radius value that makes p density-reachable from q

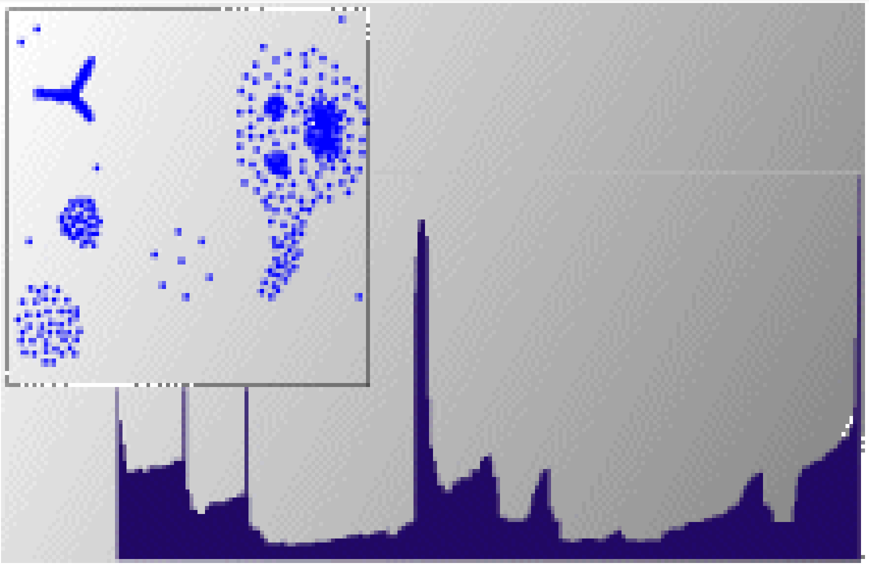
OPTICS Clustering Algorithm

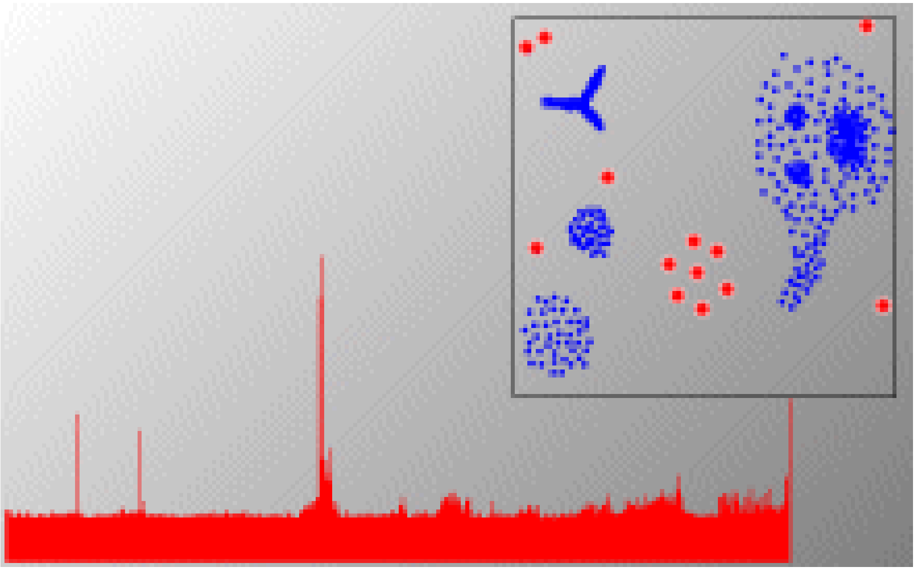
- If p is not a core object, OPTICS simply moves on to the next object in the OrderSeeds list (or the input database if OrderSeeds is empty)
- If p is a core object, then for each object, q, in the Eps-neighborhood of p, OPTICS updates its reachability-distance from p and inserts q into OrderSeeds if q has not yet been processed.
- The iteration continues until the input is fully consumed and OrderSeeds is empty

