Clustering Techniques for Large Data Sets

From the Past to the Future

Alexander Hinneburg, Daniel A. Keim
University of Halle

Introduction - Preliminary Remarks

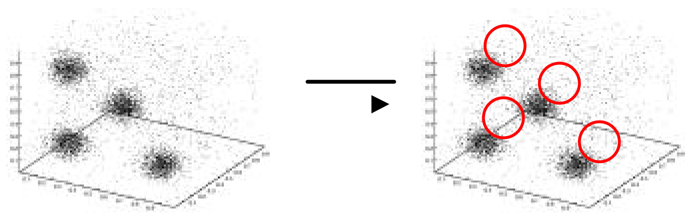
Problem: Analyze a (large) set of objects and form a smaller number of groups using the similarity and factual closeness between the objects.

Goals:

- Finding representatives for homogenous groups -> ***Data Reduction***
- Finding "natural" clusters and describe their unknown properties -> ***"natural" Data Types***
- Find useful and suitable groupings -> ***"useful" Data Classes***
- Find unusual data objects -> ***Outlier Detection***

Introduction - Preliminary Remarks

- Examples:
  - Plant / Animal classification
  - Book ordering
  - Sizes for clothing
  - Fraud detection



Introduction - Preliminary Remarks

- Goal: ***objective*** instead of ***subjective*** Clustering
- Preparations:
  - Data Representation
    - Feature Vectors, real / categorical values
    - Strings, Key Words
  - Similarity Function, Distance Matrix

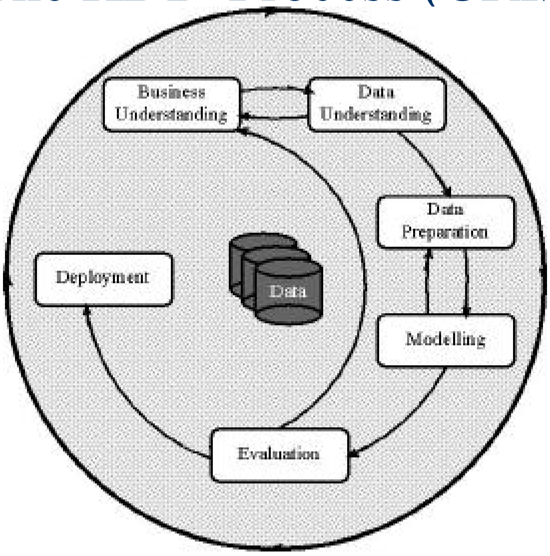
Introduction

- **Application Example:** Marketing
  - Given:
    - Large data base of customer data containing
      their properties and past buying records
  - Goal:
    - Find groups of customers with similar behavior
    - Find customers with unusual behavior

Introduction

- **Application Example:**
  Class Finding in CAD-Databases
  - Given:
    - Large data base of CAD data containing abstract feature vectors (Fourier, Wavelet, ...)
  - Goal:
    - Find homogeneous groups of similar CAD parts
    - Determine standard parts for each group
    - Use standard parts instead of special parts
      (→ reduction of the number of parts to be produced)

The KDD-Process (CRISP)



Hinneburg / Keim, PKDD 2000

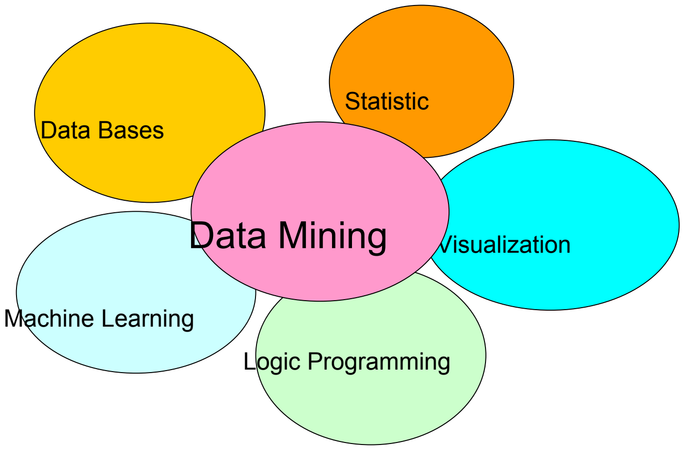
Data Mining vs. Statistic

- Algorithms scale to large data sets
- Data is used secondary for Data mining
- DM-Tools are for End-User with Background
- Strategy:
  - explorative
  - cyclic

- Many Algorithms with quadratic run-time
- Data is made for the Statistic (primary use)
- Statistical Background is often required
- Strategy:
  - conformational,
  - verifying
  - few loops

Hinneburg / Keim, PKDD 2000

Data Mining, an inter-disciplinary Research Area

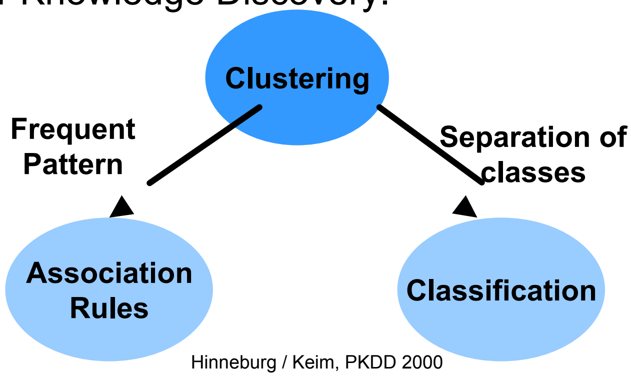


Introduction

- Related Topics
  - Unsupervised Learning (AI)
  - Data Compression
  - Data Analysis / Exploration

Role of Clustering in the KDD Process

- Clustering is beside Classification and Association Rules Mining a basic technique for Knowledge Discovery.



Hinneburg / Keim, PKDD 2000

Introduction

Problem Description

- Given:

  A data set of *N* data items with each have a *d*-dimensional data feature vector.

- Task:

  Determine a natural, useful partitioning of the data set into a number of clusters (k) and noise.

Hinneburg / Keim, PKDD 2000

Introduction

From the Past ...

- Clustering is a well-known problem in statistics [Sch 64, Wis 69, DH 73, Fuk 90]
- more recent research in
  – machine learning [Roj 96],
  – databases [CHY 96], and
  – visualization [Kei 96] ...

Introduction

... to the Future

- *Effective* and *efficient* clustering algorithms for *large high-dimensional* data sets with *high noise level*
- Requires ***Scalability*** with respect to
  – the *number of data points (**N**)*
  – the *number of dimensions (**d**)*
  – the *noise level*
- New Understanding of Problems

Overview (First Lesson)

1. Introduction
2. Clustering Methods — **From the Past ...**
   - 2.1 Model- and Optimization-based Approaches
   - 2.2 Linkage-based Methods / Linkage Hierarchies
   - 2.3 Density-based Approaches
   - 2.4 Categorical Clustering — **... to the Future**
3. Techniques for Improving the Efficiency
4. Recent Research Topics
5. Summary and Conclusions

Model-based Approaches

- Optimize the parameters for a given model

Statistic

Statistic / KDD

- K-Means/LBG
- CLARANS
- EM
- LBG-U
- K-Harmonic Means

Artificial Intelligence

- Kohonen Net/ SOM
- Neural Gas/ Hebb Learning
- Growing Networks

Model-based Methods: Statistic/KDD

- K-Means [Fuk 90]
- Expectation Maximization [Lau 95]
- CLARANS [NH 94]
- Foccused CLARANS [EKX 95]
- LBG-U [Fri 97]
- K-Harmonic Means [ZHD 99, ZHD 00]

K-Means / LBG [Fuk 90, Gra 92]

- Determine *k* prototypes (*p)* of a given data set
- Assign data points to nearest prototype

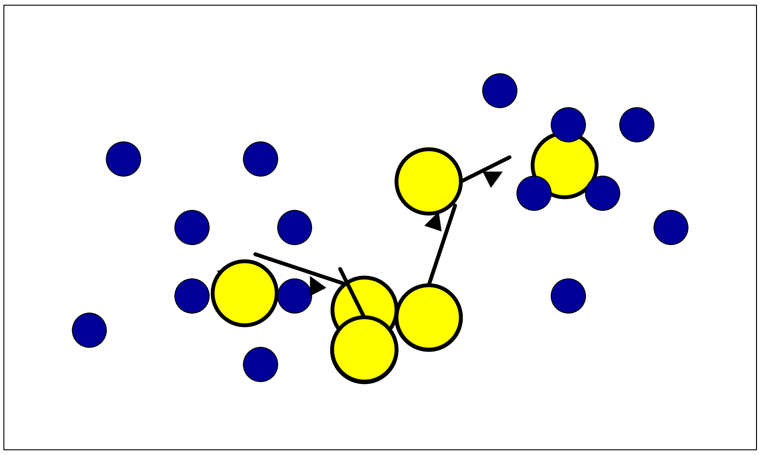
$$p \rightarrow R_p \text{ Voronoi-Set}$$

- Minimize distance criterion:

$$E(D,P) = 1/|D| \sum_{p \in P} \sum_{x \in R_p} dist(p,x)^2$$

- Iterative Algorithm
  - Shift the prototypes towards the mean of their point set
  - Re-assign the data points to the nearest prototype

K-Means: Example



Hinneburg / Keim, PKDD 2000

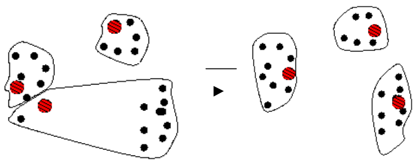
Expectation Maximization [Lau 95]

- Generalization of k-Means
  (➔ probabilistic assignment of points to clusters)
- Baisc Idea:
  - Estimate parameters of k Gaussians
  - Optimize the probability, that the mixture of parameterized Gaussians fits the data
  - Iterative algorithm similar to k-Means

Hinneburg / Keim, PKDD 2000

CLARANS [NH 94]

- Medoid Method:
  - Medoids are special data points
  - All data points are assigned to the nearest medoid



- Optimization Criterion:

$$average\_distance(c) = \sum_{m_i \in M} \sum_{o \in cluster(m_i)} dist(o, m_i)$$

Bounded Optimization [NH 94]

- CLARANS uses two bounds to restrict the optimization: *num_local, max_neighbor*
- Impact of the Parameters:
  - *num_local* ‣ Number of iterations
  - *max_neighbors* ‣ Number of tested neighbors per iteration

CLARANS

- Graph Interpretation:
  - Search process can be symbolized by a graph
  - Each node corresponds to a specific set of medoids
  - The change of one medoid corresponds to a jump to a neighboring node in the search graph
- Complexity Considerations:
  - The search graph has $\binom{N}{k}$ nodes and each node has *N*k* edges
  - The search is bound by a fixed number of jumps *(num_local)* in the search graph
  - Each jump is optimized by randomized search and costs *max_neighbor* scans over the data (to evaluate the cost function)

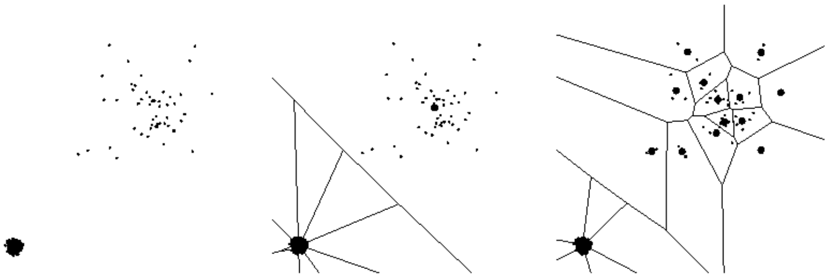
LBG-U [Fri 97]

$$\text{Utility } U(p) = E(D, P \setminus \{p\}) - E(D, P)$$
$$= \sum_{x \in R_p} dist(p_2, x) - dist(p, x)$$

$$\text{Quantization Error } E(p) = 1 / R_P \sum_{x \in R_p} dist(x, p)$$

- Pick the prototype p with min. Utility and set it near the prototype p' with max. Quantization Error .
- Run LBG again until convergence

LBG-U: Example



a) Datenmenge $\mathcal{D}$ mit 1000 Elementen

b) LBG-Kodebuch, $\sqrt{\text{MSE}}$: 0.001

c) „besseres“ Kodebuch, $\sqrt{\text{MSE}}$: 0.00009

Hinneburg / Keim, PKDD 2000

K-Harmonic Means [ZHD 99]

