DATABASE SYSTEMS GROUP

Ludwig-Maximilians-Universität München
Institut für Informatik
Lehr- und Forschungseinheit für Datenbanksysteme

LMU

Lecture notes

Knowledge Discovery in Databases

Summer Semester 2012

Lecture 9: Clustering III

Lecture: Dr. Eirini Ntoutsi
Tutorials: Erich Schubert

http://www.dbs.ifi.lmu.de/cms/Knowledge_Discovery_in_Databases_I_(KDD_I)

Sources

- Previous KDD I lectures on LMU (Johannes Aßfalg, Christian Böhm, Karsten Borgwardt, Martin Ester, Eshref Januzaj, Karin Kailing, Peer Kröger, Jörg Sander, Matthias Schubert, Arthur Zimek)

- Tan P.-N., Steinbach M., Kumar V., *Introduction to Data Mining*, Addison-Wesley, 2006

- Jiawei Han, Micheline Kamber and Jian Pei, *Data Mining: Concepts and Techniques, 3rd ed.,* Morgan Kaufmann, 2011.

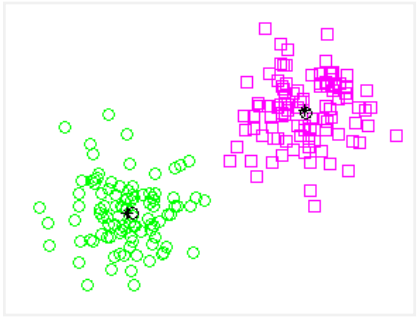
- Christoph Lippert | Data Mining in Bioinfortics | Clustering – tutorial, http://agbs.kyb.tuebingen.mpg.de/wikis/bg/tutorial_GMM.pdf

Outline

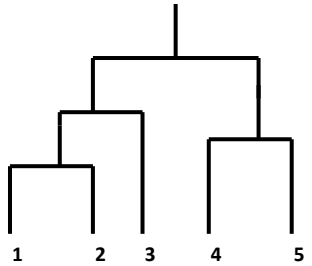
- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

Major clustering methods I

- Partitioning approaches:
  - 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

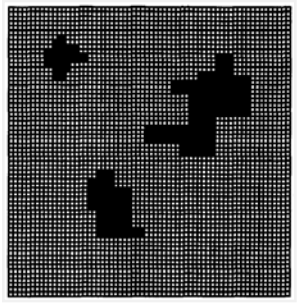


- Hierarchical approaches:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, ROCK, CHAMELEON



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

Major clustering methods II



- Grid-based approaches:
  - based on a multiple-level granularity structure
  - Typical methods: STING, WaveCluster, CLIQUE
- Model-based approaches:
  - 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 approaches:
  - Based on the analysis of frequent patterns
  - Typical methods: pCluster
- User-guided or constraint-based approaches:
  - Clustering by considering user-specified or application-specific constraints
  - Typical methods: COD (obstacles), constrained clustering

Outline

- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

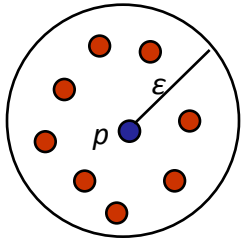
Density-based clustering

- Clusters are regions of high density surrounded by regions of low density (noise)
- 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

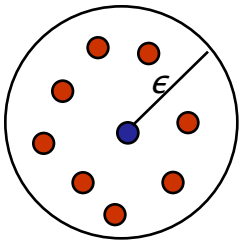
- Several interesting studies:
  - DBSCAN: Ester, et al. (KDD'96)
  - OPTICS: Ankerst, et al (SIGMOD'99).
  - DENCLUE: Hinneburg & D. Keim (KDD'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98) (more grid-based)

DBSCAN (Ester et al, KDD'96) (previous lecture)

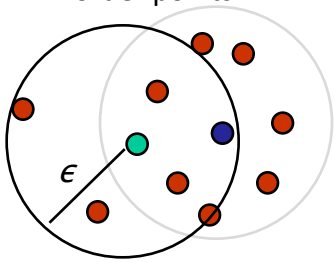
- Two parameters:
  - Eps (or ε): Maximum radius of the neighbourhood
  - MinPts: Minimum number of points in an Eps-neighbourhood of that point
- Eps-neighborhood of a point p in D
  - $N_{Eps}(p)$: {q belongs to D | dist(p,q) <= Eps}



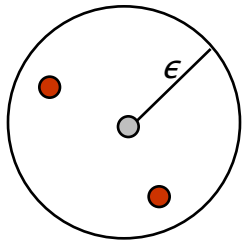
Core points

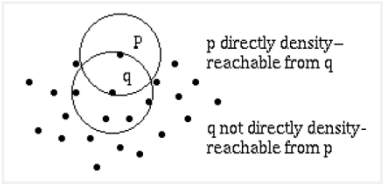


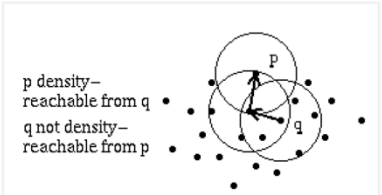
Border points

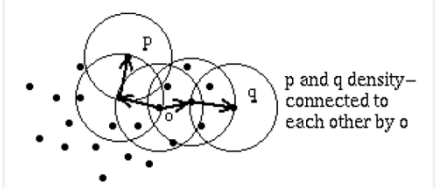


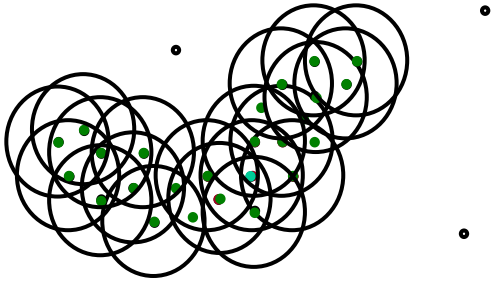
Noise









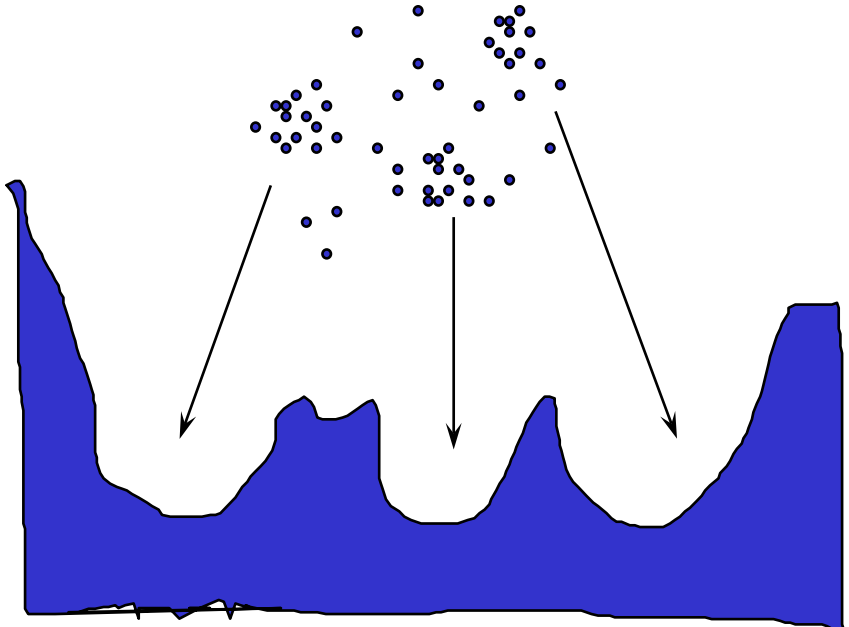


A cluster is a maximal set of density-connected points

OPTICS (Ankerst et al, SIGMOD'99)

- OPTICS: Ordering Points To Identify the Clustering Structure
  - Extension of DBSCAN
- It does not produce a clustering of a dataset explicitly, instead it produces a special ordering of the database w.r.t. its density-based clustering structure
- This cluster-ordering contains information that is equivalent to the density-based clusterings corresponding to a broad range of parameter settings

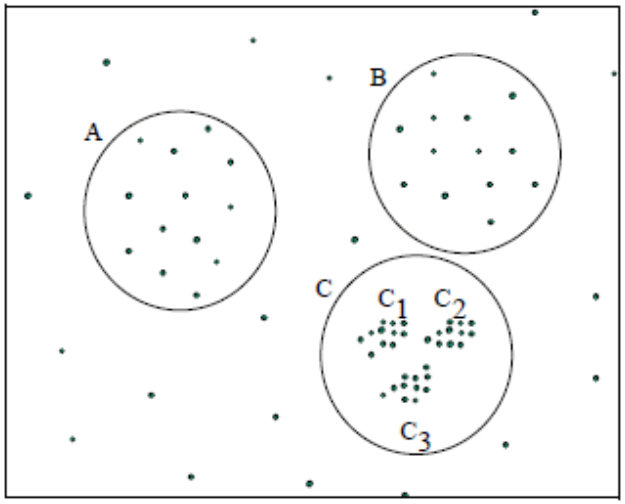
- Good for both automatic and interactive cluster analysis, including finding intrinsic clustering structure
- Can be represented graphically



Cluster-ordering of the objects

Motivation

- In many cases the intrinsic cluster structure cannot be characterized by *global* density parameters
- Different *local* densities may be needed to reveal clusters in different regions of the data space



Global densities would result in:

- A,B,C clusters

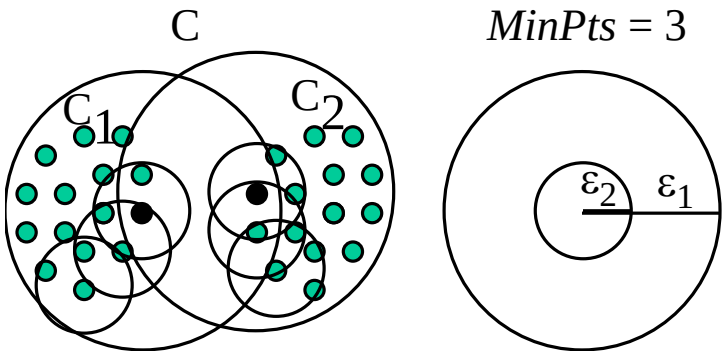
  or

- C_1, C_2, C_3 clusters

- Use a hierarchical clustering?
  - Difficult to interpret the dendrogram for large datasets
- Use DBSCAN with different parameters?
  - Infinite number of possible parameters

OPTICS: idea

- Observation: for a constant minPts value, density-based clusters w.r.t. to a lower value for ε are completely contained in density-based clusters with a higher value for ε



C is a cluster w.r.t. ε_1
C_1, C_2 are clusters w.r.t. $\varepsilon_2 < \varepsilon_1$
C contains C_1, C_2

