Clustering II: Spatial Clustering

Roadmap

- Density Based Clustering
- DBSCAN
  - Concepts
  - Algorithm
  - Comments
- Take-home messages

Density-based Approaches*

- Why Density-Based Clustering methods?
  - Discover clusters of arbitrary shape
  - Clusters – Dense regions of objects separated by regions of low density
- DBSCAN – the first density based clustering
- Other methods:
  - OPTICS – density based cluster-ordering
  - DENCLUE – a general density-based description of cluster and clustering

DBSCAN:

Density Based Spatial Clustering of Applications with Noise

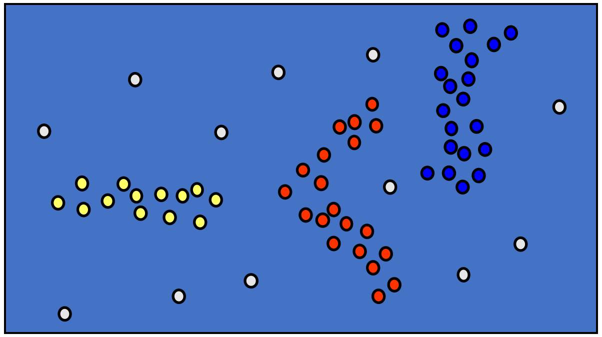
- Proposed by Ester, Kriegel, Sander, and Xu (KDD96)
- Relies on a density-based notion of cluster: A cluster is defined as a maximal set of density-connected points.
- Discovers clusters of arbitrary shape in spatial databases with noise

Visualization tool:
https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

Density-Based Clustering

- *Basic Idea*:

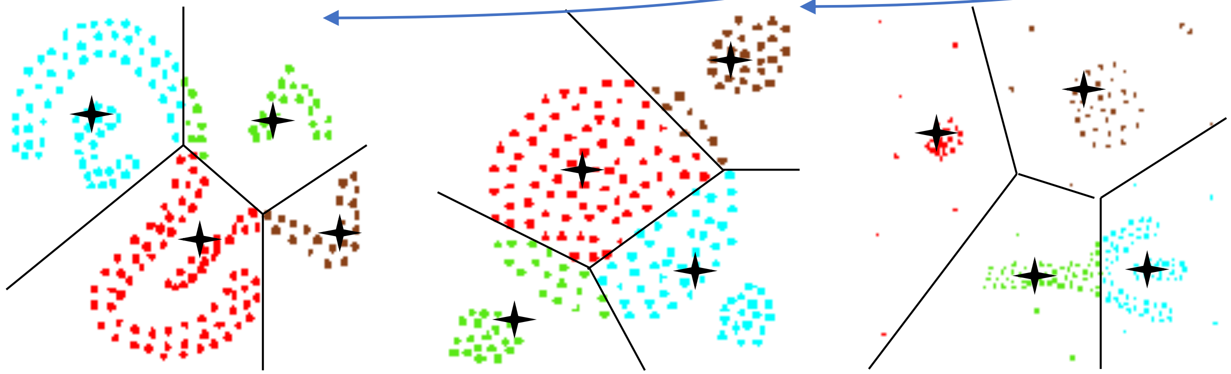
  Clusters are dense regions in the data space, separated by regions of lower object density



- Why Density-Based Clustering?

Results of a *k*-medoid algorithm for *k*=4

Are these reasonable?



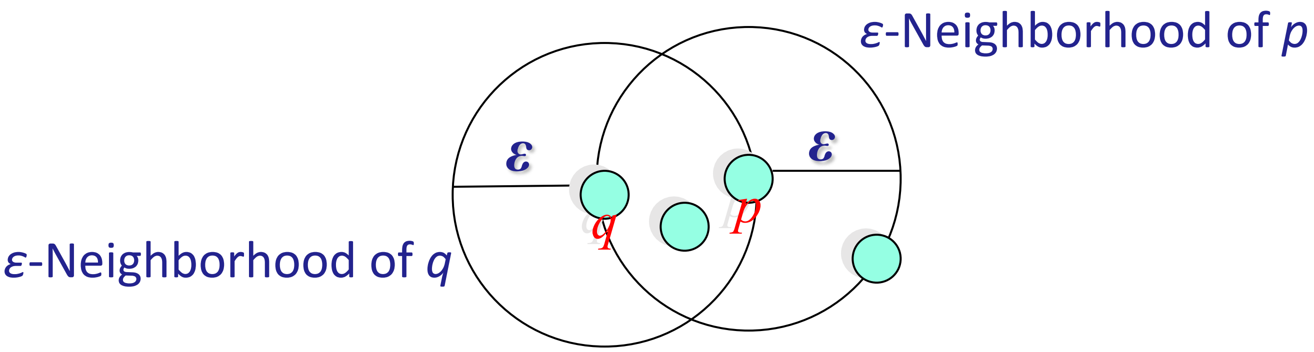
Different density-based approaches exist (see Textbook & Papers)
Here, we discuss the ideas underlying the DBSCAN algorithm

Density Based Clustering: Basic Concept

- Intuition for the formalization of the basic idea
  - For any point in a cluster, the local point density around that point has to exceed some threshold
  - The set of points from one cluster is spatially connected
- Local point density at a point p defined by two parameters
  - ε – radius for the neighborhood of point p:
    $N_\varepsilon(p) := \{q \text{ in data set } D \mid dist(p, q) \leq \varepsilon\}$
  - *MinPts* – minimum number of points in the given neighbourhood $N(p)$