- Different Optimization Function:

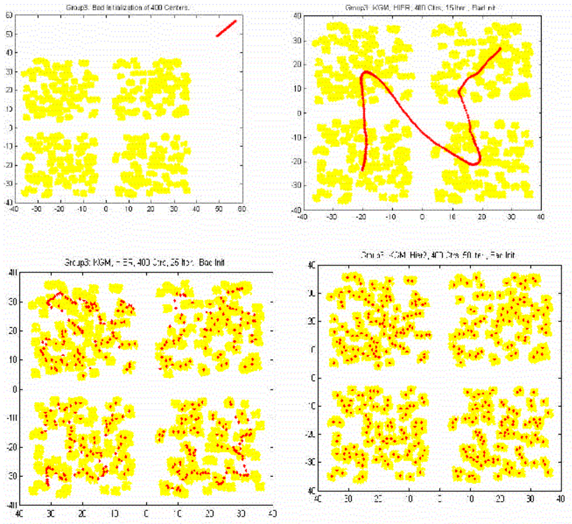
$$Perf_{KHM}(\{x_i\}_{i=1}^{N},\{m_l\}_{l=1}^{K}) = \sum_{i=1}^{N} \frac{K}{\sum_{l=1}^{K} \frac{1}{\| x_i - m_l \|^2}}$$

- Update Formula for Prototypes:

$$m_k = \frac{\sum_{i=1}^{N} \frac{1}{d_{i,k}^{\;3} \left(\sum_{l=1}^{K} \frac{1}{d_{i,l}^{\;2}}\right)^2} x_i}{\sum_{i=1}^{N} \frac{1}{d_{i,k}^{\;3} \left(\sum_{l=1}^{K} \frac{1}{d_{i,l}^{\;2}}\right)^2}} \qquad d_{i,j} = dist(x_i, m_j)$$

Hinneburg / Keim, PKDD 2000

K-Harmonic Means [ZHD 99]



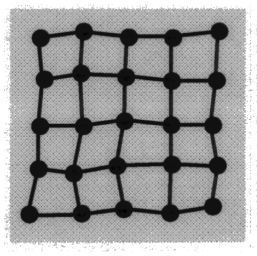
Model-based Methods: AI

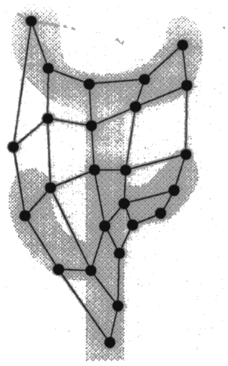
- Online Learning vs. Batch Learning
- Self-Organizing Maps [KMS+ 91, Roj 96]
- Neural Gas & Hebb. Learning [MBS 93, Fri 96]
- Growing Networks [Fri 95]

Hinneburg / Keim, PKDD 2000

Self Organizing Maps

- Self-Organizing Maps [Roj 96, KMS 91]
  - Fixed map topology (grid, line)

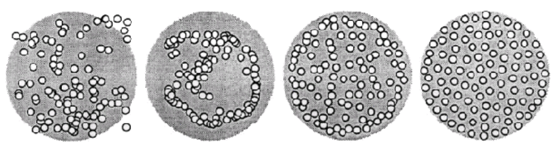




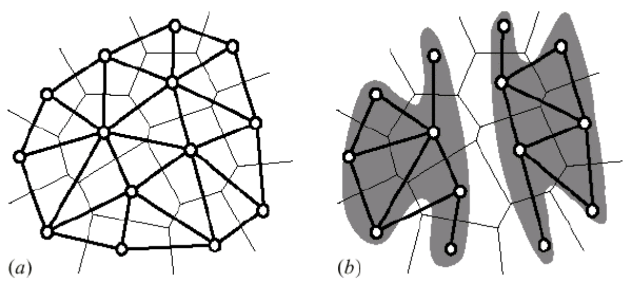
Hinneburg / Keim, PKDD 2000

Neural Gas / Hebb Learning [MBS 93, Fri 96]

- Neural Gas:



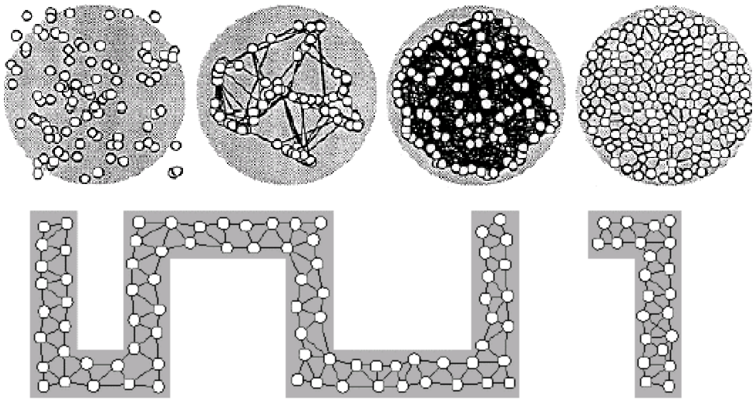
- Hebbian Learning:



Hinneburg / Keim, PKDD 2000

Neural Gas / Hebb Learning
[MBS 93, Fri 96]

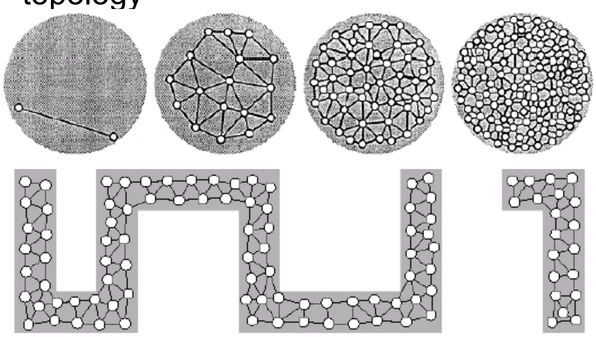
- Neural Gas & Hebbian Learning:



Hinneburg / Keim, PKDD 2000

Growing Networks

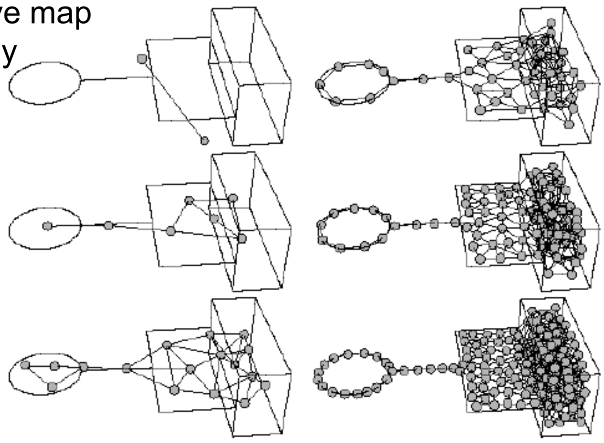
- Growing Networks [Fri 95]
  - Iterative insertion of nodes
  - Adaptive map topology



Hinneburg / Keim, PKDD 2000

Growing Networks

- Growing Networks [Fri 95]
  - Adaptive map topology



Hinneburg / Keim, PKDD 2000

Linkage-based Methods

- Hierarchical Methods:
  - Single / Complete / Centroid Linkage
  - BIRCH [ZRL 96]
- Graph Partitioning based Methods:
  - Single Linkage
  - Method of Wishart
  - DBSCAN
  - DBCLASD

Hinneburg / Keim, PKDD 2000

Linkage Hierarchies [Bok 74]

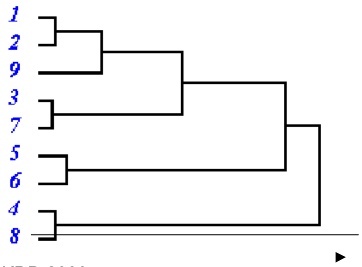
- Single Linkage (Minimum Spanning Tree)
- Complete Linkage
- Average Linkage
- Centroid Linkage (see also BIRCH)

Top-down (Dividing):

- Find the most inhomogeneous cluster and split

Bottom-up (Agglomerating):

- Find the nearest pair of clusters and merge



Single Linkage

- Distance between clusters (nodes):

$$Dist(C_1, C_2) = \min_{p \in C_1, q \in C_2} \{dist(p, q)\}$$

- Merge Step:
  Union of two subset of data points
- A single linkage hierarchy can be constructed using the Minimal Spanning Tree

Complete Linkage

- Distance between clusters (nodes):
  $$Dist(C_1, C_2) = \max_{p \in C_1, q \in C_2} \{dist(p,q)\}$$
- Merge Step:
  Union of two subset of data points
- Each cluster in a complete linkage hierarchy corresponds to a complete subgraph

Hinneburg / Keim, PKDD 2000

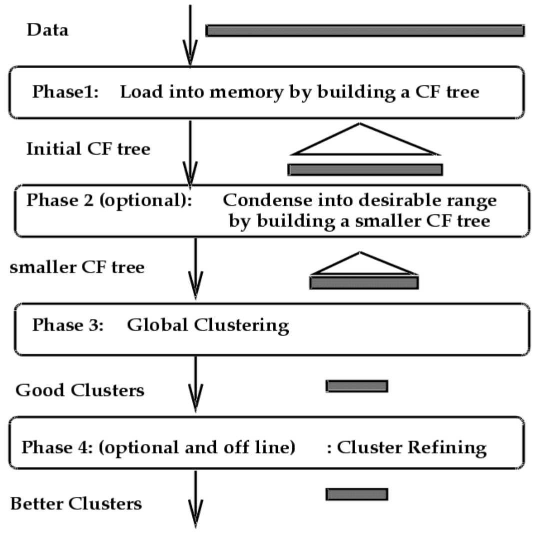
Average Linkage / Centroid Method

- Distance between clusters (nodes):
  $$Dist_{avg}(C_1, C_2) = \frac{1}{\#(C_1) \cdot \#(C_2)} \sum_{p \in C_1} \sum_{p \in C_2} dist(p,q)$$
  $$Dist_{mean}(C_1, C_2) = dist[mean(C_1), mean(C_2)]$$
- Merge Step:
  - union of two subset of data points
  - construct the mean point of the two clusters

Hinneburg / Keim, PKDD 2000

BIRCH [ZRL 96]

Clustering in BIRCH



BIRCH

Basic Idea of the CF-Tree

- Condensation of the data $\{\vec{X_i}\}$ using CF-Vectors $\mathbf{CF} = (N, \vec{LS}, SS)$

$$\vec{LS} = \sum_{i=1}^{N} \vec{X_i}, \; SS = \sum_{i=1}^{N} \vec{X_i}^{\,2}$$

- CF-tree uses sum of CF-vectors to build higher levels of the CF-tree

BIRCH

Insertion algorithm for a point x:

(1) Find the closest leaf b

(2) If x fits in b, insert x in b;
otherwise split b

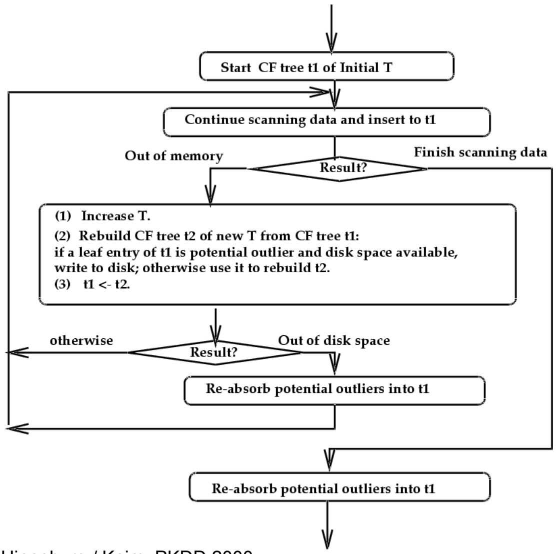
(3) Modify the path for b

(4) If tree is to large, condense the tree
by merging the closest leaves

Hinneburg / Keim, PKDD 2000

BIRCH

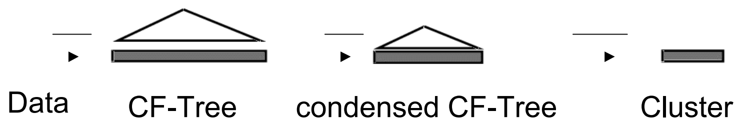
CF-Tree Construction



Hinneburg / Keim, PKDD 2000

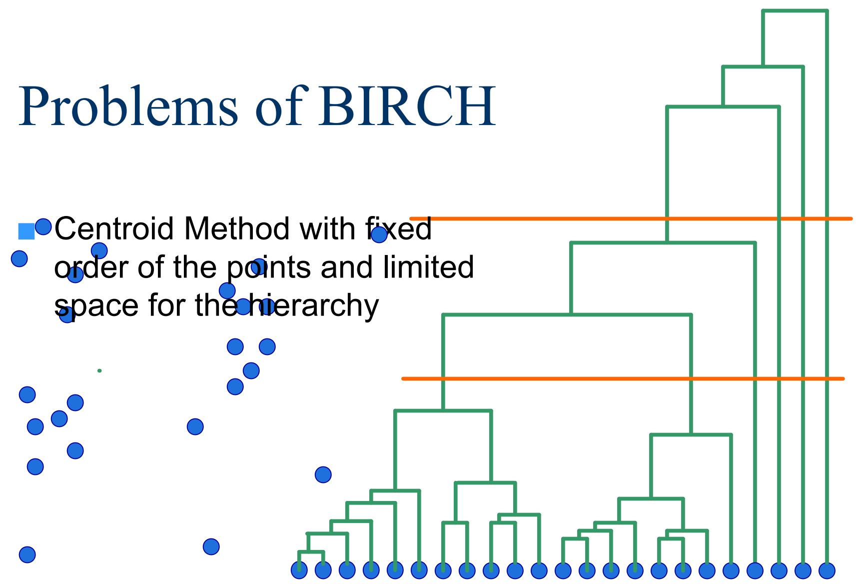
Condensing Data

- BIRCH [ZRL 96]:
  - Phase 1-2 produces a condensed representation of the data (CF-tree)
  - Phase 3-4 applies a separate cluster algorithm to the leafs of the CF-tree
- Condensing data is crucial for efficiency



Problems of BIRCH

- Centroid Method with fixed order of the points and limited space for the hierarchy

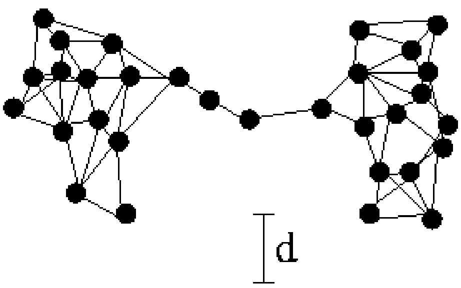


Linkage-based Methods

- Hierarchical Methods:
  - Single / Complete / Centroid Linkage
  - BIRCH [ZRL 96]
- Graph Partitioning based Methods:
  - Single Linkage [Boc 74]
  - Method of Wishart [Wis 69]
  - DBSCAN [EKS+ 96]
  - DBCLASD [XEK+ 98]

Linkage -based Methods
(from Statistics) [Boc 74]

- Single Linkage (Connected components for distance d)



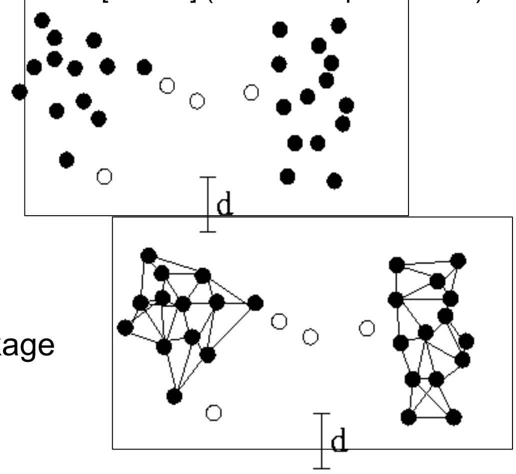
- Single Linkage + additional Stop Criterion

Linkage -based Methods [Boc 74]

- Method of Wishart [Wis 69] (Min. no. of points: c=4)

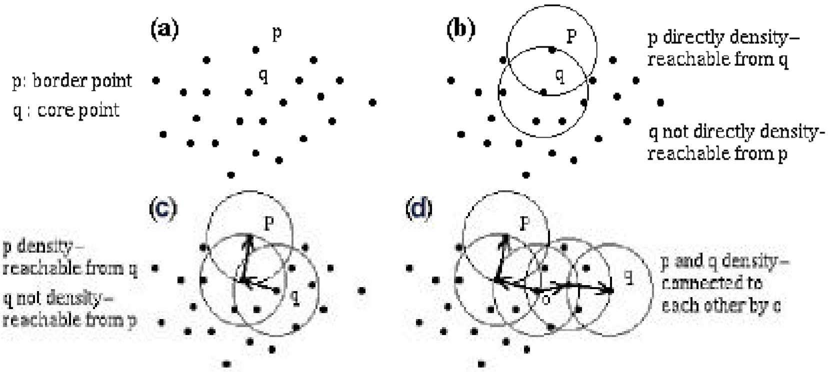
Reduce data set

Apply Single Linkage



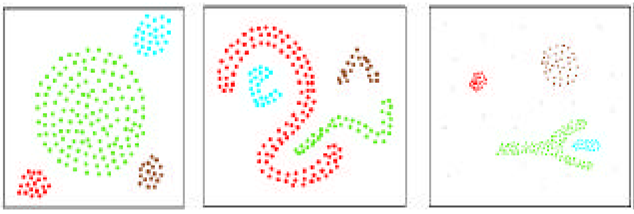
DBSCAN [EKS+ 96]

- Clusters are defined as Density-Connected Sets (wrt. MinPts, ε)



DBSCAN

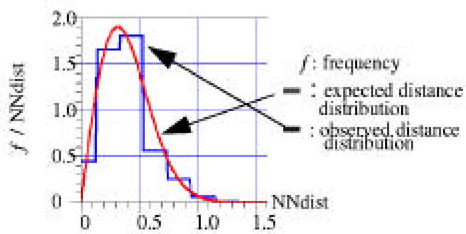
- For each point, DBSCAN determines the ε-environment and checks, whether it contains more than MinPts data points
- DBSCAN uses index structures for determining the ε-environment
- Arbitrary shape clusters found by DBSCAN



Hinneburg / Keim, PKDD 2000

DBCLASD [XEK+ 98]

- Distribution-based method
- Assumes arbitrary-shape clusters of uniform distribution
- Requires no parameters



The expected and the observed distance distributions for cluster 1

Hinneburg / Keim, PKDD 2000

DBCLASD

- Definition of a cluster *C* based on the distribution of the NN-distance *(NNDistSet)*:

  (1) *NNDistSet(C)* has the expected distribution with a required confidence level.

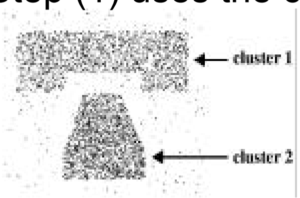
  (2) *C* is *maximal*, i.e. each extension of C by neighboring points does not fulfill condition (1). (maximality).

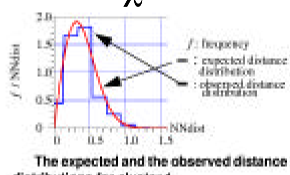
  (3) *C* is *connected*, i.e. for each pair of points (a,b) of the cluster there is a path of occupied grid cells connecting a and b (connectivity).

Hinneburg / Keim, PKDD 2000

DBCLASD

- Step (1) uses the concept of the χ^2-test





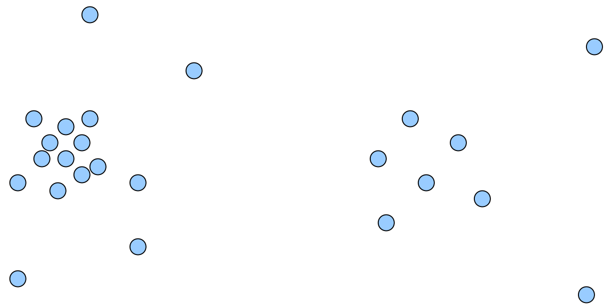
The expected and the observed distance distributions for cluster 1

- Incremental augmentation of clusters by neighboring points (order-depended)
  - unsuccessful candidates are tried again later
  - points already assigned to some cluster may switch to another cluster

Hinneburg / Keim, PKDD 2000

Linkage-based Methods

- Single Linkage + additional Stop Criteria describes the border of the Clusters



Hinneburg / Keim, PKDD 2000

OPTICS[ABK+ 99]

- DBSCAN with variable e, $0 \le e \le e_{MAX}$
- The Result corresponds to the Bottom of a hierarchy
- Ordering:
  - Reachability Distance:

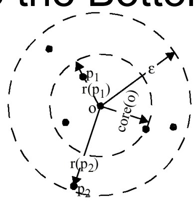


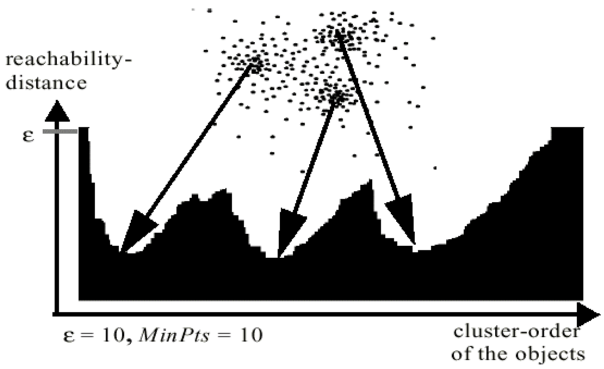
Figure 4. Core-distance(o), reachability-distances r(p_1,o), r(p_2,o) for *MinPts*=4

$$reach-dist(p,o) = \begin{cases} Undefined, if \left|N_{e_{MAX}}(o)\right| < MinPTS \\ \max\{core-dist(o), dist(o,p)\}, else \end{cases}$$

Hinneburg / Keim, PKDD 2000

OPTICS[ABK+ 99]

- Breath First Search with Priority Queue



Hinneburg / Keim, PKDD 2000

DBSCAN / DBCLASD/ OPTICS

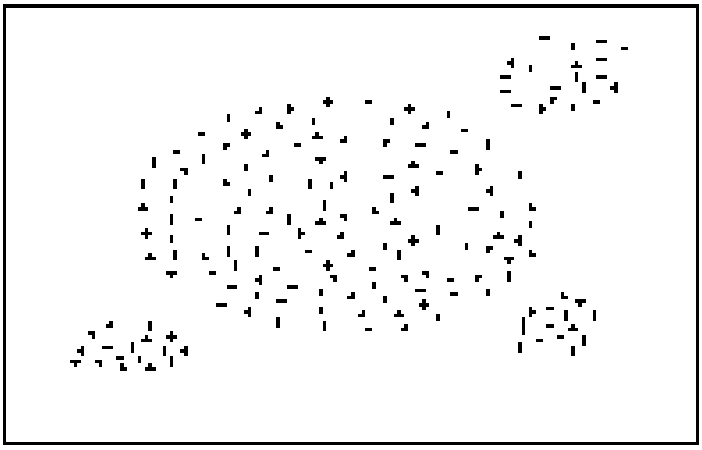
- DBSCAN / DBCLASD / OPTICS use index structures to speed-up the ε-environment or nearest-neighbor search
- the index structures used are mainly the R-tree and variants

Hinneburg / Keim, PKDD 2000

Density-based Methods

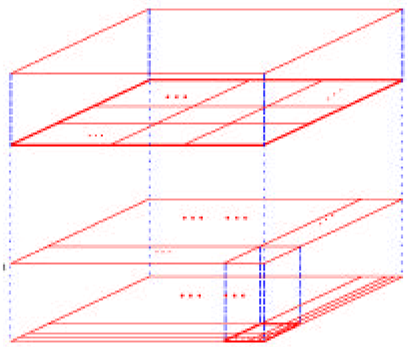
- Kernel-Density Estimation [Sil 86]
- STING [WYM 97]
- Hierarchical Grid Clustering [Sch 96]
- WaveCluster [SCZ 98]
- DENCLUE [HK 98]

Point Density



STING [WYM 97]

- Uses a quadtree-like structure for condensing the data into grid cells
  - The nodes of the quadtree contain statistical information about the data in the corresponding cells
  - STING determines clusters as the density-connected components of the grid
  - STING approximates the clusters found by DBSCAN



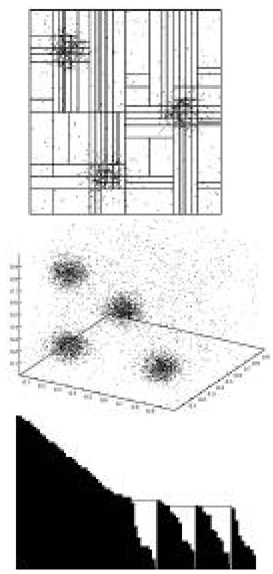
Hinneburg / Keim, PKDD 2000

Hierarchical Grid Clustering [Sch 96]

- Organize the data space as a grid-file
- Sort the blocks by their density

$$DB = \frac{p_B}{V_B} \quad \blacktriangleright \quad \langle B_{1'}, B_{2'}, \ldots B_{b'} \rangle$$

- Scan the blocks iteratively and merge blocks, which are adjacent over a (d-1)-dim. hyperplane.
- The order of the merges forms a hierarchy



Hinneburg / Keim, PKDD 2000

WaveCluster [SCZ 98]

- Clustering from a signal processing perspective using wavelets

Input: Multidimensional data objects' feature vectors
Output: clustered objects

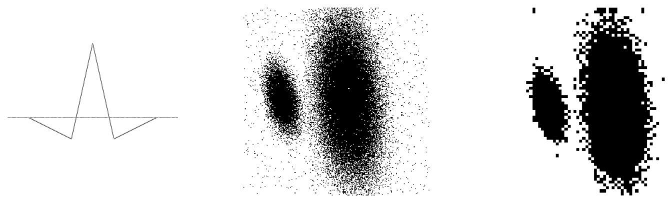
1. Quantize feature space, then assign objects to the units.
2. Apply wavelet transform on the feature space.
3. Find the connected components (clusters) in the subbands of transformed feature space, at different levels.
4. Assign label to the units.
5. Make the lookup table.
6. Map the objects to the clusters.