- Idea: extend DBSCAN so as several distance parameters are processed at the same time
- This way, density-based clusters w.r.t. different densities are constructed simultaneously

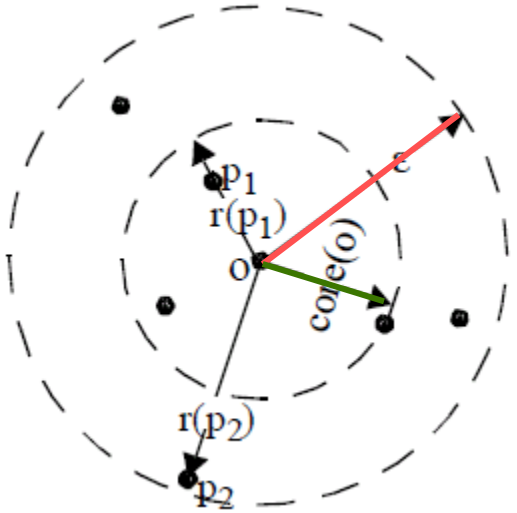
OPTICS basic notions I

- Core distance of an object p

  Let $N_\varepsilon(p)$ be the ε-neighborhood of p in D and let MinPtsdistance(p) be the distance from p to its MinPts' neighbor.

$$\text{coredistance}_{\varepsilon,MinPts}(p) = \begin{cases} UNDEFINED, & if\ |N_\varepsilon(p)| < MinPts \\ MinPtsdistance(p), & otherwise \end{cases}$$

  – Is simply the smallest distance ε' between p and an object q in its ε-neighborhood such that p would be a core point w.r.t. ε' if q was contained in $N_\varepsilon(p)$. Otherwise is undefined.



MinPts=4

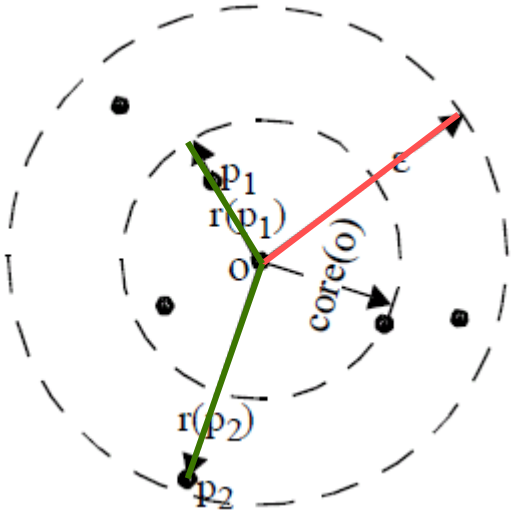
OPTICS basic notions II

- Reachability distance of an object p w.r.t. an object o

  Let $N_\varepsilon(o)$ be the ε-neighborhood of o in D.

$$\text{reachabilitydistance}_{\varepsilon,MinPts}(p,o) = \begin{cases} UNDEFINED & \text{if } |N_\varepsilon(o)| < MinPts \\ \max\{\text{coredistance}(o), dist(o,p)\}, & \text{otherwise} \end{cases}$$

  - Is the smallest distance so as p is directly density-reachable from o, if o is core.
    - It cant be smaller than coredistance(o) because otherwise o will not be core
  - it depends on the core object o w.r.t. which it is calculated.



MinPts=4

OPTICS pseudocode I


```
OPTICS (SetOfObjects, ε, MinPts, OrderedFile)
  OrderedFile.open();
  FOR i FROM 1 TO SetOfObjects.size DO
    Object := SetOfObjects.get(i);
    IF NOT Object.Processed THEN
      ExpandClusterOrder(SetOfObjects, Object, ε,
                         MinPts, OrderedFile)
  OrderedFile.close();
END; // OPTICS
```

OPTICS pseudocode II


```
ExpandClusterOrder(SetOfObjects, Object, ε, MinPts,
OrderedFile);
  neighbors := SetOfObjects.neighbors(Object, ε);
  Object.Processed := TRUE;
  Object.reachability_distance := UNDEFINED;
  Object.setCoreDistance(neighbors, ε, MinPts);
  OrderedFile.write(Object);
  IF Object.core_distance <> UNDEFINED THEN
    OrderSeeds.update(neighbors, Object);
    WHILE NOT OrderSeeds.empty() DO
      currentObject := OrderSeeds.next();
      neighbors:=SetOfObjects.neighbors(currentObject, ε);
      currentObject.Processed := TRUE;
      currentObject.setCoreDistance(neighbors, ε, MinPts);
      OrderedFile.write(currentObject);
      IF currentObject.core_distance<>UNDEFINED THEN
        OrderSeeds.update(neighbors, currentObject);
END; // ExpandClusterOrder
```

The objects contained in OrderSeeds are sorted by their reachability-distance to the closest core object, Object, from which they have been directly density-reachable.

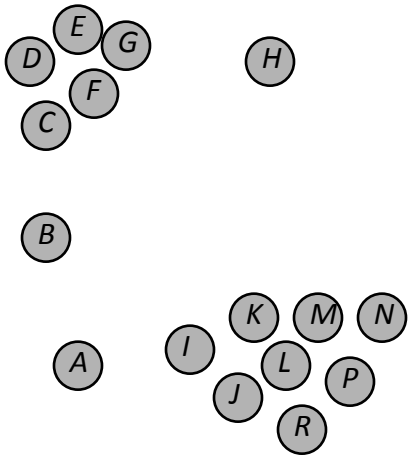
the object having the smallest reachability-distance in the seed-list is selected, currentObject

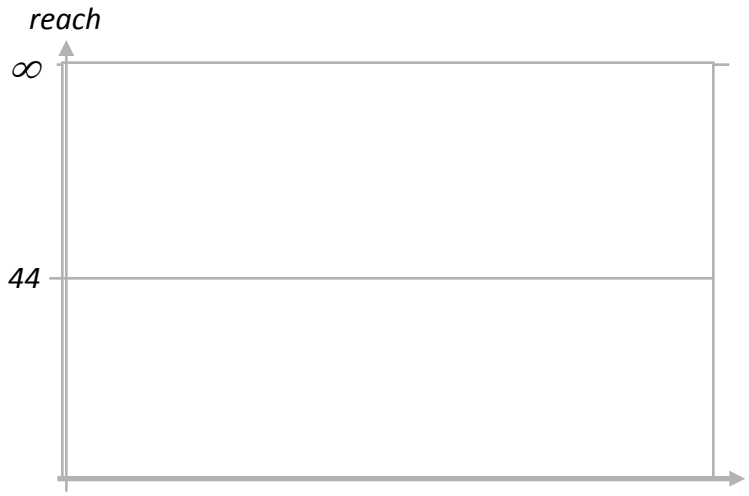
Output: OPTICS outputs the points in a particular ordering. Each point is accompanied with its coredistance and its smallest reachability distance.

```
OrderSeeds::update(neighbors, CenterObject);
  c_dist := CenterObject.core_distance;
  FORALL Object FROM neighbors DO
    IF NOT Object.Processed THEN
      new_r_dist:=max(c_dist,CenterObject.dist(Object));
      IF Object.reachability_distance=UNDEFINED THEN
        Object.reachability_distance := new_r_dist;
        insert(Object, new_r_dist);
      ELSE // Object already in OrderSeeds
        IF new_r_dist<Object.reachability_distance THEN
          Object.reachability_distance := new_r_dist;
          decrease(Object, new_r_dist);
END; // OrderSeeds::update
```

Example 1

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

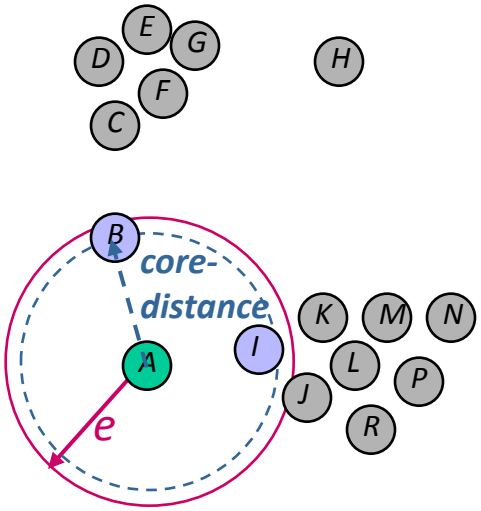




seed list:

Example 2

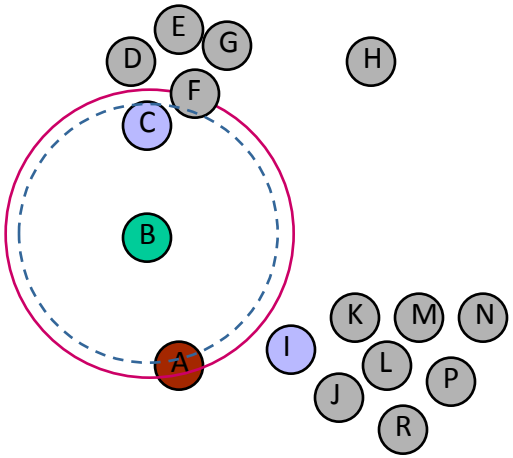
- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

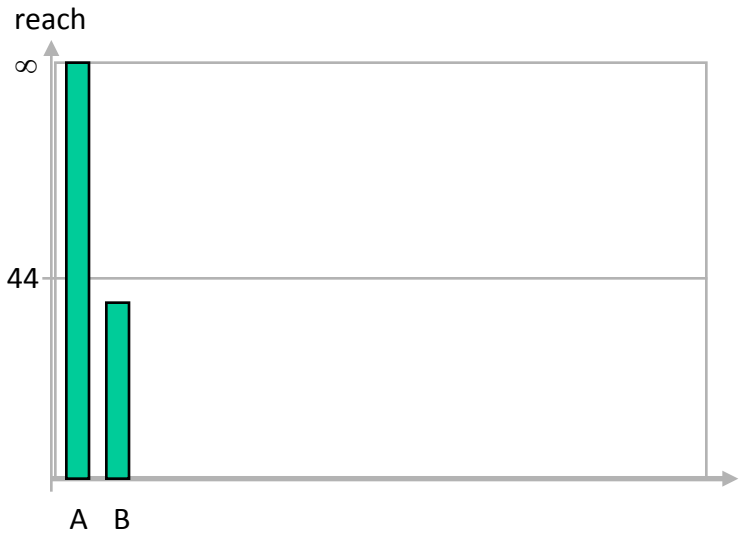


seed list: (B,40) (I, 40)

