POLITECNICO DI MILANO

Density-based, Grid-based, and Model-based Clustering

Data Mining and Text Mining (UIC 583 @ Politecnico di Milano)

Prof. Pier Luca Lanzi

Lecture outline

- Density-based clustering

Density-based clustering

What is density-based clustering?

- 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)

DBSCAN: Basic Concepts

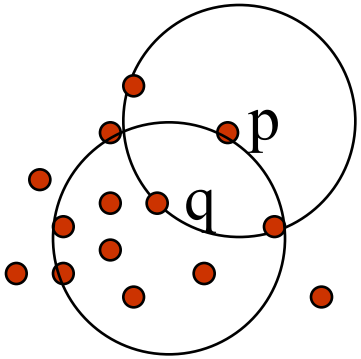
- The neighborhood within a radius ε of a given object is called the ε-neighborhood of the object
- If the ε-neighborhood of an object contains at least MinPts objects, then the object is a core object
- An object p is directly density-reachable from object q if p is within the ε-neighborhood of q and q is a core object
- An object p is density-reachable from object q if there is a chain of object p_1, …, p_n where p_1=p and p_n=q such that p_{i+1} is directly density reachable from p_i
- An object p is density-connected to q with respect to ε and MinPts if there is an object o such that both p and q are density reachable from o

DBSCAN: Basic Concepts

- Density = number of points within a specified radius (Eps)
- 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 or a border point
- A density-based cluster is a set of density-connected objects that is maximal with respect to density-reachability

