Clustering

CS 145
Fall 2015
Wei Wang

Challenges of Hierarchical Clustering Methods

- Hard to choose merge/split points
  - Never undo merging/splitting
  - Merging/splitting decisions are critical
- Do not scale well: $O(n^2)$
- What is the bottleneck when the data can't fit in memory?
- Integrating hierarchical clustering with other techniques
  - BIRCH, CURE, CHAMELEON, ROCK

BIRCH

- Balanced Iterative Reducing and Clustering using Hierarchies
- CF (Clustering Feature) tree: a hierarchical data structure summarizing object info
  - Clustering objects → clustering leaf nodes of the CF tree

Clustering Feature Vector

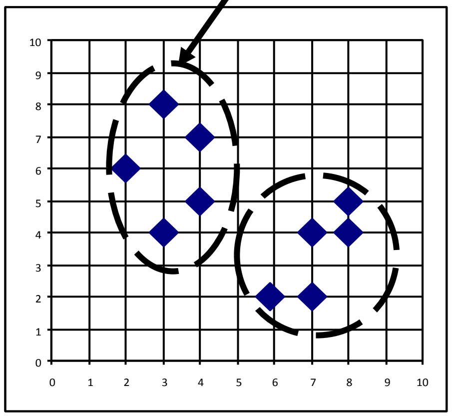
Clustering Feature: $CF = (N, \overrightarrow{LS}, \overrightarrow{SS})$

N: **Number of data points**

LS: $\sum_{i=1}^{N} \overrightarrow{X_i}$

SS: $\sum_{i=1}^{N} \overrightarrow{X_i^2}$

CF = (5, (16,30),(54,190))