ε-Neighborhood

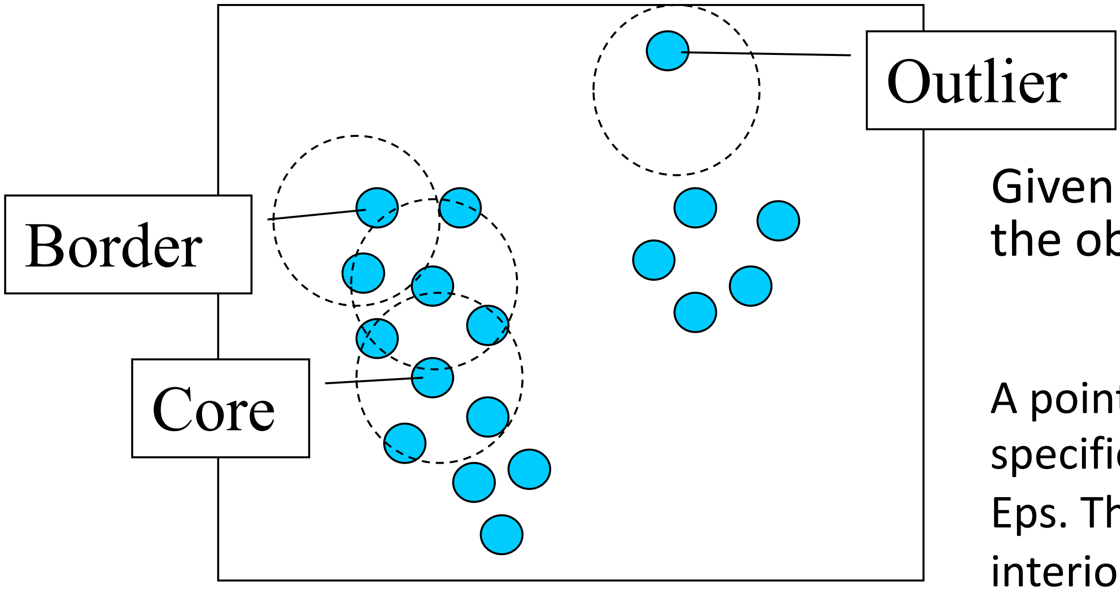
- ε-Neighborhood – Objects within a radius of ε from an object.
$$N_\varepsilon(p):\{q \mid d(p,q) \le \varepsilon\}$$
- "High density" – ε-Neighborhood of an object contains at least *MinPts* of objects.



Density of p is "high" (MinPts = 4)

Density of q is "low" (MinPts = 4)

Core, Border & Outlier



$\varepsilon = 1$ unit, MinPts = 5

Given ε and *MinPts*, we can categorize the objects into three exclusive groups.

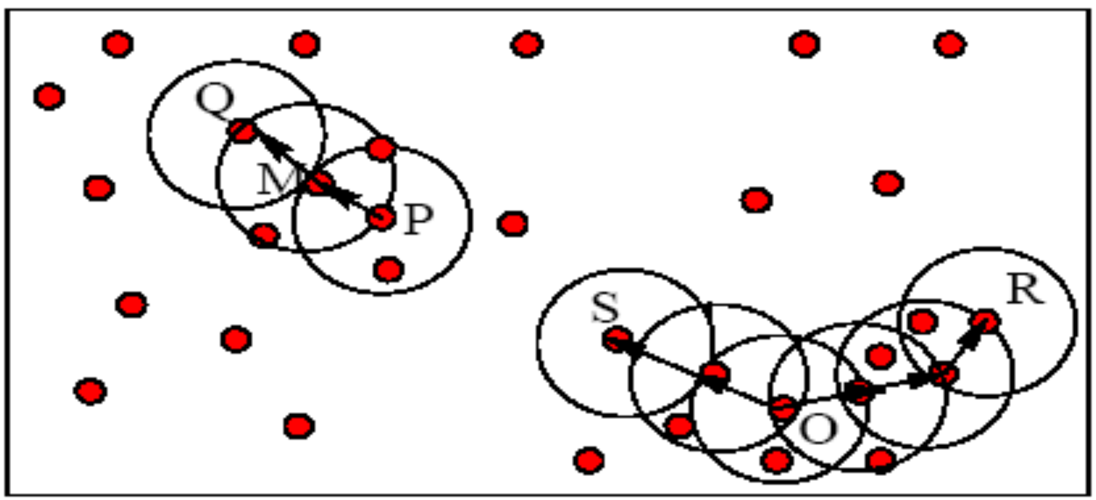
A point is a core point if it has more than a specified number of points (MinPts) within Eps. These are points that are at the interior of a cluster.

A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point.

A noise point/outlier is any point that is not a core point nor a border point.

Example

- M, P, O, and R are core objects (out of Q, M, P, S, O, R) since each of them is in an ε-neighborhood containing at least 3 points

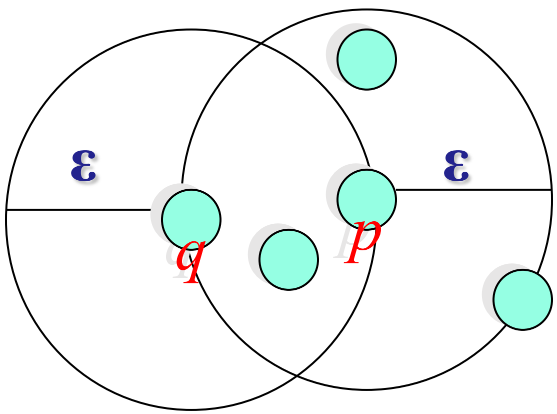


MinPts = 3

ε = radius of the circles

Density-Reachability

- Directly density-reachable
  - An object q is directly density-reachable from object p if p is a core object and q is in p's ε-neighborhood.