WaveCluster

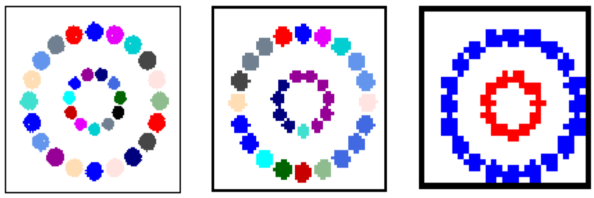
- Grid Approach
  - Partition the data space by a grid → reduce the number of data objects by making a small error
  - Apply the wavelet-transformation to the reduced feature space
  - Find the connected components as clusters
- Compression of the grid is crucial for the efficiency
- Does not work in high dimensional space!

WaveCluster

- Signal transformation using wavelets



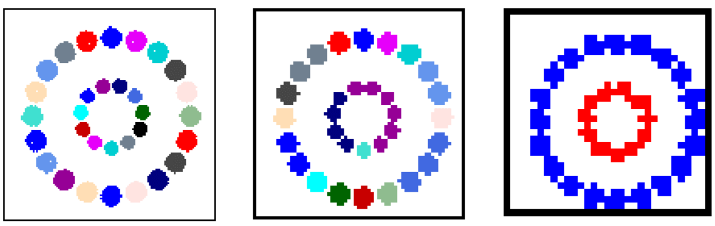
- Arbitrary shape clusters found by WaveCluster



Hinneburg / Keim, PKDD 2000

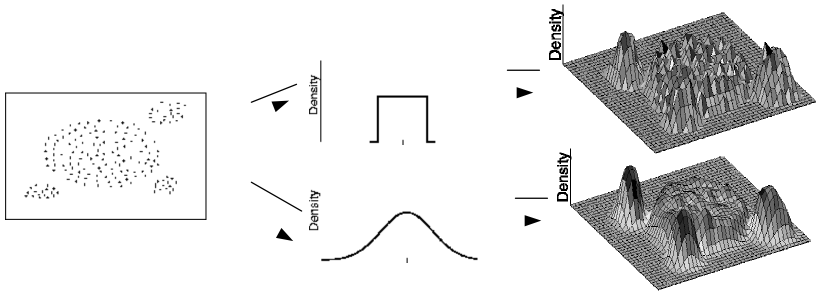
Hierarchical Variant of WaveCluster [SCZ 98]

- WaveCluster can be used to perform multiresolution clustering
- Using coarser grids, cluster start to merge



Hinneburg / Keim, PKDD 2000

Kernel Density Estimation



Data Set ***Influence Function*** ***Density Function***

Influence Function: Influence of a data point in its neighborhood

Density Function: Sum of the influences of all data points

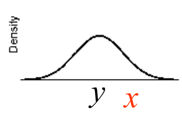
Hinneburg / Keim, PKDD 2000

Kernel Density Estimation

Influence Function

The influence of a data point *y* at a point *x* in the data space is modeled by a function $f_B^y : F^d \rightarrow \Re$,

e.g., $$f_{Gauss}^y(x) = e^{-\frac{d(x,y)^2}{2s^2}}.$$



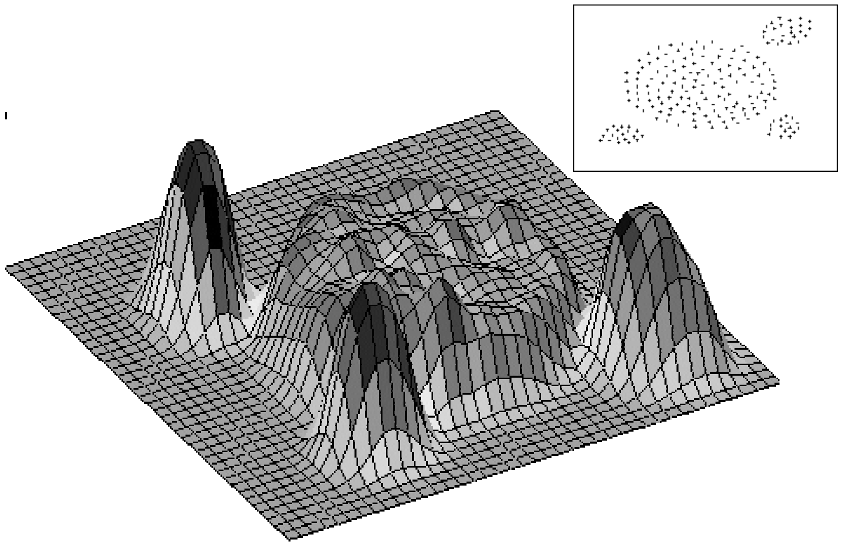
Density Function

The density at a point *x* in the data space is defined as the sum of influences of all data points x_i, i.e.

$$f_B^D(x) = \sum_{i=1}^{N} f_B^{x_i}(x)$$

Hinneburg / Keim, PKDD 2000

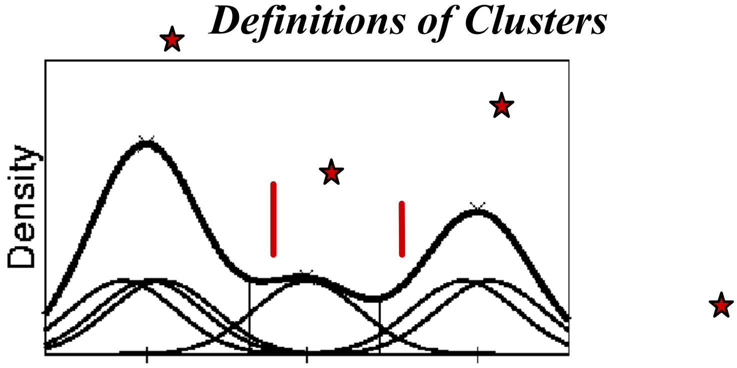
Kernel Density Estimation



Hinneburg / Keim, PKDD 2000

DENCLUE [HK 98]

Definitions of Clusters



Density Attractor/Density-Attracted Points ( )

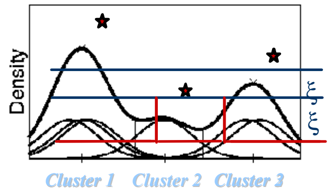
- local maximum of the density function
- density-attracted points are determined by a gradient-based hill-climbing method

Hinneburg / Keim, PKDD 2000

DENCLUE

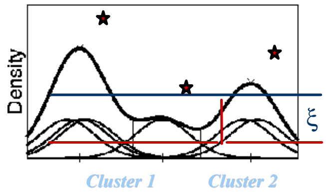
Center-Defined Cluster

A center-defined cluster with density-attractor x* ($f_B^D(x^*) > x$) is the subset of the database which is density-attracted by x*.



Multi-Center-Defined Cluster

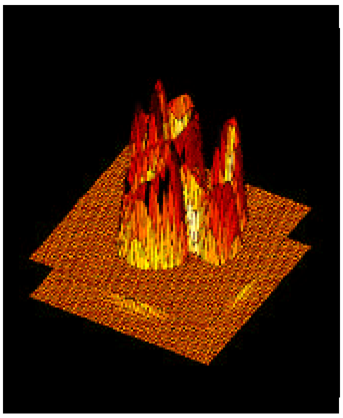
A multi-center-defined cluster consists of a set of center-defined clusters which are linked by a path with significance ξ.

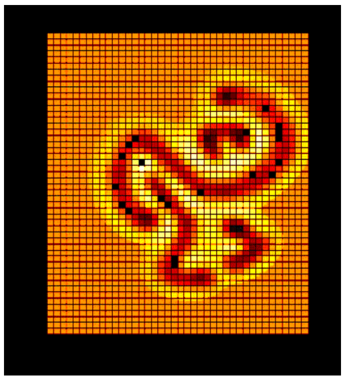


Hinneburg / Keim, PKDD 2000

DENCLUE

Impact of different Significance Levels (x)



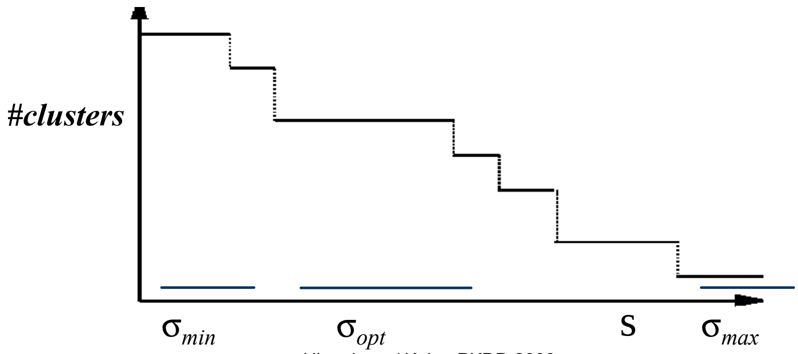


Hinneburg / Keim, PKDD 2000

DENCLUE

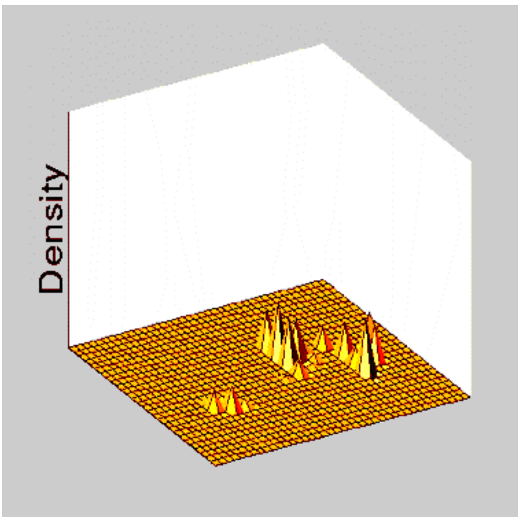
Choice of the Smoothness Level (σ)

Choose σ such that *number of density attractors* is constant for a long interval of σ!



