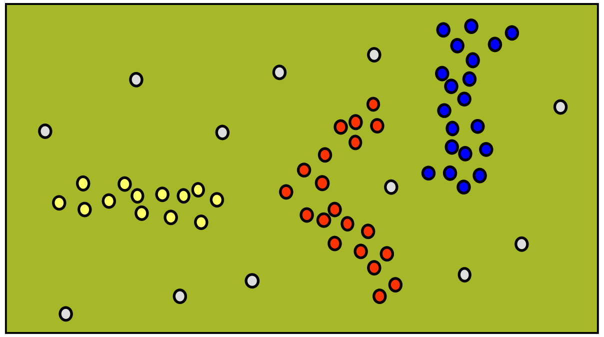
CLUSTERING

DBSCAN

DBSCAN

- DBSCAN ≡ Density Based Spatial Clustering of Applications with Noise
- Originally proposal to handle spatial data
- It uses the idea of density
- Density = number of points within a specified radius (Epsilon ε)
- Idea: a cluster has a much higher density of points than outside of the cluster



- outliers are points in low dense areas
- number of clusters is not a parameter

Definitions

epsilon

- The ε-neighborhood of a point p is the set of points whose distance from p is at most ε

  i.e., $N_\varepsilon(p) = \{q \in D \mid dist(p,q) <= \varepsilon\}$

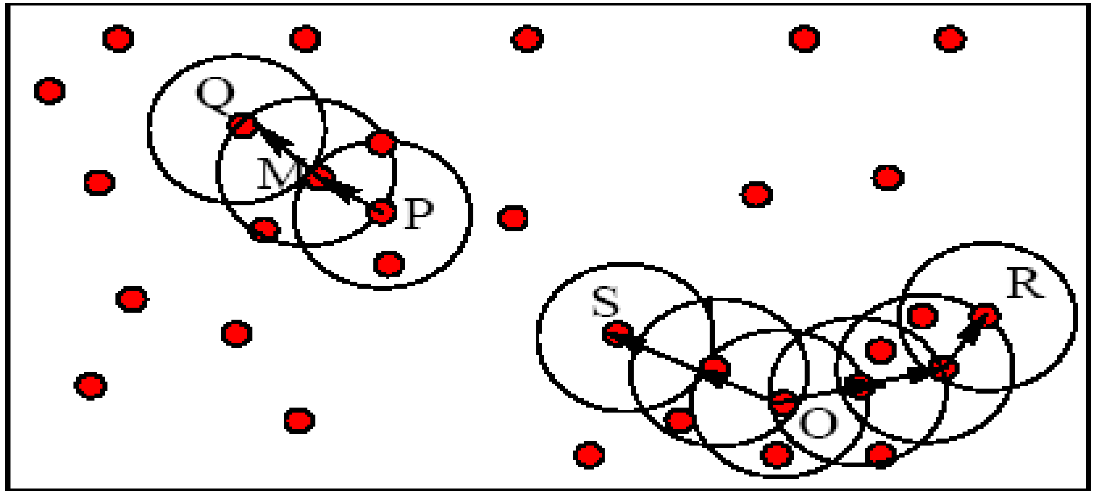
- Point p is a core point if the ε-neighborhood of p contains at least a minimum number, MinPts, of points

  i.e., $|N_\varepsilon(p)| \geq$ MinPts

- Two parameters***:***
  - $\boldsymbol{\varepsilon}$ (epsilon)
  - ***MinPts***

