PAPER REVIEW

A DENSITY-BASED ALGORITHM FOR DISCOVERING CLUSTERS IN LARGE SPATIAL DATABASES WITH NOISE

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu

KDD-96 (https://www.dbs.ifi.lmu.de/Publikationen/Papers/KDD-96.final.frame.pdf)

Oktsa Dwika Rahmashari
665020043-6

OVERVIEW

1. Introduction
2. Clustering Algorithm
3. A Density Based Notion of Clusters
4. DBSCAN
5. Performance Evaluation
6. Conclusion

Introduction

01

Clustering algorithms need to meet certain requirements when applied to large spatial databases

02

Existing clustering algorithms do not meet all these requirements.

03

New clustering algorithm called DBSCAN (Density Based Spatial Clustering of Applications with Noise), which is designed to discover clusters of arbitrary shape.

Clustering Algorithm

Partitioning

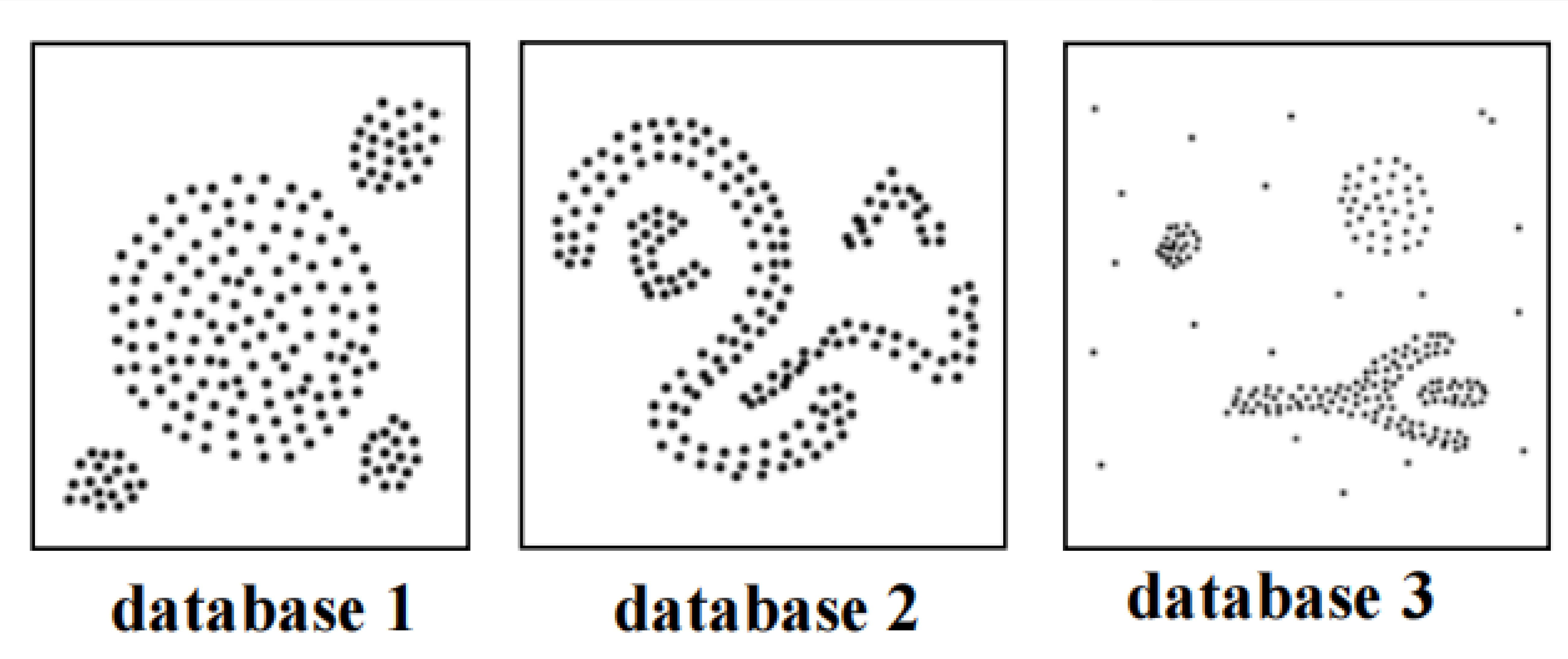
- Construct a partition of a database of **n** objects into a set of **k** clusters
- Each cluster is represented by the gravity center (k-means) or by one of the objects located near its center (k-medoid).

