

# **Data Mining Cluster Analysis: Basic Concepts and Algorithms**

---

Lecture Notes for Chapter 7

Introduction to Data Mining, 2<sup>nd</sup> Edition

by

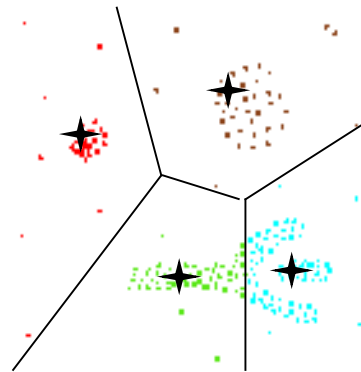
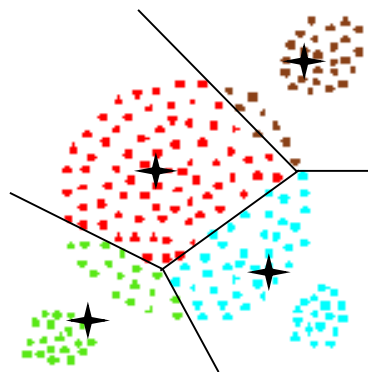
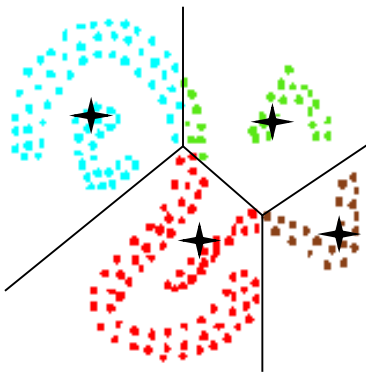
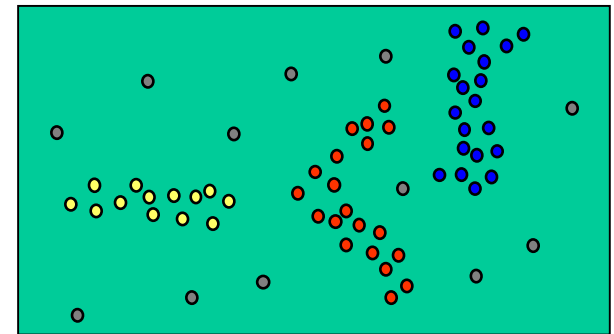
Tan, Steinbach, Karpatne, Kumar

With additions from Michael Houle (NII) and Aidong Zhang (UBuffalo)