Example

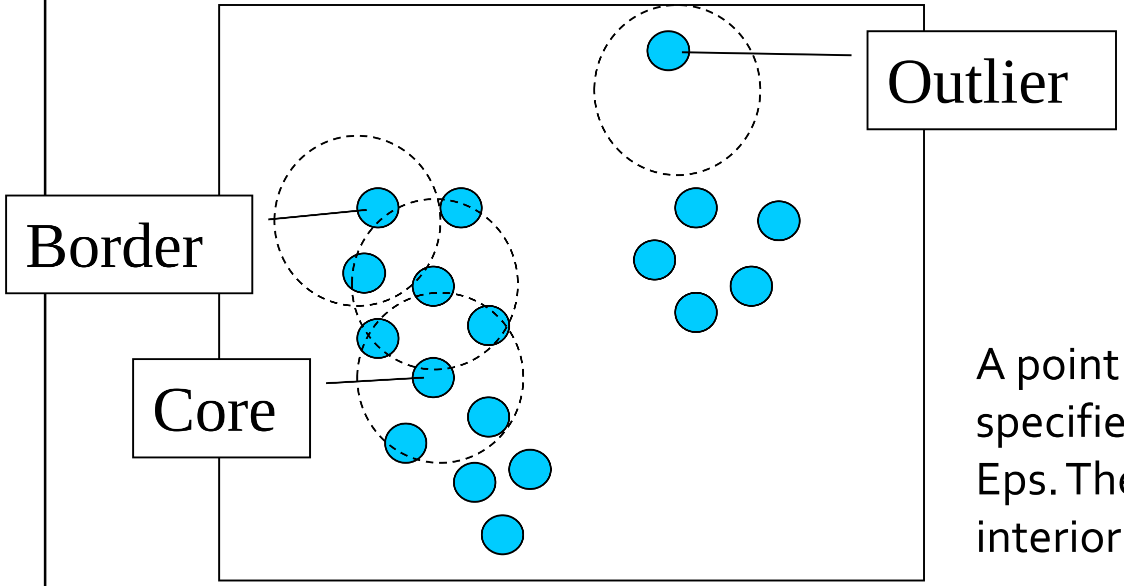
- M, P, O, and R are core point since each is in an Eps neighborhood containing at least 3 points



MinPts = 3 (self counts)

Eps=radius of the circles

Types of Points: Core, Border & Outlier



ε = 1 unit, MinPts = 5

Given ε and *MinPts*, 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.

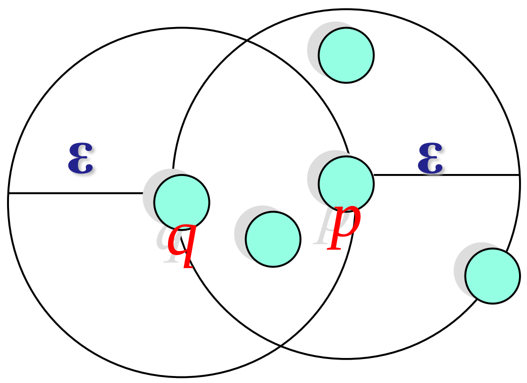
A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point.

Points inside a cluster

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

Directly Density-Reachability

- A point q is **directly density-reachable** from point p if
  1) p is a core point, and
  2) q is in the ε-neighborhood of p.



MinPts = 4

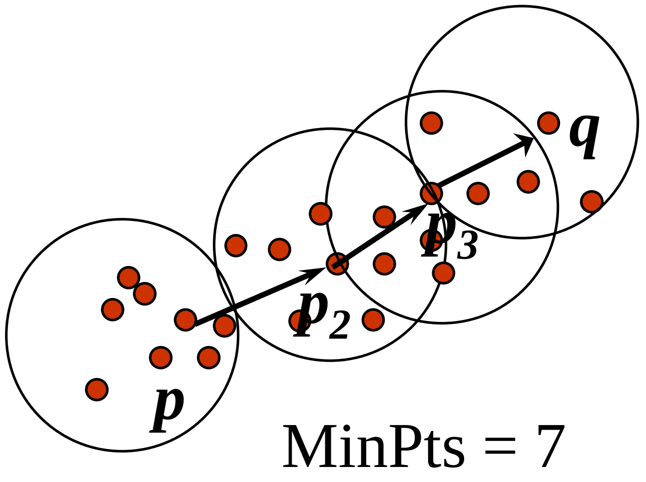
- q is directly density-reachable from p
- p is not directly density- reachable from q?