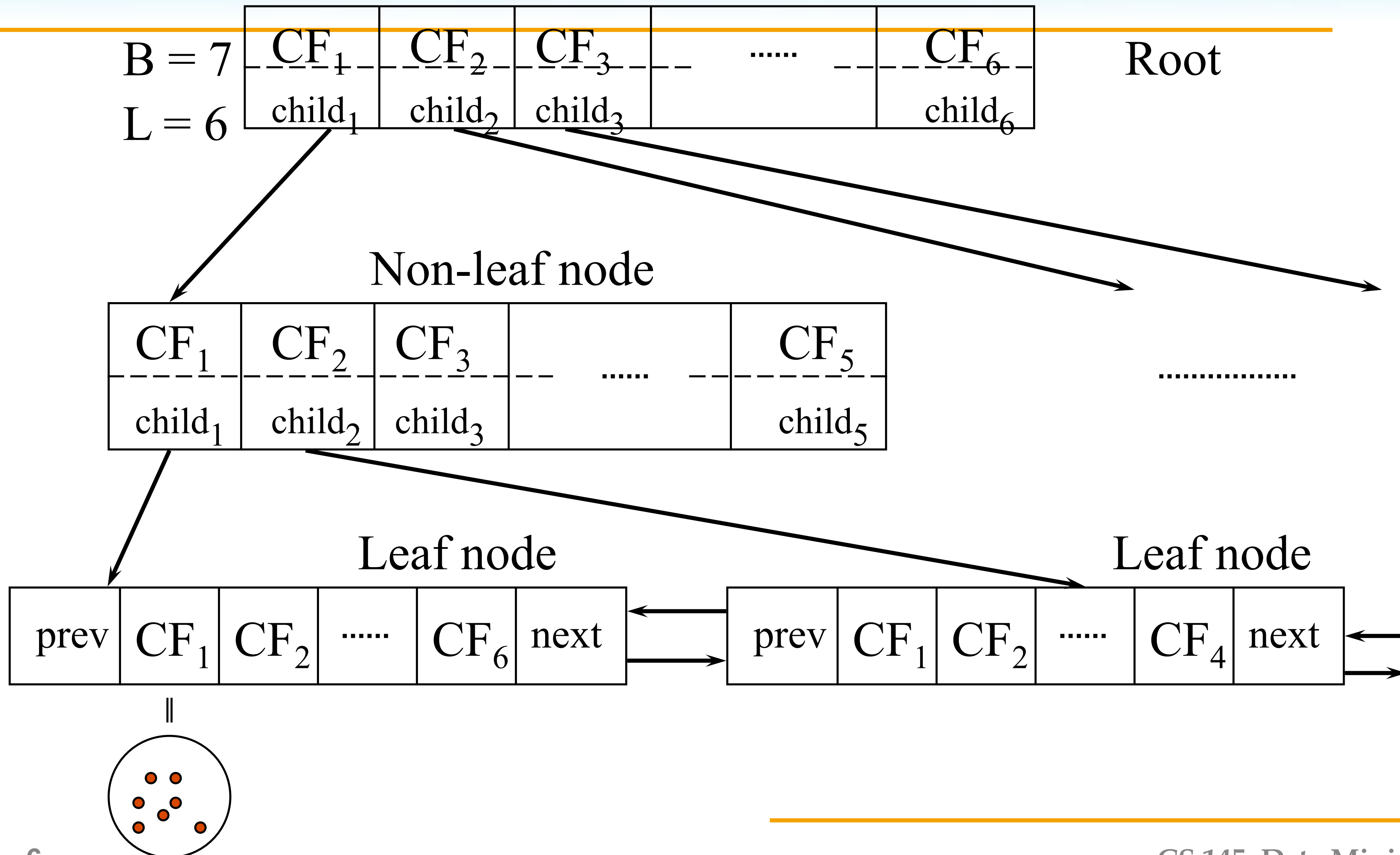


(3, 4)
(2, 6)
(4, 5)
(4, 7)
(3, 8)

CF-tree in BIRCH

- Clustering feature:
  - Summarize the statistics for a subcluster: the 0^{th}, 1^{st} and 2^{nd} moments of the subcluster
  - Register crucial measurements for computing cluster and utilize storage efficiently
- A CF tree: a height-balanced tree storing the clustering features for a hierarchical clustering
  - A nonleaf node in a tree has descendants or “children”
  - The nonleaf nodes store sums of the CFs of children

CF Tree



B = 7
L = 6
CF1 child1
CF2 child2
CF3 child3
......
CF6 child6
Root
Non-leaf node
CF1 child1
CF2 child2
CF3 child3
......
CF5 child5
....
Leaf node
prev CF1 CF2 ...... CF6 next
Leaf node
prev CF1 CF2 ...... CF4 next

Parameters of A CF-tree

- Branching factor: the maximum number of children
- Threshold: max diameter of sub-clusters stored at the leaf nodes

BIRCH Clustering

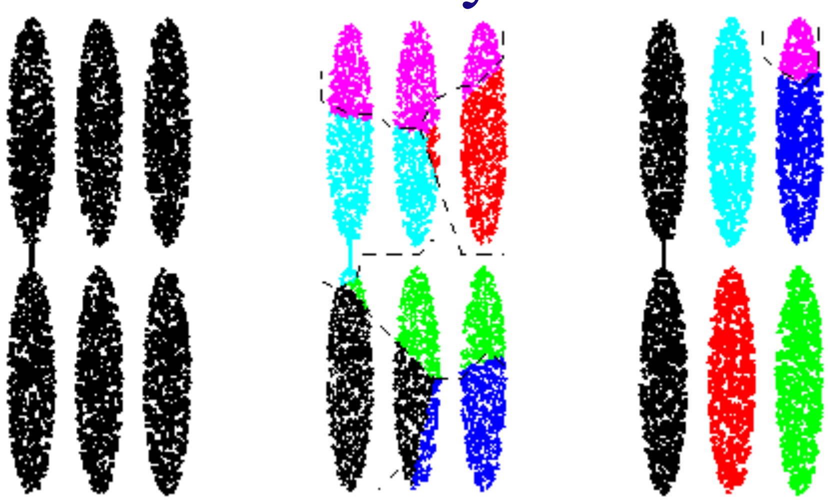
- Phase 1: scan DB to build an initial in-memory CF tree (a multi-level compression of the data that tries to preserve the inherent clustering structure of the data)
- Phase 2: use an arbitrary clustering algorithm to cluster the leaf nodes of the CF-tree

Pros & Cons of BIRCH

- Linear scalability
  - Good clustering with a single scan
  - Quality can be further improved by a few additional scans
- Can handle only numeric data
- Sensitive to the order of the data records

Drawbacks of Square Error Based Methods

- One representative per cluster
  - Good only for convex shaped having similar size and density
- A number of clusters parameter k
  - Good only if k can be reasonably estimated