Hierarchical

- Used to create a hierarchical decomposition of a dataset **D**.
- represented by a dendrogram, a tree that iteratively splits **D** into smaller subsets until each subset consists of only one object.

A density based approach has the capability of identifying clusters of any shape

A Density Based Notion of Clusters



Density within cluster is high, but density between cluster is low

A Density Based Notion of Clusters

Parameter

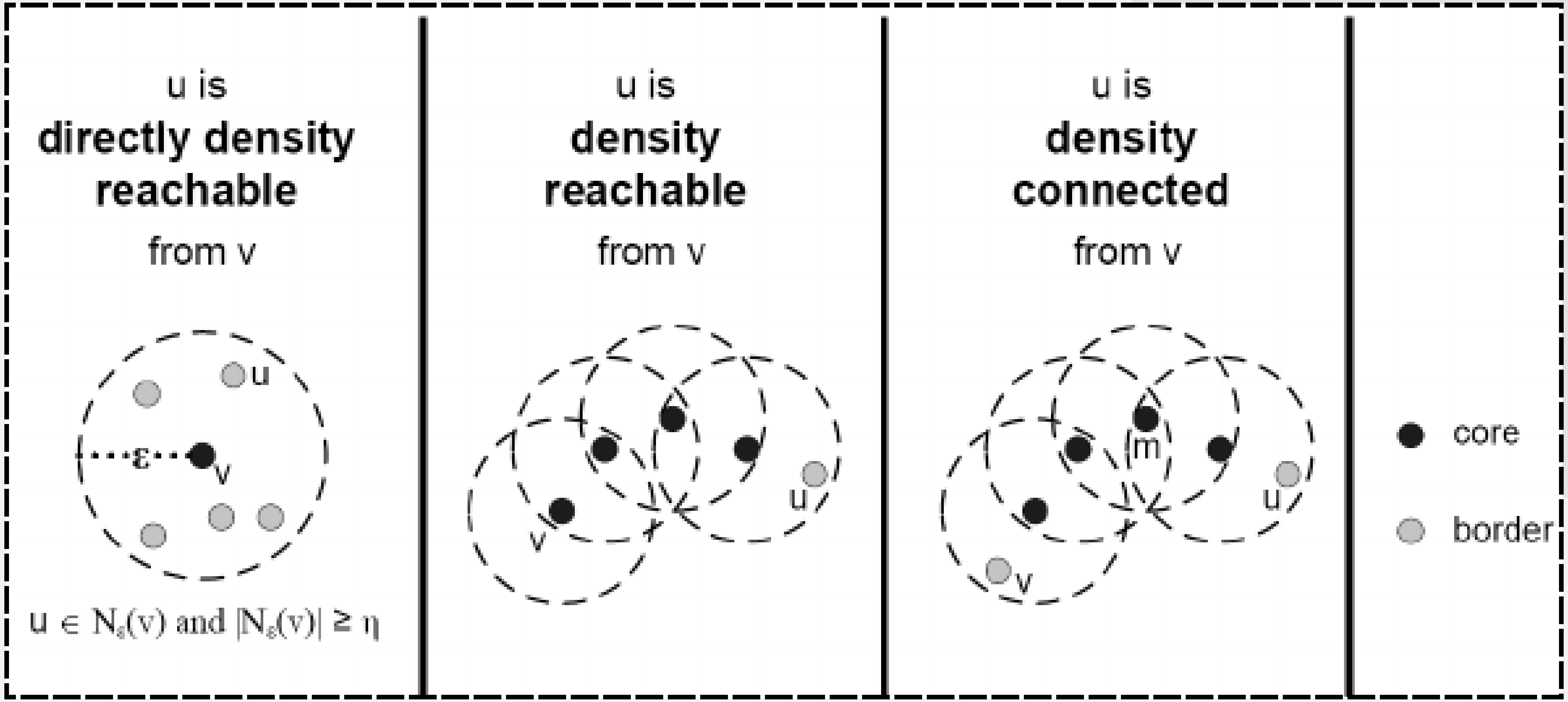
Eps

MinPts

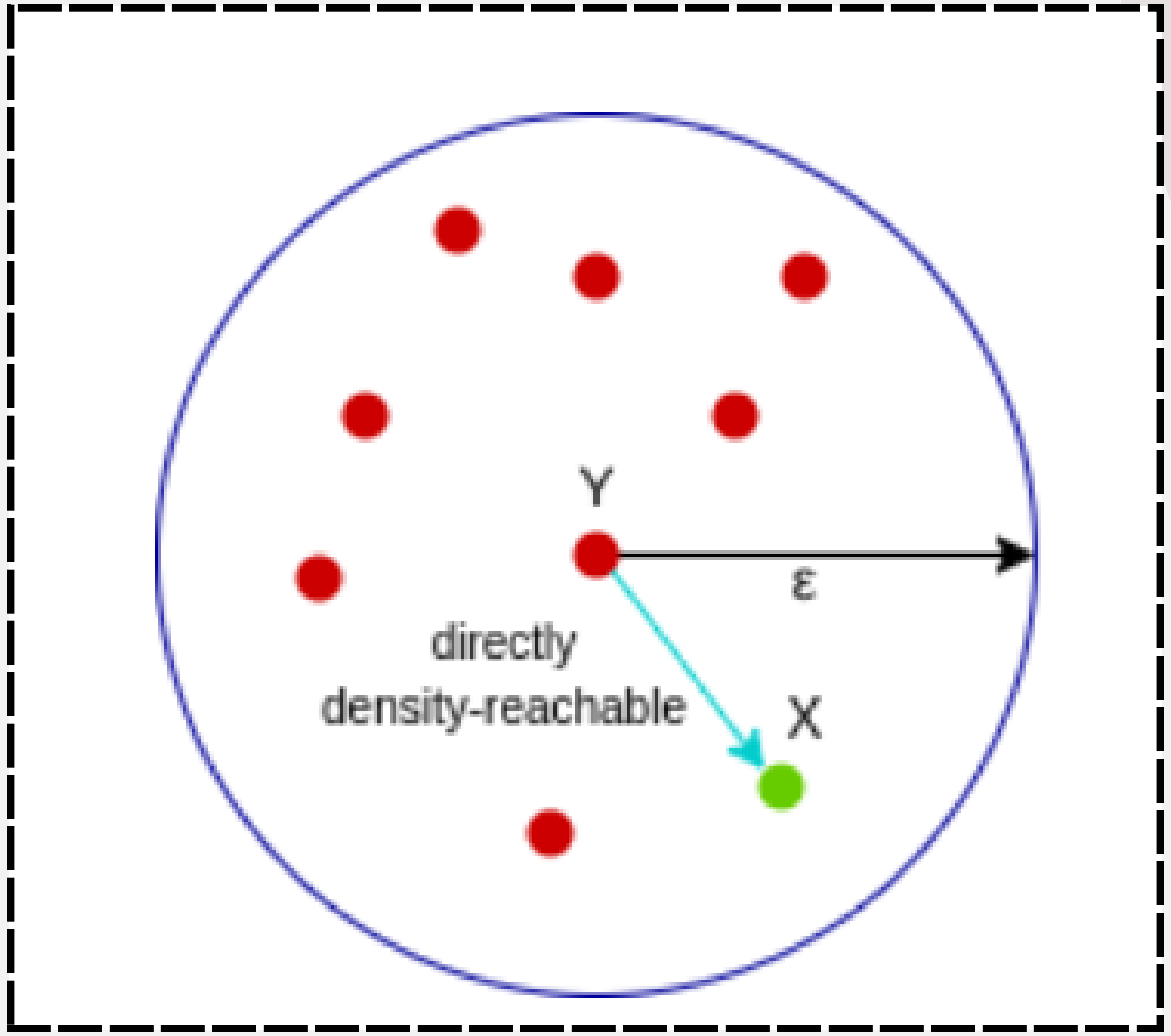
- **Eps or Epsilon** is the radius of the circle to be created around each data point to check the density
- **MinPts or minPoints** is the minimum number of data points required inside that circle

Core Point	Border Point	Noise
$\lvert N_{Eps} \rvert \geq MinPts$	$\lvert N_{Eps} \rvert < MinPts$	$\lvert N_{Eps} \rvert = \{\}$