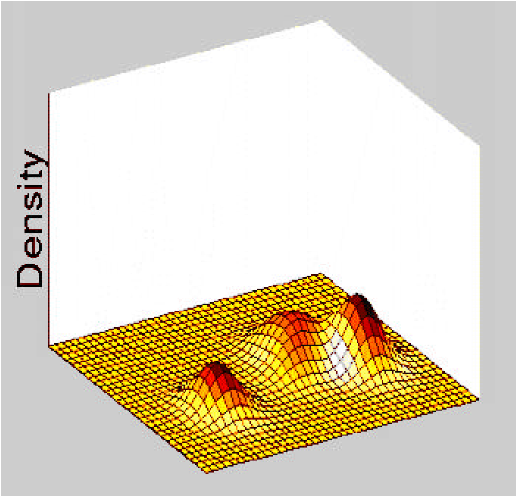
Hinneburg / Keim, PKDD 2000

Building Hierarchies (σ)



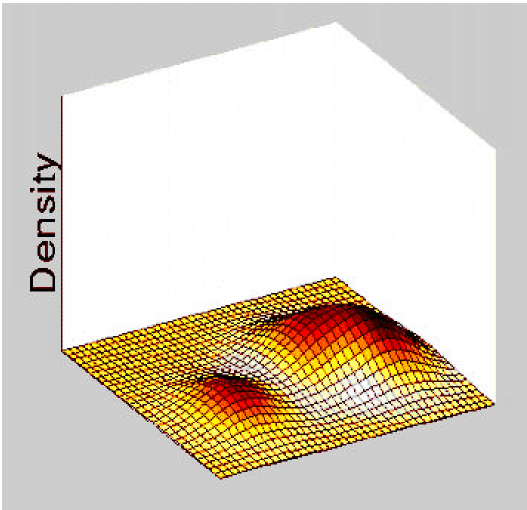
Hinneburg / Keim, PKDD 2000

DENCLUE
Variation of the Smoothness Level (s)



Hinneburg / Keim, PKDD 2000

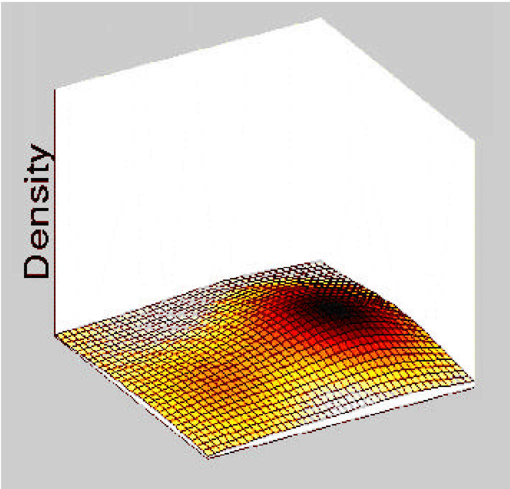
DENCLUE
Variation of the Smoothness Level (s)



Hinneburg / Keim, PKDD 2000

DENCLUE

Variation of the Smoothness Level (s)



Hinneburg / Keim, PKDD 2000

DENCLUE

Noise Invariance

Assumption: Noise is uniformly distributed in the data space

Lemma:

The density-attractors do not change when increasing the noise level.

Idea of the Proof:

- partition density function into signal and noise

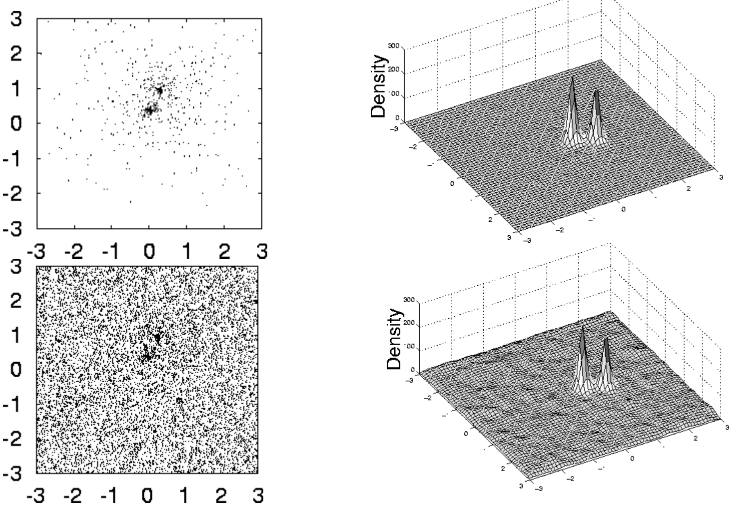
$$f^D(x) = f^{D_C}(x) + f^N(x)$$

- density function of noise approximates a constant $(f^N(x) \approx const.)$

Hinneburg / Keim, PKDD 2000

DENCLUE

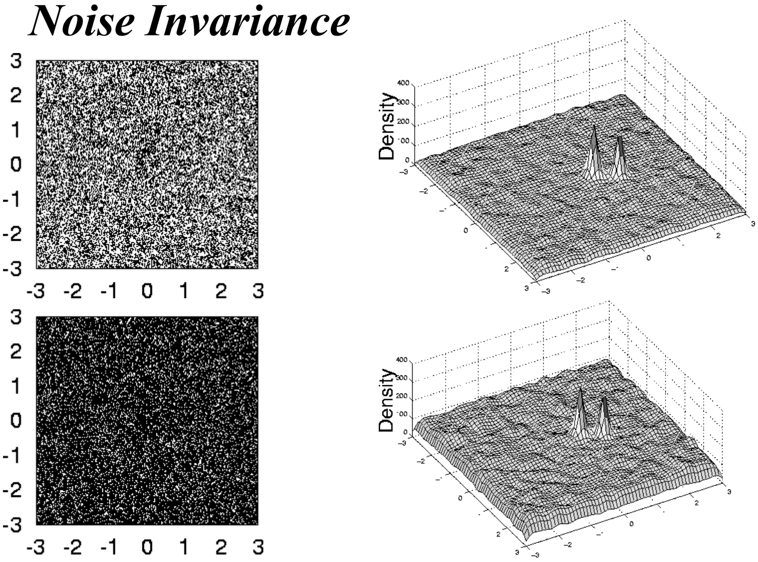
Noise Invariance



Hinneburg / Keim, PKDD 2000

DENCLUE

Noise Invariance



Hinneburg / Keim, PKDD 2000

DENCLUE

Local Density Function

Definition

The local density $\hat{f}_B^D(x)$ is defined as

$$\hat{f}_B^D(x) = \sum_{x_i \in near(x)} f_B^{x_i}(x) \; .$$

Lemma (Error Bound)

If $near(x) = \{x_i \in D \mid d(x, x_i) \le ks\}$, the error is bound by:

$$Error = \sum_{x_i \in D,\, d(x_i, x) > ks} e^{-\frac{d(x, x_i)^2}{2s^2}} \le \|\{x_i \in D \mid d(x, x_i) > ks\}\| \cdot e^{-\frac{k^2}{2}}$$

Hinneburg / Keim, PKDD 2000

Clustering on Categorical Data

- STIRR [GKR 2000], [GKR 98]
- ROCK [GRS 99]
- CACTUS [GGR 99]

Hinneburg / Keim, PKDD 2000

Overview (Second Lesson)

1. Introduction
2. Clustering Methods
   Model-, Linkage-, Density- based Approaches
3. Techniques Improving the Efficiency
   3.1 Multi-Dimensional Indexing
   3.2 Grid-based Approaches
   3.3 Sampling
4. Recent Research Topics
   4.1 Outlier Detection
   4.2 Projected Clustering
4. Summary and Conclusions

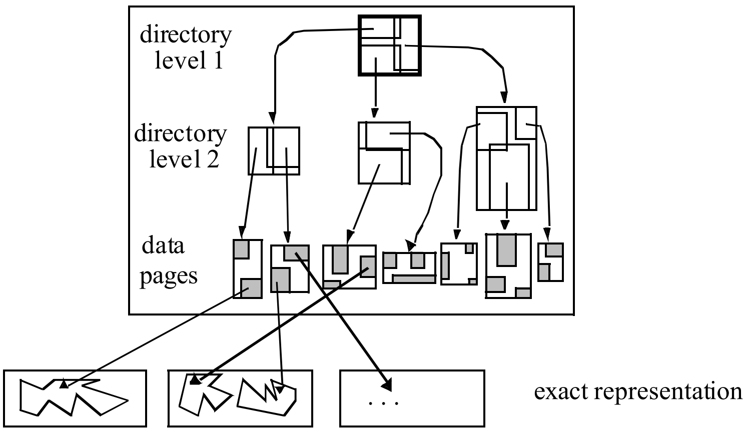
Improving the Efficiency

- Multi-dimensional Index Structures
  R-Tree, X-Tree, VA-File
- Grid Structures
- Sampling

Indexing [BK 98]

- Cluster algorithms and their index structures
  - BIRCH: CF-Tree [ZRL 96]
  - DBSCAN: R*-Tree [Gut 84]
    X-Tree [BKK 96]
  - STING: Grid / Quadtree [WYM 97]
  - WaveCluster: Grid / Array [SCZ 98]
  - DENCLUE: B^{+}-Tree, Grid / Array [HK 98]

R-Tree: [Gut 84]
The Concept of Overlapping Regions



Variants of the R-Tree

Low-dimensional

- R+-Tree [SRF 87]
- R*-Tree [BKSS 90]
- Hilbert R-Tree [KF94]

High-dimensional

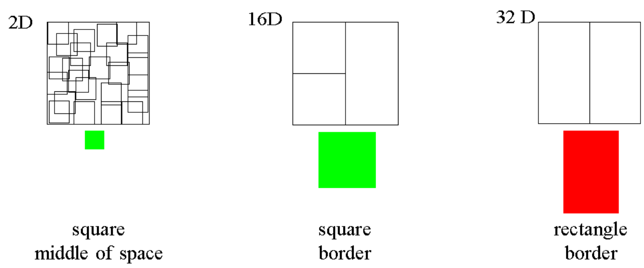
- TV-Tree [LJF 94]
- X-Tree [BKK 96]
- SS-Tree [WJ 96]
- SR-Tree [KS 97]

Hinneburg / Keim, PKDD 2000

Effects of High Dimensionality

Location and Shape of Data Pages

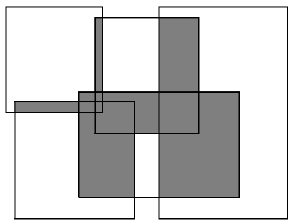
- Data pages have large extensions
- Most data pages touch the surface of the data space on most sides

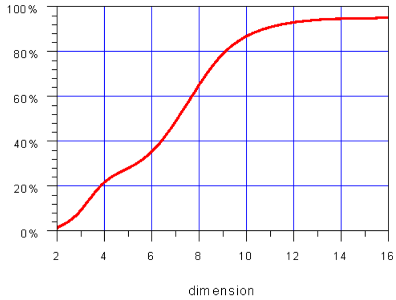


Hinneburg / Keim, PKDD 2000

The X-Tree [BKK 96]
(eXtended-Node Tree)

- Motivation:
  Performance of the R-Tree degenerates in high dimensions
- Reason: overlap in the directory

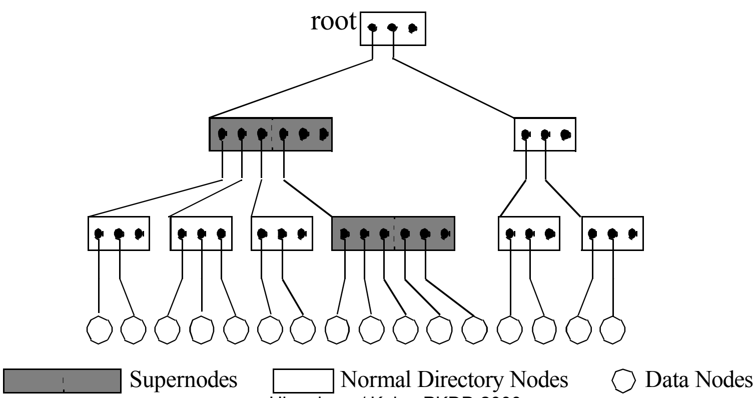




Hinneburg / Keim, PKDD 2000

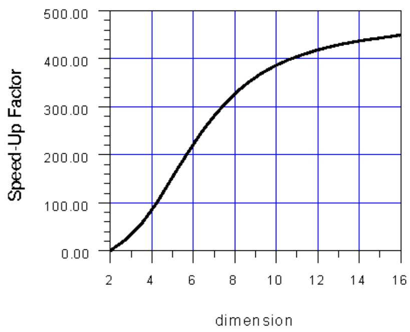
The X-Tree

- X-tree avoids overlap in the directory by using
  - an overlap-free split
  - the concept of supernodes