Density-Reachability

- Density-reachable (directly and indirectly):
- Def: A point q is **density-reachable** from point p with respect to ε and MinPts if there is a chain of points $p_1, p_2, \ldots, p_n$ such that

  $p_1 = p$, $p_n = q$, and

  p_{i+1} is directly density-reachable from p_i wrt ε and MinPts , for $1 \leq i \leq n - 1$

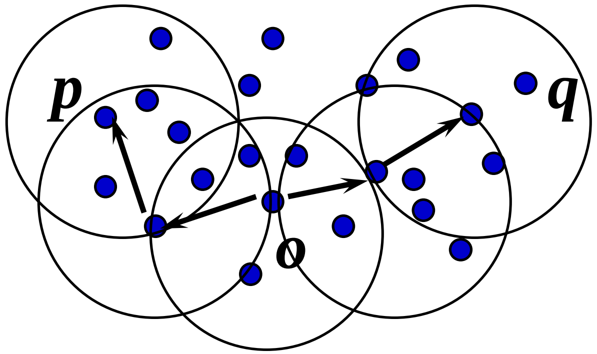


- Is q density-reachable from p?
- Is p density- reachable from q?

Density Connectivity

- Not all points in a cluster are density-reachable from each other
- So, density-reachable is not good enough to describe clusters
- Def: A point p is **density-connected** to point q (wrt ε and MinPts) if there is a point o such that both p and q are density-reachable from o (wrt ε and MinPts)

i.e., two points p and q are density-connected if they are both density-reachable from a given point o.