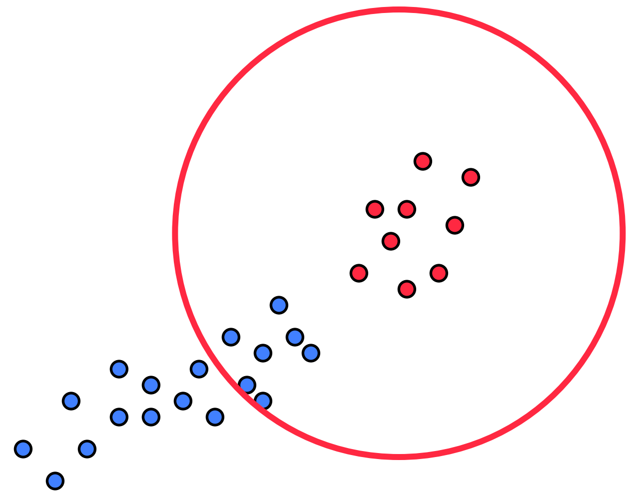


(a) (b) (c)

(a) (b)

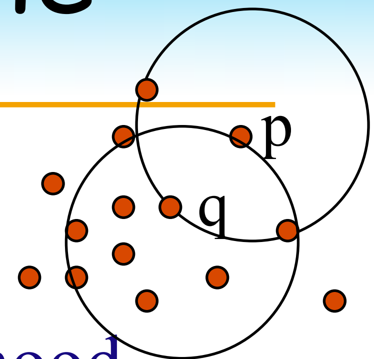
Drawback of Distance-based Methods

- Hard to find clusters with irregular shapes
- Hard to specify the number of clusters
- Heuristic: a cluster must be dense



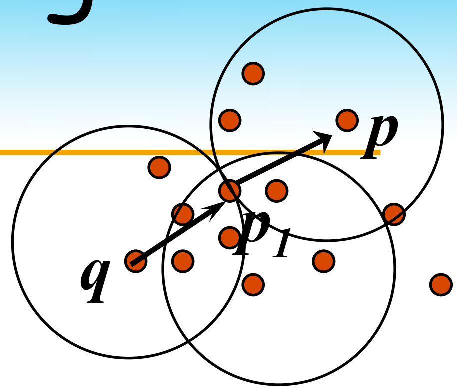
Directly Density Reachable

MinPts = 3
Eps = 1 cm

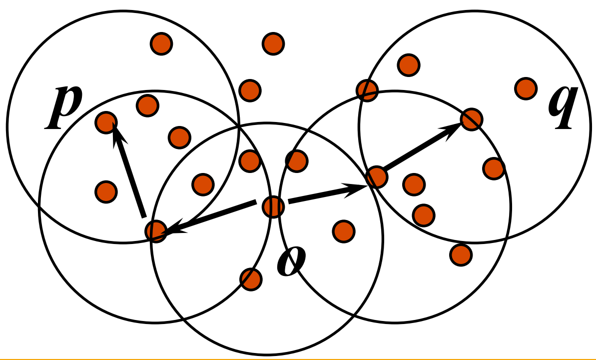


- Parameters
  - Eps: Maximum radius of the neighborhood
  - MinPts: Minimum number of points in an Eps-neighborhood of that point
  - NEps(p): $\{q \mid dist(p,q) \leq Eps\}$
- Core object p: $|Neps(p)| \geq MinPts$
- Point q directly density-reachable from p iff $q \in Neps(p)$ and p is a core object

Density-Based Clustering: Background (II)