Example 3

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

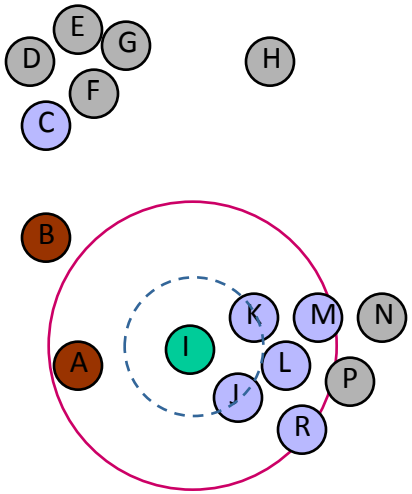


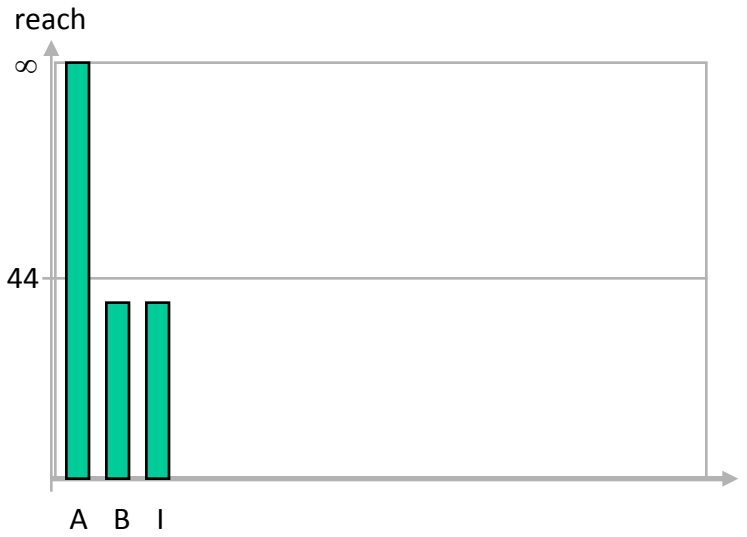


seed list: (I, 40) (C, 40)

Example 4

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

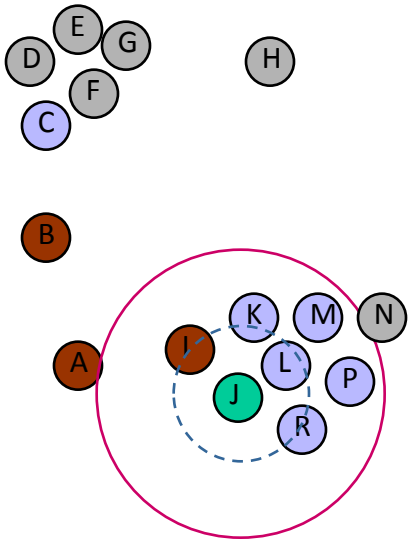


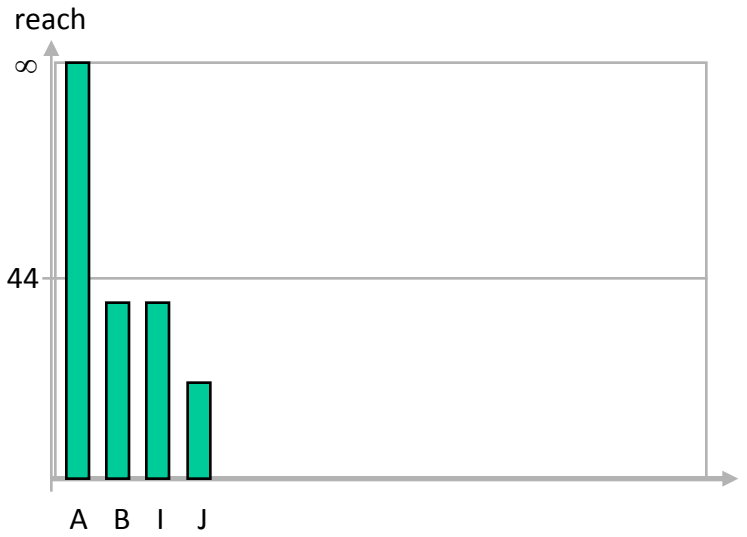


seed list: (J, 20) (K, 20) (L, 31) (C, 40) (M, 40) (R, 43)

Example 5

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

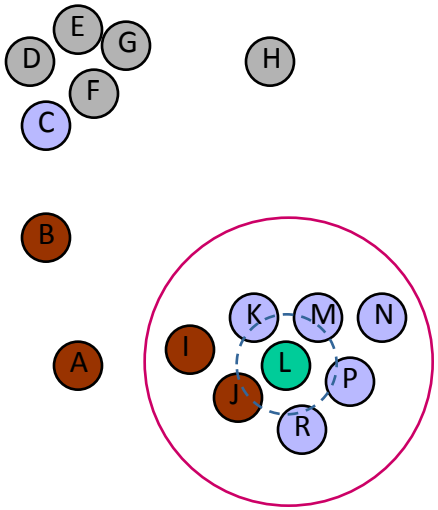


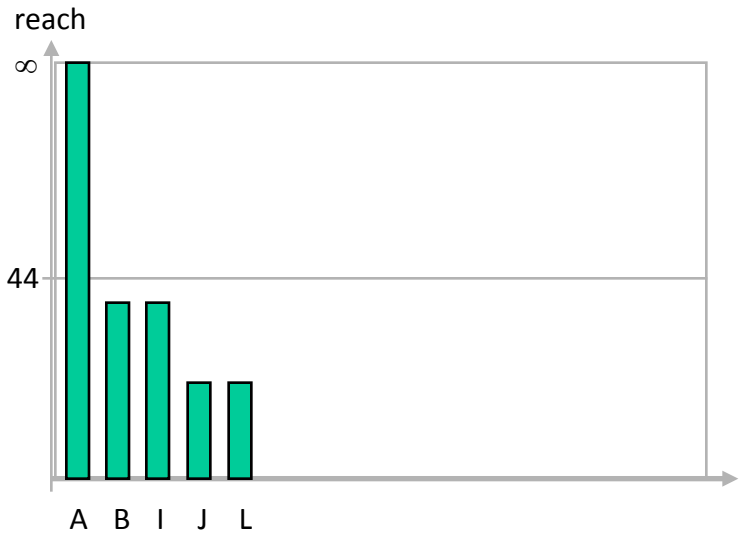


seed list: (L, 19) (K, 20) (R, 21) (M, 30) (P, 31) (C, 40)

Example 6

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

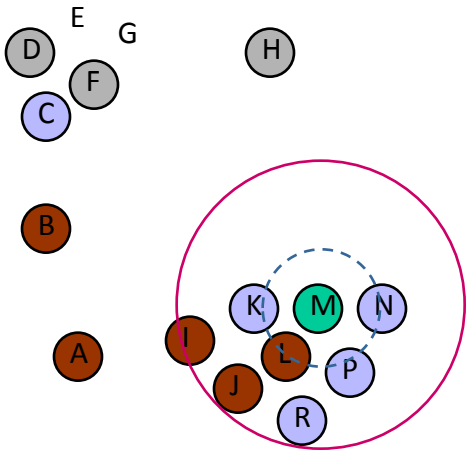


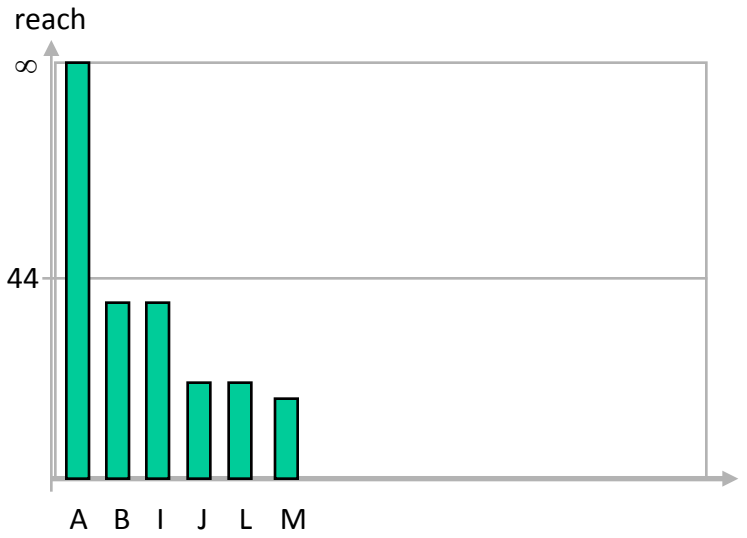


seed list: (M, 18) (K, 18) (R, 20) (P, 21) (N, 35) (C, 40)

Example 7

- Example Database (2-dimensional, 16 points)
- $\varepsilon = 44$, MinPts = 3

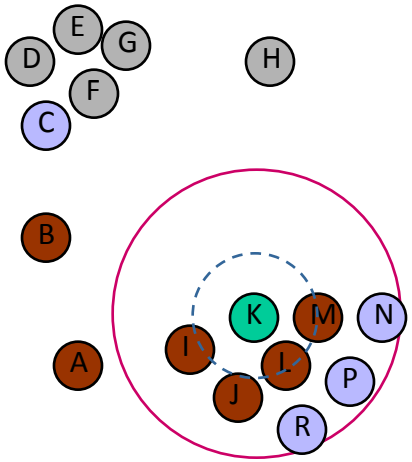


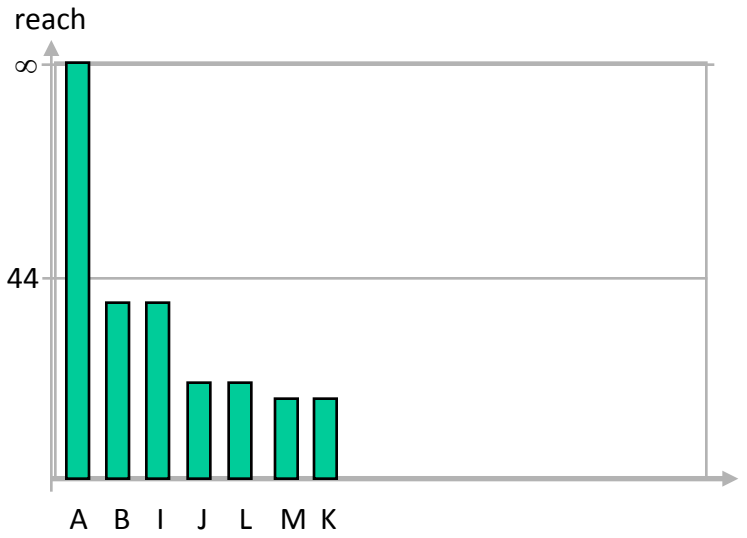


seed list: (K, 18) (N, 19) (R, 20) (P, 21) (C, 40)