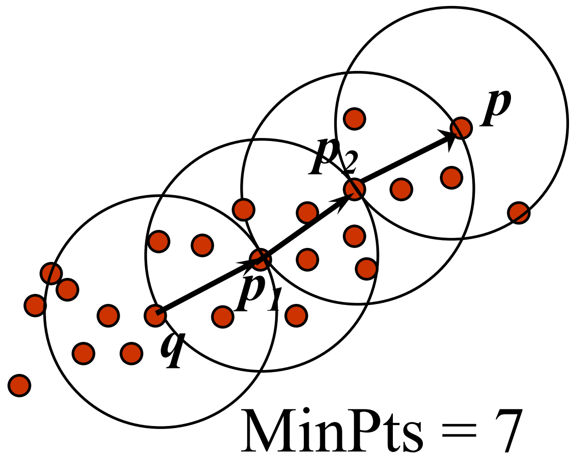


MinPts = 4

- q is directly density-reachable from p
- p is not directly density-reachable from q
- Density-reachability is asymmetric.

Density-reachability

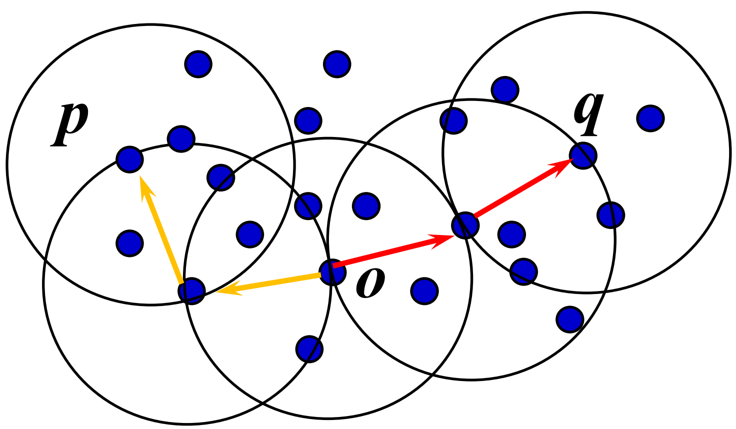
- Density-Reachable (directly and indirectly):
  - A point p is directly density-reachable from p_2;
  - p_2 is directly density-reachable from p_1;
  - p_1 is directly density-reachable from q;
  - $p \leftarrow p_2 \leftarrow p_1 \leftarrow q$ form a chain.



- p is (indirectly) density-reachable from q
- Is q not density-reachable from p? Yes, check whether p_2 is directly density-reachable from p

Density-Reachability ➜ Density-Connectivity

- Density-Reachable is not symmetric
  - not good enough to describe clusters
- Density-Connected
  - A pair of points *p* and *q* are density-connected if they are commonly density-reachable from a point *o*.