Density-Reachable & Density-Connected

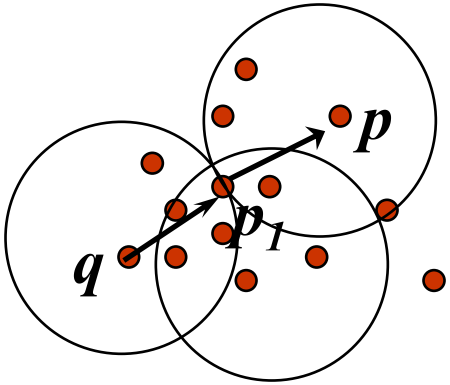
Directly density-reachable



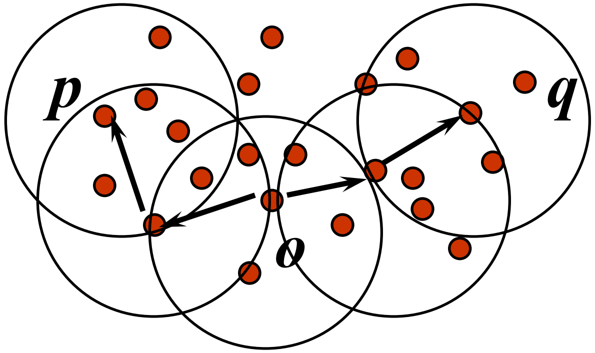
MinPts = 5

Eps = 1 cm

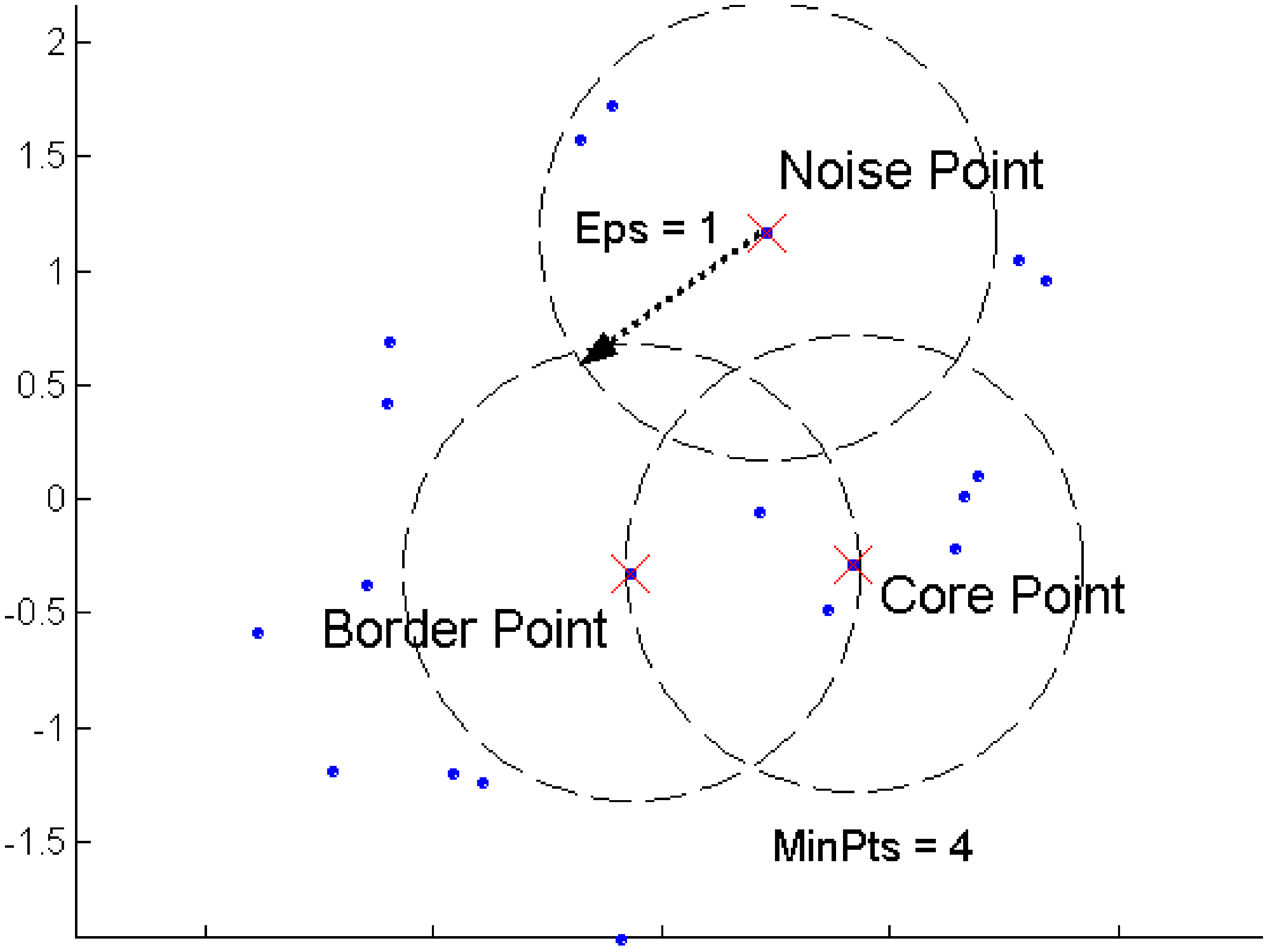
Density-reachable



Density-connected



DBSCAN: Core, Border, and Noise Points



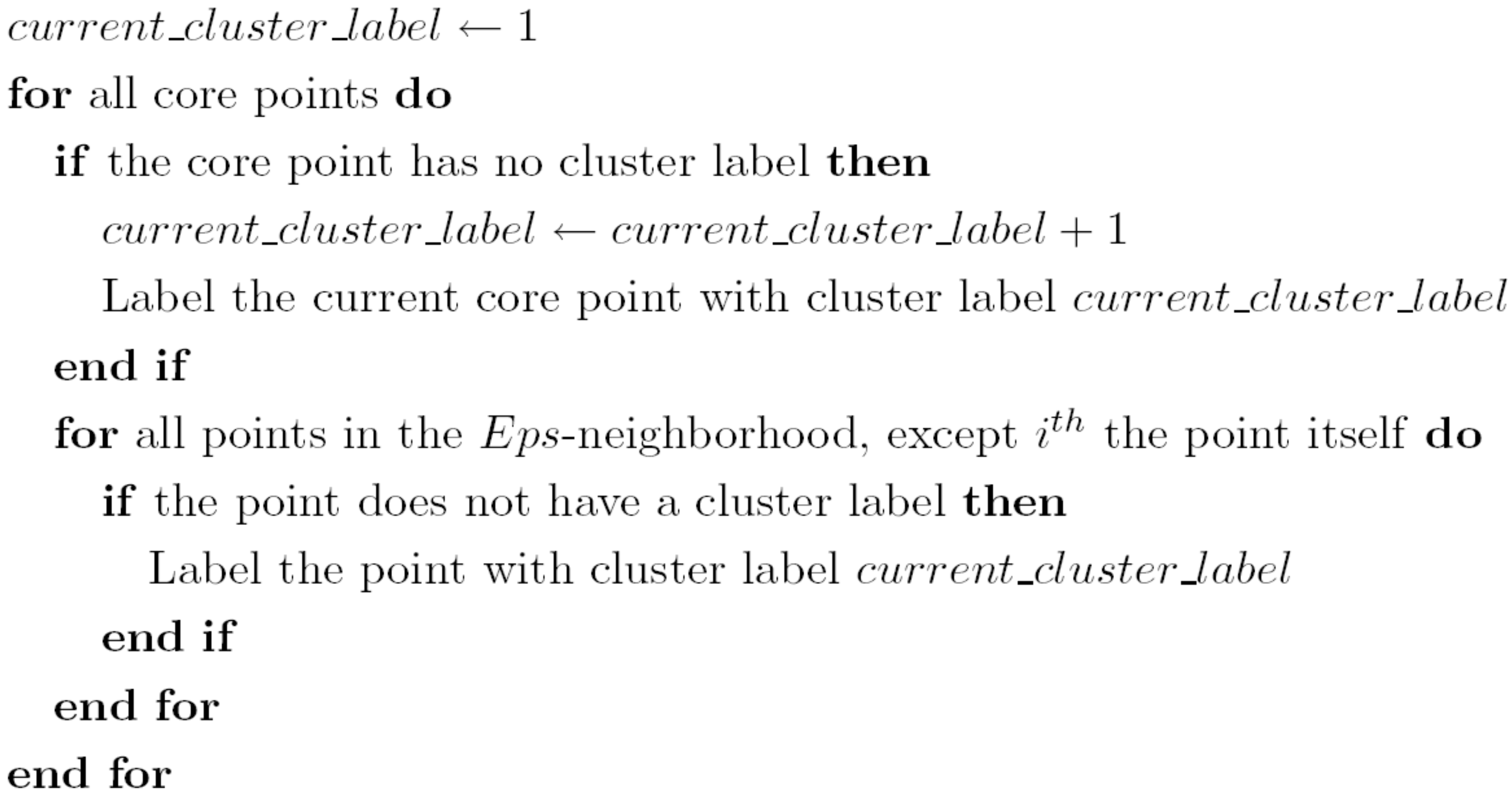
DBSCAN
Density Based Spatial Clustering

- 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

- The Algorithm
  - Arbitrary select a point p
  - Retrieve all points density-reachable from p given 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

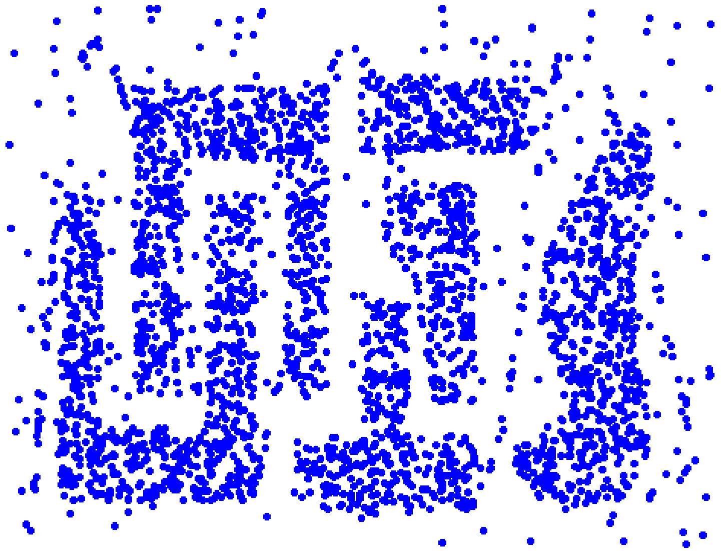
DBSCAN Algorithm

- ❑ Eliminate noise points
- ❑ Perform clustering on the remaining points

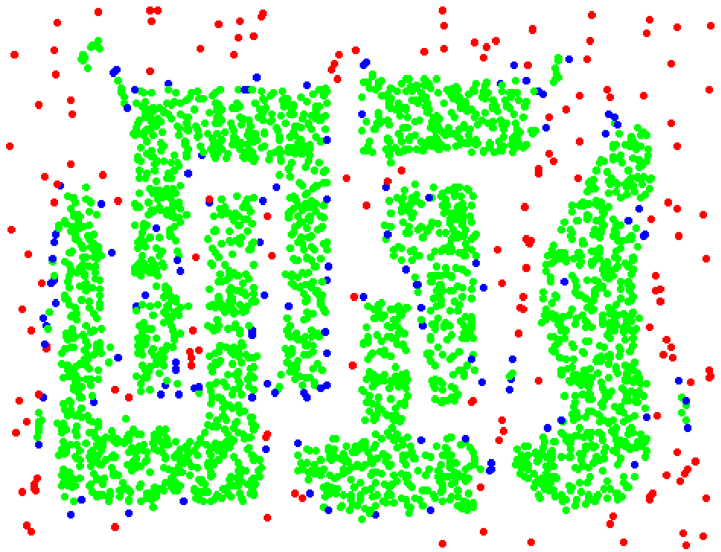


```
$current\_cluster\_label \leftarrow 1$
for all core points do
    if the core point has no cluster label then
        $current\_cluster\_label \leftarrow current\_cluster\_label + 1$
        Label the current core point with cluster label $current\_cluster\_label$
    end if
    for all points in the $Eps$-neighborhood, except $i^{th}$ the point itself do
        if the point does not have a cluster label then
            Label the point with cluster label $current\_cluster\_label$
        end if
    end for
end for
```