A Density Based Notion of Clusters



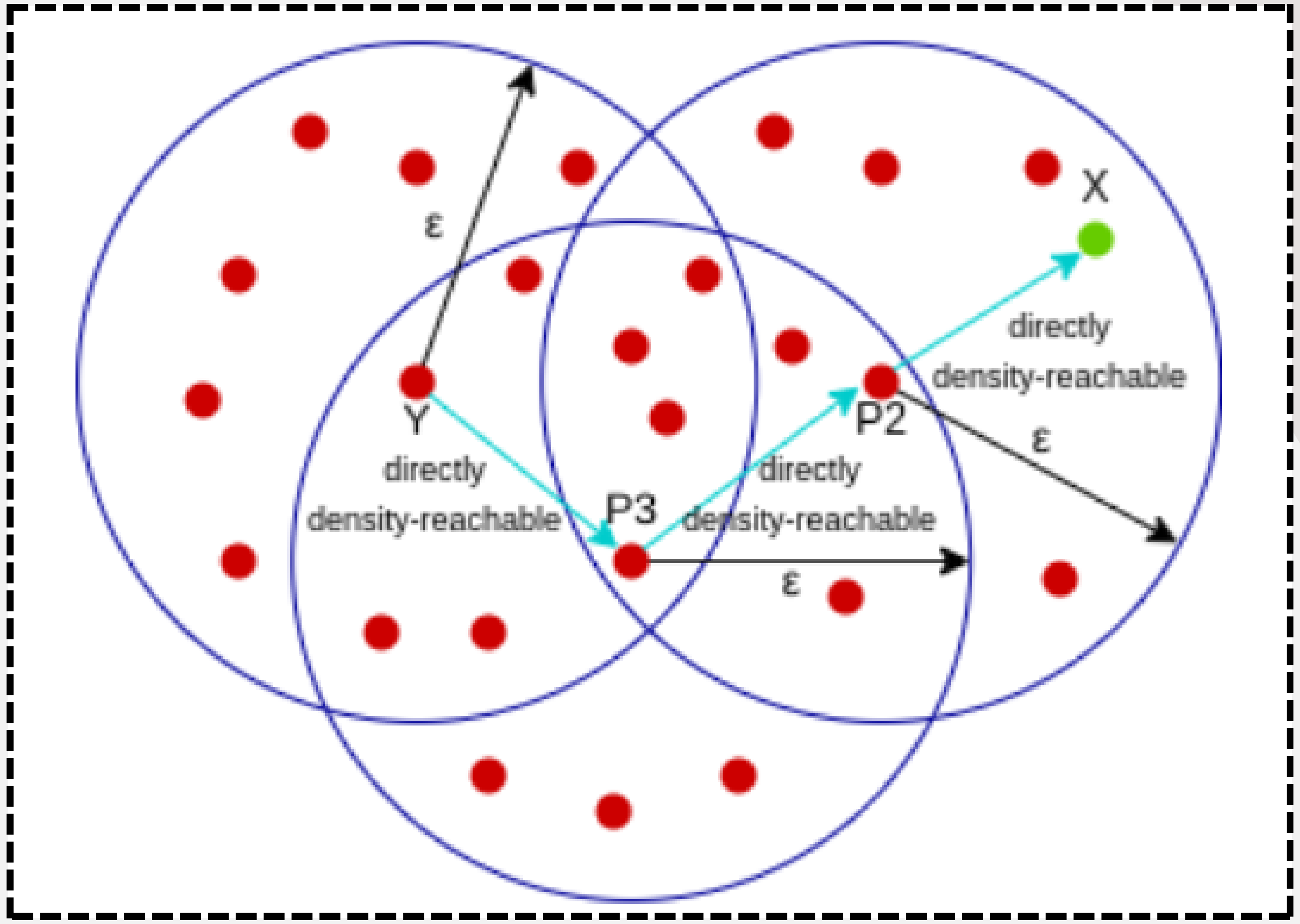
A Density Based Notion of Clusters



Directly Density-Reachable: **X** is directly density-reachable from point **Y** w.r.t epsilon, minPoints if;