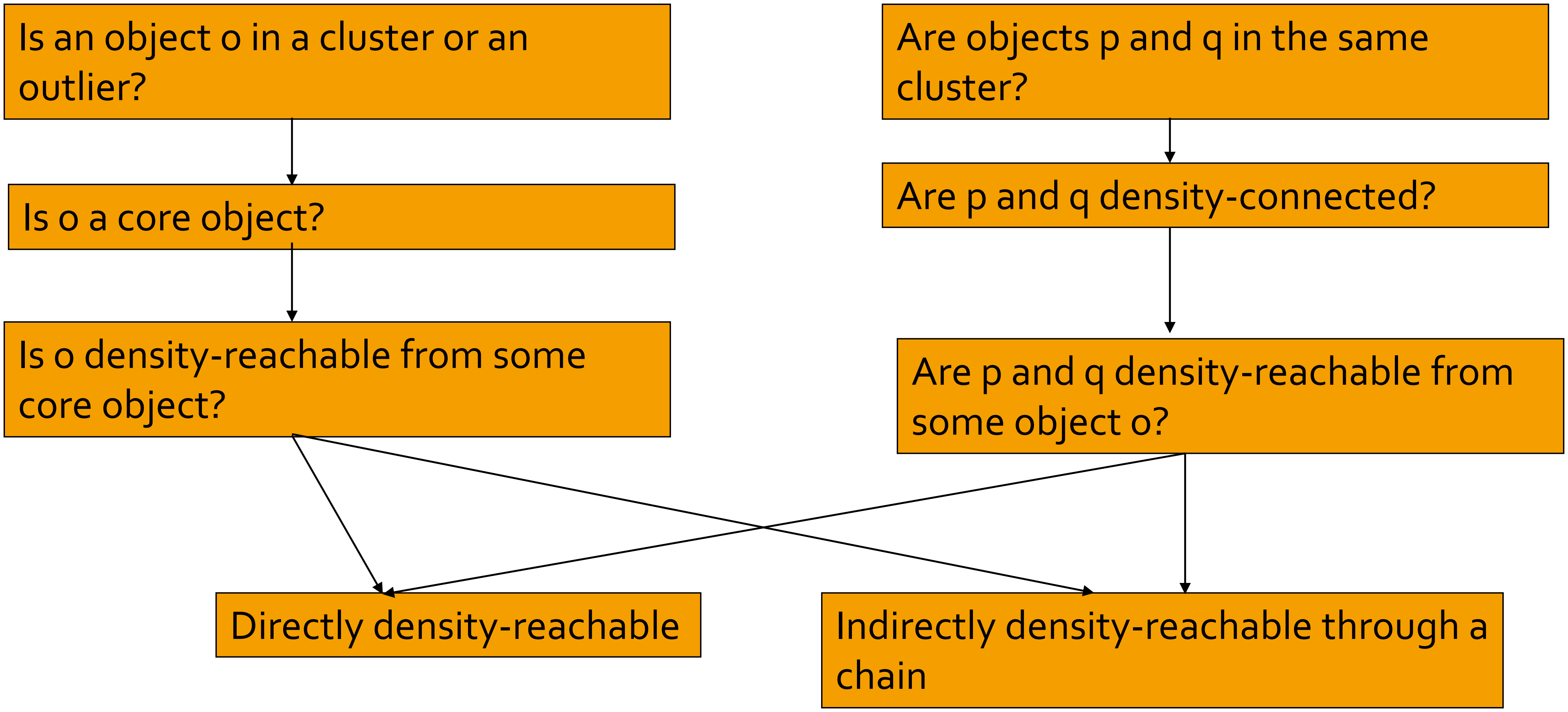


Formal Description of Cluster

- DBSCAN defines a cluster as a set of density-connected points which is maximal wrt density-reachability
- Noise is any point in the dataset which does not belong to any of the clusters
- Def. (cluster): Given a data set D, parameter ε and threshold MinPts. A **cluster** C is a subset of D satisfying the two conditions:
  1. $\forall$ p, q $\in$ C, p and q are density-connected. (**connectivity**)
  2. $\forall$ p, q $\in$ D, if p $\in$ C and q is density-reachable from p, then q $\in$ C. (**maximal-ity**)

p is a core point.

Review of Concepts



Outline of the DBSCAN Algorithm

Input: The data set D

Parameter: ε, MinPts

For each object p in D
    if p is a core object and not processed then
        C = retrieve all objects density-reachable from p
        mark all objects in C as processed
        report C as a cluster
    end if

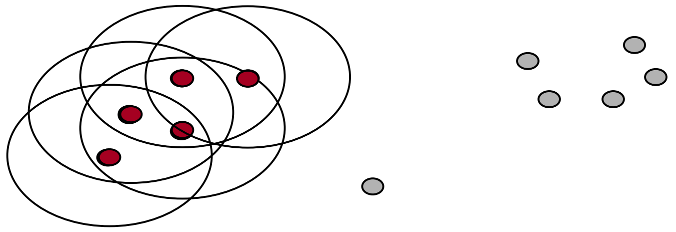
End For

DBSCAN: The Algorithm

- Arbitrarily 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 dataset
- Continue the process until all of the points have been processed.

DBSCAN Algorithm – Example

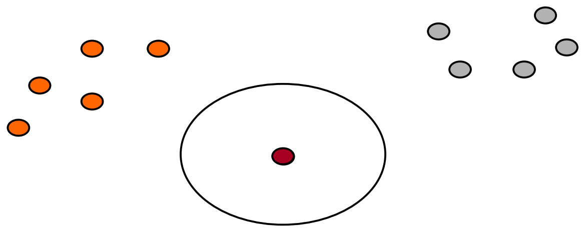
- Parameter
  - ε = 2 cm
  - *MinPts* = 3



- Arbitrarily select a point p
- Retrieve all points density-reachable from p
- 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 dataset
- Continue the process until all of the points have been processed.