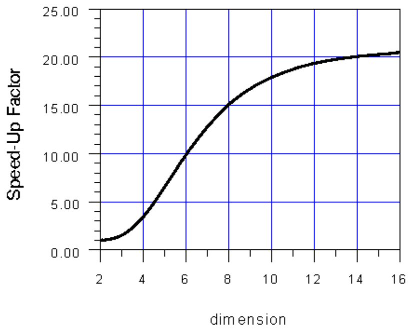


Hinneburg / Keim, PKDD 2000

Speed-Up of X-Tree over the R*-Tree



Point Query



10 NN Query

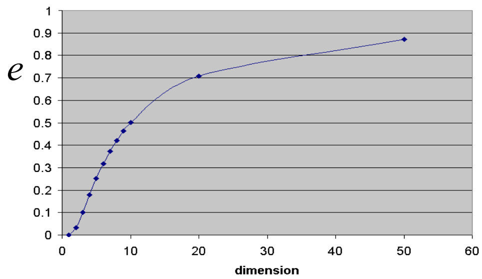
Hinneburg / Keim, PKDD 2000

Effects of High Dimensionality

Selectivity of Range Queries

- The selectivity depends on the volume of the query

$$e = \sqrt[d]{Vol_{cube}}$$



selectivity = 0.1 %

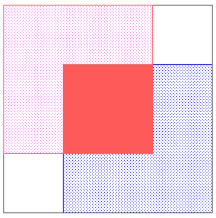
Þ **no fixed** e**-environment (as in DBSCAN)**

Hinneburg / Keim, PKDD 2000

Effects of High Dimensionality

Selectivity of Range Queries

- In high-dimensional data spaces, there exists a region in the data space which is affected by ANY range query (assuming uniformly distributed data)



Þ **difficult to build an efficient index structure**

Þ **no efficient support of range queries (as in DBSCAN)**

Efficiency of NN-Search[WSB 98]

- **Assumptions:**
  - A cluster is characterized by a geometrical form (MBR) that covers all cluster points
  - The geometrical form is convex
  - Each Cluster contains at least two points
- **Theorem:** For any clustering and partitioning method there is a dimensionality d for which a sequential scan performs better.

VA File [WSB 98]

- Vector Approximation File:
  - Compressing Vector Data: each dimension of a vector is represented by some bits
    ▶ partitions the space into a grid
  - Filtering Step: scan the compressed vectors to derive an upper and lower bound for the NN-distance ▶ Candidate Set
  - Accessing the Vectors: test the Candidate Set

Multi-Dimensional Grids

- Difference to Indexes:
  Allow Partitions with one Data Point
- Collect statistical Information about regions in the Data Space
- Filter Noise from the clustered data
- Used by:
  - STING [WYM 97]
  - WaveCluster [SCZ 98]
  - DENCLUE [HK 98]

Multi-Dimensional Grids

- General Idea:

$$Coding\ Function : R^d \rightarrow N$$

- Two Implementations:
  - Array: Stores all Grid Cells,
    - prohibitive for large d
  - Tree, Hash Structure: stores only the used Grid Cells,
    - works for all dimensions
    - drawback: mostly all data point are in a single cell, when the dimensionality is high

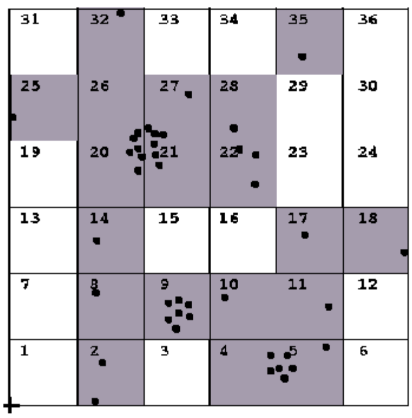
Hinneburg / Keim, PKDD 2000

Multi-Dimensional Grids

- Connecting Grid Cells:
  - the number of neighboring cells grows exponentially with the dimensionality
    ► Testing if the cell is used is prohibitive
  - Connect only the highly populated cells drawback: highly populated cells are unlikely in high dimensional spaces

Hinneburg / Keim, PKDD 2000

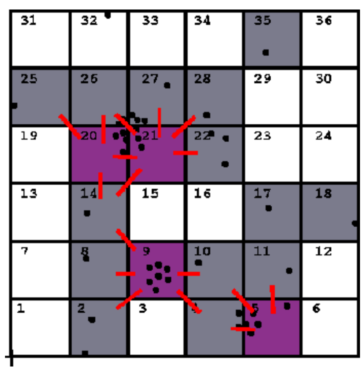
CubeMap



Data Structure based on regular cubes for storing the data and efficiently determining the density function

Hinneburg / Keim, PKDD 2000

DENCLUE Algorithm



DENCLUE (D, σ, ξ)

(a) $MBR \leftarrow DetermineMBR(D)$

(b) $C_p \leftarrow DetPopCubes(D, MBR, s)$

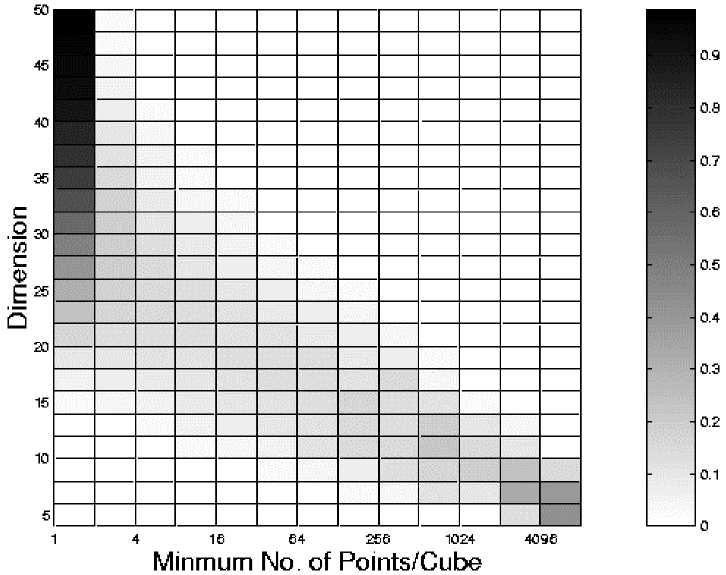
$C_{sp} \leftarrow DetHighlyPopCubes(C_p, x_c)$

(c) $map, C_r \leftarrow ConnectMap(C_p, C_{sp}, s)$

(d) $clusters \leftarrow DetDensAttractors(map, C_r, s, x)$

Hinneburg / Keim, PKDD 2000

Multi-Dimensional Grids

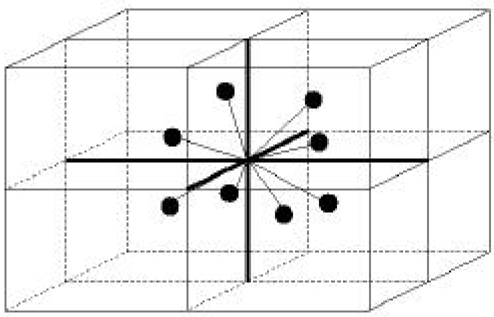


Hinneburg / Keim, PKDD 2000

Effects of High Dimensionality

Number of Neighboring cells

- Probability that Cutting Planes partition clusters increases



Þ **cluster can not be identified using the grid**

Hinneburg / Keim, PKDD 2000

Complexity of Clustering Data with Noise

Lemma:

The worst case time compexity for a correct clustering of highdimensional data with a constant percentage of noise is ***superlinear***, when the number of datapoints is $N < 2^d$.

Idea of the Proof:

Assumption: - database contains O(N) noise
- noise is read first (worst case)

Observation: - no constant access possible for noisy highdimensional, nonredundent data

⇨ noise (linear in N) has to be read multiple times

Sampling

- R*-Tree Sampling [EKX 95]
- Density Based Sampling [PF 00]
  - Uses a grid combined with a hash structure to approximate the density
  - Generates a uniform (random) Sample, if most grid-cells have only one data point.
- Sampling uses the redundancy in the data; however, the redundancy of high dimensional data decreases

Recent Research Topics

- Outlier Detection [KN 98,99,00], [RRS 00], [BKN+99, 00]
- Projected Clustering [AGG+ 98] [AMW+ 99], [AY 00][HK 99],[HKW 99]

Outlier

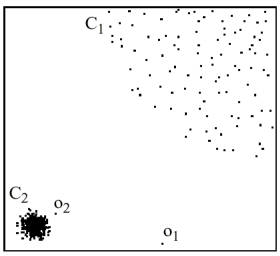
- Definition: (Hawkins-Outlier) An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism.

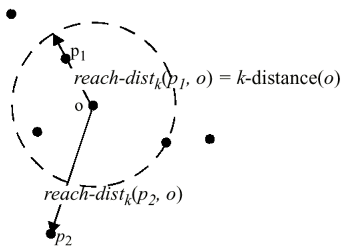
Distance Based Outliers

- Definition 1[KN 00] : Given a Database D, the Object o is an Outlier, iff
  $$p \cdot \#(D) \leq \#\{o' \mid o' \in D, dist(o,o') > D\}, 0 \leq p \leq 1$$
- Definition 2[RRS 00]: An Object o is an Outlier iff there exists not more than n-1 objects o' with
  $$nn-dist^k(o') > nn-dist^k(o)$$
- Both groups proposed efficient algorithms for multi-dimensional data with d<6
- The Algorithms base on Grids or Indexes.

Density Based Outliers

- Local instead Global Definition:





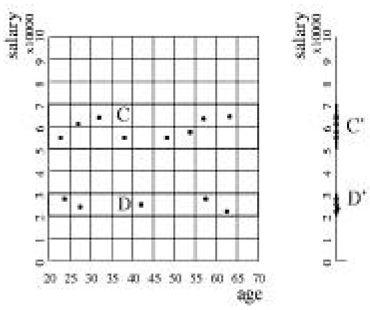
- The Outlier-Definition base on the average density in the neighborhood of a point, see reachability distance in the OPTICS paper.
- The Performance depends on the used index

Projected Clustering

- CLIQUE [AGG+ 98]
- ProClust / OrClust [AMW+ 99],[AY 00]
- OptiGrid / HD-Eye [HK 99],[HKW 99]

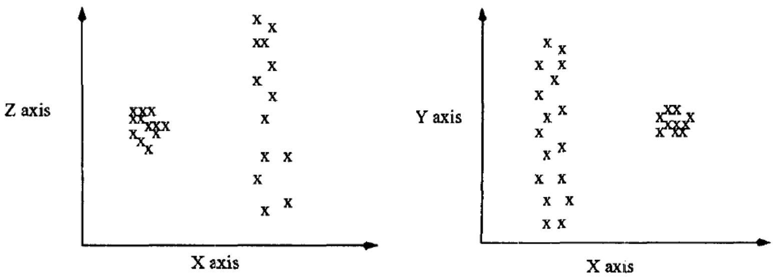
CLIQUE [AGG+ 98]

- Subspace Clustering
- Monotonicity Lemma:
  If a collection of points *S* is a cluster in a *k*-dimensional space, then S is also part of a cluster in any (k-1)-dimensional projection of this space.
- Bottom-up Algorithm for determining the projections



ProClust [AMW+ 99]

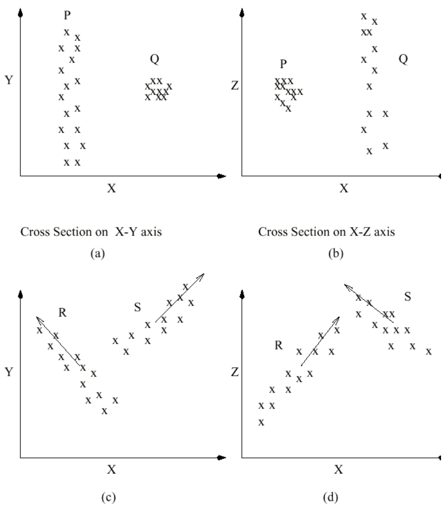
- Based on k-Means with a usability criterion for the dimensions



Hinneburg / Keim, PKDD 2000

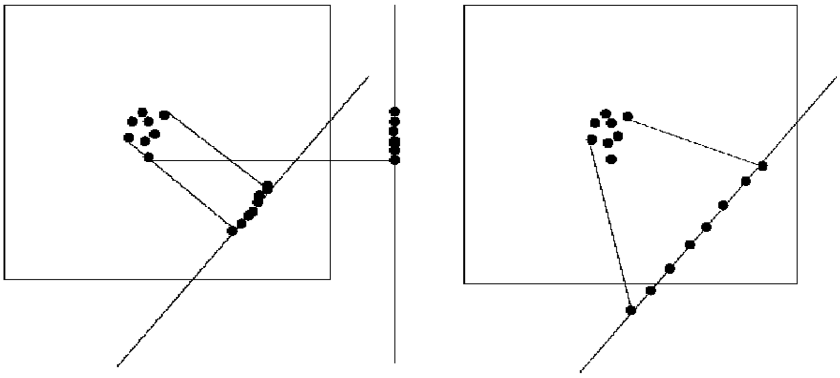
OR CLUST[AY 00]