# Density Based Clustering

## ★ *Basic Idea:*

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



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

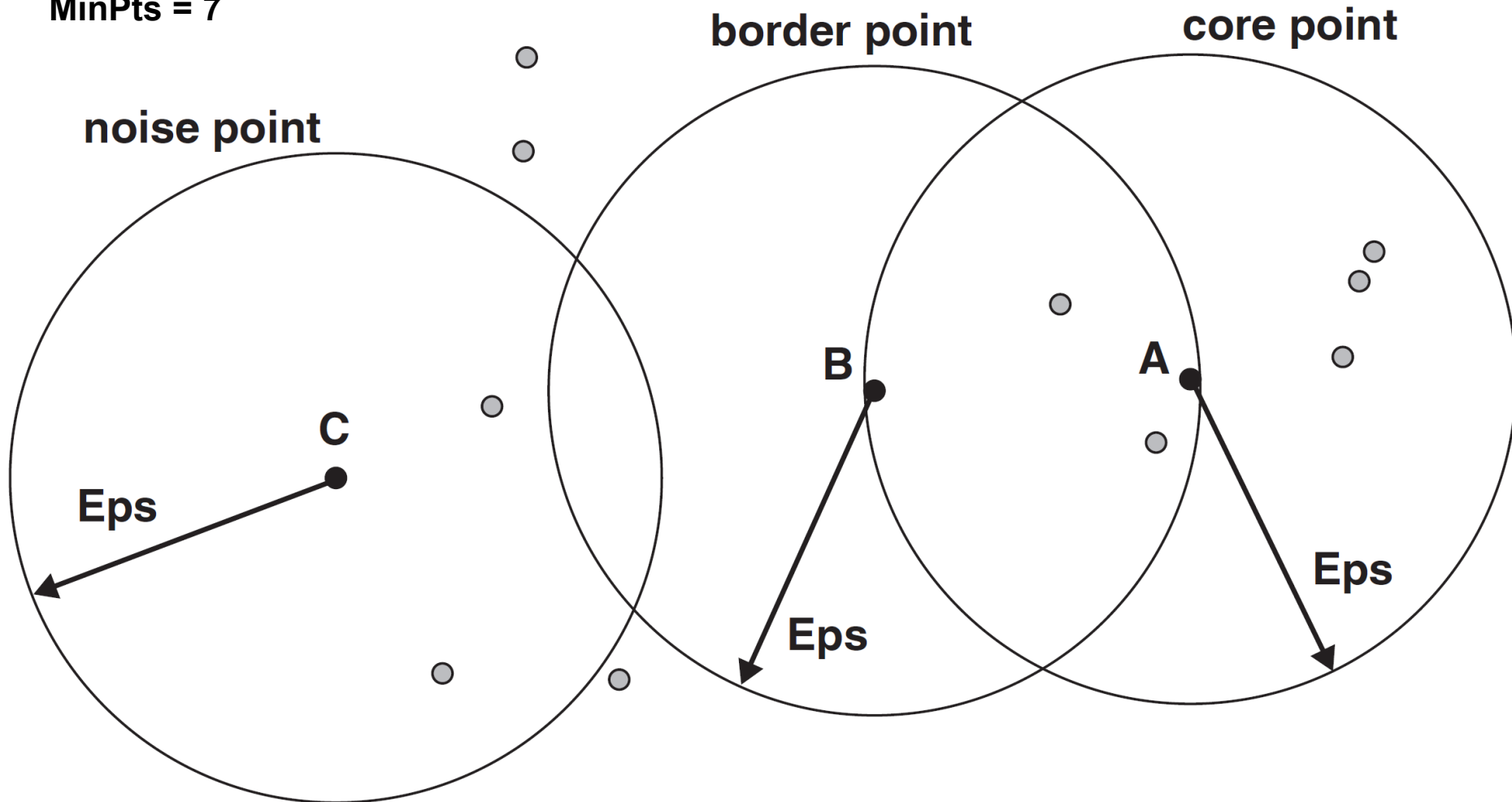
# DBSCAN

---

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a **core point** if it has at least a specified number of points (MinPts) within Eps
    - ◆ These are points that are at the interior of a cluster
    - ◆ Counts the point itself
  - A **border point** is not a core point, but is in the neighborhood of a core point
  - A **noise point** is any point that is not a core point or a border point