DBSCAN Algorithm – Example

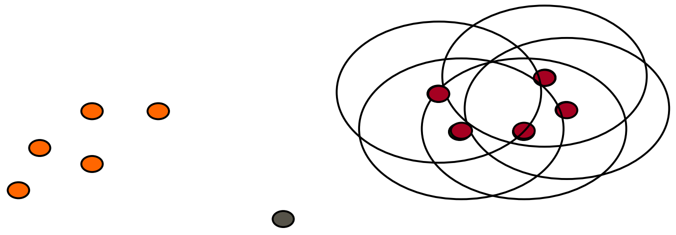
- Parameter
  - ε = 2 cm
  - *MinPts* = 3



- Arbitrarily select a point p
- Retrieve all points density-reachable from p
- 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 dataset
- Continue the process until all of the points have been processed.

DBSCAN Algorithm – Example

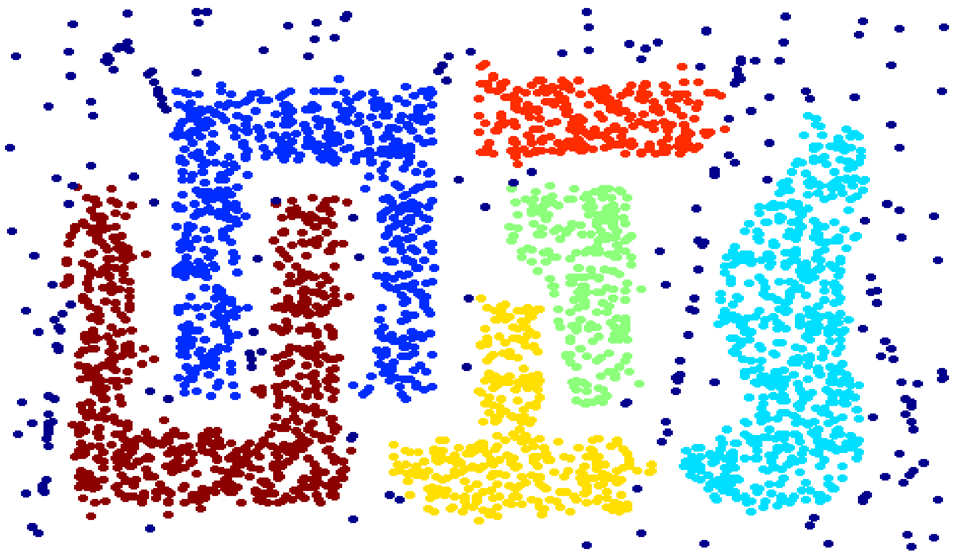
- Parameter
  - ε = 2 cm
  - *MinPts* = 3



- Arbitrarily select a point p
- Retrieve all points density-reachable from p
- 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 dataset
- Continue the process until all of the points have been processed.

When DBSCAN Works Well

Original Points



Clusters

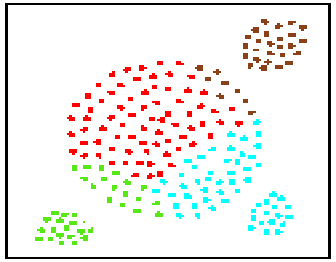
- DBSCAN works well when cluster **densities** do not vary a lot.
- Can handle clusters of different shapes and sizes
- Resistant to Noise

Performance Evaluation compared with CLARANS

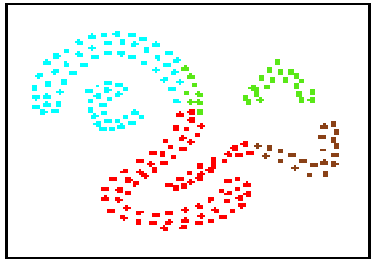
- DBSCAN outperformed CLARANS by a factor of more than 100