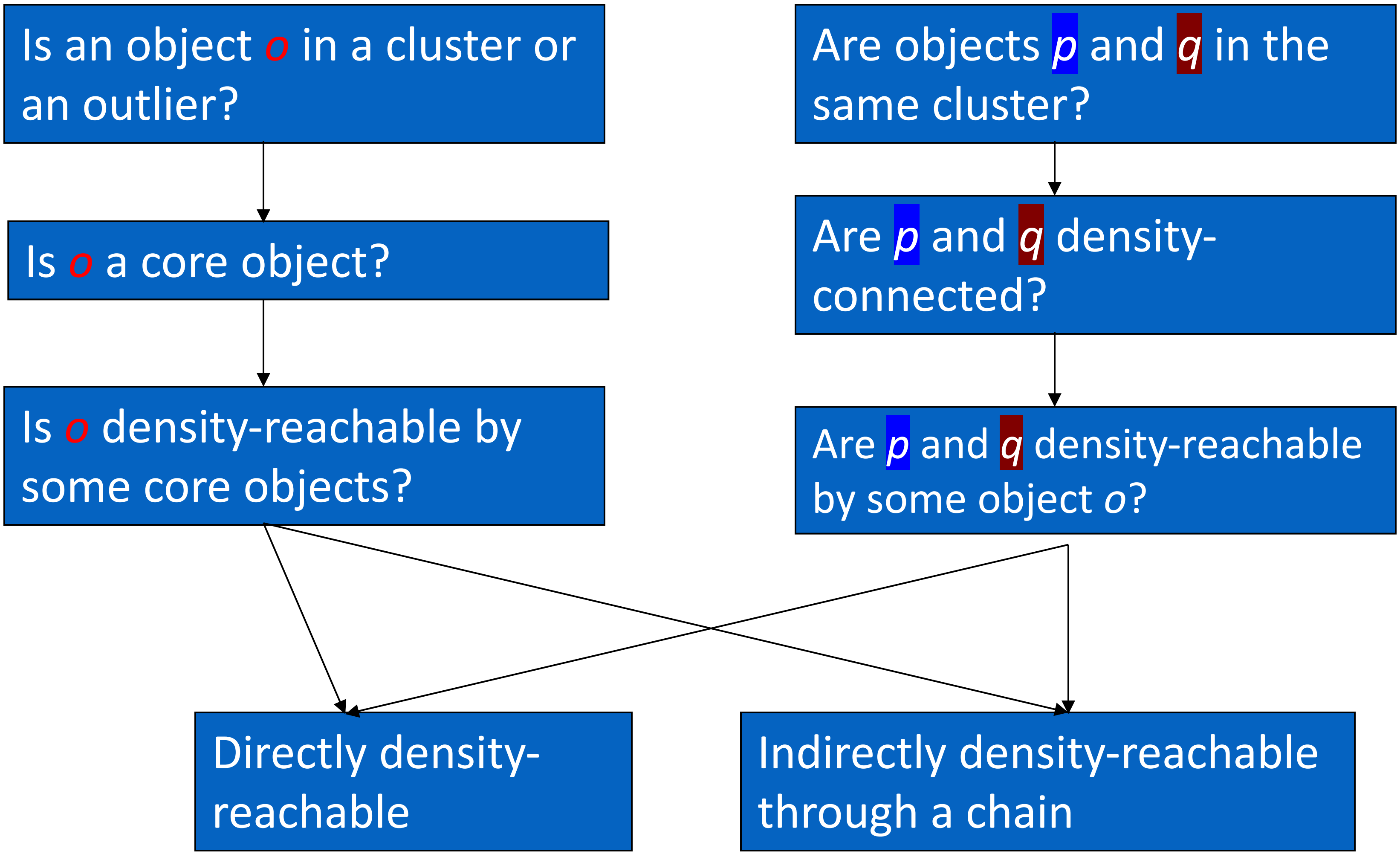
- Density-connectivity is symmetric

Formal Description of Cluster

- Given a data set *D*, parameter ε and threshold MinPts.
- A cluster *C* is a subset of objects (with core and border points) satisfying two criteria:
  - *Connected:* ∀ *p*,*q* ∈ *C*: *p* and *q* are density-connected.
  - *Maximal:* ∀ *p*,*q*: if *p* ∈ *C* and if *q* is density-reachable from *p*, then *q* ∈ *C*.

Density-reachable ---> *p* is a core point

Review of Concepts

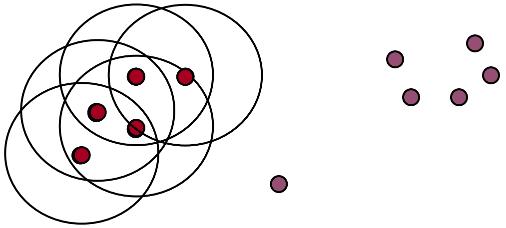


DBSCAN: The Algorithm

1. Arbitrary select a point *p*
2. Retrieve all points density-reachable from *p* wrt *Eps* (ε) and *MinPts*.
3. If *p* is a core point, a cluster is formed.
4. If *p* is a border point, no points are density-reachable from *p* and DBSCAN visits the next point of the database.
5. Continue the process until all of the points have been processed.

DBSCAN Algorithm: Example

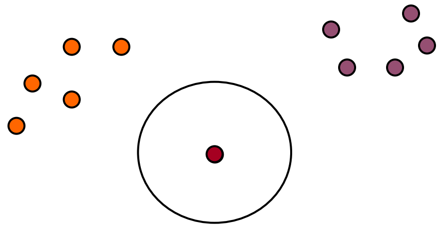
- Parameter
  - ε = 2 cm
  - *MinPts* = 3



for each $o \in D$ **do**
  if o is not yet classified **then**
    if o is a core-object **then**
      collect all objects density-reachable from o
      and assign them to a new cluster.
    else
      assign o to NOISE

DBSCAN Algorithm: Example

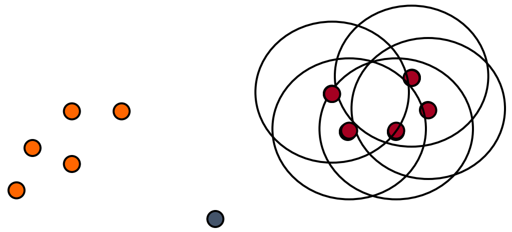
- Parameter
  - ε = 2 cm
  - *MinPts* = 3



for each $o \in D$ **do**
  if o is not yet classified **then**
    if o is a core-object **then**
      collect all objects density-reachable from o
      and assign them to a new cluster.
    else
      assign o to NOISE

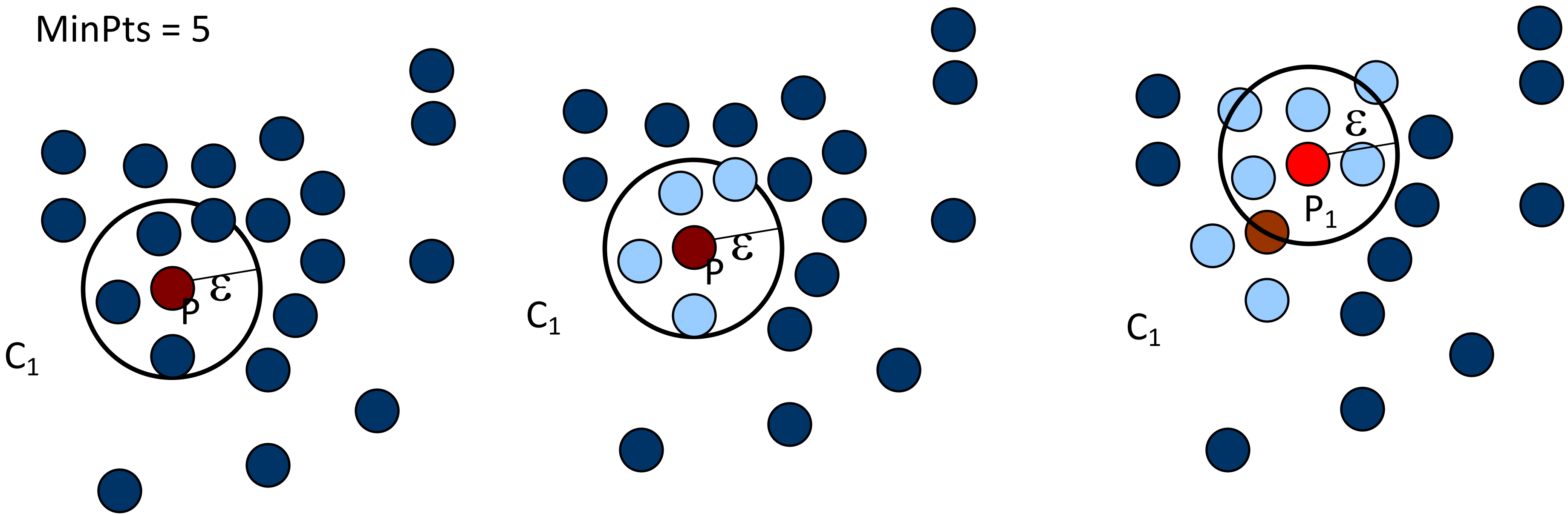
DBSCAN Algorithm: Example

- Parameter
  - ε = 2 cm
  - *MinPts* = 3



for each $o \in D$ **do**
    if o is not yet classified **then**
        if o is a core-object **then**
            collect all objects density-reachable from o
            and assign them to a new cluster.
        else
            assign o to NOISE