DBSCAN: Core, Border and Noise Points



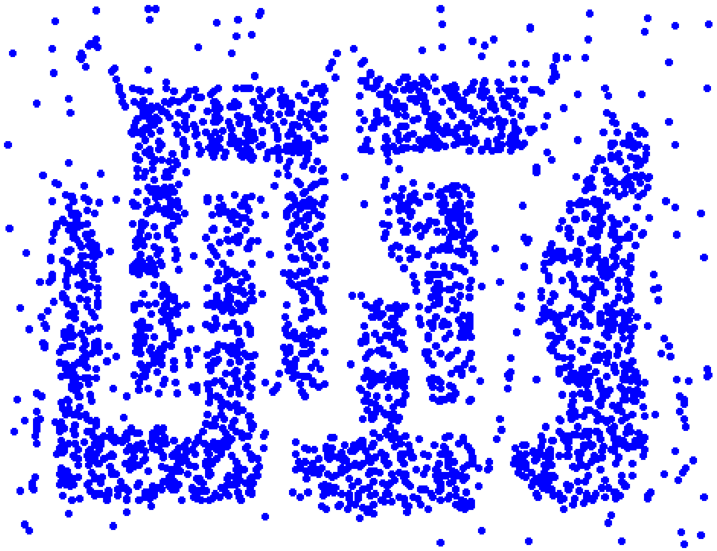
Original Points



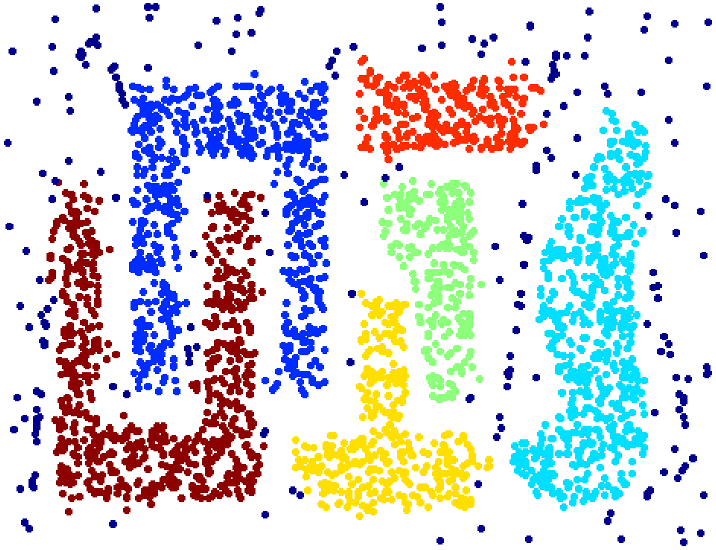
Point types: core, border and noise

Eps = 10, MinPts = 4

When DBSCAN Works Well



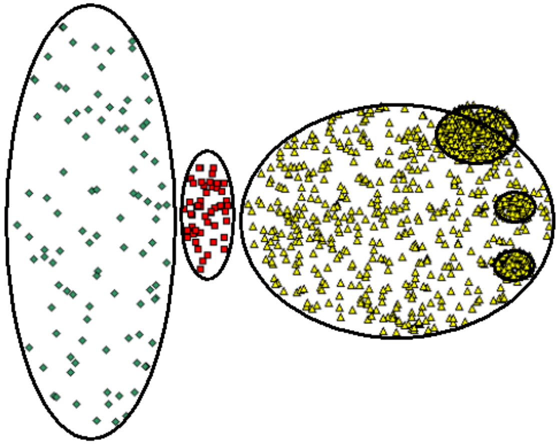
Original Points



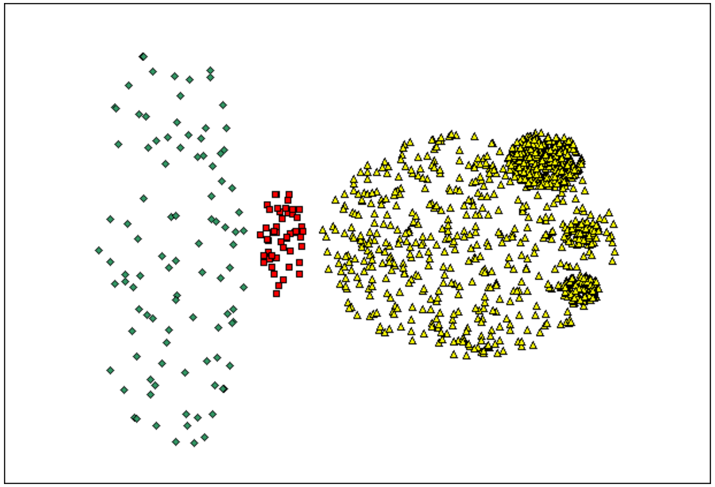
Clusters

- ❑ Resistant to Noise
- ❑ Can handle clusters of different shapes and sizes

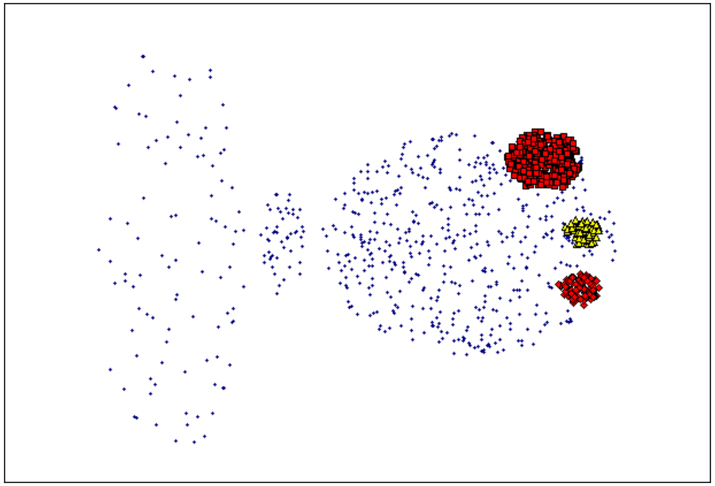
When DBSCAN Does NOT Work Well



Original Points



(MinPts=4, Eps=9.75).