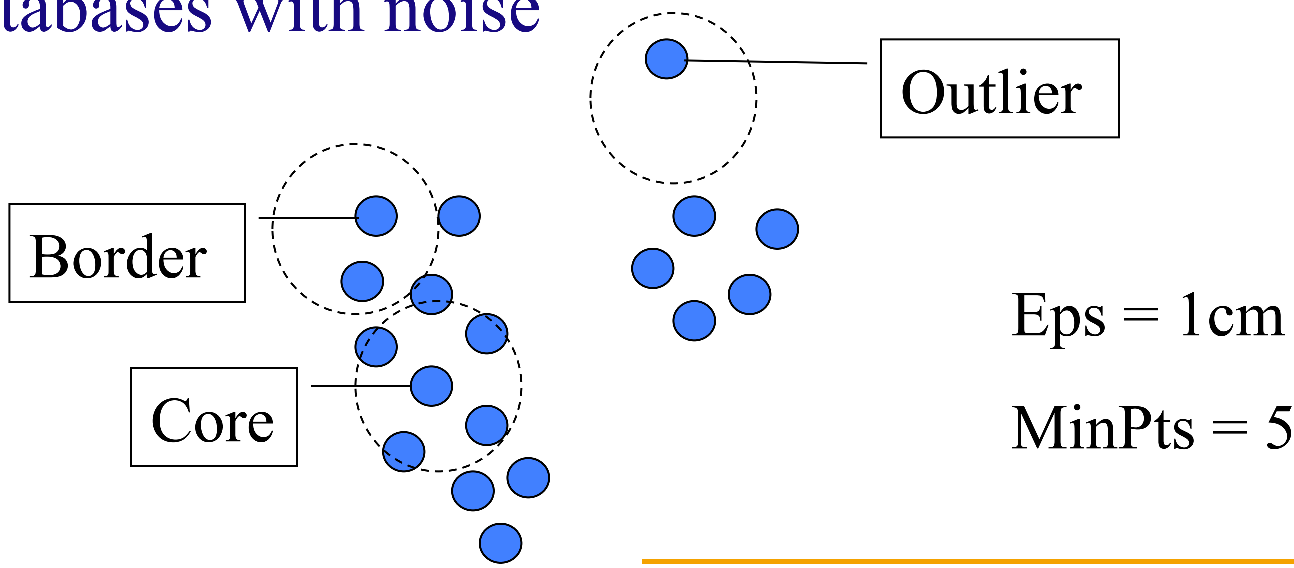


- Density-reachable
  - Directly density reachable $p_1 \rightarrow p_2$, $p_2 \rightarrow p_3$, …, $p_{n-1} \rightarrow p_n$ ➔ p_n density-reachable from p_1
- Density-connected
  - Points p, q are density-reachable from o ➔ p and q are density-connected



DBSCAN

- A cluster: a maximal set of density-connected points
  - Discover clusters of arbitrary shape in spatial databases with noise

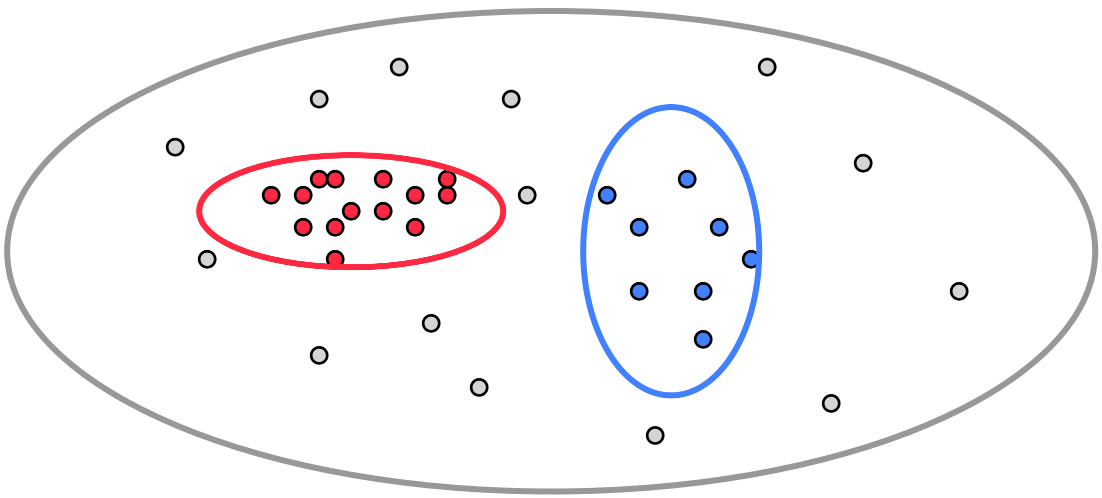


DBSCAN: the Algorithm

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

Problems of DBSCAN

- Different clusters may have very different densities
- Clusters may be in hierarchies