More elaborations

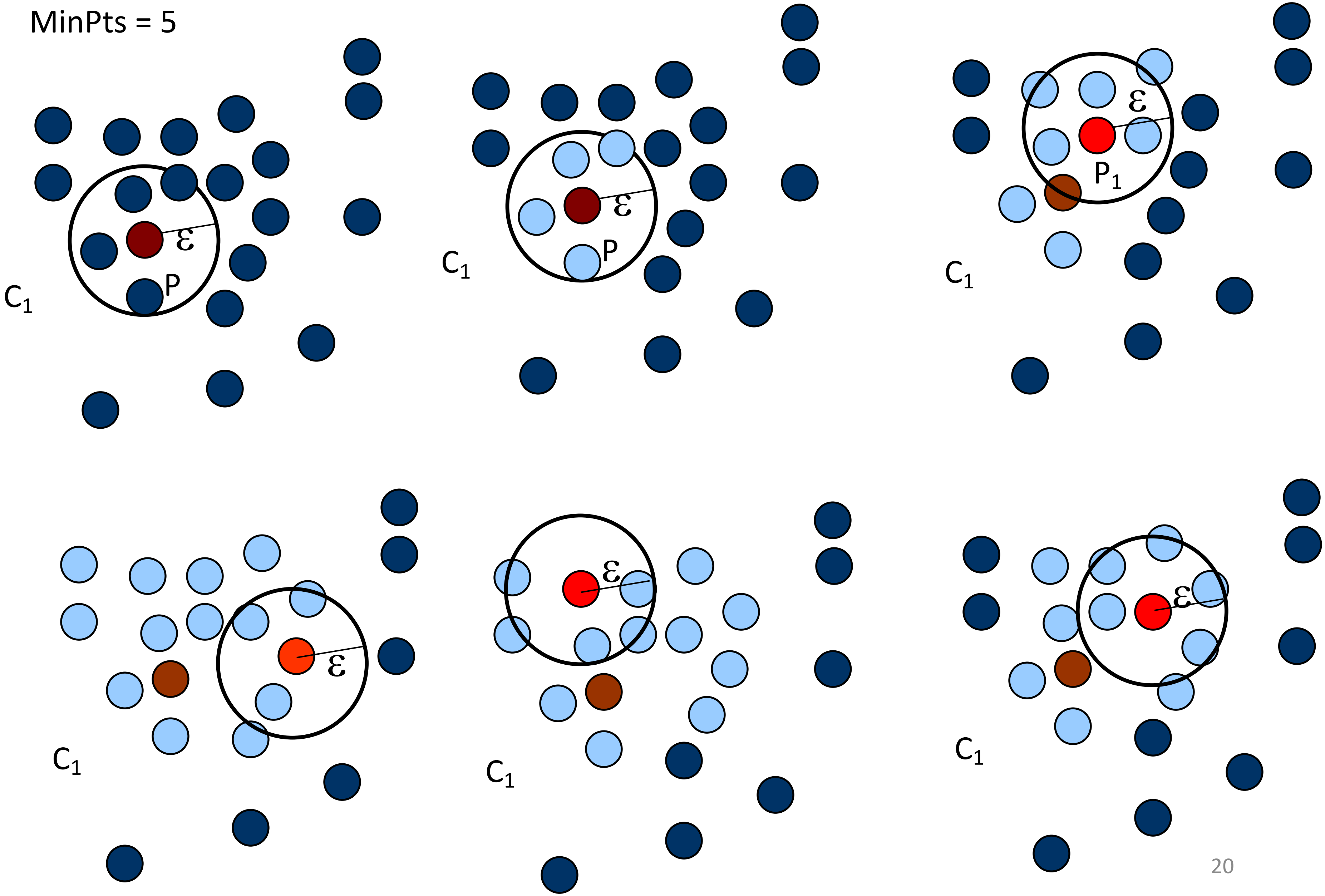


1. Check the ε-neighborhood of p;
2. If p has less than MinPts neighbors then mark p as outlier and continue with the next object
3. Otherwise mark p as processed and put all the neighbors in cluster C

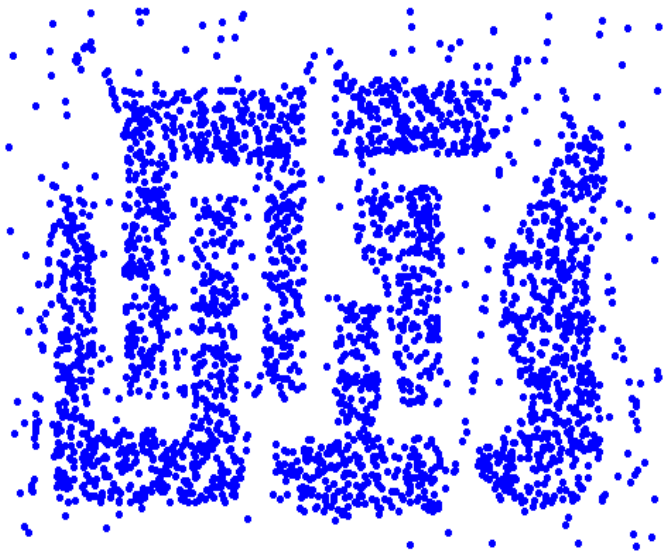
1. Check the unprocessed objects in C (light blue dots)
2. If no core object, return C
3. Otherwise, randomly pick up one core object p_1, mark p_1 as processed, and put all unprocessed neighbors of p_1 in cluster C