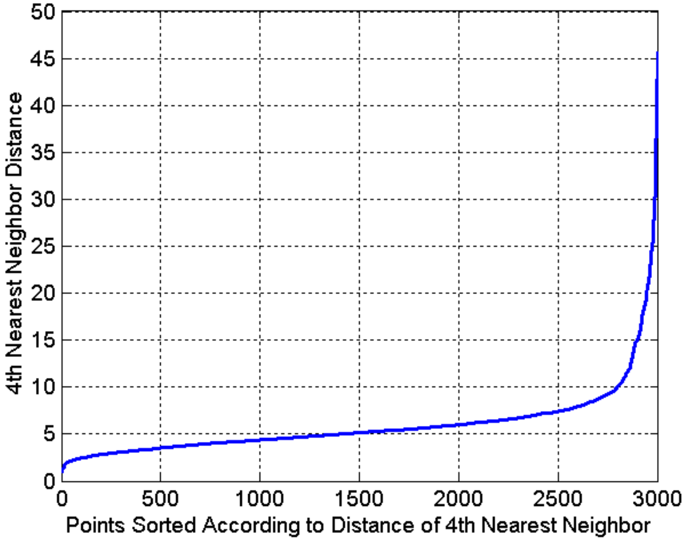


(MinPts=4, Eps=9.92)

- Varying densities
- High-dimensional data

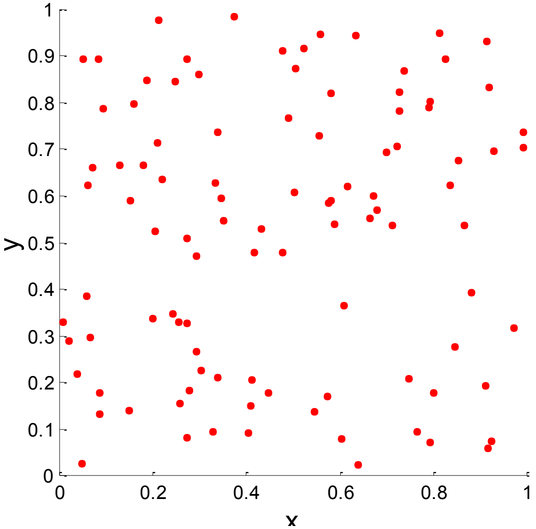
DBSCAN: Determining EPS and MinPts

- ❑ Idea is that for points in a cluster, their kth nearest neighbors are at roughly the same distance
- ❑ Noise points have the kth nearest neighbor at farther distance
- ❑ So, plot sorted distance of every point to its kth nearest neighbor

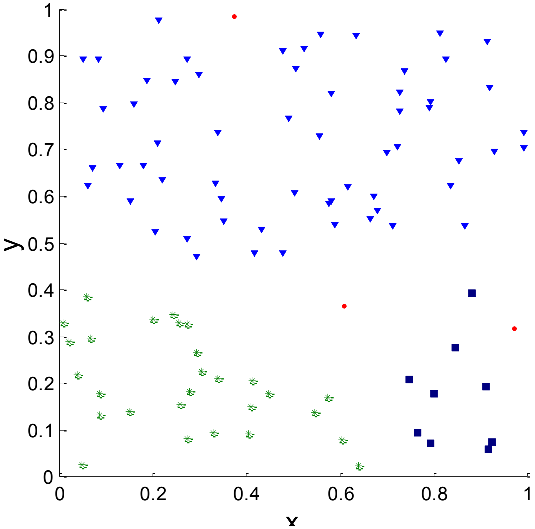


Clusters found in Random Data

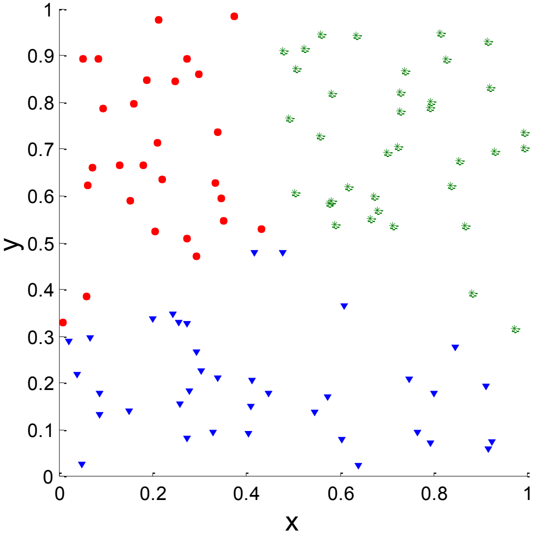
Random Points



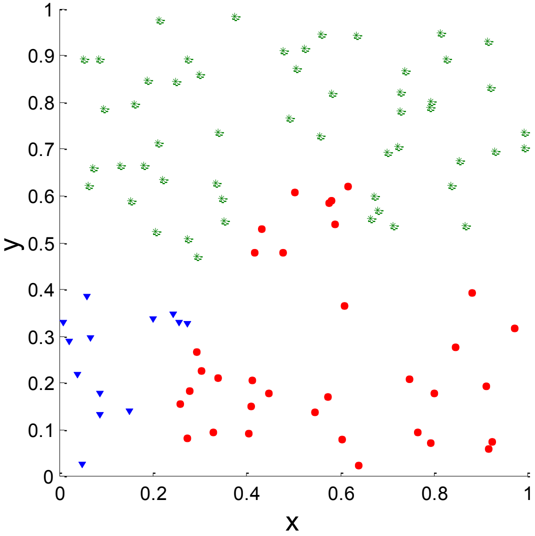
DBSCAN



K-means



Complete Link

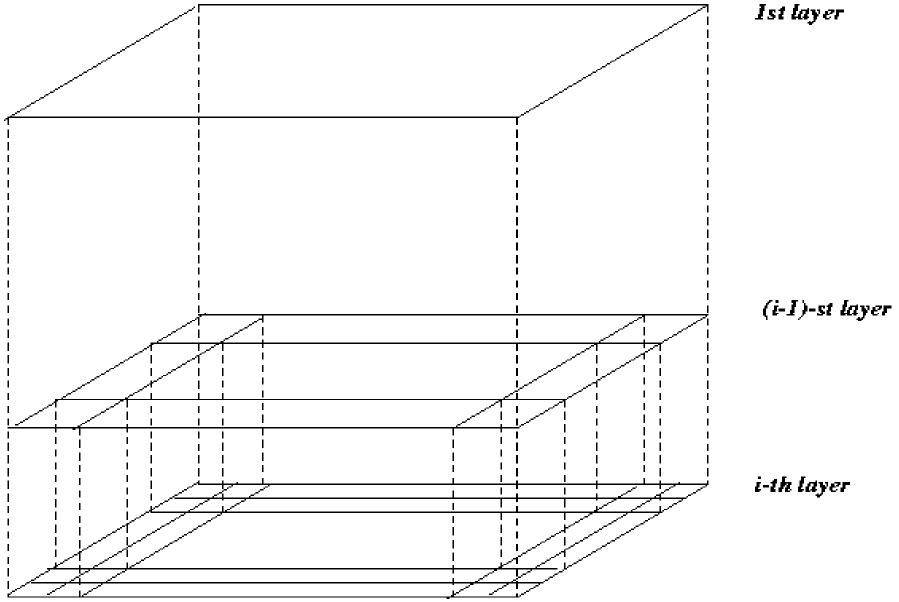


Grid-based clustering

Grid-Based Clustering Method

- Using multi-resolution grid data structure

- STING: A Statistical Information Grid Approach
  - The spatial area area is divided into rectangular cells
  - There are several levels of cells corresponding to different levels of resolution



The STING Clustering Method

- 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

Comments on STING

- 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

Model-based clustering

What is model-based clustering?