# DBSCAN: Core, Border, and Noise Points

MinPts = 7

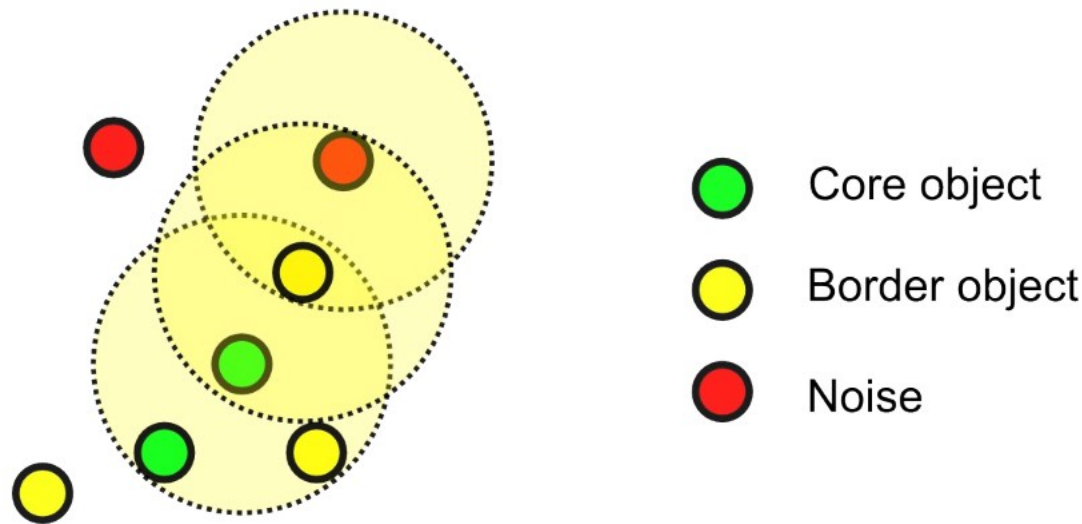


---

# Density Based Clustering

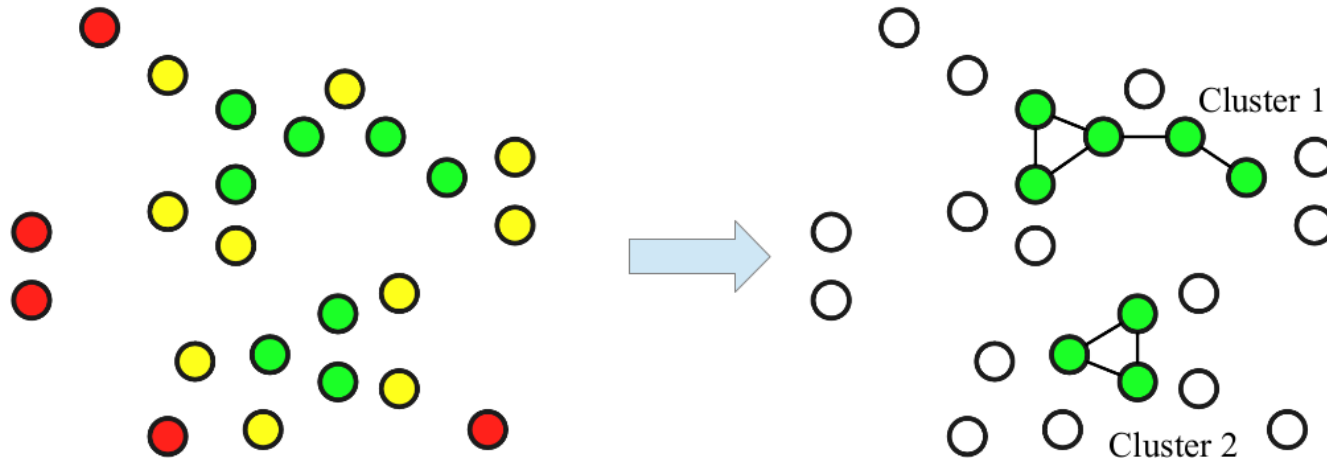
Step 1: label points as core (dense), border and noise

- Based on thresholds  $R$  (radius of neighborhood) and  $\text{min\_pts}$  (min number of neighbors)



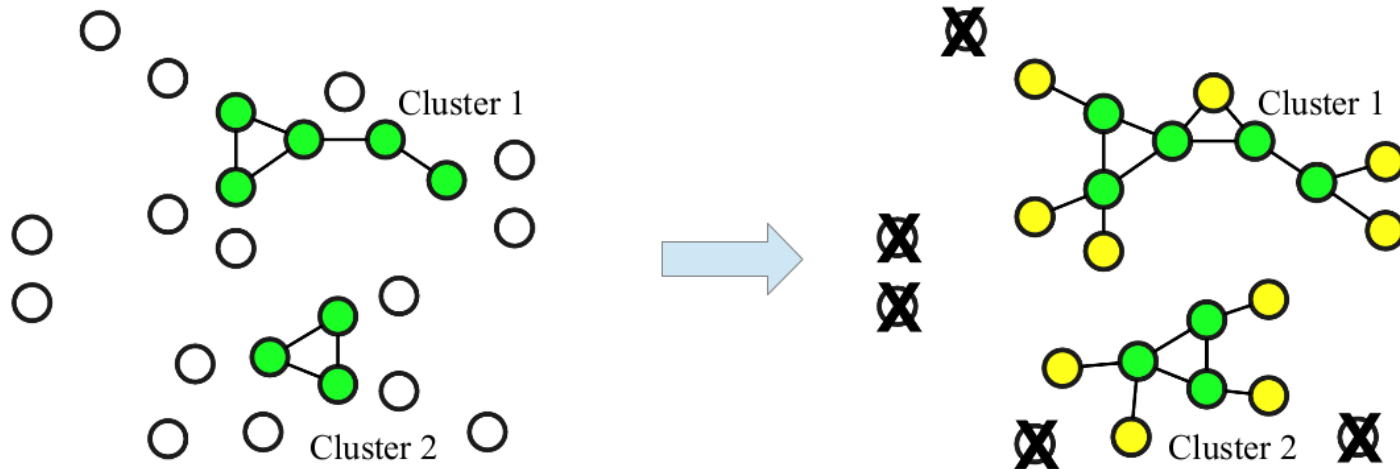
# Density Based Clustering

Step 2: connect core objects that are neighbors, and put them in the same cluster