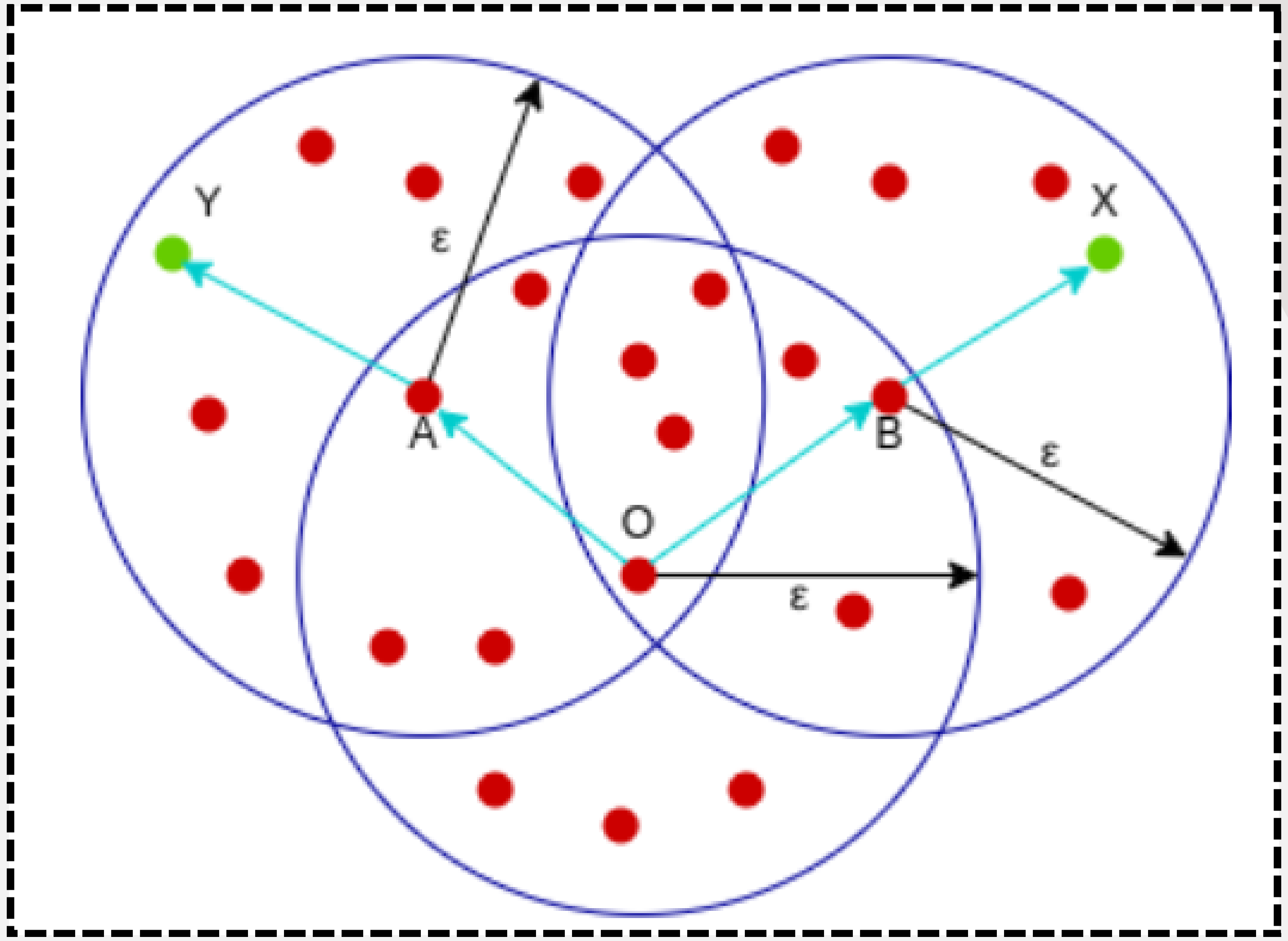
1. **X** belongs to the neighbourhood of **Y**
2. **Y** is a core point.

A Density Based Notion of Clusters



Density reachable: **X** is density-reachable from point **Y** w.r.t epsilon, minPoints if there is a chain of points p1, p2, p3, ..., pn and p1=X and pn=Y such that pi+1 is directly density-reachable from pi.

A Density Based Notion of Clusters



Density connected: A point **X** is density-connected from point **Y** w.r.t epsilon and minPoints if there exists a point **O** such that both **X** and **Y** are density-reachable from O w.r.t to epsilon and minPoints.