- The foundation of the probability-based clustering approach is based on a so-called finite mixture model.
- Based on the assumption that data are generated by a mixture of underlying probability distribution

- A mixture is a set of *k* probability distributions, each of which governs the attribute values distribution of a cluster.
- Attempt to optimize the fit between the data and some mathematical model

- Typical methods
  - Statistical approach
    EM (Expectation maximization), AutoClass
  - Machine learning approach
    COBWEB, CLASSIT
  - Neural network approach
    SOM (Self-Organizing Feature Map)

Expectation Maximization (EM) Clustering

- A popular iterative refinement algorithm
- An extension to k-means
  - Assign each object to a cluster according to a weight (probability distribution)
  - New means are computed based on weighted measures
- General idea
  - Start with an initial estimate of the parameter vector
  - Iteratively rescores the patterns against the mixture density produced by the parameter vector
  - The rescored patterns are used to update the parameter updates
  - Patterns belonging to the same cluster, if they are placed by their scores in a particular component
- Algorithm converges fast but may not be in global optima

The EM (Expectation Maximization) Algorithm

- Initially, randomly assign k cluster centers
- Iteratively refine the clusters based on two steps
  - Expectation step: assign each data point X_i to cluster C_i with the following probability

$$P(X_i \in C_k) = p(C_k|X_i) = \frac{p(C_k)p(X_i|C_k)}{p(X_i)},$$

    where $p(x_i|C_k)=N(m_k,E_k(x_i))$ follows the normal distribution. This step calculates the probability of cluster membership of x_i for each C_k

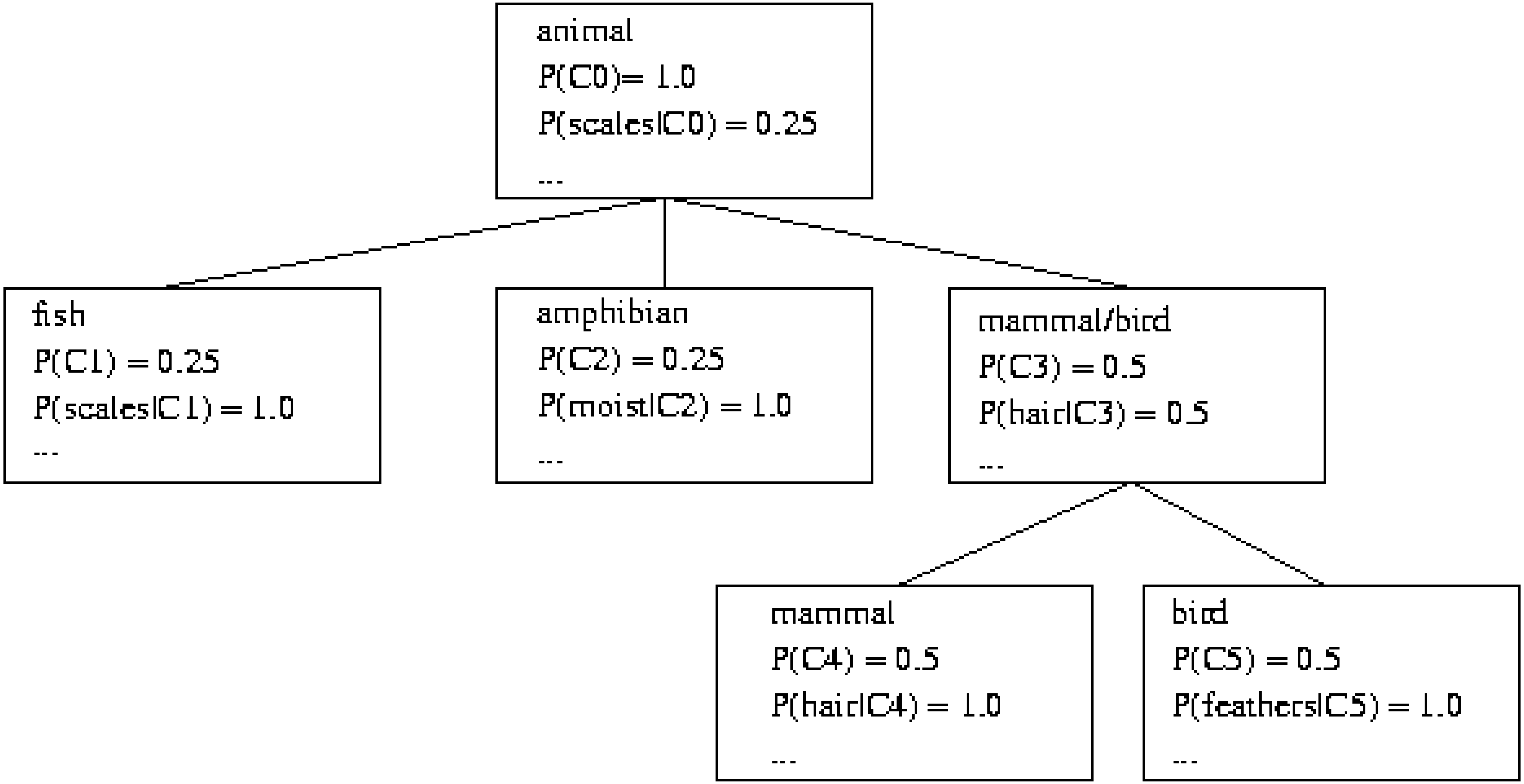
  - Maximization step: Estimation of model parameters

$$m_k = \frac{1}{N}\Sigma_{i=1}^{N}\frac{X_i P(X_i \in C_k)}{\Sigma_j P(X_i \in C_j)}.$$

Conceptual Clustering

- Conceptual clustering
  - A form of clustering in machine learning
  - Finds characteristic description for each concept (class)
  - Proposed as a means of discovering `understandable' patterns in data (Michalski 1980)
  - Yields a clustering dendrogram called classification tree that characterizes each cluster with a probabilistic description.
- COBWEB (Fisher'87)
  - A popular a simple method of incremental conceptual learning
  - Creates a hierarchical clustering in the form of a classification tree
  - Each node refers to a concept and contains a probabilistic description of that concept

COBWEB Clustering Method



COBWEB: The Category Utility Function

- The COBWEB algorithm operates based on the so-called category utility function (CU) that measures clustering quality.
- If we partition a set of objects into m clusters, then the CU of this particular partition is

$$\frac{\sum_{k=1}^{m} P(C_k)\left[\sum_i \sum_j P(A_i = V_{ij} \mid C_k)^2 - \sum_i \sum_j P(A_i = V_{ij})^2\right]}{m}$$

- For a given object in cluster C_k, if we guess its attribute values according to the probabilities of occurring, then the expected number of attribute values that we can correctly guess is

$$\sum_i \sum_j P(A_i = V_{ij} \mid C_k)^2$$

COBWEB: Operations

- The COBWEB algorithm constructs a classification tree incrementally by inserting the objects into the classification tree one by one.
- When inserting an object into the classification tree, the COBWEB algorithm traverses the tree top-down starting from the root node.
- At each node, the COBWEB algorithm considers 4 possible operations and select the one that yields the highest CU function value: insert, create, merge, split.