# Density Based Clustering

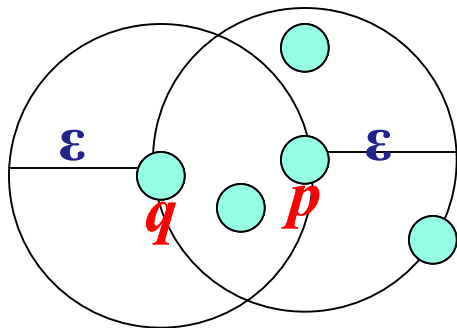
Step 3: associate border objects to (one of) their core(s), and remove noise



# 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  $\epsilon$ -neighborhood.



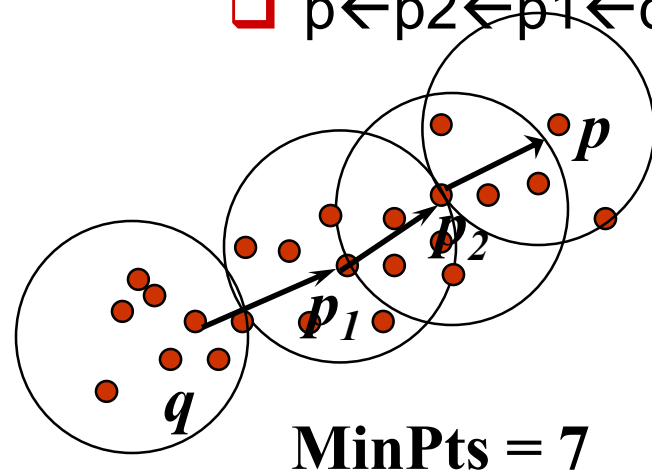
**MinPts = 4**

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

# Density-Reachability

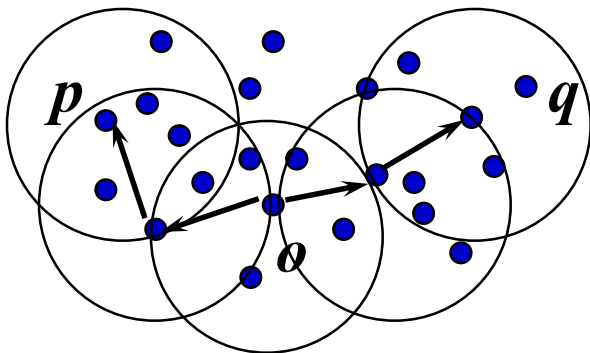
## ■ Density-reachable is not symmetric

- 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



# DBSCAN Algorithm

---

---