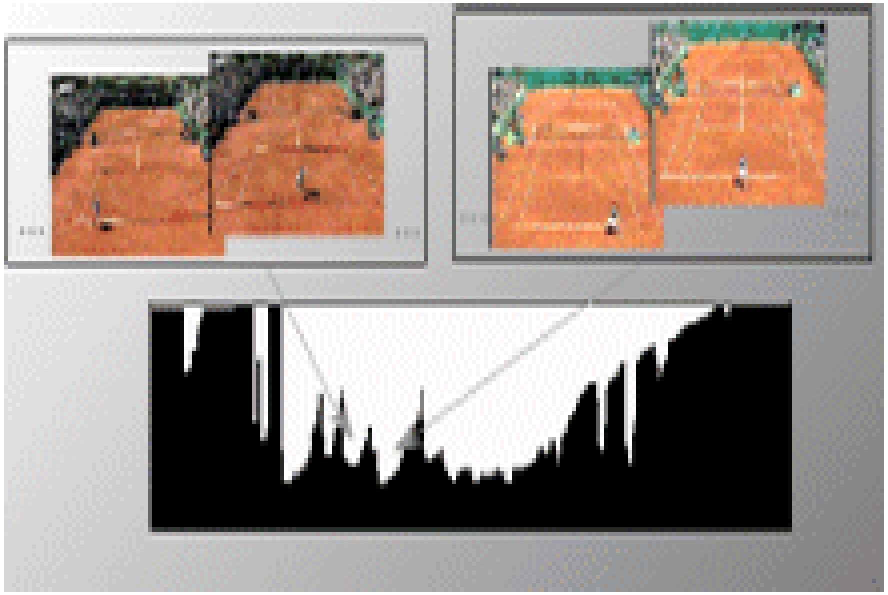

Reachability
-distance
undefined
ε
ε
ε
Cluster-order
of the objects

Density-Based Clustering: OPTICS & Its Applications









Outline

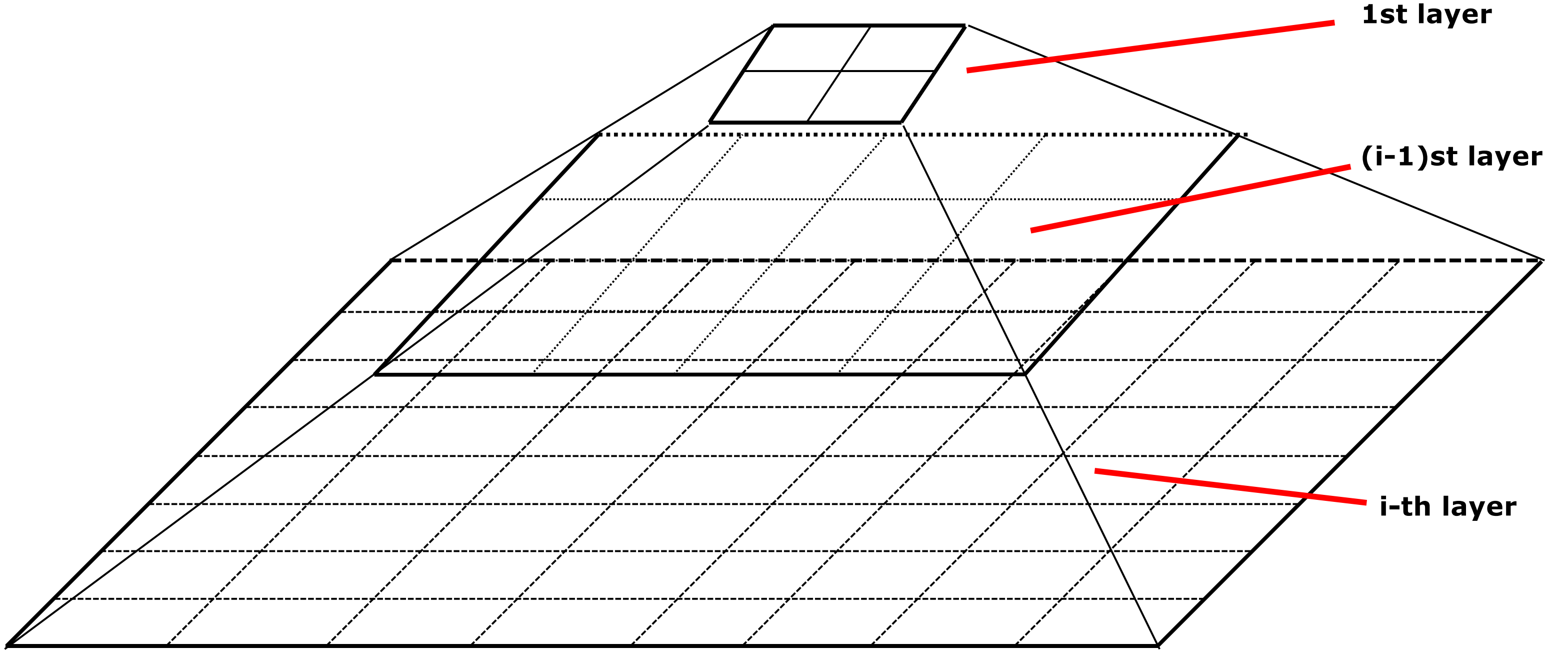
- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based
- **Grid-Based Methods**
- Evaluation of Clustering
- Summary