Example 8

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

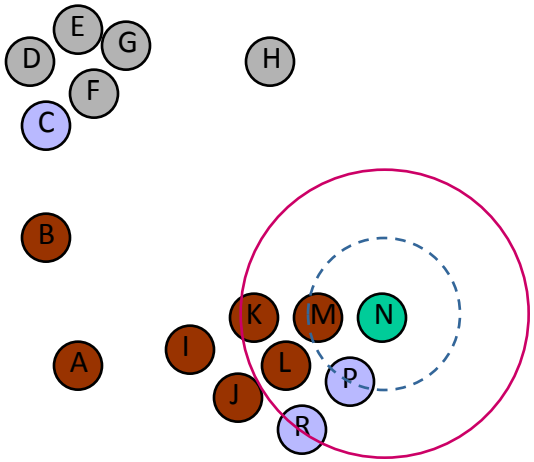


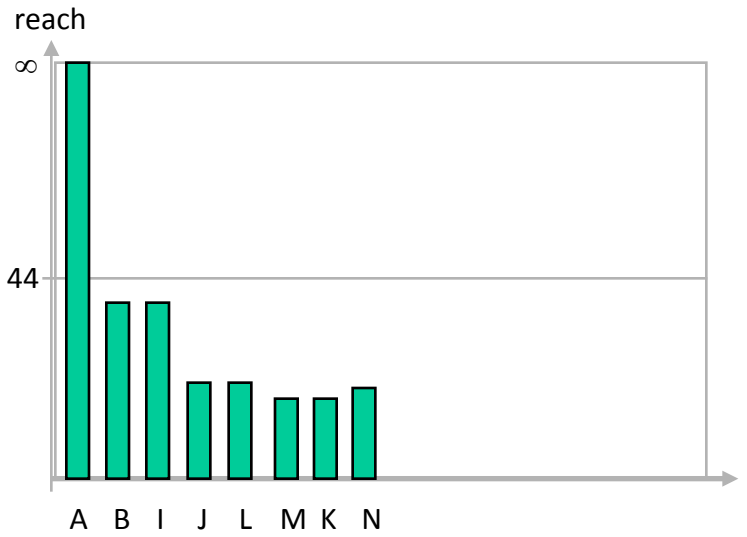


seed list: (N, 19) (R, 20) (P, 21) (C, 40)

Example 9

- Example Database (2-dimensional, 16 points)
- $\varepsilon = 44$, MinPts = 3

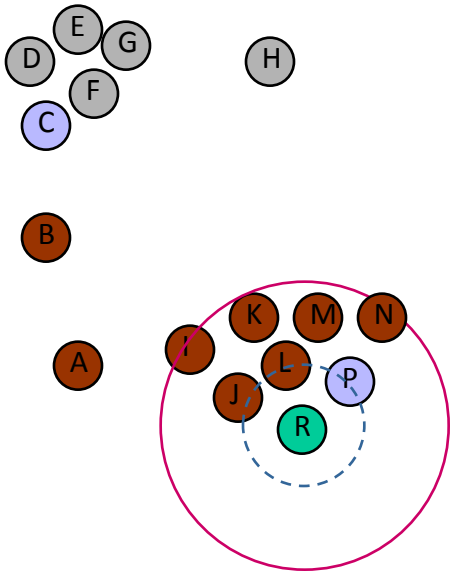


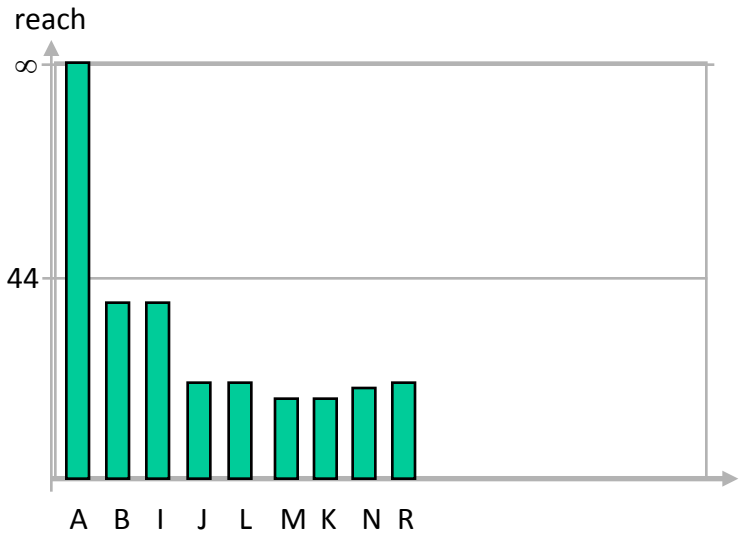


seed list: (R, 20) (P, 21) (C, 40)

Example 10

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

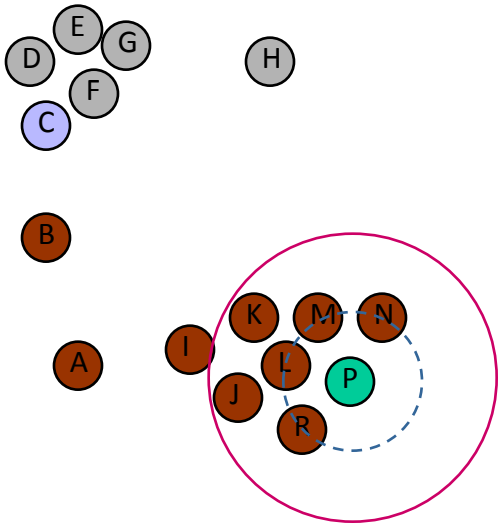


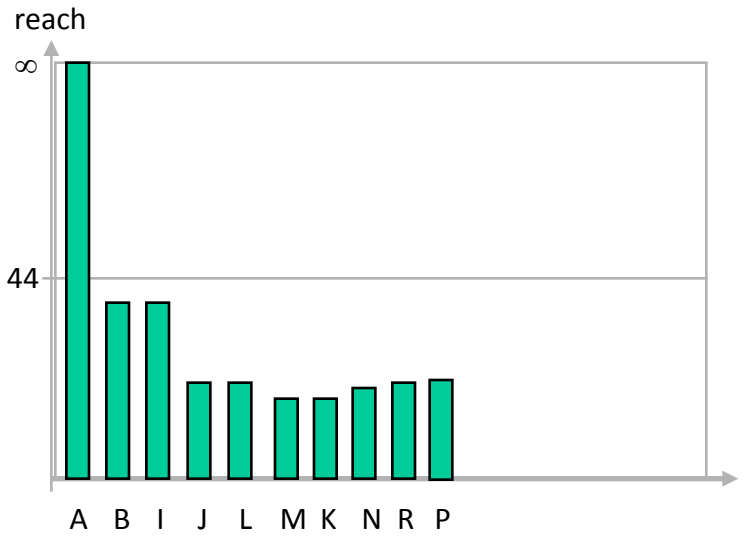


seed list: (P, 21) (C, 40)

Example 11

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

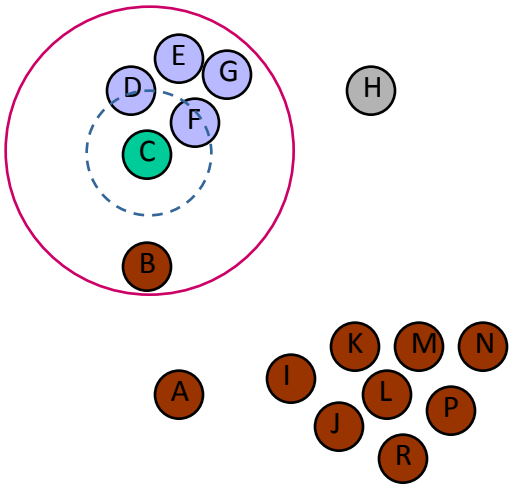


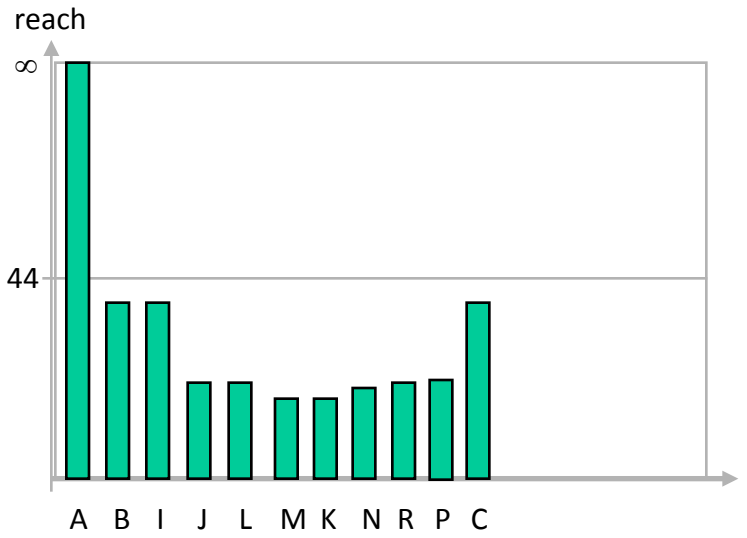


seed list: (C, 40)

Example 12

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

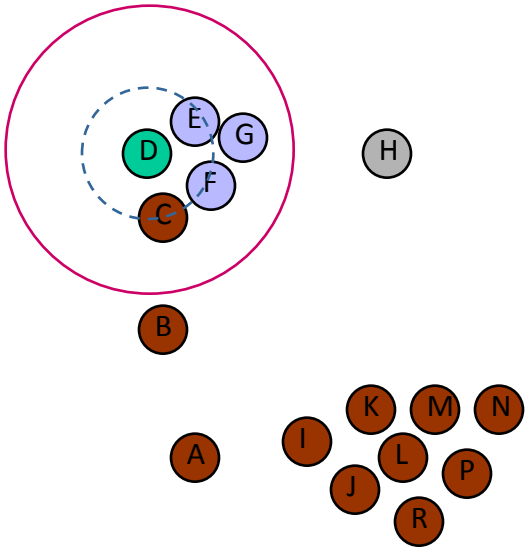


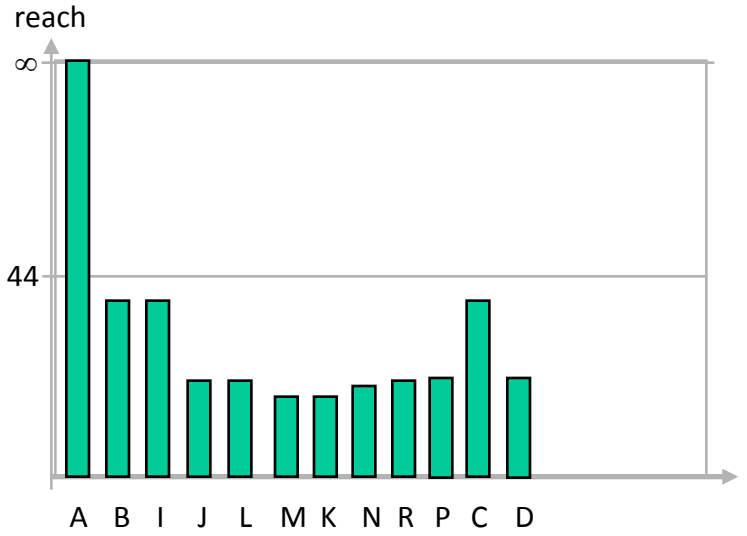


seed list: (D, 22) (F, 22) (E, 30) (G, 35)