- Accuracy

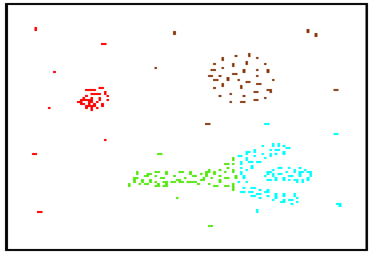
CLARANS:



database 1

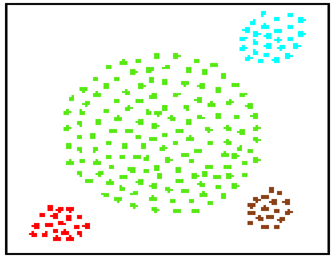


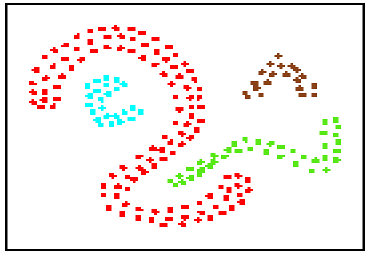
database 2

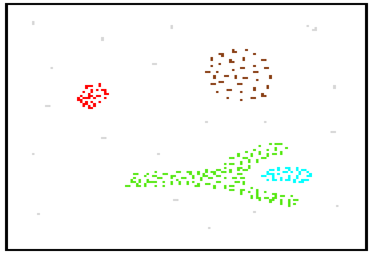


database 3

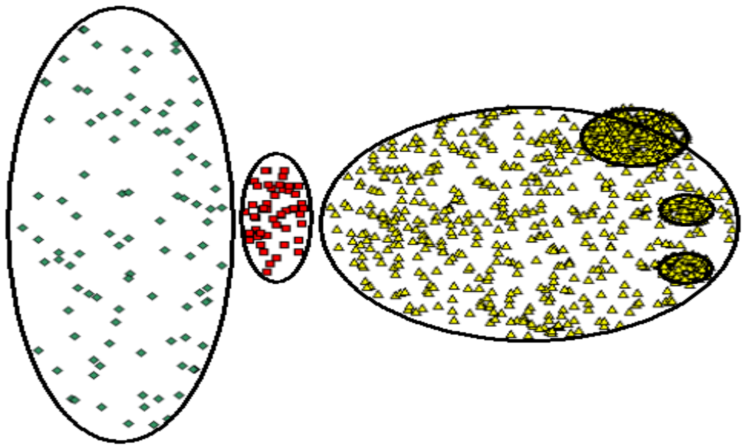
DBSCAN:





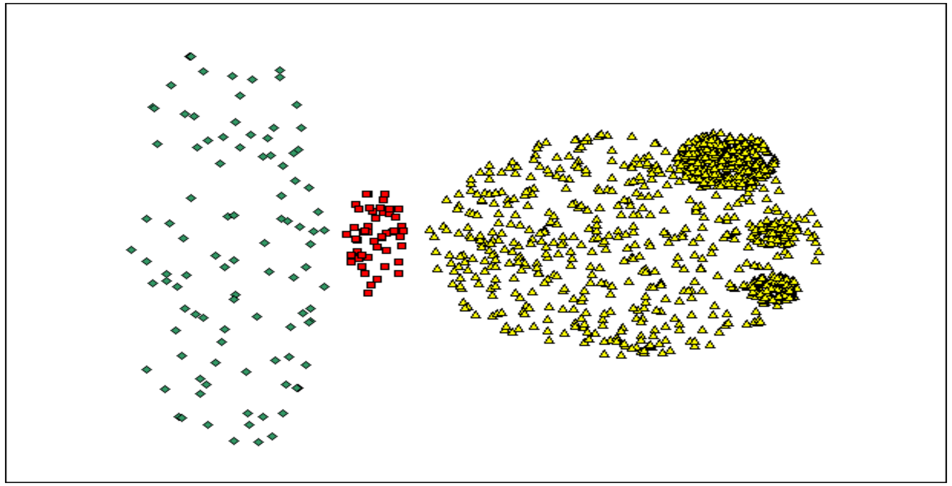


When DBSCAN Does NOT Work Well

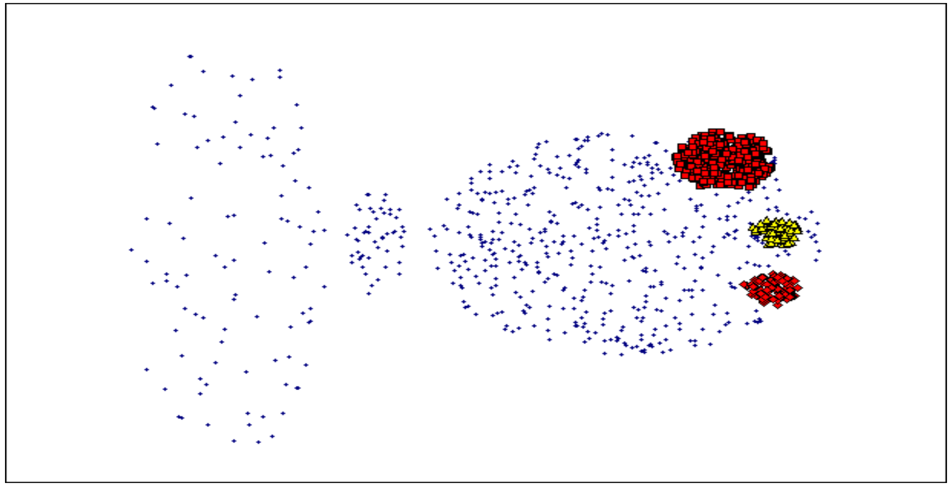


Original Points

- Varying densities
- High-dimensional data



(MinPts=4 Eps=large value).



(MinPts=4, Eps=small value; min density increases)