More elaborations

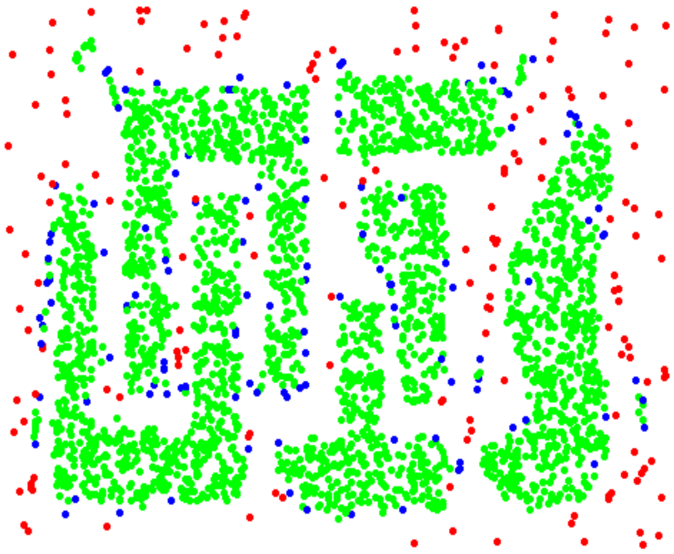
MinPts = 5



Pictorial Example



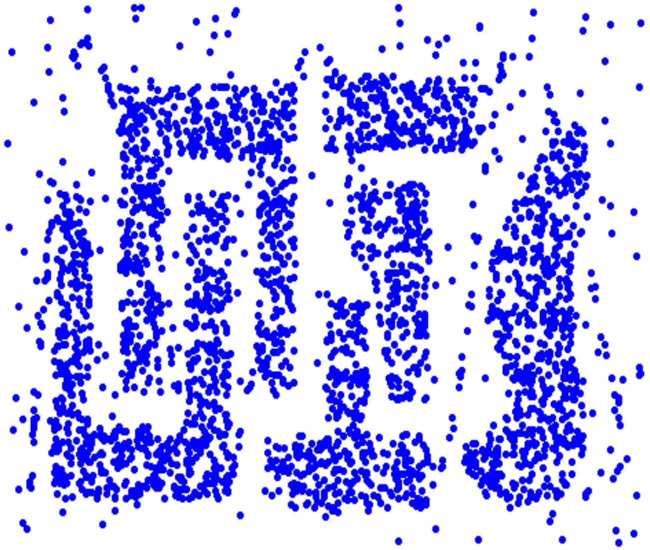
Original Points



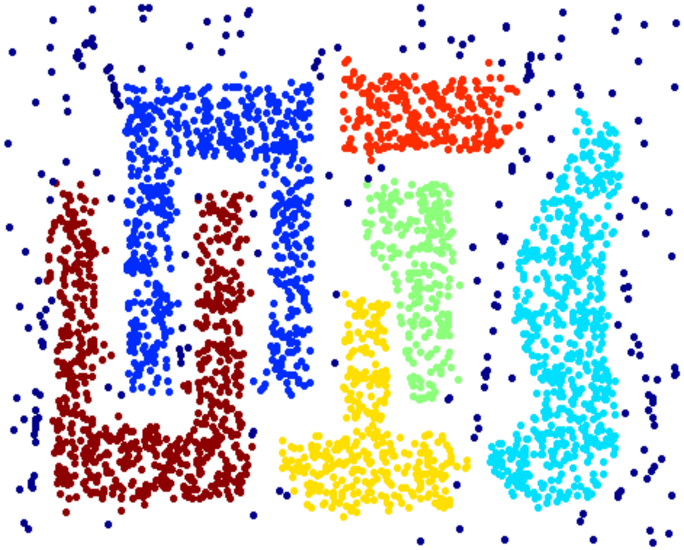
Point types: core, border and outliers

ε = 10, MinPts = 4

When DBSCAN Works Well…



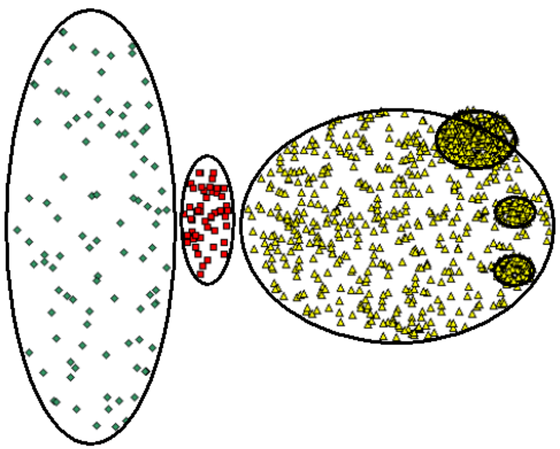
Original Points



Clusters

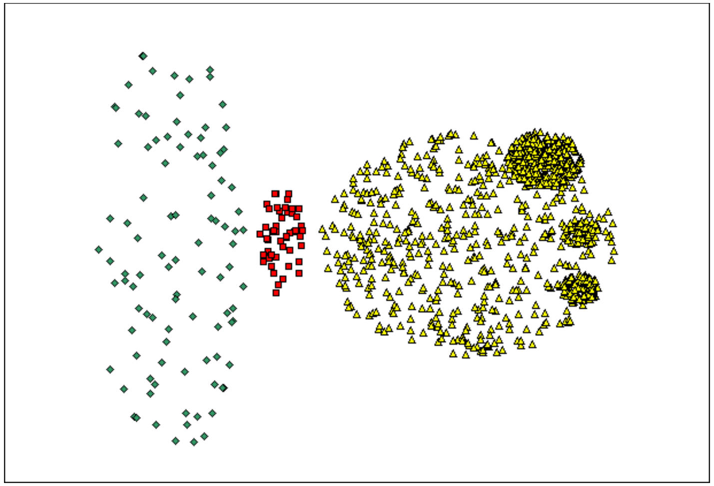
- **Resistant to Noise**