Grid-Based Clustering Method

- Using multi-resolution grid data structure
- Several interesting methods
  - STING (a STatistical INformation Grid approach) by Wang, Yang and Muntz (1997)
  - WaveCluster by Sheikholeslami, Chatterjee, and Zhang (VLDB'98)
    - A multi-resolution clustering approach using wavelet method
  - CLIQUE: Agrawal, et al. (SIGMOD'98)
    - Both grid-based and subspace clustering

STING: A Statistical Information Grid Approach

- Wang, Yang and Muntz (VLDB'97)
- The spatial area is divided into rectangular cells
- There are several levels of cells corresponding to different levels of resolution



STING Clustering Algorithm

- Each cell at a high level is partitioned into a number of smaller cells in the next lower level
- Statistical info of each cell is calculated and stored beforehand and is used to answer queries
- Parameters of higher level cells can be easily calculated from parameters of lower level cell
  - count, mean, s, min, max
  - type of distribution—normal, uniform, etc.
- Use a top-down approach to answer spatial data queries
- Start from a pre-selected layer—typically with a small number of cells
- For each cell in the current level compute the confidence interval

STING Algorithm and Its Analysis

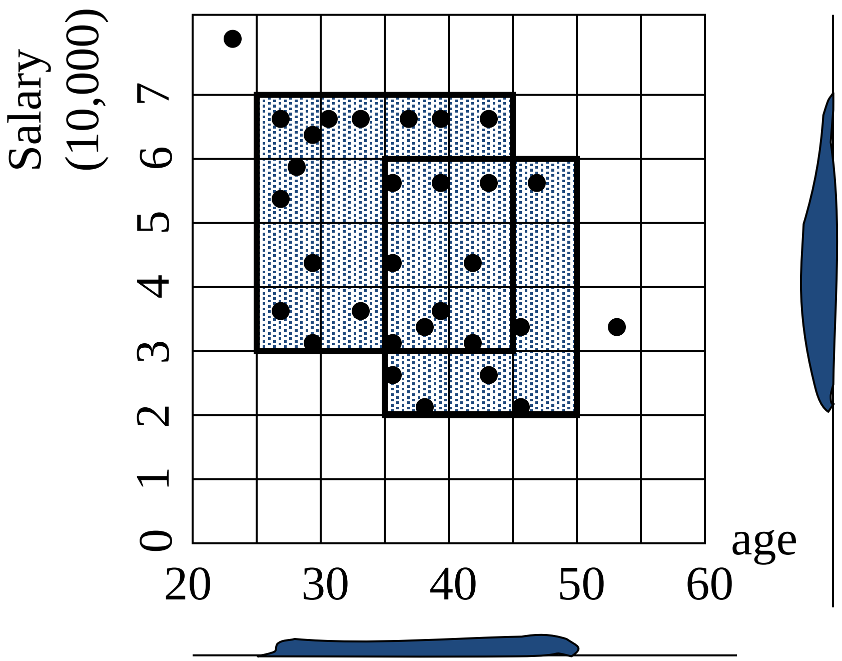
- Remove the irrelevant cells from further consideration
- When finish examining the current layer, proceed to the next lower level
- Repeat this process until the bottom layer is reached
- Advantages:
  - Query-independent, easy to parallelize, incremental update
  - $O(K)$, where K is the number of grid cells at the lowest level
- Disadvantages:
  - All the cluster boundaries are either horizontal or vertical, and no diagonal boundary is detected