Cluster

A cluster is a non-empty subset of the database that satisfies two conditions: **maximality** and **connectivity**.

1) $\forall$ p, q: if p $\in$ C and q is density-reachable from p wrt. Eps and MinPts, then q $\in$ C. (Maximality)

2) $\forall$ p, q $\in$ C: p is density-connected to q wrt. EPS and MinPts. (Connectivity)

Noise

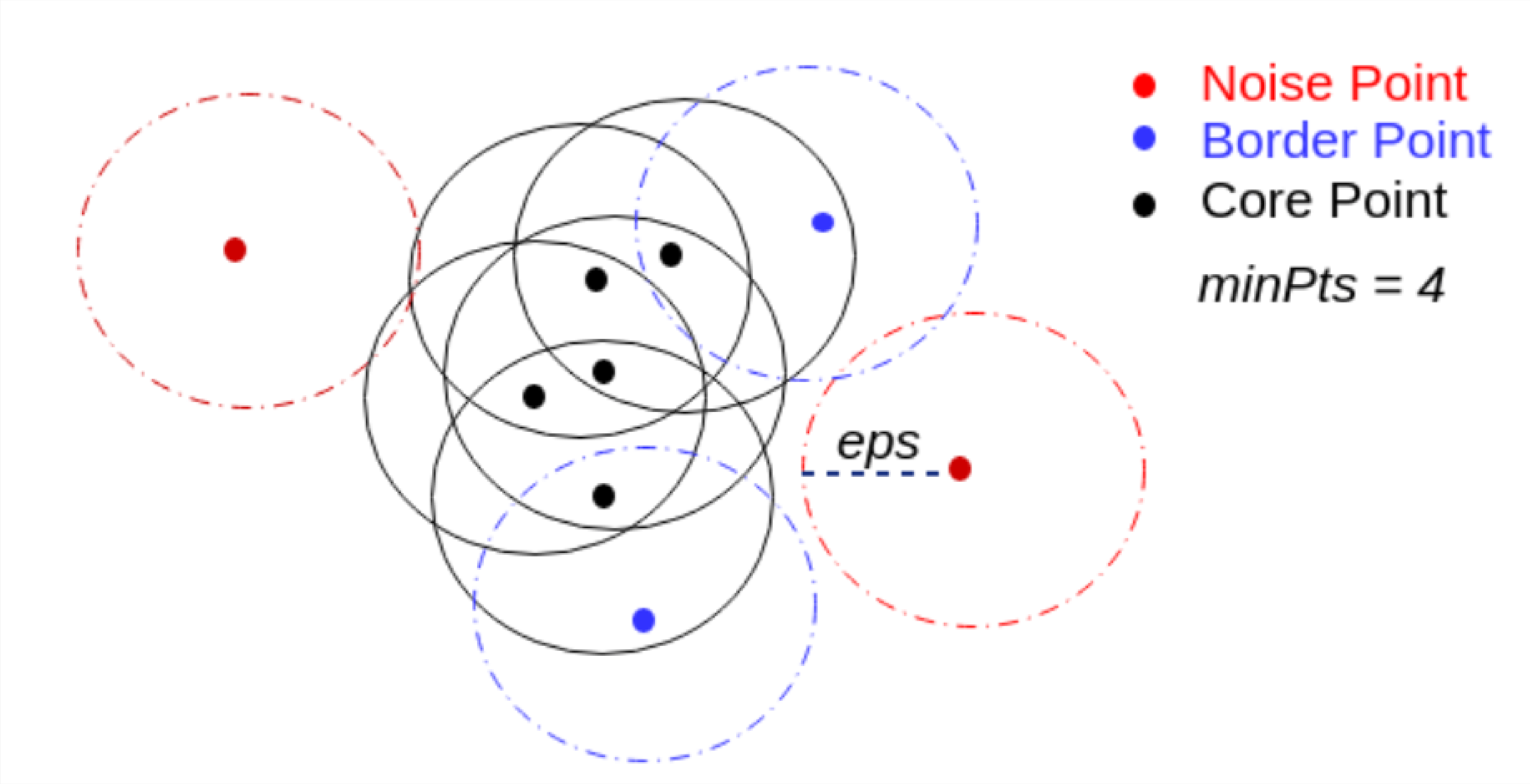
Noise is defined as the set of points in the database that do not belong to any of the clusters