More on Conceptual Clustering

- Limitations of COBWEB
  - The assumption that the attributes are independent of each other is often too strong because correlation may exist
  - Not suitable for clustering large database data – skewed tree and expensive probability distributions
- CLASSIT
  - an extension of COBWEB for incremental clustering of continuous data
  - suffers similar problems as COBWEB
- AutoClass (Cheeseman and Stutz, 1996)
  - Uses Bayesian statistical analysis to estimate the number of clusters
  - Popular in industry

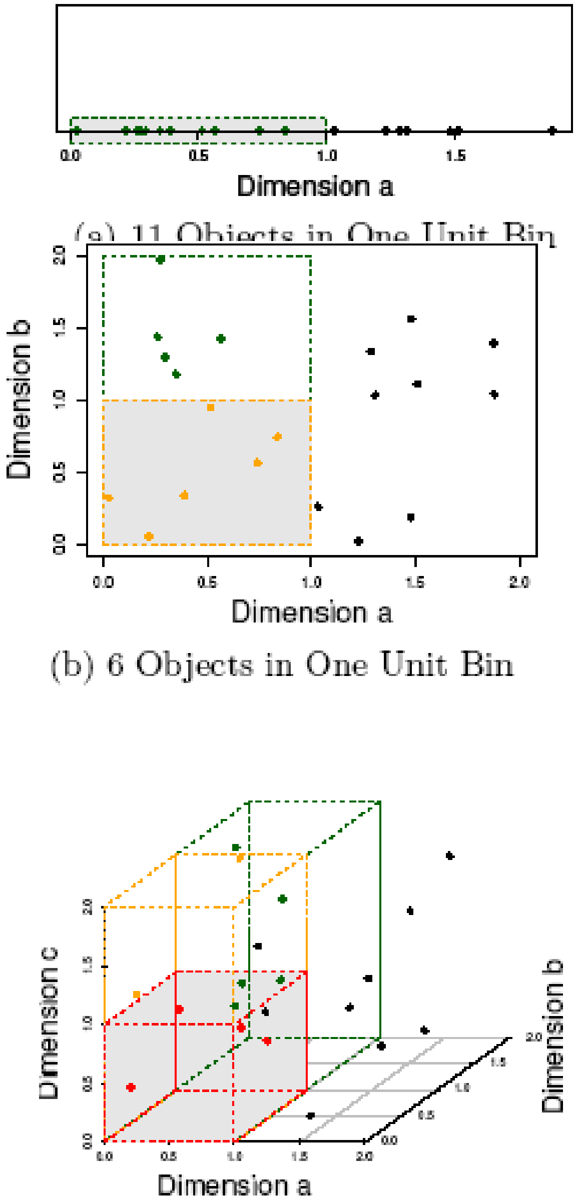
Curse of dimensionality

Clustering High-Dimensional Data

- Many applications: text documents, DNA micro-array data
- Major challenges:
  - Many irrelevant dimensions may mask clusters
  - Distance measure becomes meaningless due to equi-distance
  - Clusters may exist only in some subspaces
- Methods
  - Feature transformation: only effective if most dimensions are relevant: PCA & SVD useful only when features are highly correlated/redundant
  - Feature selection: wrapper or filter approaches, useful to find a subspace where the data have nice clusters
  - Subspace-clustering: find clusters in all the possible subspaces, e.g., CLIQUE, ProClus, and frequent pattern-based clustering

The Curse of Dimensionality

- ❑ Data in only one dimension is relatively packed
- ❑ Adding a dimension “stretch” the points across that dimension, making them further apart
- ❑ Adding more dimensions will make the points further apart—high dimensional data is extremely sparse
- ❑ Distance measure becomes meaningless
  due to equi-distance



(a) 11 Objects in One Unit Bin

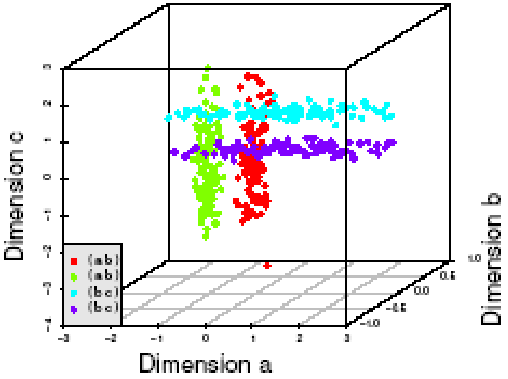
(b) 6 Objects in One Unit Bin

(c) 4 Objects in One Unit Bin

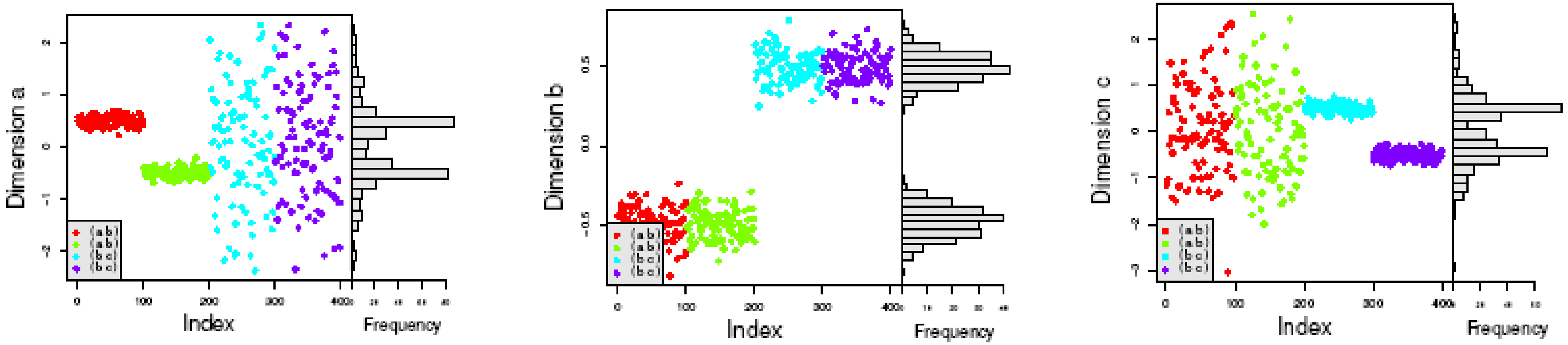
(Graphs adapted from Parsons et al. KDD Explorations 2004)

Why Subspace Clustering?