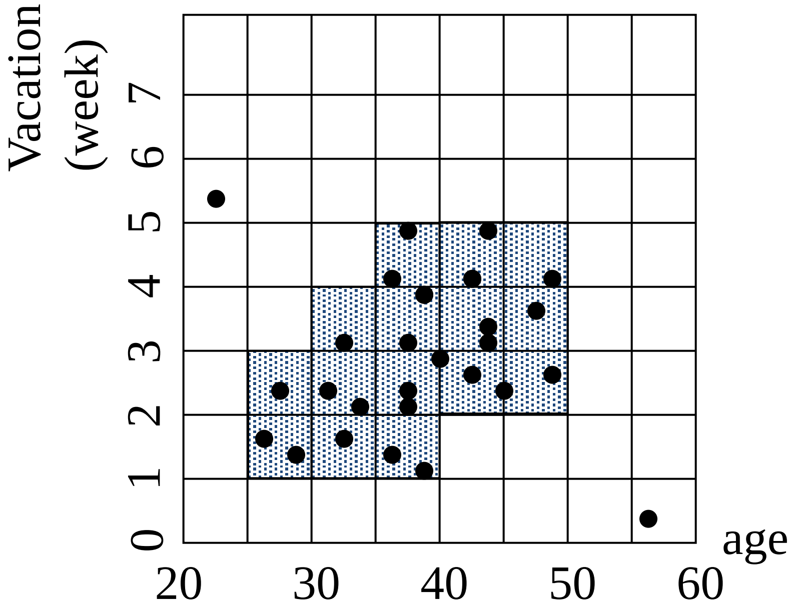
CLIQUE (Clustering In QUEst)

- Agrawal, Gehrke, Gunopulos, Raghavan (SIGMOD'98)
- Automatically identifying subspaces of a high dimensional data space that allow better clustering than original space
- CLIQUE can be considered as both density-based and grid-based
  - It partitions each dimension into the same number of equal length interval
  - It partitions an m-dimensional data space into non-overlapping rectangular units
  - A unit is dense if the fraction of total data points contained in the unit exceeds the input model parameter
  - A cluster is a maximal set of connected dense units within a subspace