**Input:** The data set  $D$

**Parameter:**  $\epsilon$ , 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**

**else mark  $p$  as outlier**

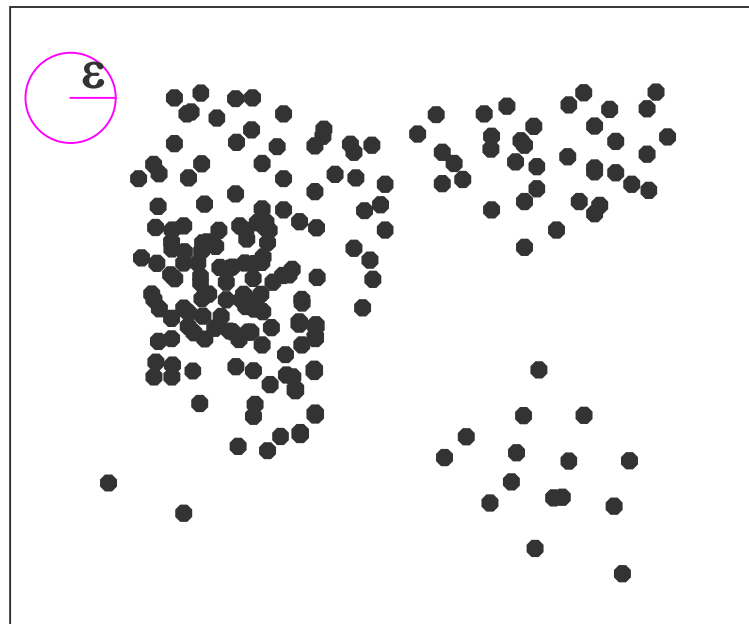
**end if**

**End For**

# DBSCAN

## ❖ Example:

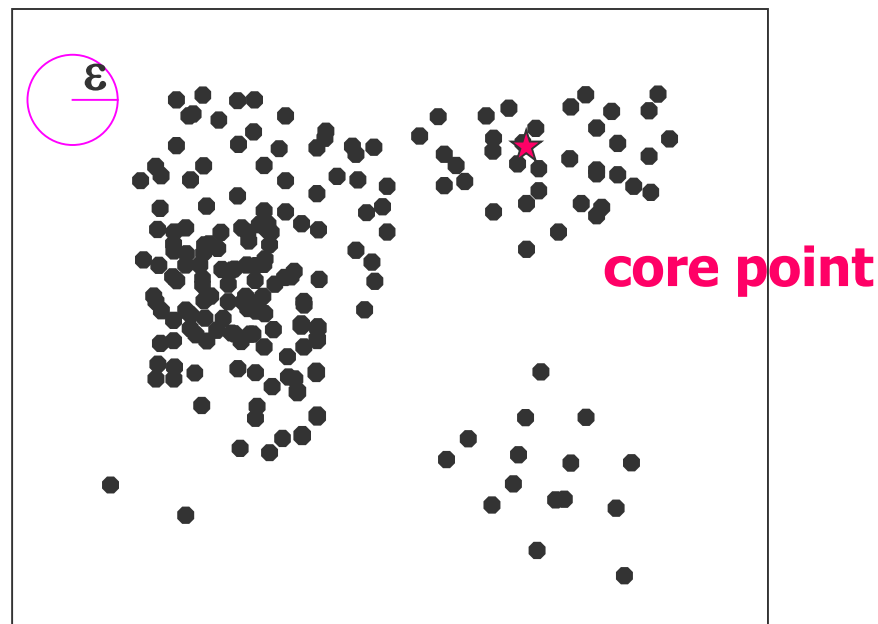
- ❖ Radius  $\varepsilon$  as shown below (Euclidean distance).
- ❖ Minimum support  $m = 7$ .
- ❖ What are the clusters?



# DBSCAN

## ❖ Example:

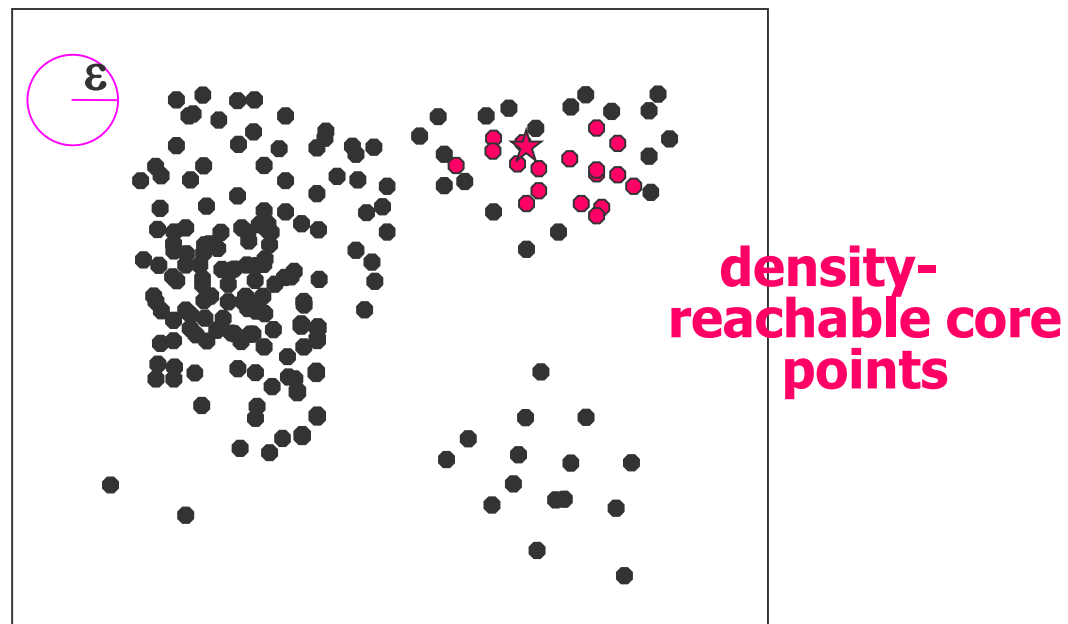
- ❖ Radius  $\varepsilon$  as shown below (Euclidean distance).
- ❖ Minimum support  $m = 7$ .
- ❖ What are the clusters?