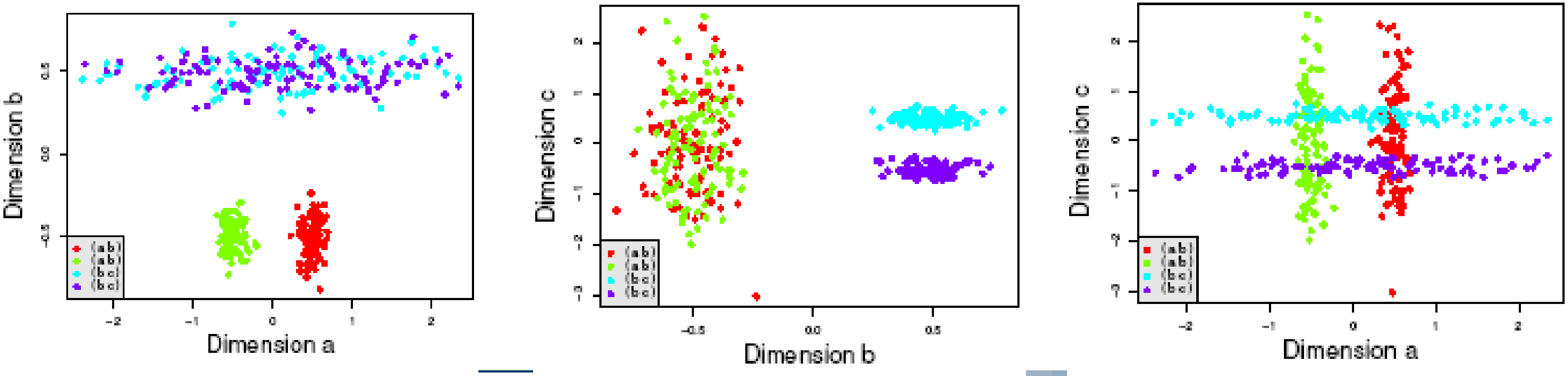
(adapted from Parsons et al. SIGKDD Explorations 2004)



- ❑ Clusters may exist only in some subspaces
- ❑ Subspace-clustering: find clusters in all the subspaces



(a) Dimension a (b) Dimension b (c) Dimension c



(a) Dims a & b (b) Dims b & c (c) Dims a & c

CLIQUE (Clustering In QUEst)

- 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

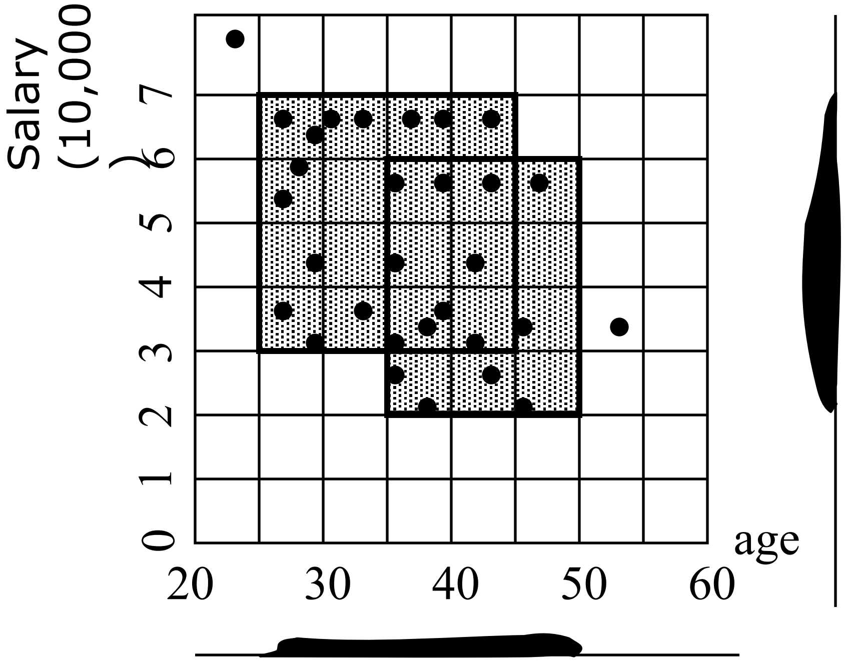
CLIQUE: The Major Steps

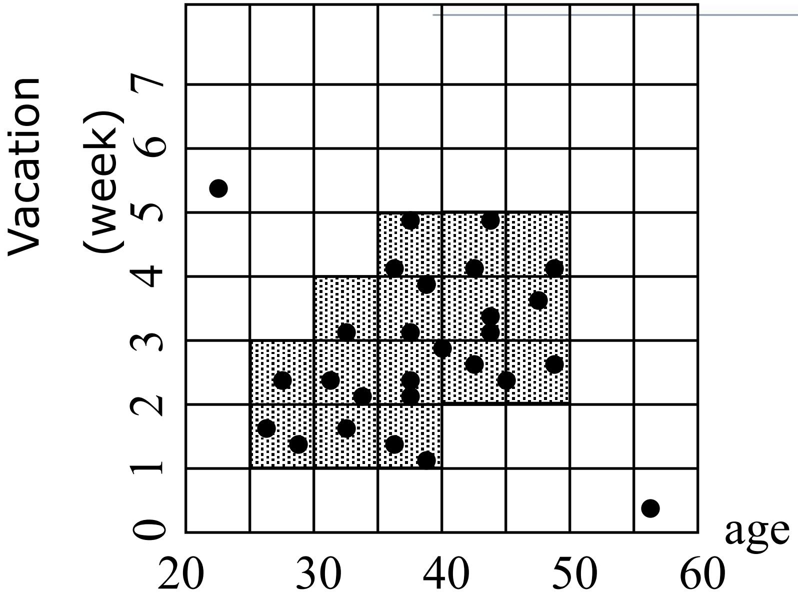
- Partition the data space and find the number of points that lie inside each cell of the partition.