Example 12

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

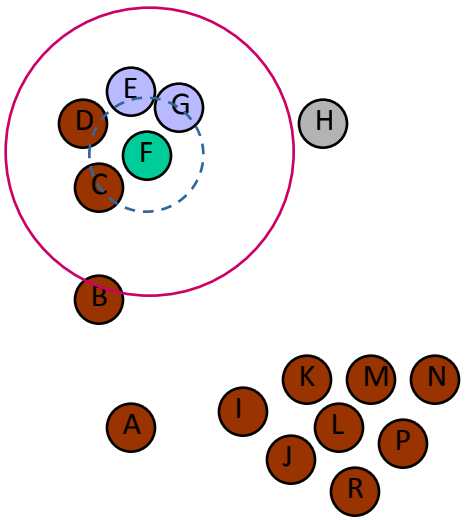


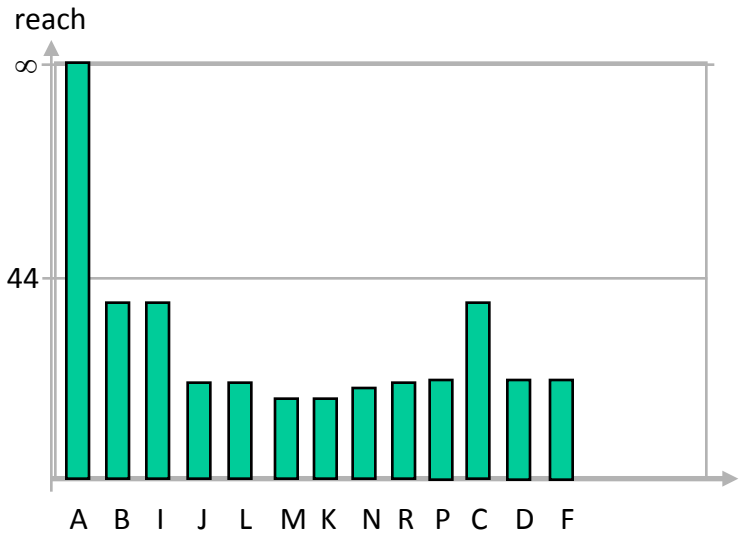


seed list: (F, 22) (E, 22) (G, 32)

Example 14

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

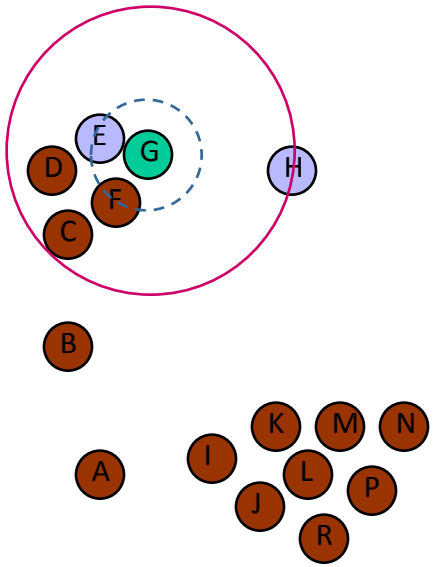


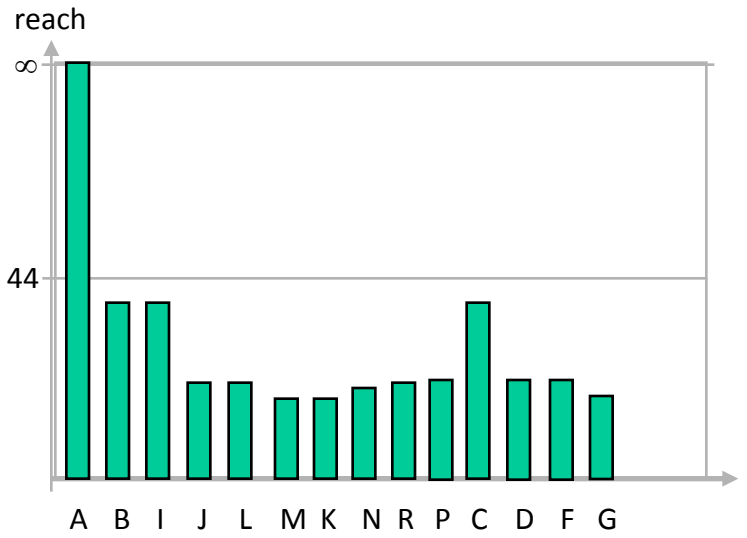


seed list: (G, 17) (E, 22)

Example 15

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

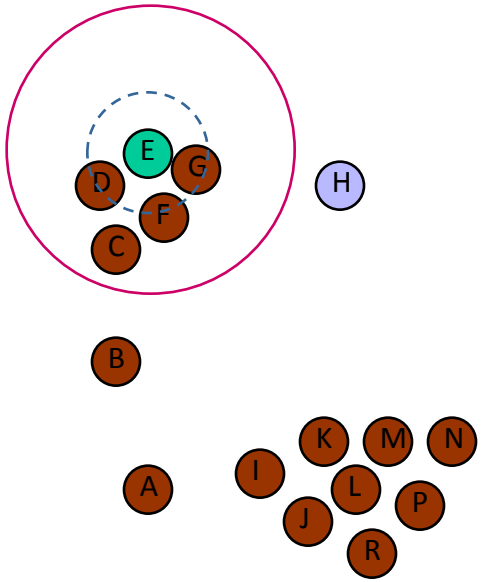


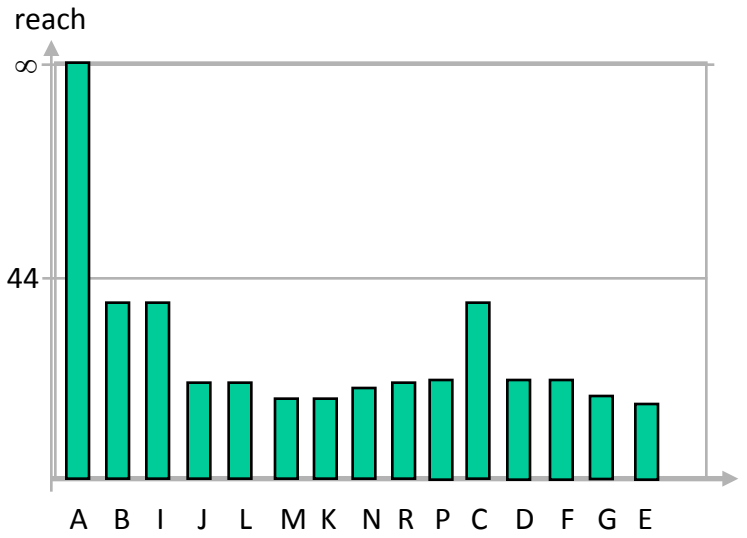


seed list: (E, 15) (H, 43)

Example 16

- Example Database (2-dimensional, 16 points)
- ε= 44, MinPts = 3

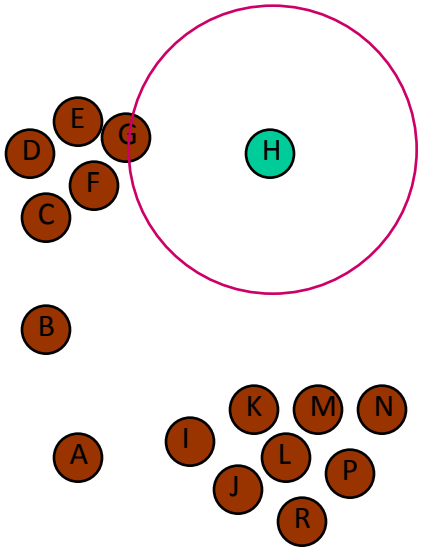


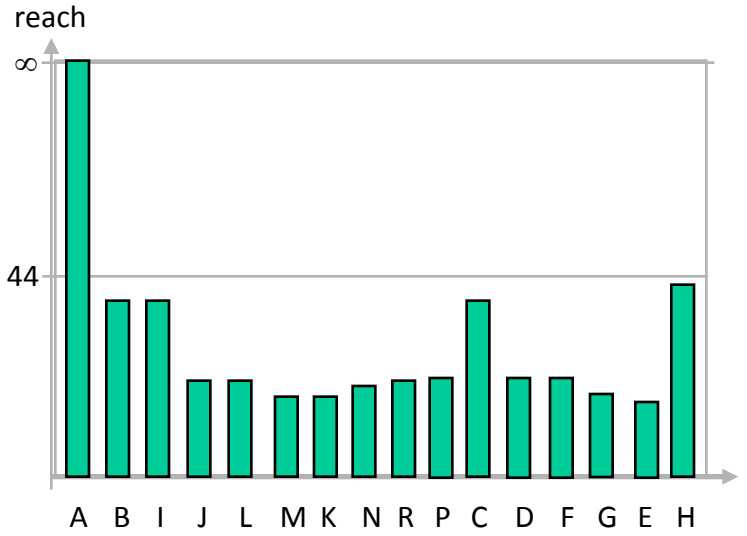


seed list: (H, 43)