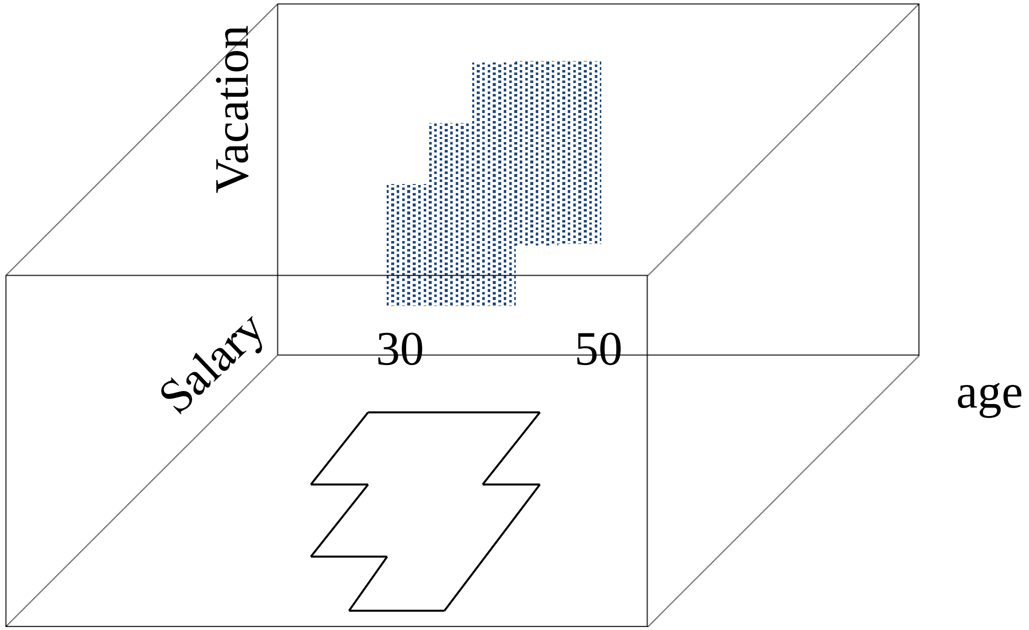
CLIQUE: The Major Steps

- Partition the data space and find the number of points that lie inside each cell of the partition.
- Identify the subspaces that contain clusters using the Apriori principle
- Identify clusters
  - Determine dense units in all subspaces of interests
  - Determine connected dense units in all subspaces of interests.
- Generate minimal description for the clusters
  - Determine maximal regions that cover a cluster of connected dense units for each cluster
  - Determination of minimal cover for each cluster


Salary
(10,000)
0
1
2
3
4
5
6
7
20
30
40
50
60
age


Vacation
(week)
0
1
2
3
4
5
6
7
20
30
40
50
60
age

τ = 3


Vacation
Salary
30
50
age

Strength and Weakness of CLIQUE

- Strength
  - Automatically finds subspaces of the highest dimensionality such that high density clusters exist in those subspaces
  - Insensitive to the order of records in input and does not presume some canonical data distribution
  - Scales linearly with the size of input and has good scalability as the number of dimensions in the data increases
- Weakness
  - The accuracy of the clustering result may be degraded at the expense of simplicity of the method

Outline

- Cluster Analysis: Basic Concepts
- Partitioning Methods
- Hierarchical Methods
- Density-Based
- Grid-Based Methods
- **Evaluation of Clustering**
- Summary

Assessing Clustering Tendency

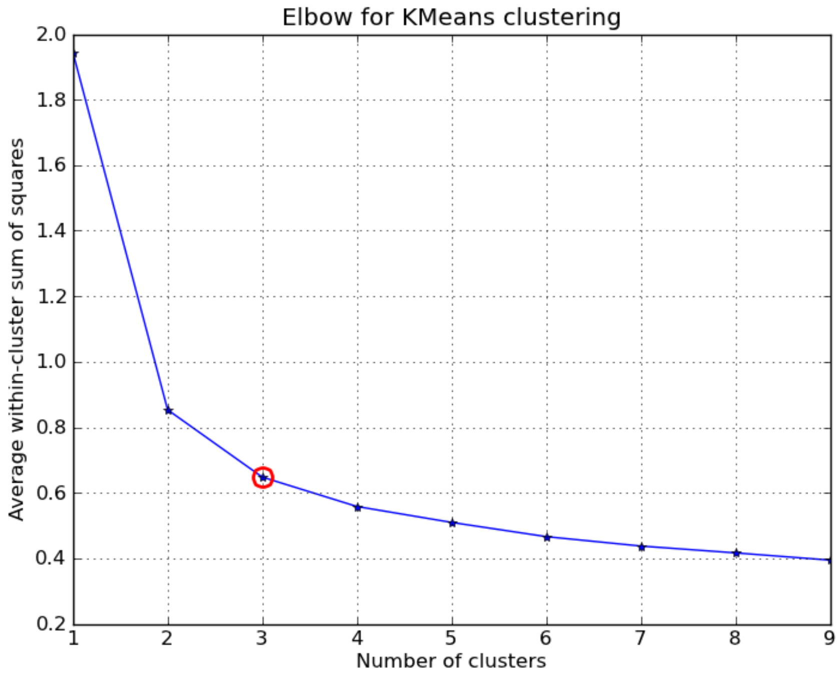
- Assess if non-random structure exists in the data by measuring the probability that the data is generated by a uniform data distribution
- Test spatial randomness by statistic test: Hopkins Static
  - Given a dataset D regarded as a sample of a random variable o, determine how far away o is from being uniformly distributed in the data space
  - Sample n points, $p_1 \dots p_n$, uniformly from D. For each p_i, find its nearest neighbor in D: $x_i = \min\{\text{dist}(p_i, v)\}$ where v in D
  - Sample n points, $q_1 \dots q_n$, uniformly from D. For each q_i, find its nearest neighbor in $D - q_i$: $y_i = \min\{\text{dist}(q_i, v)\}$ where v in D and $v \neq q_i$