# DBSCAN

## ❖ Example:

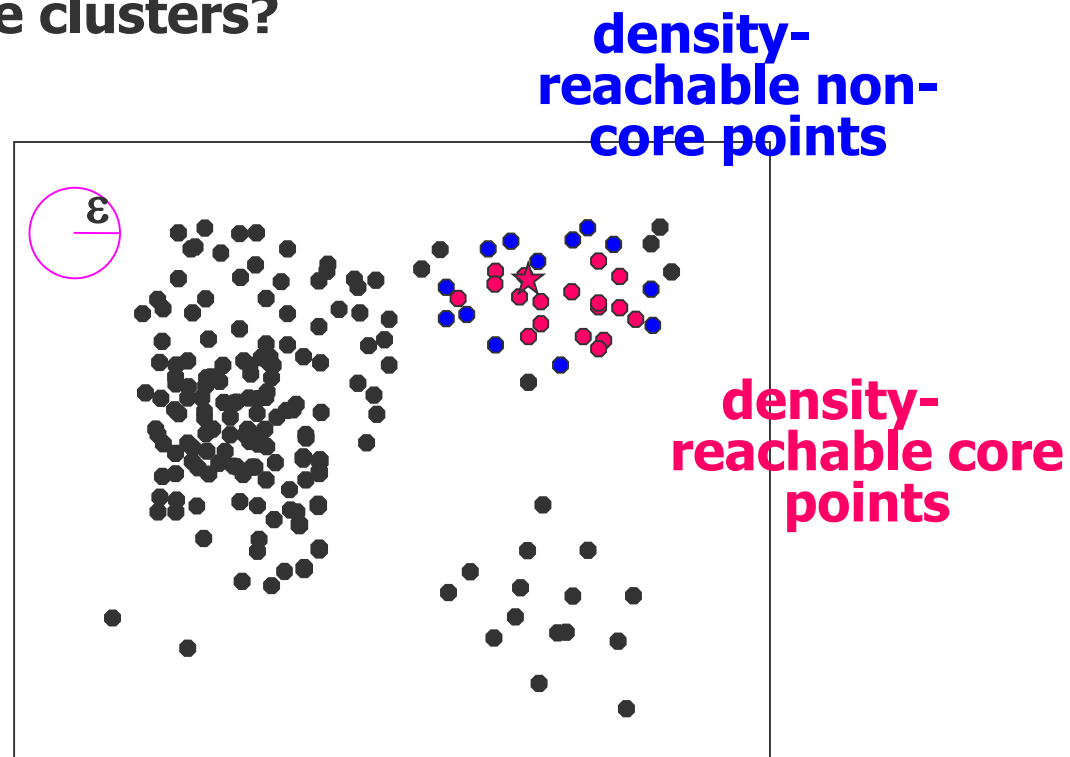
- ❖ Radius  $\varepsilon$  as shown below (Euclidean distance).
- ❖ Minimum support  $m = 7$ .
- ❖ What are the clusters?