DBSCAN: Sensitive to Parameters

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

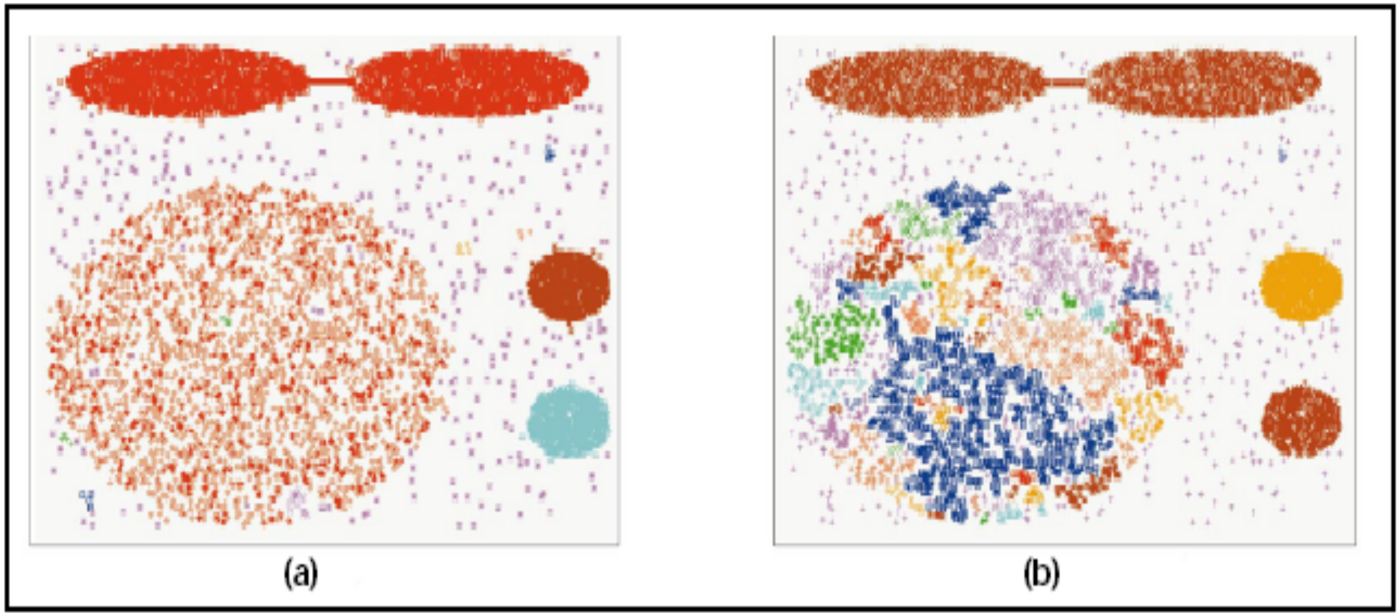
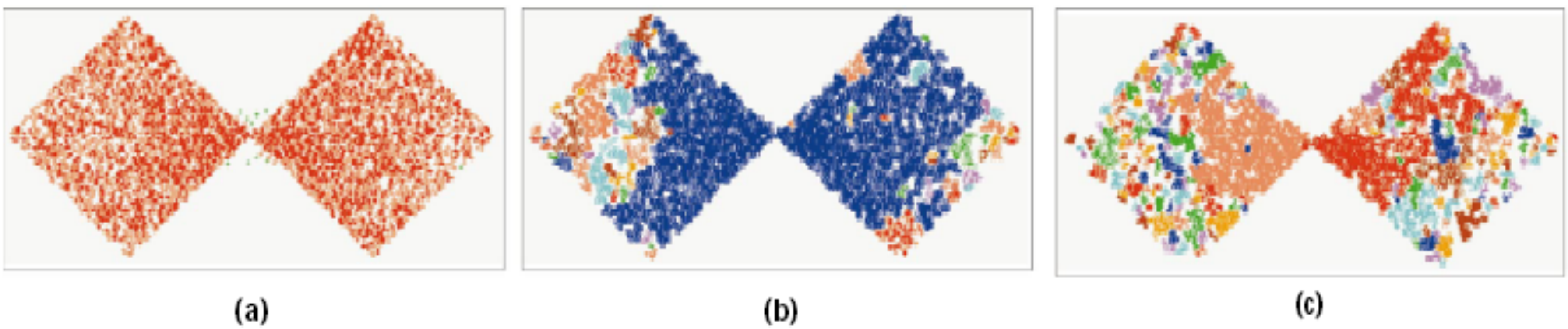


Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.