- **Can handle clusters of different shapes and sizes**

When DBSCAN Does NOT Work Well...

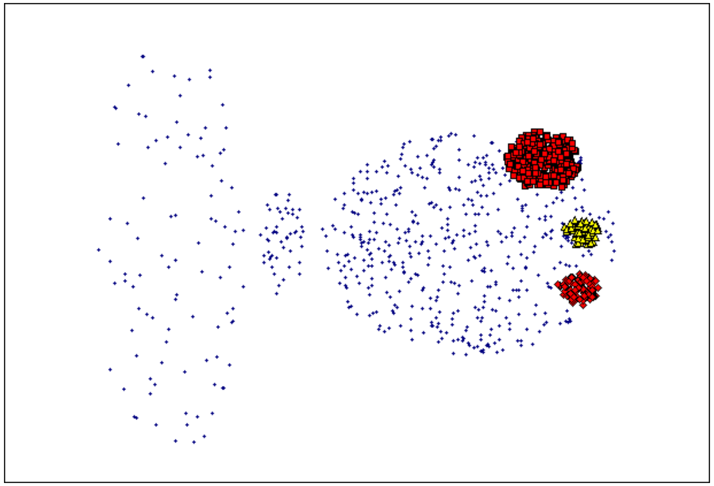


Original Points

- **Cannot handle varying densities!**
- **Sensitive to parameters!**



(MinPts=4, Eps=9.92).



(MinPts=4, Eps=9.75)

DBSCAN: Sensitive to Parameters

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

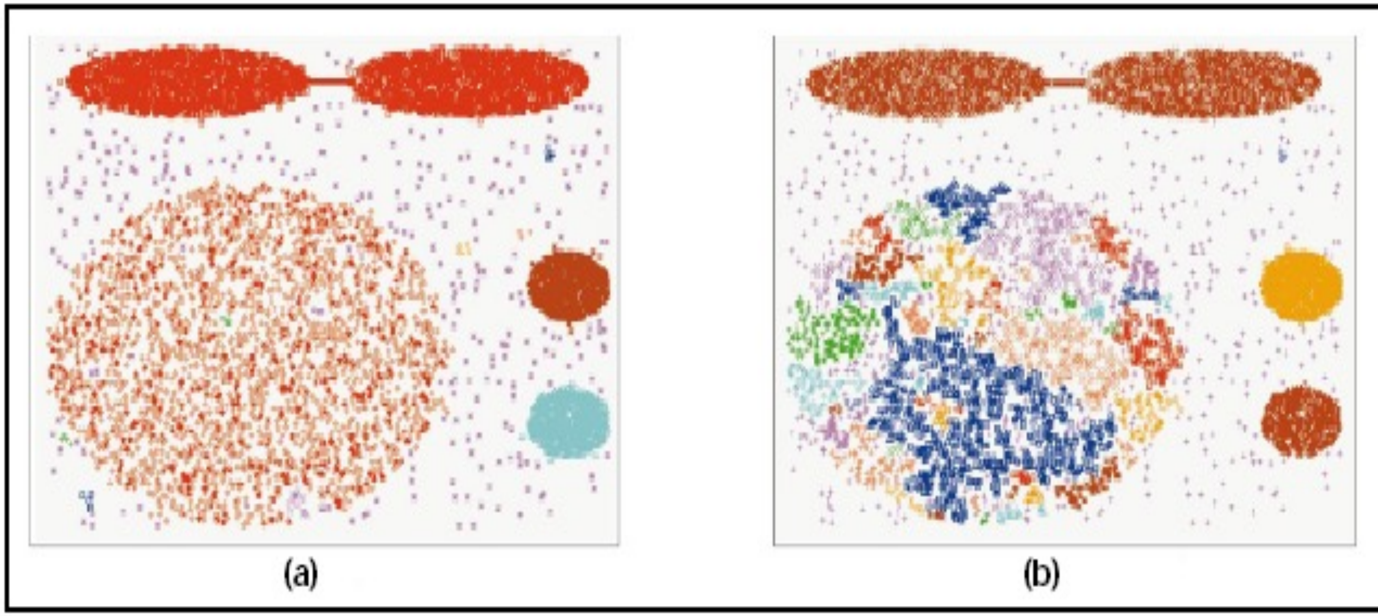
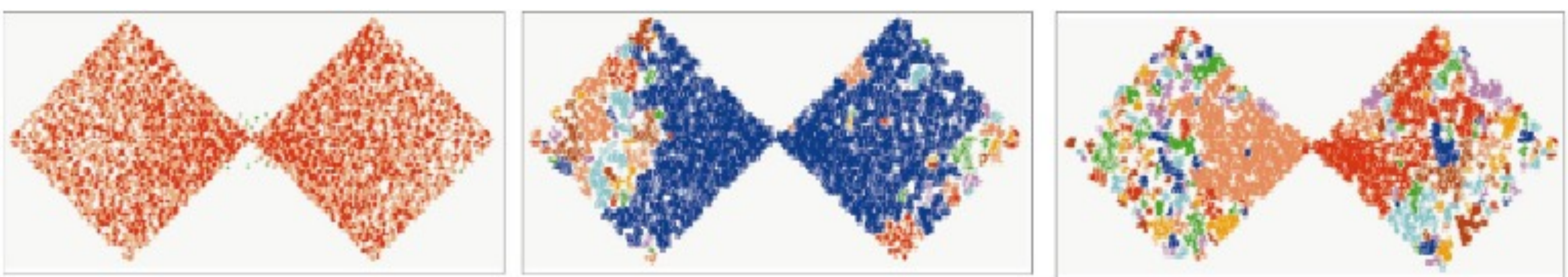