# DBSCAN

## ❖ Example:

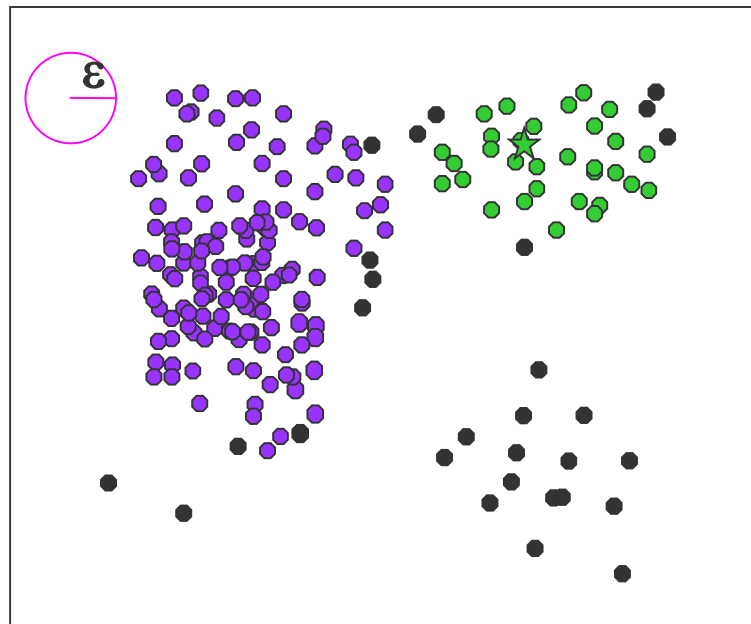
- ❖ Radius  $\varepsilon$  as shown below (Euclidean distance).
- ❖ Minimum support  $m = 7$ .
- ❖ What are the clusters?



# DBSCAN

## ❖ Example:

- ❖ Radius  $\varepsilon$  as shown below (Euclidean distance).
- ❖ Minimum support  $m = 7$ .
- ❖ 2 clusters in this example.
- ❖ Lower-right grouping not dense enough to form a cluster.