Example 17

- Example Database (2-dimensional, 16 points)
- $\varepsilon = 44$, MinPts = 3

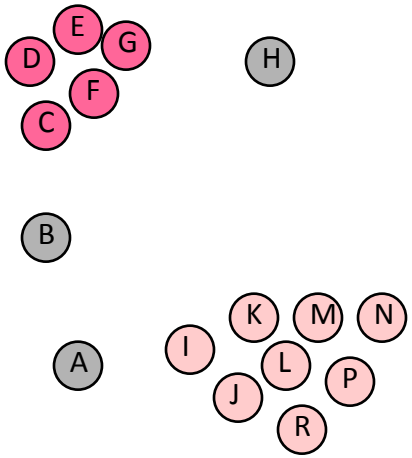


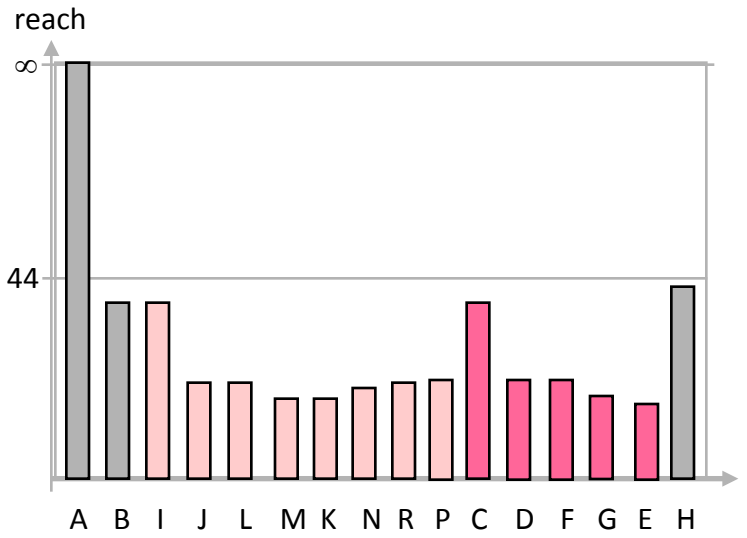


seed list: -

Example 18

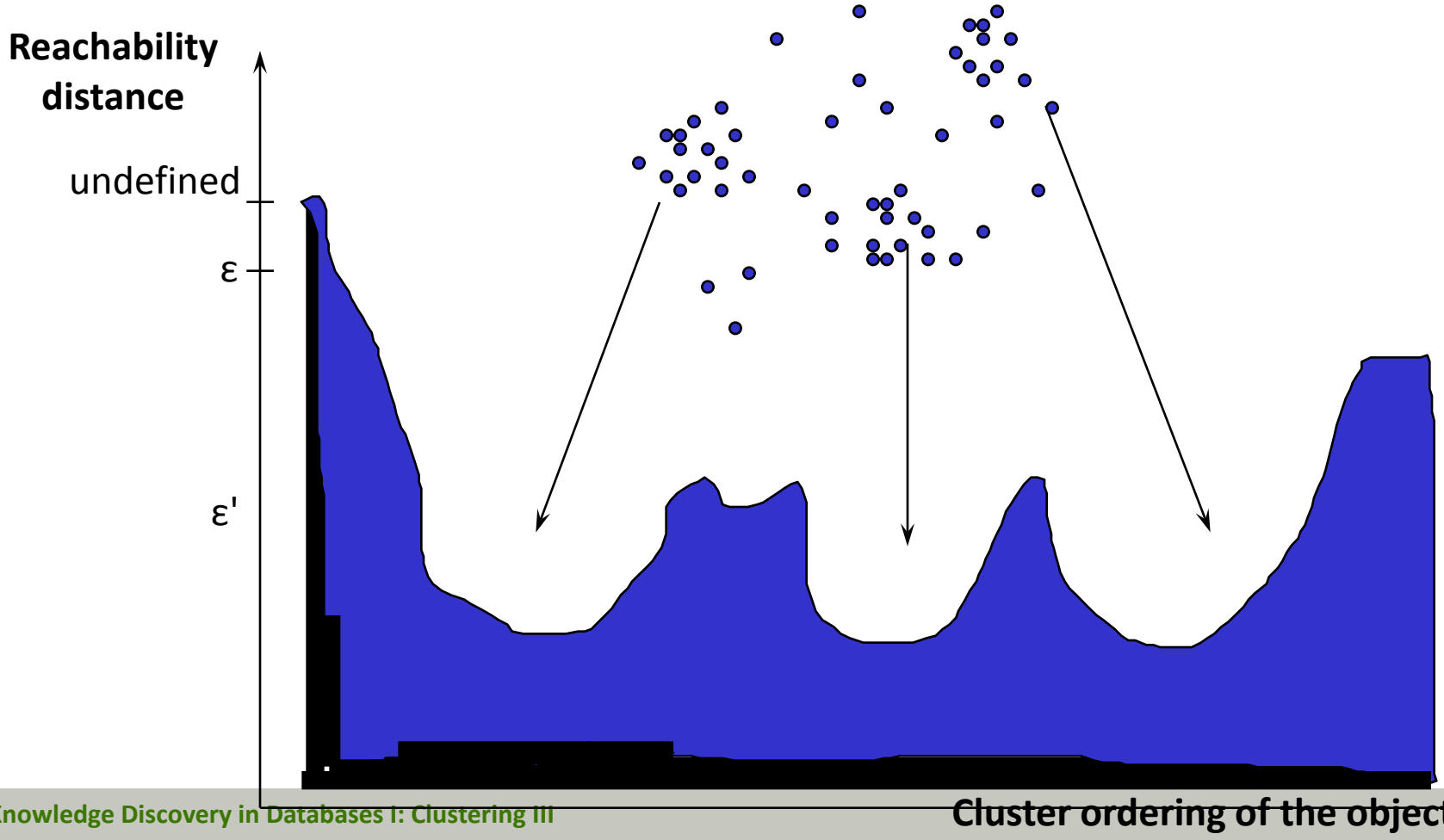
- Example Database (2-dimensional, 16 points)
- $\varepsilon = 44$, MinPts = 3



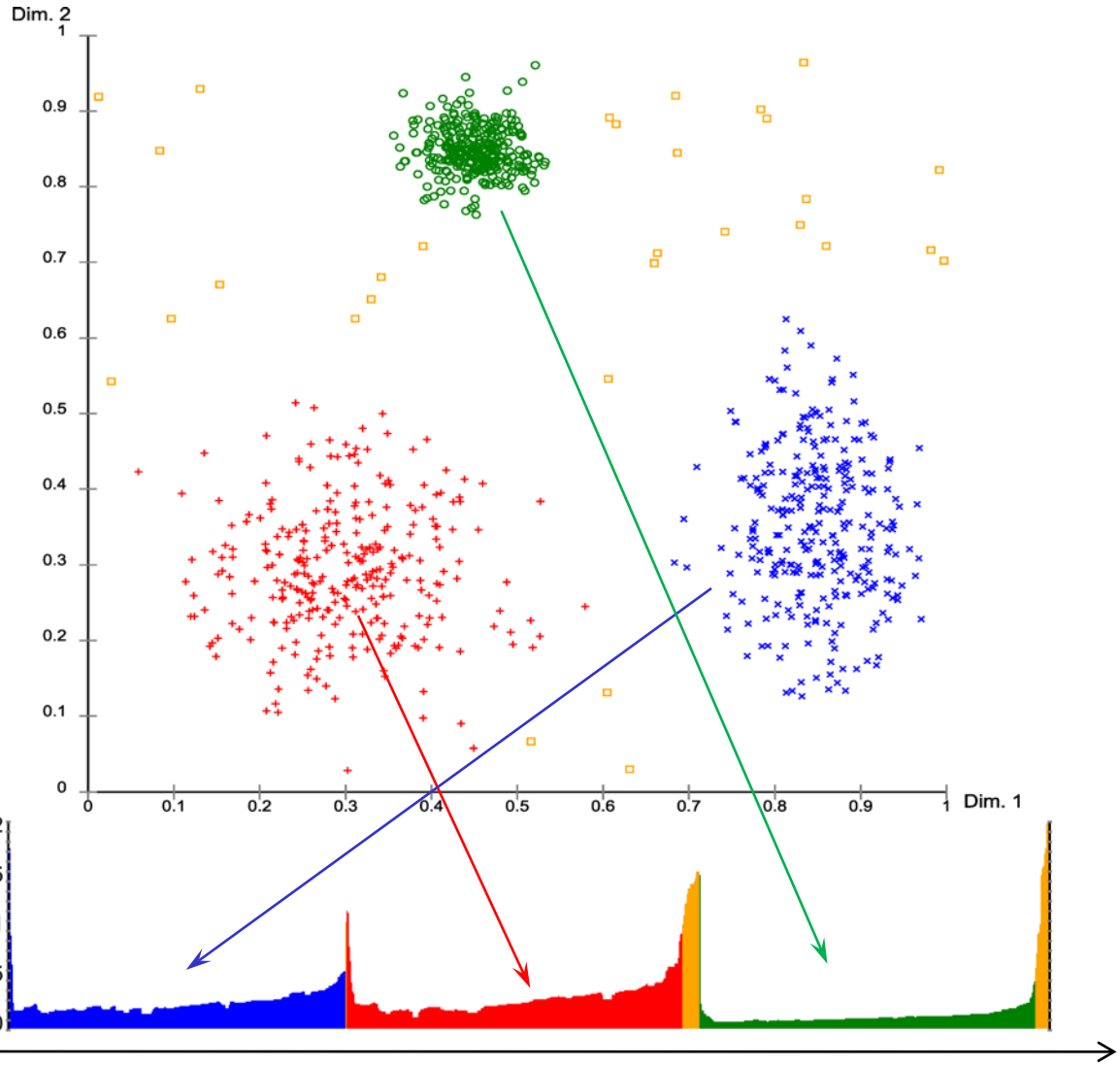


Reachability plot

- A 2D plot of objects ordering (x-axis) and reachability distance (y-axis)
- Clusters correspond to valleys in the plot since their cluster members have a low reachability distance to their nearest neighbor.
  - –The deeper the valley the denser the cluster