DBSCAN

```
DBSCAN(Dataset, Eps, MinPts)
    Initialize an empty list of clusters
    For each unvisited point P in the Dataset
        Mark P as visited
        Find all points within distance Eps of P and store them in a new cluster
        If the cluster has at least MinPts points
            Expand the cluster by adding more points from the neighborhood

    Return the list of clusters
```

DBSCAN



Performance Evaluation

SEQUOIA 2000 benchmark.

figure 5: Clusterings discovered by CLARANS

Performance Evaluation

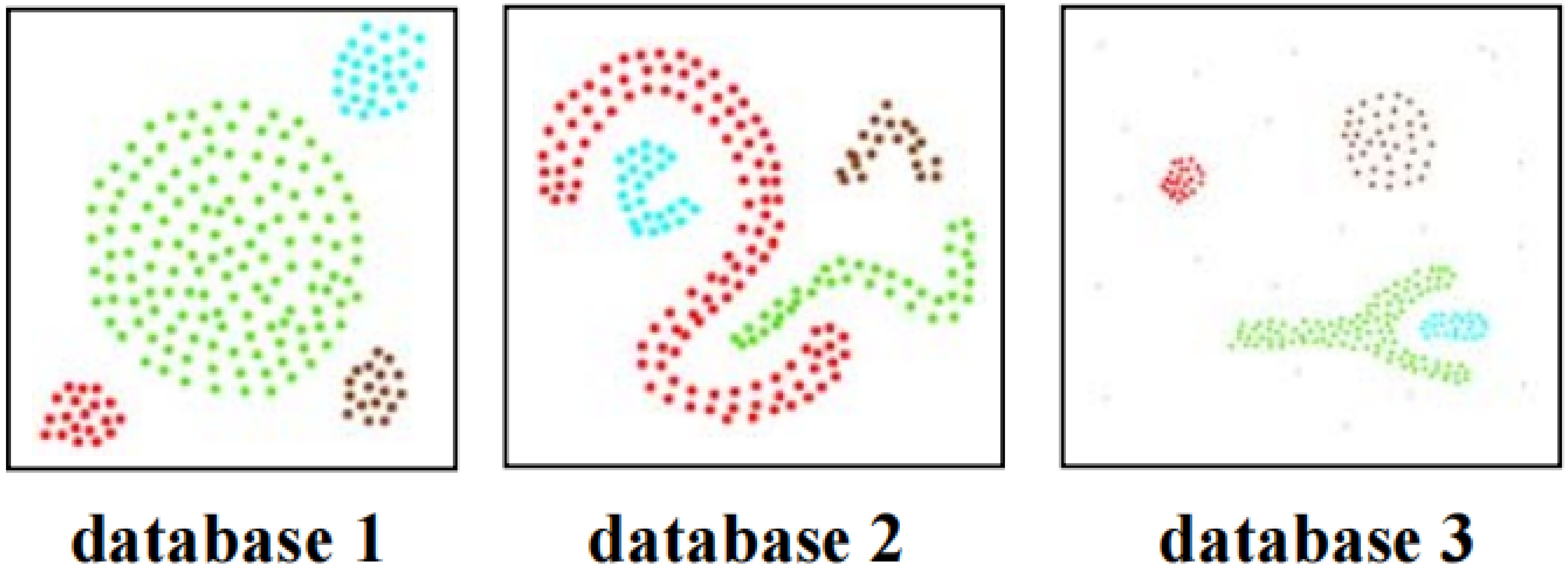


figure 6: Clusterings discovered by DBSCAN

Performance Evaluation

Table 1: run time in seconds

number of points	1252	2503	3910	5213	6256
DBSCAN	3.1	6.7	11.3	16.0	17.8
CLAR-ANS	758	3026	6845	11745	18029
number of points	7820	8937	10426	12512	
DBSCAN	24.5	28.2	32.7	41.7	
CLAR-ANS	29826	39265	60540	80638	

- DBSCAN is significantly faster than CLARANS for all numbers of points.
- The run time for DBSCAN increases as the number of points increases, but the increase is not consistent.
- The run time for CLARANS increases significantly as the number of points increases.