- Based on k-Means with Local Principal Component Analyses.
- Prototypes with orientation vector



Hinneburg / Keim, PKDD 2000

Contracting Projections



Contracting Projection *Non-contracting Projection*

Upper Bound Property

Lemma:

Let *P(x)=Ax* be a contracting projection, P(D) the projection of the data set D and $\hat{f}^{P(D)}(x')$ an estimate of the density at a point $x' \in P(S)$. Then,

$$\forall x \in S \text{ with } P(x) = x' : \hat{f}^{P(D)}(x') \geq \hat{f}^{D}(x).$$

Proof: $\forall x, y \in S$

$$\|P(x) - P(y)\| = \|A(x-y)\| \leq \|A\| \cdot \|x-y\| \leq \|x-y\|$$

Cutting Plane

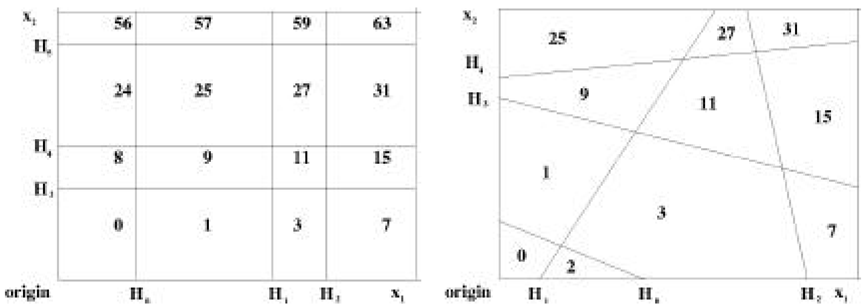
The *Cutting Plane* is a set of points y such that $\sum_{i=1}^{d} w_i \cdot y_i = 1$

The cutting plane defines a *Partitioning Function H(x)* for all points *x* of the data space

$$H(x) = \begin{cases} 1 & , \sum_{i=1}^{d} w_i x_i \geq 1 \\ 0 & , else \end{cases}$$

Hinneburg / Keim, PKDD 2000

Multi-dimensional Grids



Orthogonal Grid *Non-orthogonal Grid*

Coding Function $c(x) = \sum_{i=1}^{k} 2^i \cdot H_i(x)$

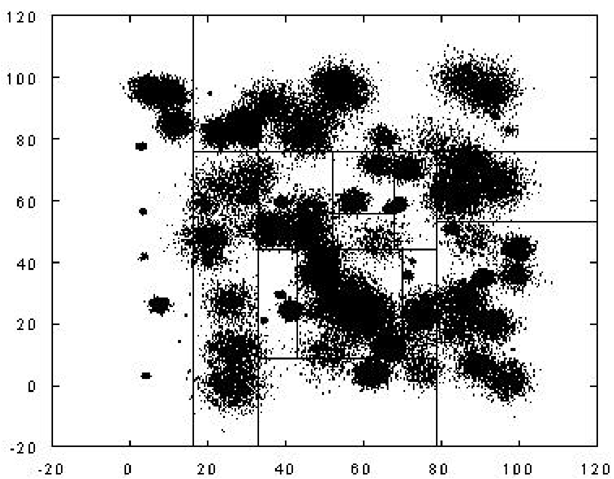
Hinneburg / Keim, PKDD 2000

The OptiGrid Algorithm [HK 99]

1. Determine a set of contracting projection $\{P_0, \ldots P_k\}$
2. Determine the best q Cutting Planes $\{H_0, \ldots H_q\}$ in the projections
3. If there are no good Cutting Planes exit; <u>otherwise:</u>
4. Determine a multi-dim. Grid based on $\{H_0, \ldots H_k\}$
5. Find Clusters C_i in the Grid by determining highly-populated grid cells
6. For each C_i : OptiGrid(C_i)

Hinneburg / Keim, PKDD 2000

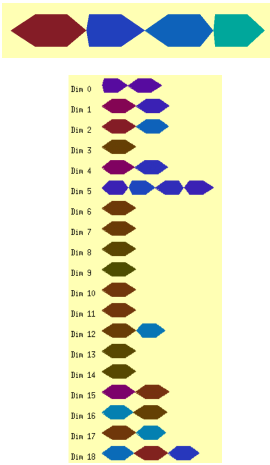
Example Partitioning



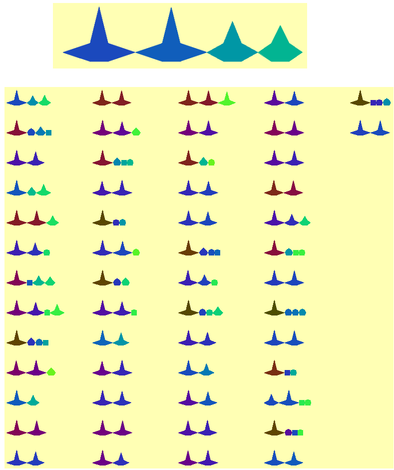
Hinneburg / Keim, PKDD 2000

Integration of Visual and Automated Data Mining

Icons for one-dim. Projections

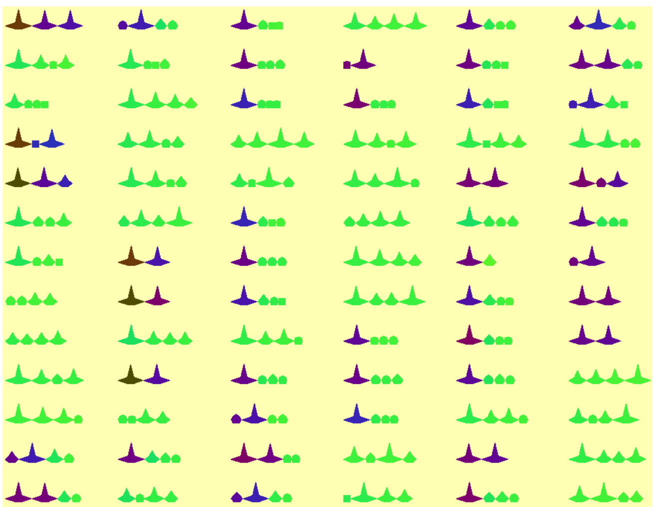


Icons for two-dim. Projections



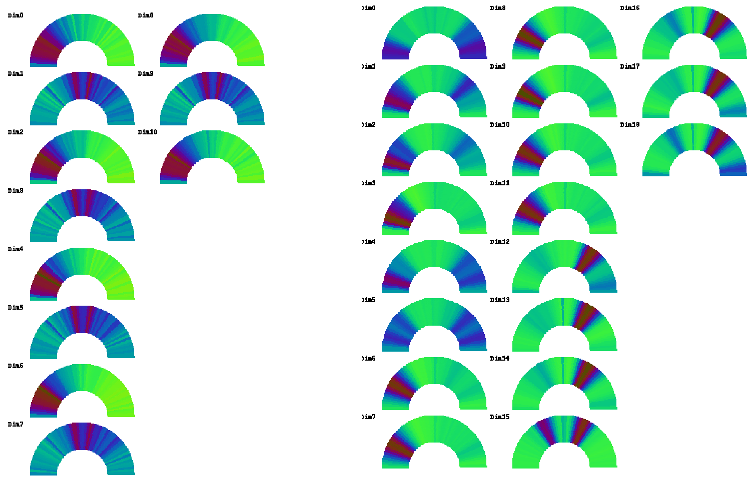
Hinneburg / Keim, PKDD 2000

Integration of Visual and Automated Data Mining



Hinneburg / Keim, PKDD 2000

Integration of
Visual and Automated Data Mining

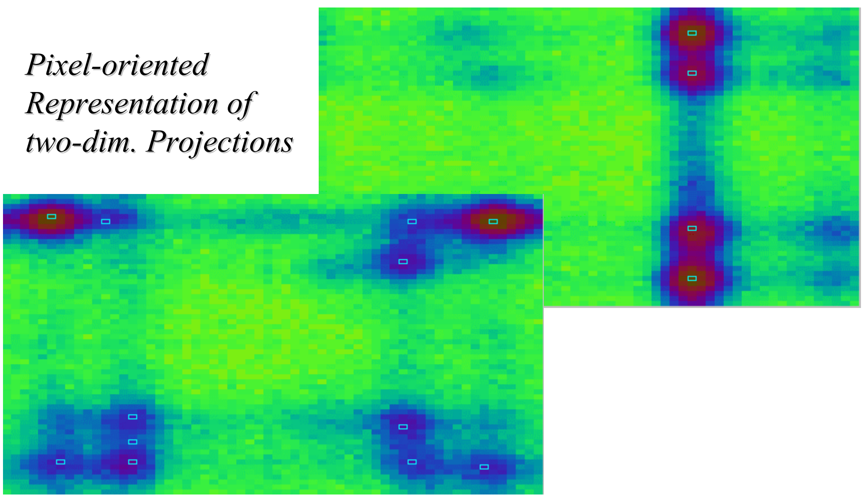


Pixel-oriented Representation of one-dimensional Projections

Hinneburg / Keim, PKDD 2000

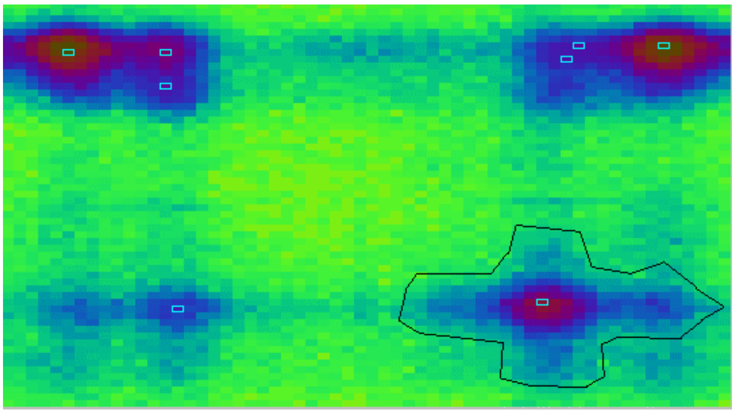
Integration of
Visual and Automated Data Mining

Pixel-oriented Representation of two-dim. Projections



Hinneburg / Keim, PKDD 2000

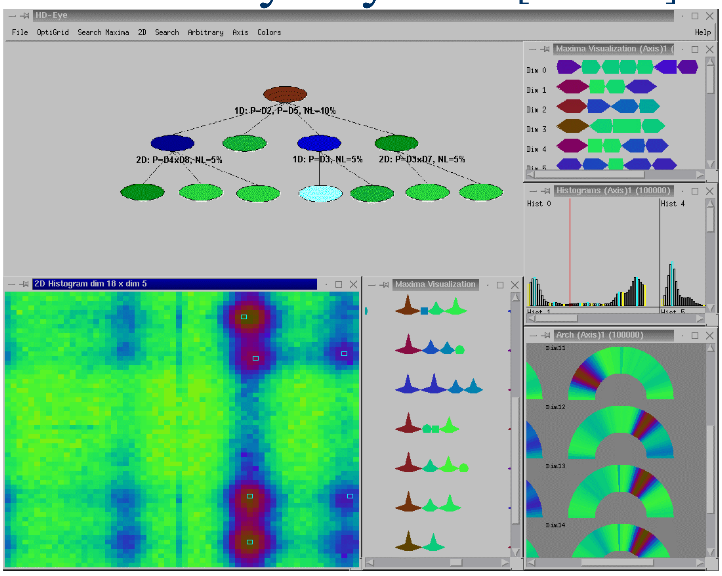
Integration of Visual and Automated Data Mining



Interactive Specification of Cutting Planes in 2D Projections

Hinneburg / Keim, PKDD 2000

The HD-Eye System [HK 99a]



Hinneburg / Keim, PKDD 2000

Summary and Conclusions

- A number of *effective* and *efficient* Clustering Algorithms is available for *small to medium size* data sets and *small dimensionality*
- **Efficiency** suffers severely for large dimensionality *(d)*
- **Effectiveness** suffers severely for large dimensionality (*d*), especially in combination with a high *noise level*