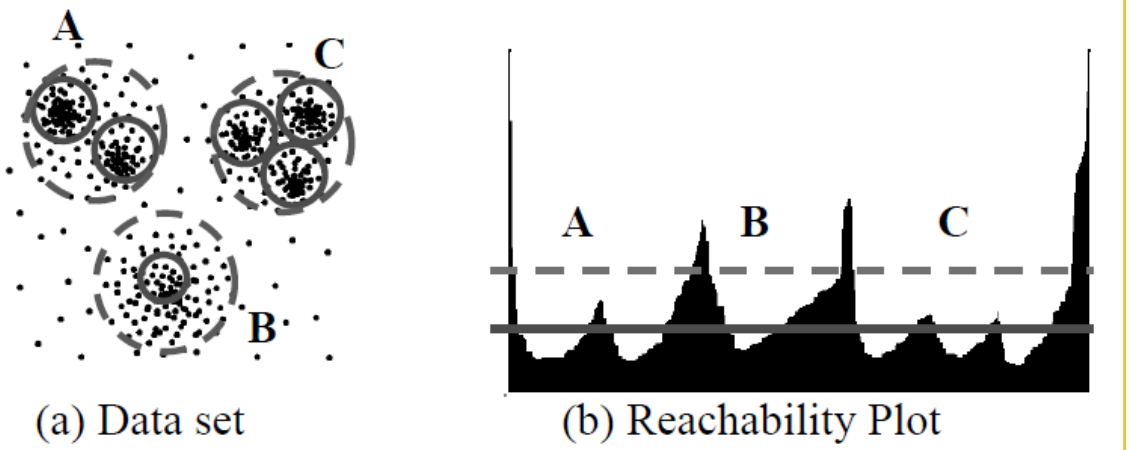
Reachability plot: another example



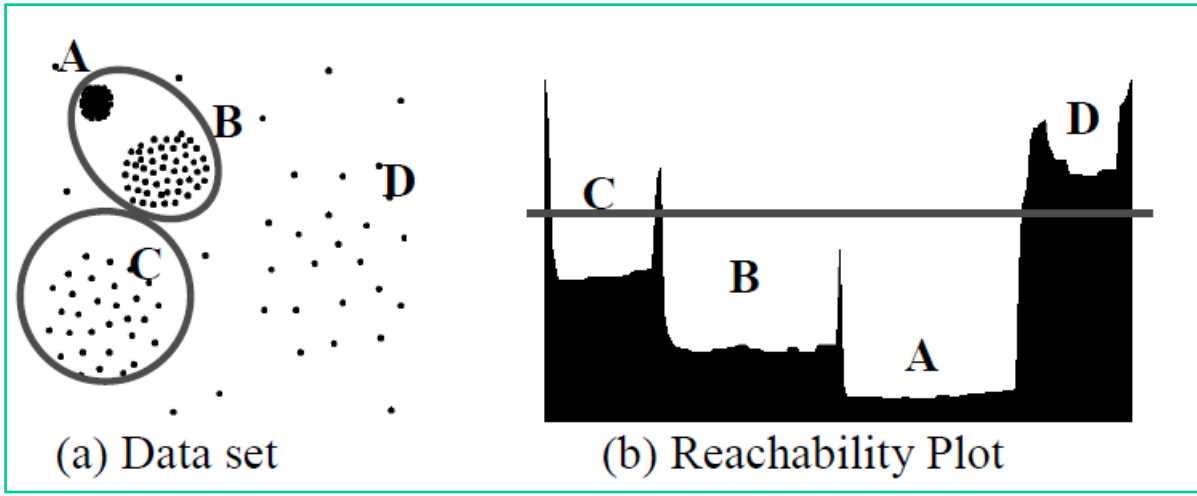
Reachability plot: an example with hierarchical clusters

Reachability plot: how to obtain a clustering?

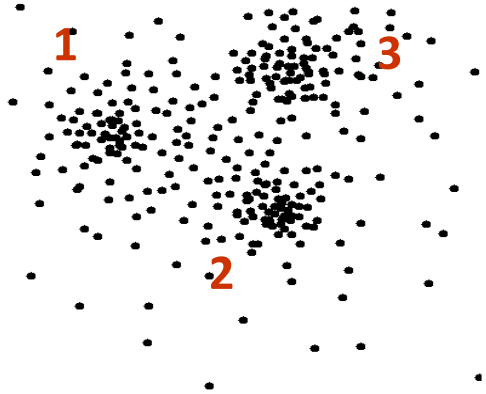
- Draw a horizontal line to obtain a clustering.



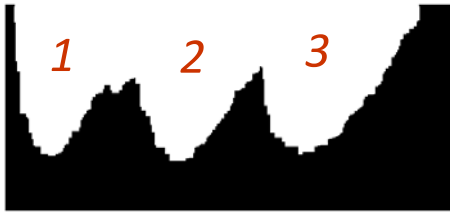
Depending on the data distribution, the lower the line is the more clusters would emerge



Parameters effect



MinPts = 10, ε = 10

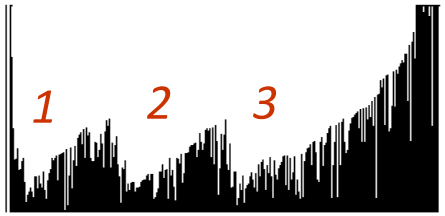


optimal parameters

MinPts = 10, ε = 5

smaller ε

MinPts = 2, ε = 10



smaller MinPts

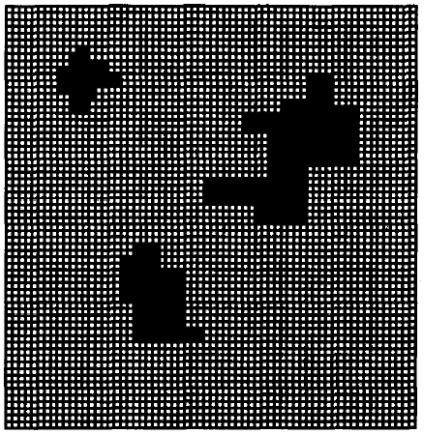
- Cluster ordering is robust to the parameter values ε and MinPts
  - Good results when parameter values are “large enough”
  - the smaller ε, the more objects may have undefined reachability distance
  - the smaller MinPts, the more jagged the plot looks
- Also, cluster ordering is independent from the dimension of the dataset

Outline

- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

Grid-based methods

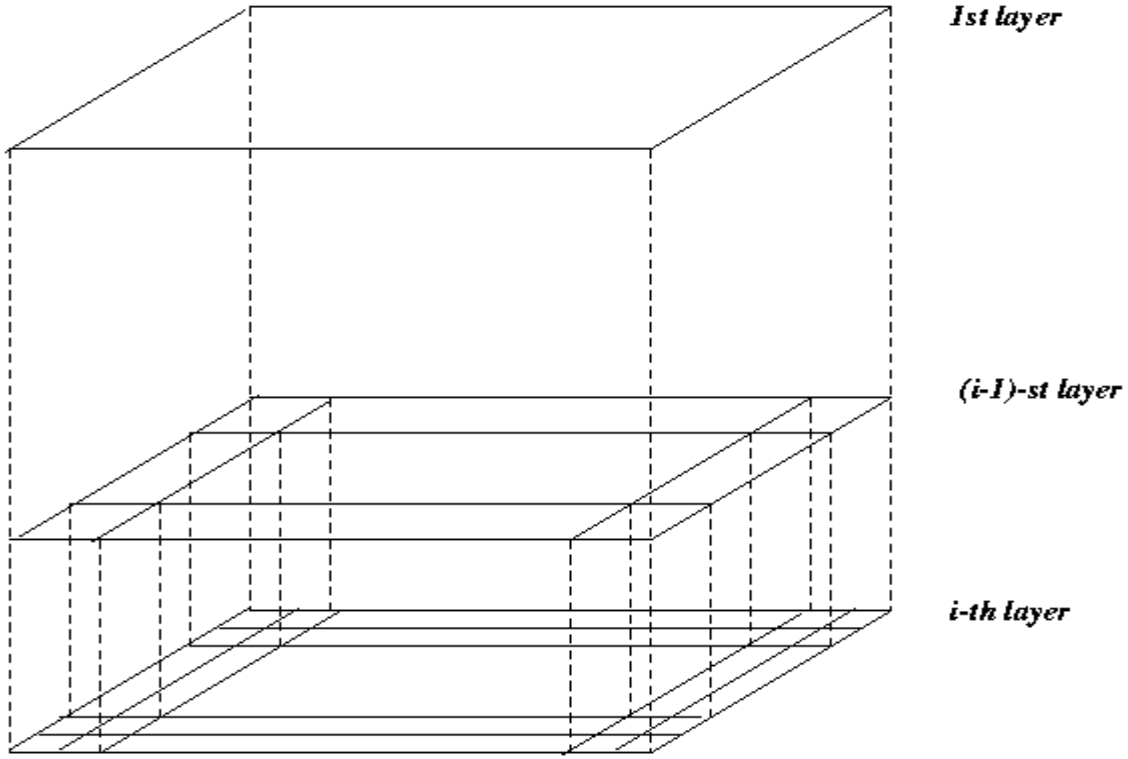
- A grid structure is used to capture the density of the dataset.
  - A cluster is a set of connected dense cells
  - STING (a STatistical INformation Grid approach) by Wang, Yang and Muntz (VLDB'97)
  - WaveCluster by Sheikholeslami, Chatterjee, and Zhang (VLDB'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98)
    - for high-dimensional data



- Appealing features
  - No assumption on the number of clusters
  - Discovering clusters of arbitrary shapes
  - Ability to handle outliers

STING I

- The spatial area is divided into rectangular cells
- There are several levels of cells corresponding to different levels of resolution



STING II

- 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 pre-computed and stored beforehand for query answering
- Parameters of higher level cells can be easily calculated from parameters of lower level cells
  - Count, mean, standard deviation, min, max
  - Type of distribution—normal, uniform, etc

STING III

- A top-down approach
- Start from a pre-selected layer—typically with a small number of cells
- 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

STING: overview

- 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

Outline

- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

Model-based clustering

- Assumption: data have been generated by a statistical process
- Goal: find the statistical model that best fits the data
- Each cluster can be seen as one distribution
  - e.g., Gaussian distribution
- Objects are assumed to be independent samples from their cluster distribution
- A particular kind of statistical model: Gaussian mixture models
- Procedure: decide on the model and find the parameters of that model from the data

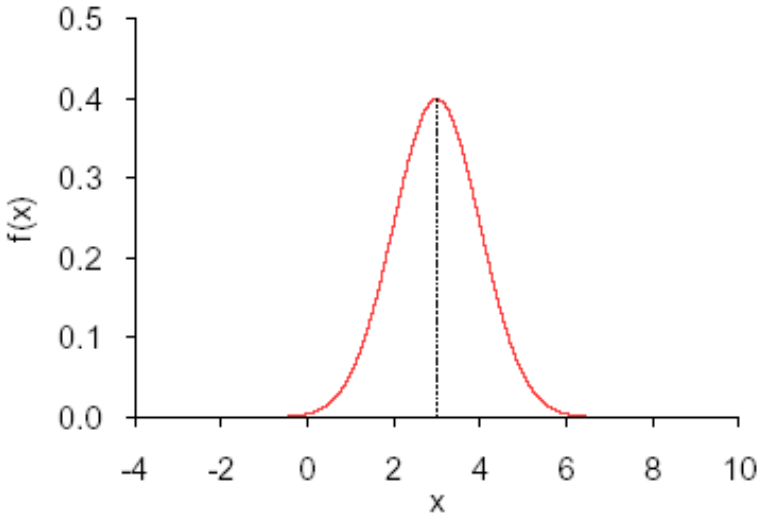
Gaussian Mixture Models

- Objects are points $x = (x_1, ..., x_d)$ in a Euclidean vector space
- Data are independent and identically distributed samples from a mixture of k distributions
- Each cluster is a multivariate Gaussian distribution
- Each cluster is represented by
  - Mean (centroid) μ_c
  - d x d covariance matrix Σ_c for the points in cluster c
- Probability density function of a Gaussian distribution