- Identify the subspaces that contain clusters using the Apriori principle

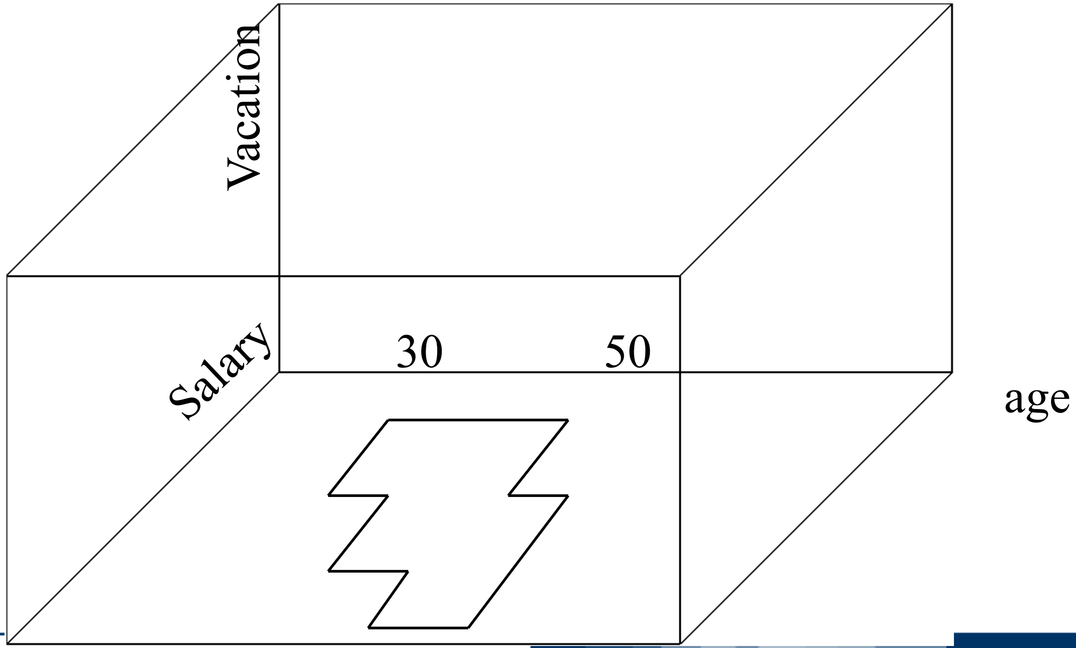
- Identify clusters
  - Determine dense units in all subspaces of interests
  - Determine connected dense units in all subspaces of interests.

- 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

Outliers

What Is Outlier Discovery?

- What are outliers?
  - The set of objects are considerably dissimilar from the remainder of the data
  - Example: Sports: Michael Jordon, Wayne Gretzky, ...
- Problem: Define and find outliers in large data sets
- Applications:
  - Credit card fraud detection
  - Telecom fraud detection
  - Customer segmentation
  - Medical analysis

Outlier Discovery: Statistical Approaches

- Assume a model underlying distribution that generates data set (e.g. normal distribution)

- Use discordancy tests depending on
  - data distribution
  - distribution parameter (e.g., mean, variance)
  - number of expected outliers

- Drawbacks
  - most tests are for single attribute
  - In many cases, data distribution may not be known

Outlier Discovery: Distance-Based Approach

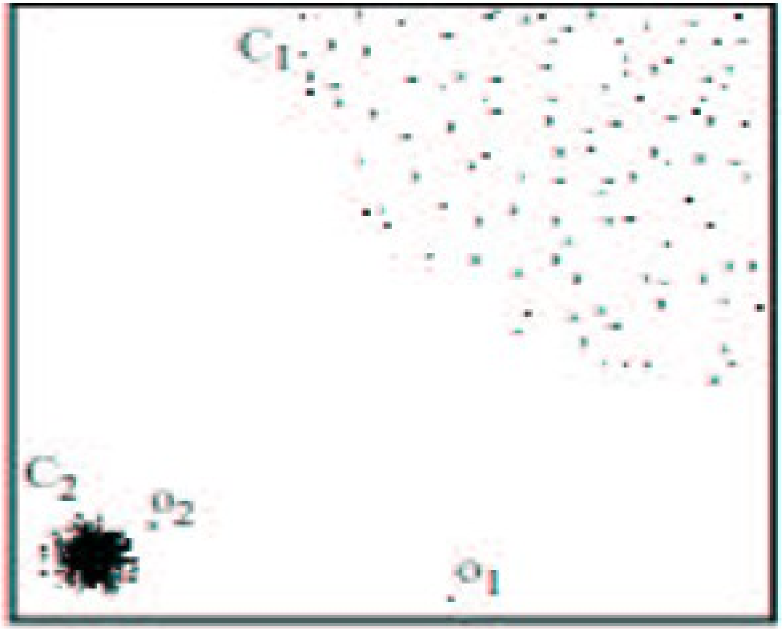
- ❑ Introduced to counter the main limitations imposed by statistical methods
- ❑ We need multi-dimensional analysis without knowing data distribution

- ❑ Distance-based outlier: A DB(p, D)-outlier is an object O in a dataset T such that at least a fraction p of the objects in T lies at a distance greater than D from O

- ❑ Algorithms for mining distance-based outliers
  - Index-based algorithm
  - Nested-loop algorithm
  - Cell-based algorithm

Density-Based Local Outlier Detection

- Distance-based outlier detection is based on global distance distribution
- It encounters difficulties to identify outliers if data is not uniformly distributed
- Ex. C1 contains 400 loosely distributed points, C2 has 100 tightly condensed points, 2 outlier points o1, o2
- Distance-based method cannot identify o2 as an outlier
- Need the concept of local outlier



Local outlier factor (LOF)

- Assume outlier is not crisp
- Each point has a LOF

Outlier Discovery: Deviation-Based Approach

- Identifies outliers by examining the main characteristics of objects in a group
- Objects that “deviate” from this description are considered outliers
- Sequential exception technique
  - simulates the way in which humans can distinguish unusual objects from among a series of supposedly like objects
- OLAP data cube technique
  - uses data cubes to identify regions of anomalies in large multidimensional data

Summary

Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is

- For cluster analysis, the analogous question is how to evaluate the "goodness" of the resulting clusters?

- But "clusters are in the eye of the beholder"!

- Then why do we want to evaluate them?
  - To avoid finding patterns in noise
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To compare two clusters

Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following three types.
  - External Index: Used to measure the extent to which cluster labels match externally supplied class labels (Entropy)
  - Internal Index: Used to measure the goodness of a clustering structure without respect to external information (SSE)
  - Relative Index: Used to compare two different clusterings or clusters (Often an external or internal index)

- Sometimes these are referred to as criteria instead of indices
  - However, sometimes criterion is the general strategy and index is the numerical measure that implements the criterion.

Final Comment on Cluster Validity

"The validation of clustering structures is the most difficult and frustrating part of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage."

Algorithms for Clustering Data, Jain and Dubes

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, density-based methods, grid-based methods, and model-based methods
- ❑ Outlier detection and analysis are very useful for fraud detection, etc. and can be performed by statistical, distance-based or deviation-based approaches
- ❑ There are still lots of research issues on cluster analysis

Problems and Challenges

- Considerable progress has been made in scalable clustering methods
    - Partitioning: k-means, k-medoids, CLARANS
    - Hierarchical: BIRCH, ROCK, CHAMELEON
    - Density-based: DBSCAN, OPTICS, DenClue
    - Grid-based: STING, WaveCluster, CLIQUE
    - Model-based: EM, Cobweb, SOM
    - Frequent pattern-based: pCluster
    - Constraint-based: COD, constrained-clustering

- Current clustering techniques do not address all the requirements adequately, still an active area of research