Open Research Issues

- *Efficient Data Structures* for large ***N*** and large ***d***
- *Clustering Algorithms* which work *effectively* for large ***N***, large ***d*** and large *Noise Levels*
- *Integrated Tools* for an Effective Clustering of High-Dimensional Data *(combination of automatic, visual and interactive clustering techniques)*

References

- [AGG+ 98] R. Aggrawal, J. Gehrke, D. Gunopulos, P. Raghavan, *Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications*, Proc. ACM SIGMOD Int. Conf. on Managment of Data, pp. 94-105, 1998
- [AMW+ 99] Charu C. Aggarwal, Cecilia Magdalena Procopiuc, Joel L. Wolf, Philip S. Yu, Jong Soo Park: *Fast Algorithms for Projected Clustering*. SIGMOD Conference 1999: 61-72.
- [AY 00] Charu C. Aggarwal, Philip S. Yu: *Finding Generalized Projected Clusters In High Dimensional Spaces*. SIGMOD Conference 2000: 70-81.
- [Boc 74] H.H. Bock, *Autmatic Classification*, Vandenhoeck and Ruprecht, Göttingen, 1974
- [BK 98] S. Berchtold, D.A. Keim, *High-Dimensional Index Structures, Database Support for Next Decade's Applications,* ACM SIGMOD Int. Conf. on Management of Data, 1998.
- [BBK 98] S. Berchtold, C. Böhm, H-P. Kriegel, *The Pyramid-Technique: Towards Breaking the Curse of Dimensionality*, Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 142-153, 1998.
- [BKK 96] S. Berchtold, D.A. Keim, H-P. Kriegel, *The X-Tree: An Index Structure for High-Dimensional Data*, Proc. 22th Int. Conf. on Very Large Data Bases, pp. 28-39, 1996.
- [BKK 97] S. Berchtold, D. Keim, H-P. Kriegel, *Using Extended Feature Objects for Partial Similarity Retrieval*, VLDB Journal, Vol.4, 1997.
- [BKN+ 99] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, Jörg Sander: *OPTICS-OF: Identifying Local Outliers.* PKDD 1999: 262-270

- [BKN+ 00] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, Jörg Sander: *LOF: Identifying Density-Based Local Outliers.* SIGMOD Conference 2000: 93-104
- [BKSS 90] N. Beckmann., h-P. Kriegel, R. Schneider, B. Seeger, *The R*-tree: An Efficient and Robust Access Method for Points and Rectangles*, Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 322-331, 1990.
- [CHY 96] Ming-Syan Chen, Jiawei Han, Philip S. Yu: *Data Mining: An Overview from a Database Perspective*. TKDE 8(6), pp. 866-883, 1996.
- [EKS+ 96] M. Ester, H-P. Kriegel, J. Sander, X. Xu, *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*, Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, 1996.
- [EKSX 98] M. Ester, H-P. Kriegel, J. Sander, X. Xu, *Clustering for Mining in Large Spatial Databases*, Special Issue on Data Mining, KI-Journal, ScienTec Publishing, No. 1, 1998.

- [EKSX 98] M. Ester, H-P. Kriegel, J. Sander, X. Xu, *Clustering for Mining in Large Spatial Databases*, Special Issue on Data Mining, KI-Journal, ScienTec Publishing, No. 1, 1998.

- [EKX 95] M. Ester, H-P. Kriegel, X. Xu, *Knowlege Discovery in Large Spatial Databases: Focusing Techniques for Efficient Class Identification*, Lecture Notes in Computer Science, Springer 1995.
- [EKX 95b] M. Ester, H-P. Kriegel, X. Xu, *A Database Interface for Clustering in Large Spatial Databases*, Proc. 1st Int. Conf. on Knowledge Discovery and Data Mining, 1995.
- [EW 98] M. Ester, R. Wittmann, *Incremental Generalization for Mining in a Data Warehousing Environment*, Proc. Int. Conf. on Extending Database Technology, pp. 135-149, 1998.

- [DE 84] W.H. Day and H. Edelsbrunner, *Efficient algorithms for agglomerative hierachical clustering methods*, Journal of Classification, 1(1):7-24, 1984.
- [DH 73] R.O. Duda and P.E. Hart, *Pattern Classifaication and Scene Analysis*, New York; Wiley and Sons , 1973.
- [Gra 92] R.M. Gray, *Vector Qunatization and Signal Compression*, Kluwer Academic Press, 1992.
- [GRS 98] Sudipto Guha,Rajeev Rastogi, Kyuseok Shim, *CURE: An Efficient Clustering Algorithm for Large Databases,* Proceedings ACM SIGMOD International Conference on Management of Data, 1998, Seattle, Washington, USA, ACM Press, 1998, pp. 73-84.
- [Fuk 90] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, San Diego, CA, Academic Press 1990.
- [Fri 95] B. Fritzke, *A Growing Neural Gas Network Learns Topologies*, in G. Tesauro, D.S. Touretzky and T.K. Leen (eds.) *Advances in Neural Information Processing Systems 7*, MIT Press, Cambridge MA, 1995.
- [Fri 96] B. Fritzke, *Unsupervised ontogenetic networks, in Handbook of Neural Computation*, IOP Publishing and Oxford University Press, 1996.
- [Fri 97] B. Fritzke, *The LBG-U method for vector quantization - an improvement over LBG inspired from neural networks*, Neural Processing Letters (1997), Vol.5, No. 1, also appeared as internal report IRINI 97-0.
- [FH 75] K. Fukunaga and L.D. Hosteler, *The Estimation of the Gradient of a density function with Applications in Pattern Recognition*, IEEE Trans. Info. Thy., IT-21, 32-40, 1975.

- [GGR 99] Venkatesh Ganti, Johannes Gehrke, Raghu Ramakrishnan: *CACTUS - Clustering Categorical Data Using Summaries*. KDD 1999: 73-83.
- [GKR 98] David Gibson, Jon M. Kleinberg, Prabhakar Raghavan: *Clustering Categorical Data: An Approach Based on Dynamical Systems*. in Proc. 24th Int. Conf on Very Large Data Bases VLDB 1998: 311-322.
- [GKR 2000] David Gibson, Jon M. Kl512-521einberg, Prabhakar Raghavan: *Clustering Categorical Data: An Approach Based on Dynamical Systems*.VLDB Journal 2000, 8, pp.222-236.
- [GRS 99] Sudipto Guha, Rajeev Rastogi, Kyuseok Shim: ROCK: *A Robust Clustering Algorithm for Categorical Attributes*. Proceedings of the 15th International Conference on Data Engineering, 23-26 March 1999, pp.512-521.
- [HAK 2000] Hinneburg A., Aggarwal C., Keim D.A.:*What is the nearest neighbor in highdimensional spaces*, in Proc. 26th Int. Conf on Very Large Data Bases, Cairo, 2000.
- [HK 98] A. Hinneburg, D.A. Keim, *An Efficient Approach to Clustering in Large Multimedia Databases with Noise*, Proc. 4th Int. Conf. on Knowledge Discovery and Data Mining, 1998.
- [HK 99] A., Keim D.A.:*Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering* , in Proc. 25th Int. Conf. on Very Large Bases, Edinburgh, 1999, pp.506-517.
- [HKW 99] Hinneburg A., Keim D.A., Wawryniuk M.:HD-Eye: *Visual Mining High-dimensional Data* , in IEEE Computer Graphics and Applications, Sept/Oct. 1999, Vol. 19 (5), pp.22-31.
- [Jag 91] J. Jagadish, *A Retrieval Technique for Similar Shapes*, Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 208-217, 1991.

- [Kei 96] D.A. Keim, *Databases and Visualization*, Tutorial on ACM SIGMOD Int. Conf. on Management of Data, 1996.
- [KMN 97] M.Kearns, Y. Mansour and A. Ng, *An Information-Theoretic Analysis of Hard and Soft Assignment Methods for Clustering*, Proc. 13th Conf. on Uncertainty in Artificial Intelligence, pp. 282-293, 1997, Morgan Kaufmann.
- [KMS+ 91] T. Kohonen, K. Mäkisara, O.Simula and J. Kangas, *Artificaial Networks*, Amsterdam 1991.
- [KN 98] Edwin M. Knorr, Raymond T. Ng: *Algorithms for Mining Distance-Based Outliers in Large Datasets.* VLDB 1998: 392-403.
- [KN 99] Edwin M. Knorr, Raymond T. Ng: *Finding Intensional Knowledge of Distance-Based Outliers*. VLDB 1999:211-222.
- [KN 00] Edwin M. Knorr, Raymond T. Ng, V. Tucakov: *Distance-Based Outliers: Algorithms and Applications*.VLDB Journal 8(3-4): 237-253 (2000).
- [Lau 95] S.L. Lauritzen, *The EM algorithm for graphical association models with missing data,* Computational Statistics and Data Analysis, 19:191-201, 1995.
- [MBS 93] T. M. Martinetz S.G. Berkovich, K.J. Schulten, *Neural-gas network for vector quantization and ist application to time-series prediction*, IEEE Trans. Neural Networks, 4, 1993, pp. 558-5569.
- [Mur 84] F. Murtagh, *Complexities of hierarchic clustering algorithms: State of the art*, Computational Statistics Quarterly, 1:101-113, 1984.
- [MG 93] R. Mehrotra, J. Gary, *Feature-Based Retrieval of Similar Shapes*, Proc. 9th Int. Conf. on Data Engeneering, April 1993.
- [NH 94] R.T. Ng, J. Han, *Efficient and Effective Clustering Methods for Spatial Data Mining*, Proc. 20th Int. Conf. on Very Large Data Bases, pp. 144-155, 1994.
- [PF 00] Christopher R. Palmer and Christos Faloutsos, *Density Biased Sampling: An Improved Method for Data Mining and Clustering*, SIGMOD, Dallas, TX, May 2000, pp.82-92.

- [Roj 96] R. Rojas, *Neural Networks - A Systematic Introduction*, Springer Berlin, 1996.
- [RRS 00] Sridhar Ramaswamy, Rajeev Rastogi, Kyuseok Shim: *Efficient Algorithms for Mining Outliers from Large Data Sets.* SIGMOD Conference 2000: 427-438
- [Sch 64] P. Schnell, *A Method for Discovering Data-Groups*, Biometrica 6, 47-48, 1964.
- [Sil 86] B.W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, 1986.
- [Sco 92] D.W. Scott, *Multivariate Density Estimation*, Wiley and Sons, 1992.
- [Sch 96] E. Schikuta, *Grid clustering: An efficient hierarchical method for very large data sets*, Proc. 13th Conf. on Patter Recognition, Vol. 2 IEEE Computer Society Press, pp. 101-105, 1996.
- [SCZ 98] G.Sheikholeslami, S. Chatterjee and A. Zhang, *WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases*, Proc. 24th Int. Conf. on Very Large Data Bases, 1998.
- [Wis 69] D. Wishart, *Mode Analysis: A Generalisation of Nearest Neighbor, which reducing Chaining Effects*, in A. J. Cole (Hrsg.), 282-312, 1969.
- [WYM 97] W. Wang, J. Yang, R. Muntz, *STING: A Statistical Information Grid Approach to Spatial Data Mining*, Proc. 23rd Int. Conf. on Very Large Data Bases 1997.
- [XEK+ 98] X. Xu, M. Ester, H-P. Kriegel and J. Sander., *A Distribution-Based Clustering Algorithm for Mining in Large Spatial Databases*, Proc. 14th Int. Conf. on Data Engineering (ICDE'98), Orlando, FL, 1998, pp. 324-331.
- [ZHD 99] Zhang Bin, Hsu Meichun, Dayal Umeshwar, *K-Harmonic Means - A Clustering Algorithm,* HP Tech. Rep HPL-1999-124.
- [ZHD 00] Zhang Bin, Hsu Meichun, Dayal Umeshwar, *K-Harmonic Means - A Spatial Clustering Algorithm With Boosting,* International Workshop on Temporal, Spatial and Spatio- Data Mining, TSDM2000.
- [ZRL 96] T. Zhang, R. Ramakrishnan and M. Livny, *An Efficient Data Clustering Method for Very Large Databases*. Proc. ACM SIGMOD Int. Conf. on Managment of Data, pp. 103-114, 1996