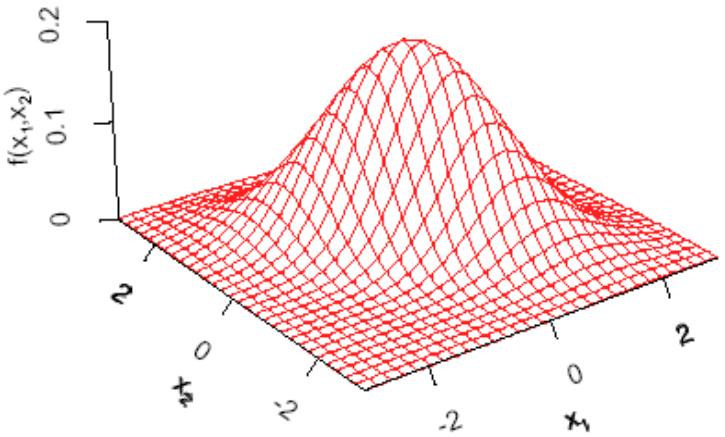
$$P(x \mid c) = \frac{1}{\sqrt{(2\pi)^d \mid \Sigma_c \mid}} e^{-\frac{1}{2}\cdot(x-\mu_c)^T \cdot \Sigma_c^{-1} \cdot (x-\mu_c)}$$

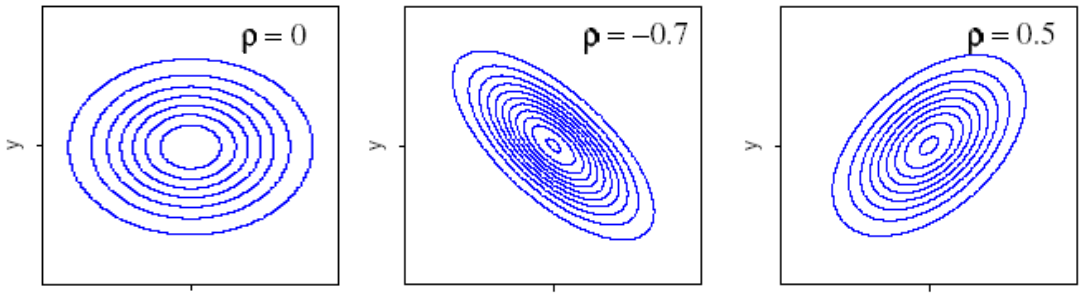
Multivariate normal distribution

Univariate normal distribution



Bivariate normal distribution





No covariance

Negative covariance

Positive covariance

Gaussian Mixture Models

- Probability of a cluster c

$$\mathrm{P}(c_l) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{P}(c_l | x_i)$$

- Probability of observing an object x_i

$$\mathrm{P}(x_i) = \sum_{l=1}^{k} \mathrm{P}(c_l)\mathrm{P}(x_i | c_l)$$

  - where $P(x_i | c_l)$ is given by the probability density function of the Gaussian distribution

Gaussian Mixture Models

- If the objects are generated in an independent manner, the probability of the whole set of objects X, |X|=N, is just the product of the probabilities of each x_i in X:

$$\mathcal{L} = \prod_{i=1}^{N} \mathrm{P}(x_i)$$

$$= \prod_{i=1}^{N} \sum_{l=1}^{k} \mathrm{P}(c_l)\mathrm{P}(x_i|c_l)$$

- Using statistical methods, we can estimate the parameters of these distributions from the data, and thus describe the clusters.

Gaussian Mixture Models clustering

- How can we partition the data?
  - Choose the most likely cluster assignment of each object

$$\operatorname*{argmax}_l \mathrm{P}(c_l|x_i) = \operatorname*{argmax}_l \mathrm{P}(c_l)\mathrm{P}(x_i|c_l)$$

- How to estimate the efficient statistics of each cluster?
  - Use Expectation – Maximization (EM) algorithm
  - Original algorithm by [Dempster, Laird and Rubin, 1977]
  - A general method for method for finding the maximum-likelihood estimate of a data distribution, when the data is partially missing or hidden.
    - In our case, data are fully observed
    - The cluster assignments of an object x_i though can be seen as hidden variables

EM algorithm I

- Initialize cluster assignments
- Two alternating steps:
  - E-step

  re-estimate the expected-values of the hidden data (cluster assignments) under the current estimate of the model
  - M-step

  re-estimate the model parameters such that the likelihood according to the current estimate of the complete data is maximized
- Until convergence

$$\frac{\mathcal{L}_{new}}{\mathcal{L}_{old}} < 1 + \epsilon$$

EM algorithm II

- E-step: re-estimate the expected-values of the hidden data (cluster assignments) under the current estimate of the model

$$\mathrm{P}^{new}(c_l|x_i) = \mathrm{P}(c_l)\mathrm{P}(x_i|c_l)$$

EM algorithm III

- M-step: re-estimate the model parameters such that the likelihood according to the current estimate of the complete data is maximized
  - Cluster densities:

$$\mathrm{P}^{new}(c_l) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{P}^{new}(c_l|x_i)$$

  - Cluster means:

$$\mu_l^{new} = \frac{\sum_{i=1}^{N} x_i \mathrm{P}^{new}(c_l|x_i)}{\sum_{i=1}^{N} \mathrm{P}^{new}(c_l|x_i)}$$

  - Cluster covariances:

$$\Sigma_l^{new} = \frac{\sum_{i=1}^{N} (x_i - \mu_l^{new})(x_i - \mu_l^{new})' \mathrm{P}^{new}(c_l|x_i)}{\sum_{i=1}^{N} \mathrm{P}^{new}(c_l|x_i)}$$

EM and k-Means

- EM is similar to the k-Means algorithm (previous lecture)
- k-Means for Euclidean data is a special case of EM for spherical Gaussian distributions with equal covariance matrices, but different means
- E-step (EM) → assign each object to a cluster step (k-Means)
  - In EM each object is assigned to a cluster with a probability
- M-step (EM) → compute cluster centroids step (k-Means)
  - In EM, the computation of the mean also considers the fact that each object belong to a distribution with a certain probability

EM (Gaussian Mixture Models) overview

- EM can be slow
- Not practical for models with a large number of components
- Problematic when clusters contain only a few points or if the points are nearly co-linear
- The choice of the exact model to use
- Difficulties with noise and outliers

+ More general than k-Means and fuzzy c-Means because they can use distributions of various types
+ Thus, it can find clusters of different sizes and elliptical shapes
+ It is easy to characterize the produced clusters

Outline

- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

Clustering

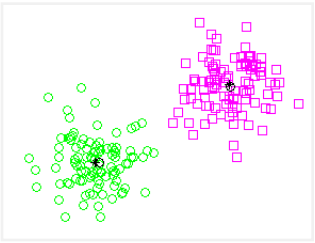
- A cluster is a set of data objects that are similar to one another within the same cluster and dissimilar to the objects in other clusters
- Cluster analysis: Find similarities between data according to the characteristics found in the data and group similar data objects into clusters
- Key points in clustering
  - Similarity/ distance function
  - Learning algorithm
- An unsupervised learning task
  - No clues on the number of clusters, nor in the characteristics of these clusters
- Important DM task: as a stand-alone tool or as a preprocessing step
- A large amount of algorithms
  - Partitioning methods
  - Hierarchical methods
  - Density-based methods
  - Model-based methods
  - ....

Clustering methods I

Partitioning methods

- Construct a partition of a database D of n objects into a set of k clusters
  - Each object belongs to exactly one cluster (hard clustering)
  - The number of clusters k is given in advance
- The partition should optimize the chosen partitioning criterion
  - e.g., minimize the intra-cluster variance, i.e., the sum of the squared distances from each data point to its cluster center.
- k-Means: choose a set of k points $\{c_1, c_2,\ldots,c_k\}$ in the d-dimensional space to form clusters $\{C_1, C_2,\ldots,C_k\}$ such that the following quantity is minimized

$$Cost(C) = \sum_{i=1}^{k} \sum_{x \in C_i} (x - c_i)^2$$

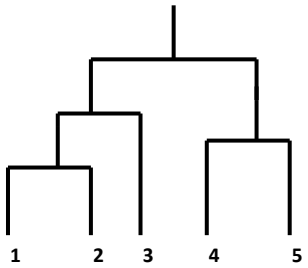


- k-Means (centroid) – k-Medoids (medoid)
- Other methods that scale to large datasets, e.g. CLARA, CLARANS

Clustering methods II

Hierarchical methods

- Create a hierarchical decomposition of the dataset. Not a single clustering but a set of nested clusters organized as a hierarchical tree (dendrogram)
- 2 ways: Agglomerative (bottom up) – Divisive (top down)
- How to merge (split) and when to stop?
- Inter-cluster similarity:
  - single link,
  - complete link,
  - group average,
  - centroid,
  - Ward's method,
  - ….



Clustering methods III

Density-based clustering

- Clusters are regions of high density surrounded by regions of low density (noise)
- Density is measured locally in Eps-neighborhood
- DBSCAN
  - minPts, Eps parameters
  - Core points, border points, noise points
  - Direct reachability, reachability, connectivity, cluster
- OPTICS
  - Cluster ordering
  - Core distance, reachability distance
  - Reachability plot

Clustering methods IV

Grid-based clustering

- A grid structure is used to capture the dataset distribution
- Work in the grid, after mapping the points to the grid

Model-based clustering

- Data have been generated by a statistical process, the goal is to find the statistical model that best fits the data
- EM algorithm

Outline

- Introduction
- A categorization of major clustering methods
- Density-based methods cont'
- Grid-based methods
- Model-based methods
- An overview of clustering
- Things you should know
- Homework/tutorial

Things you should know

- Density-based methods cont'
  - OPTICS
    - core-distance, reachability distance
    - Reachability plot
- Grid-based methods
  - STING
- Model-based methods
  - EM

Homework/ Tutorial

Tutorial: Tutorial this Thursday on clustering

Homework:

- Try OPTICS in Weka, Elki
  - Can you interpret the reachability plot?
  - What if you change some parameter, ε, MinPts?

Suggested reading:

- Tan P.-N., Steinbach M., Kumar V., *Introduction to Data Mining*, Addison-Wesley, 2006 (Chapter 9).
- Han J., Kamber M., Pei J. *Data Mining: Concepts and Techniques 3rd ed., Morgan Kaufmann, 2011* (Chapter 10)