Assessing Clustering Tendency

- Assess if non-random structure exists in the data by measuring the probability that the data is generated by a uniform data distribution
- Test spatial randomness by statistic test: Hopkins Static
  - Calculate the Hopkins Statistic: $H = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}$
  - If D is uniformly distributed, $\sum x_i$ and $\sum y_i$ will be close to each other and H is close to 0.5. If D is highly skewed, H is close to 0

Determine the Number of Clusters

- Empirical method
  - # of clusters ≈$\sqrt{n}/2$ for a dataset of n points
- Elbow method
  - Use the turning point in the curve of sum of within cluster variance w.r.t the # of clusters



Determine the Number of Clusters

- Cross validation method
  - Divide a given data set into m parts
  - Use $m - 1$ parts to obtain a clustering model
  - Use the remaining part to test the quality of the clustering
    - E.g., For each point in the test set, find the closest centroid, and use the sum of squared distance between all points in the test set and the closest centroids to measure how well the model fits the test set
  - For any k > 0, repeat it m times, compare the overall quality measure w.r.t. different k's, and find # of clusters that fits the data the best

Measuring Clustering Quality

- Extrinsic: supervised, i.e., the ground truth is available
  - Compare a clustering against the ground truth using certain clustering quality measure
  - E.g., BCubed precision and recall metrics
- Intrinsic: unsupervised, i.e., the ground truth is unavailable
  - Evaluate the goodness of a clustering by considering how well the clusters are separated, and how compact the clusters are
  - E.g., Silhouette coefficient

Measuring Clustering Quality: Extrinsic Methods

- Clustering quality measure: $Q(C, C_g)$, for a clustering C given the ground truth C_g
- Q is good if it satisfies the following 4 essential criteria
  - Cluster homogeneity: the purer, the better
  - Cluster completeness: should assign objects belong to the same category in the ground truth to the same cluster
  - Rag bag: putting a heterogeneous object into a pure cluster should be penalized more than putting it into a rag bag (i.e., “miscellaneous” or “other” category)
  - Small cluster preservation: splitting a small category into pieces is more harmful than splitting a large category into pieces

Outline

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

Summary

- Cluster analysis groups objects based on their similarity
- Measure of similarity can be computed for various types of data
- Clustering algorithms can be categorized into partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods
- Partitioning-based clustering algorithms: K-means and K-medoids
- Hierarchical clustering algorithms: Birch and Chameleon, and there are also probabilistic hierarchical clustering algorithms
- Density-based clustering algorithms: DBSCAN, OPTICS, and DENCLU
- Grid-based clustering methods: STING and CLIQUE, where CLIQUE is also a subspace clustering algorithm
- Quality of clustering results can be evaluated in various ways