DBSCAN: Sensitive to Parameters

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

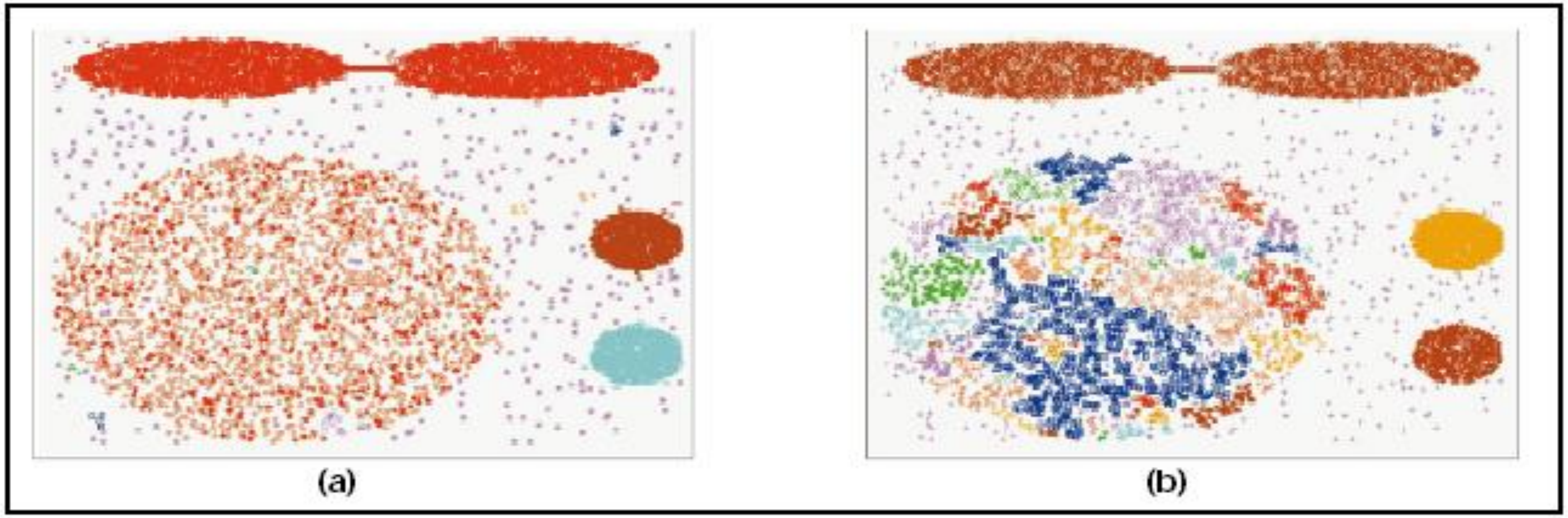
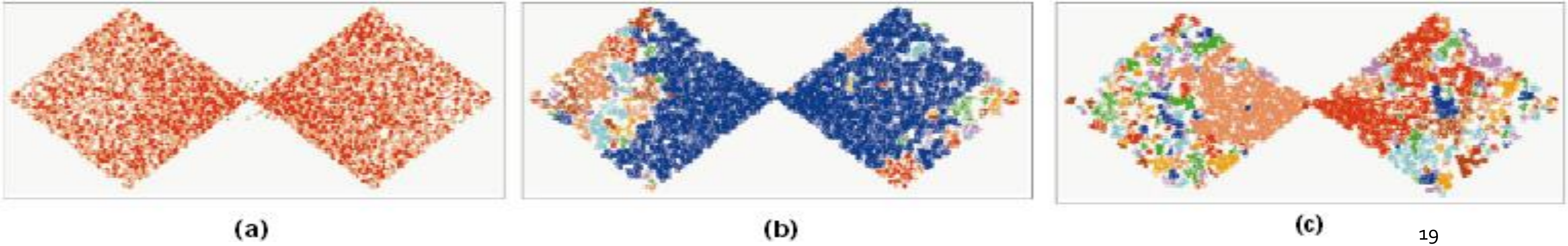
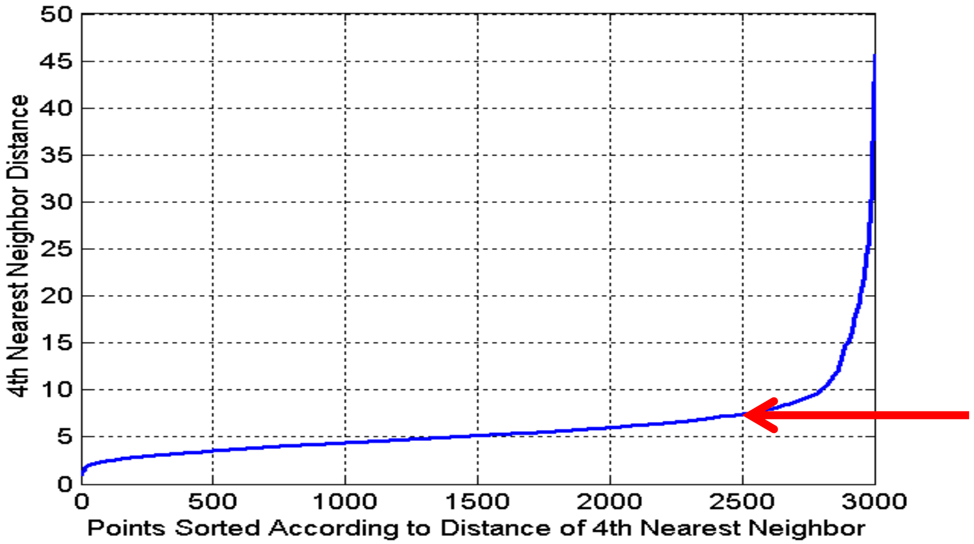


Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.



DBSCAN: Heuristics for determining EPS and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor (e.g., k=4)
- Find the distance d where there is a "knee" in the curve
  - ➢ Eps = d, MinPts = k



Eps ~ 7-10
MinPts = 4

Summary

- Advantages
  - clusters can have arbitrary shape and size
  - number of clusters is determined automatically
  - not very sensitive to noise
  - supports outlier detection
  - the second most used clustering algorithm after K-means
- Disadvantages
  - parameters selection can be tricky
  - can be sensitive to input parameter setting
  - has problems of identifying clusters of varying densities
  - does not work well in high-dimensional datasets