DBSCAN online Demo:

OPTICS: A Cluster-Ordering Method (1999)

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

OPTICS: Some Extension from DBSCAN

- Index-based: k = # of dimensions, N: # of points
  - Complexity: O(N*logN)
- Core Distance of an object p: the smallest value ε such that the ε-neighborhood of p has at least MinPts objects

  Let $N_\varepsilon(p)$: ε-neighborhood of p, ε is a distance value

  $\text{Core-distance}_{\varepsilon,\ MinPts}(p)$ = Undefined if $card(N_\varepsilon(p)) < MinPts$

  MinPts-distance(p), otherwise
- Reachability Distance of object p from core object q is the min radius value that makes p density-reachable from q

  $\text{Reachability-distance}_{\varepsilon,\ MinPts}(p, q)$ =

  Undefined if q is not a core object

  max(core-distance(q), distance (q, p)), otherwise

Core Distance & Reachability Distance

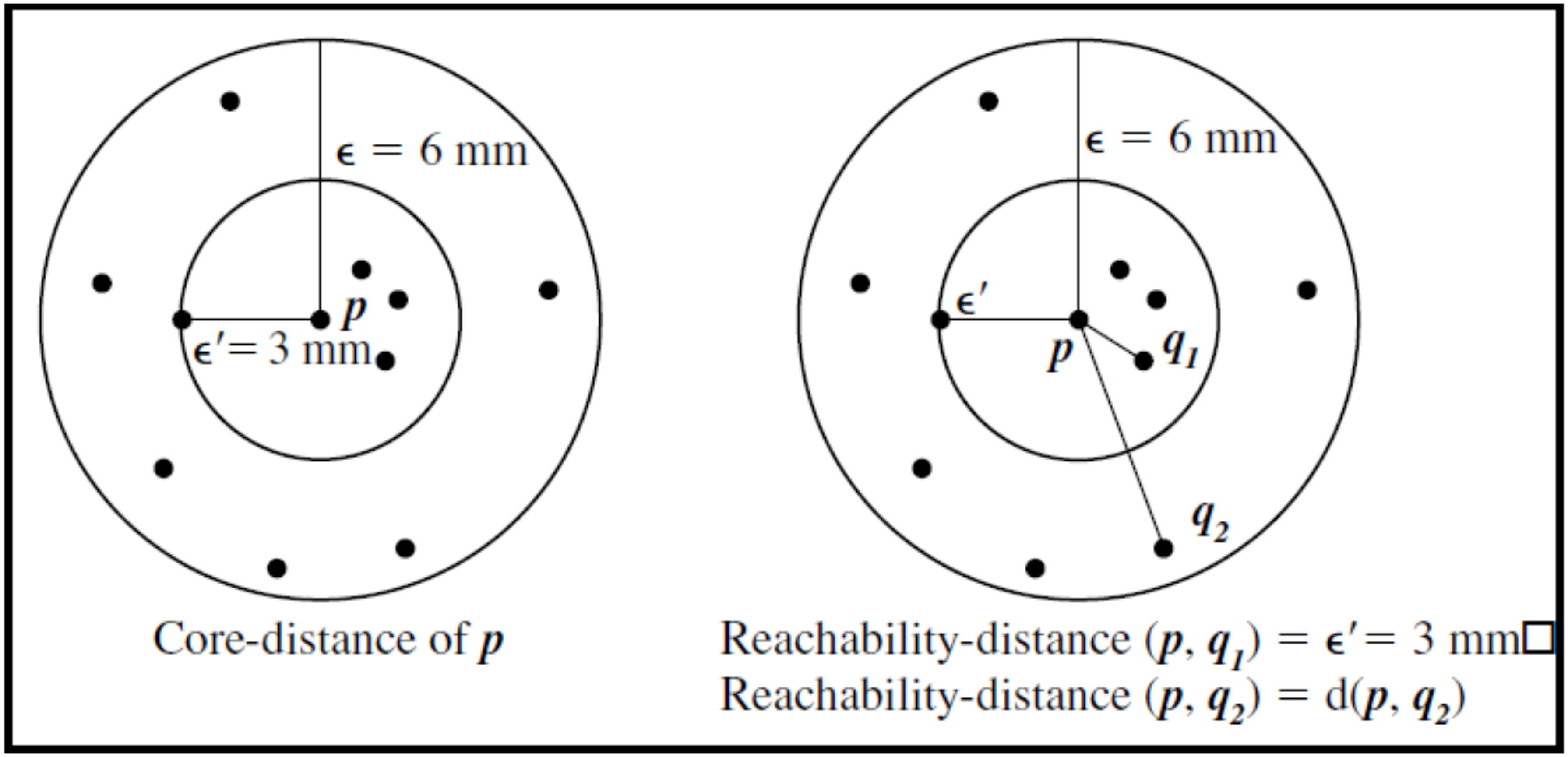
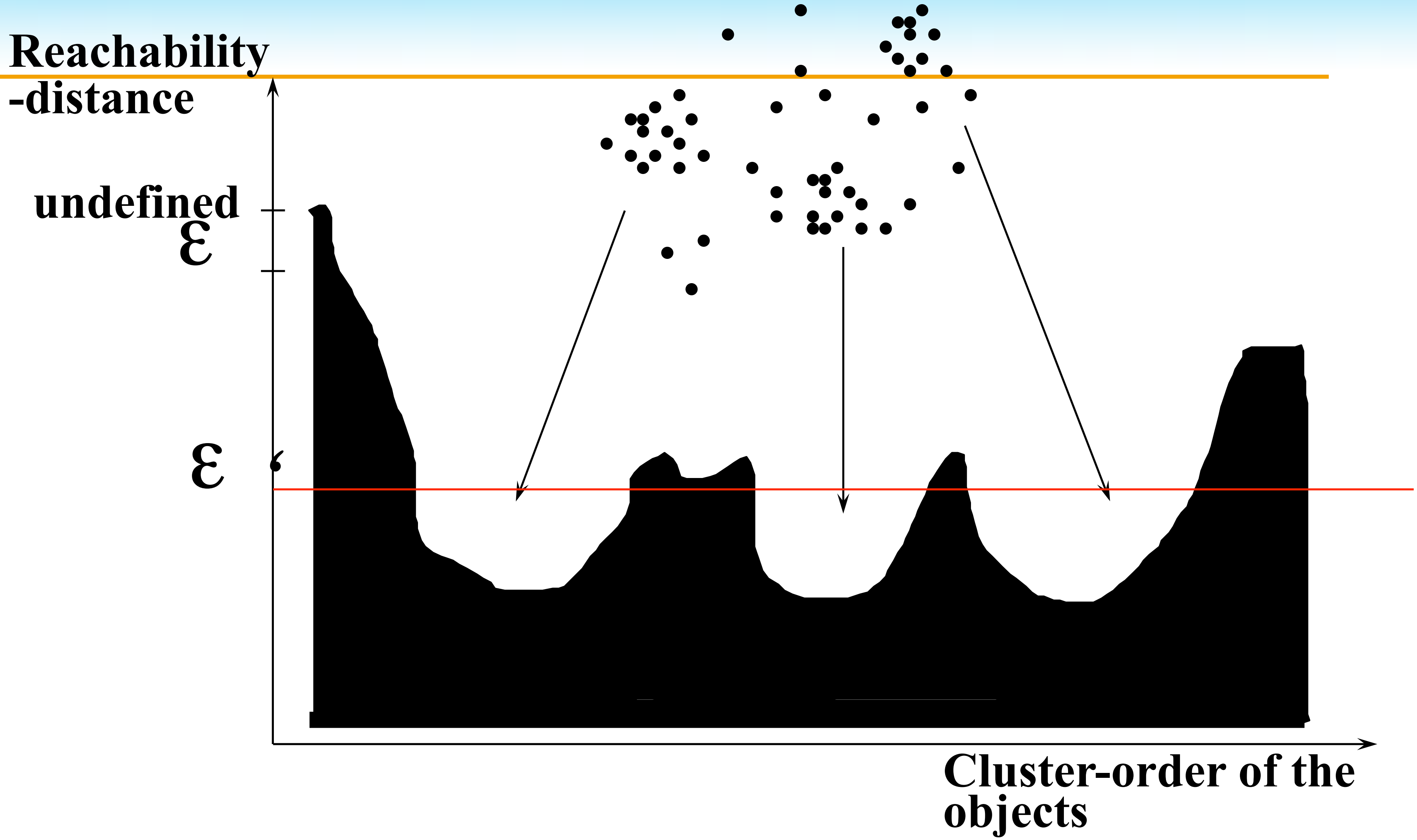


Figure 10.16: OPTICS terminology. Based on [ABKS99].



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

Density-Based Clustering: OPTICS & Applications

demo: http://www.dbs.informatik.uni-muenchen.de/Forschung/KDD/Clustering/OPTICS/Demo