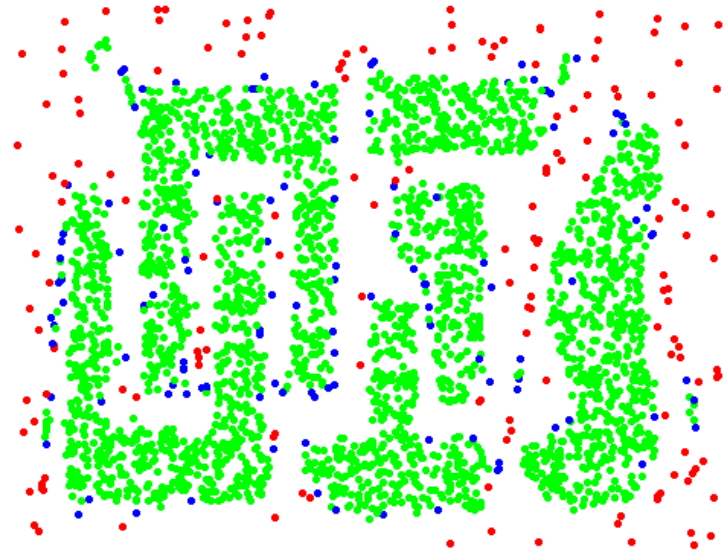


# DBSCAN: Core, Border and Noise Points

---



Original Points



Point types: **core**,  
**border** and **noise**

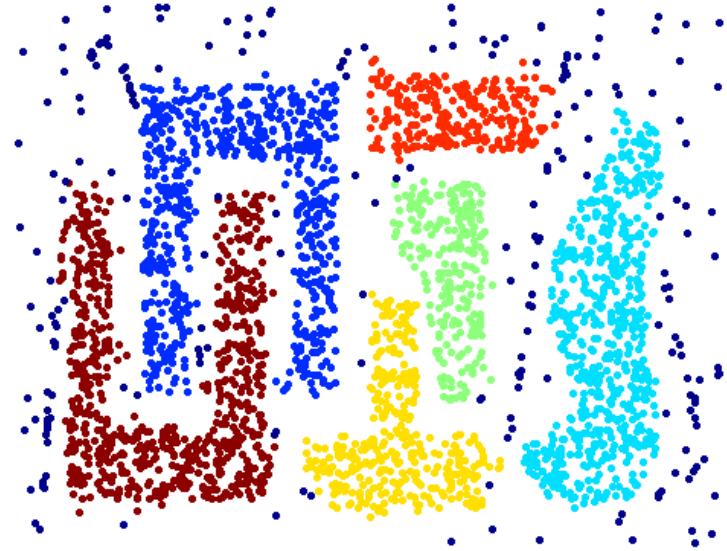
Eps = 10, MinPts = 4

# When DBSCAN Works Well

---



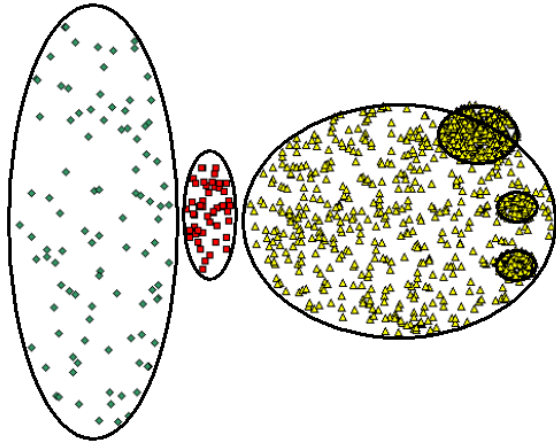
Original Points



Clusters

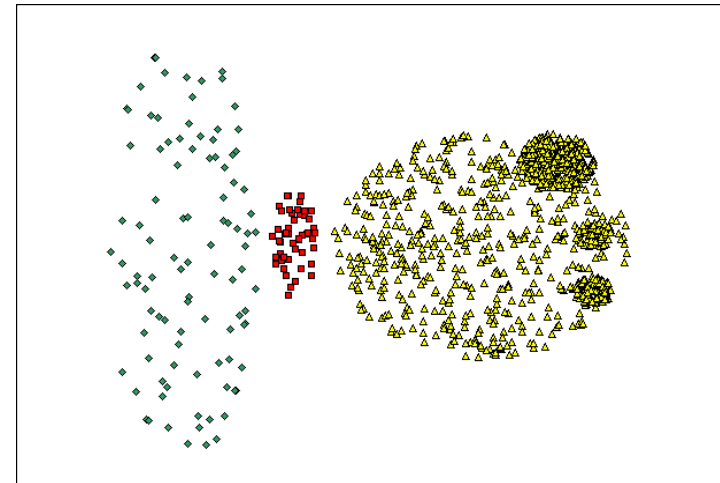
- Resistant to Noise
- Can handle clusters of different shapes and sizes

# When DBSCAN Does NOT Work Well

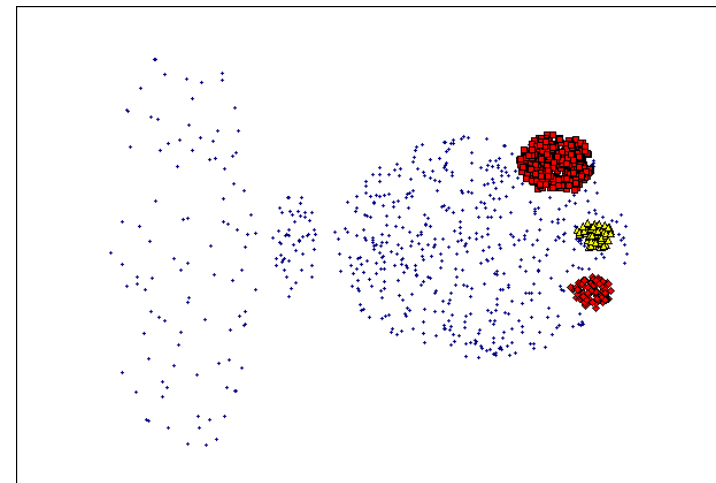


Original Points

- Varying densities
- High-dimensional data



(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.92)

# DBSCAN: Determining EPS and MinPts

- Idea is that for points in a cluster, their  $k^{\text{th}}$  nearest neighbors are at roughly the same distance
- Noise points have the  $k^{\text{th}}$  nearest neighbor at farther distance
- So, plot sorted distance of every point to its  $k^{\text{th}}$  nearest neighbor