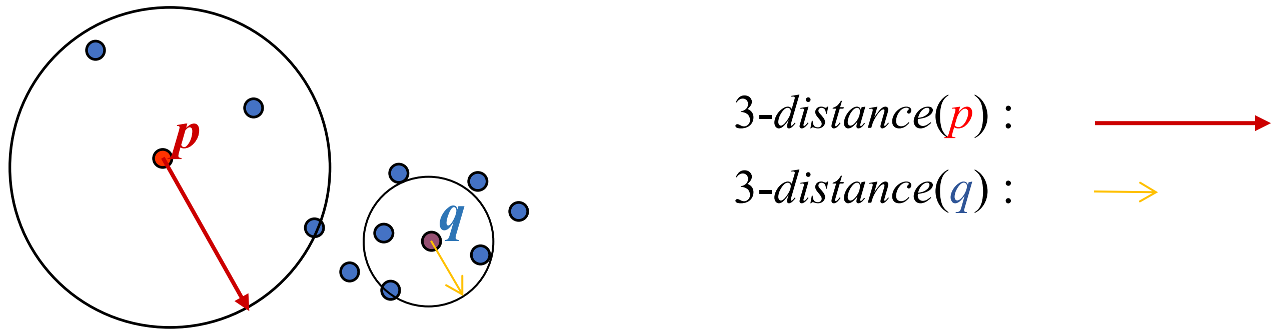
Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.



(a) (b) (c)

Determining the Parameters ε and *MinPts*

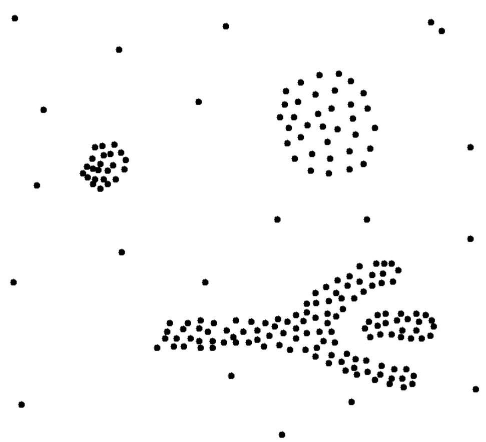
- Cluster: Point density higher than specified by ε and *MinPts*
- Idea: Use the point density of the least dense cluster in the data set as parameters – but how to determine this?
- Heuristic: look at the distances to the *k*-nearest neighbors

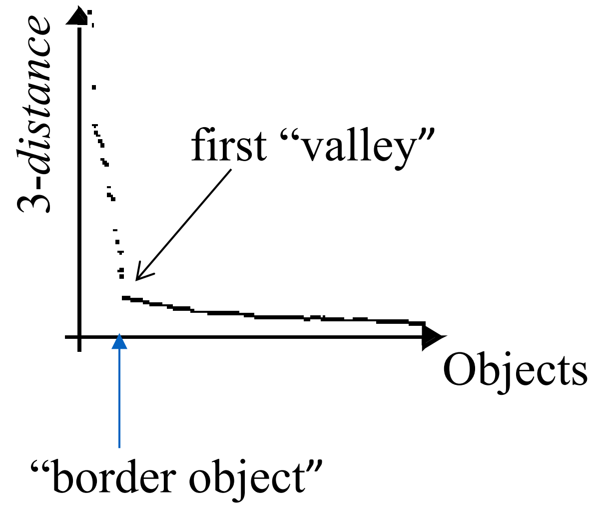


- Function *k-distance*(*p*): distance from *p* to the its *k*-nearest neighbor
- *k-distance plot*: *k*-distances of ALL objects, sorted in decreasing order

Determining the Parameters ε and *MinPts*

- Example *k*-distance plot





- Heuristic method:
  - Fix a value for *MinPts* (default: $2 \times d - 1$, where d denotes dimensionality of data space)
  - User selects “border object” *o* from the *MinPts-distance* plot; ε is set to *MinPts-distance*(*o*)

Density Based Clustering: Discussion

- Advantages
  - Clusters can have arbitrary shape and size
  - Number of clusters is determined automatically
  - Can separate clusters from surrounding noise
- Disadvantages
  - Input parameters may be difficult to determine
  - In some situations very sensitive to input parameter setting

Take-home Messages

- Cluster analysis groups objects based on their similarity and has wide applications
- Density-based clustering takes into considerations of other important concepts, i.e., (density and connectivity) vs (intra-cluster and inter-cluster similarity).
- There are still lots of research issues on cluster analysis, such as semi-supervised clustering, subspace clustering, etc.
- Yet it is always a topic of interest for emerging applications
  - Clustering in a social network graph
  - Spatial clustering of GPS data
  - Spatial clustering of farming data

Acknowledgement

- Slides/Materials of
  - https://www.cse.buffalo.edu/faculty/azhang/
- Photos from Internet