Conclusion

- **DBSCAN is significantly more effective in discovering clusters of arbitrary shape than the well-known algorithm CLARANS**
- **Future research should consider extending DBSCAN to handle extended objects such as polygons in spatial databases**

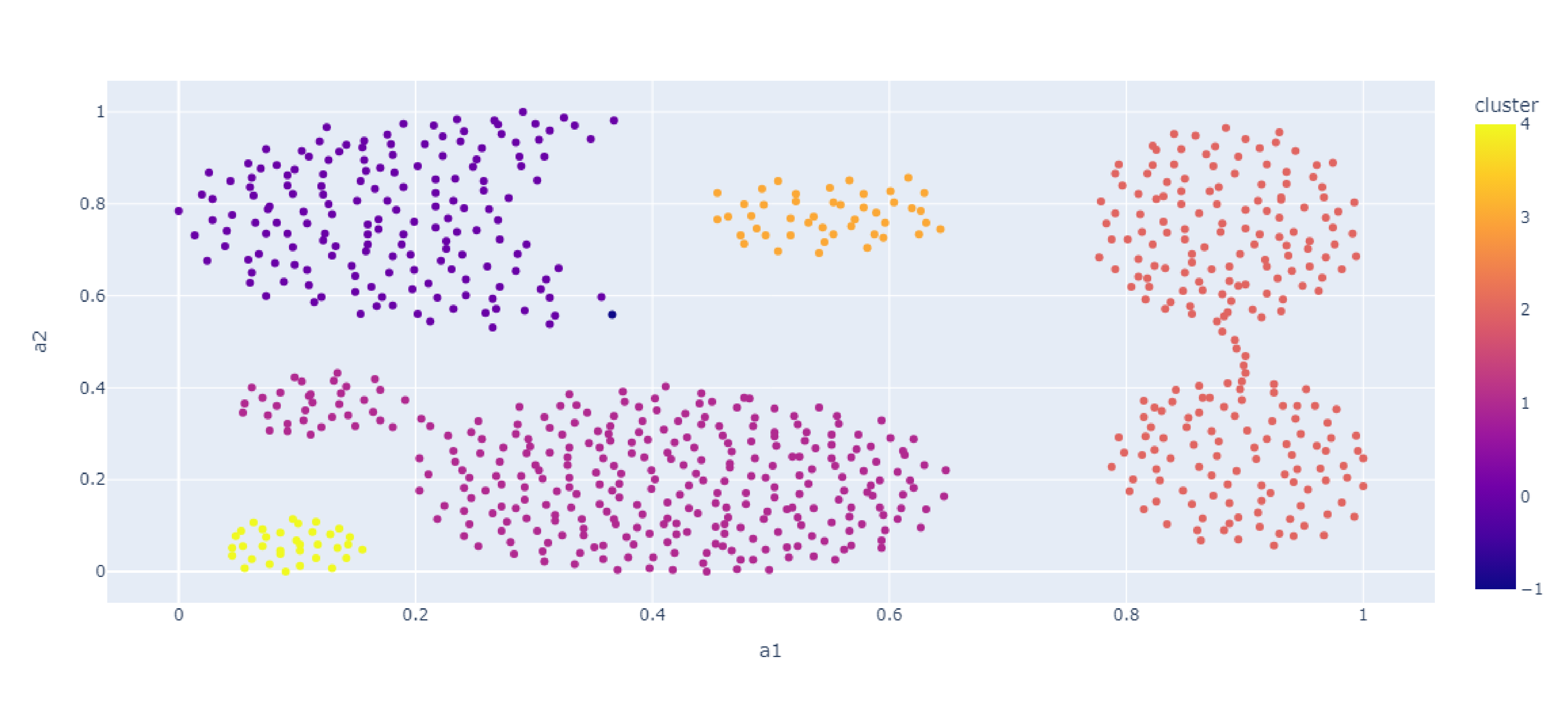
APPLICATION OF DBSCAN

Dataset : Aggregation benchmark

Sources :

- Title: dbscan: Fast Density-Based Clustering with R
- Authors: Michael Hahsler, Matthew Piekenbrock, Derek Doran
- https://doi.org/10.18637/jss.v091.i01

APPLICATION OF DBSCAN



code

THANK YOU