Clustering
Part 3

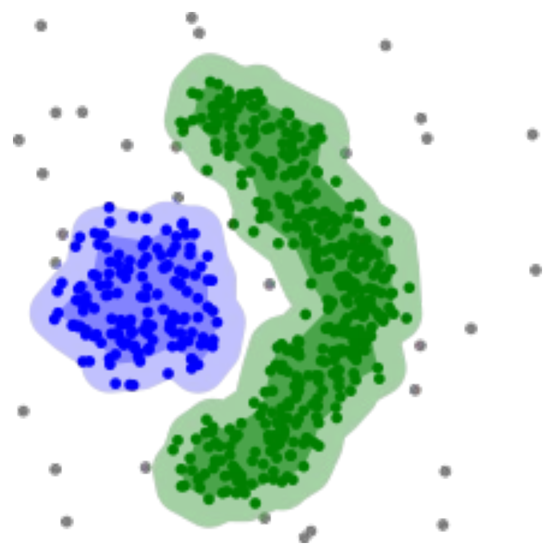
Mohammed Brahimi & Sami Belkacem

Outline

- **Overview of Clustering**
- **Major Clustering Approaches**
  - **K-means Clustering**
  - **Hierarchical Clustering**
  - **DBSCAN Clustering**
- **Cluster Evaluation**

Density-based Clustering

- Density-based methods use **density** to discover clusters of any shape.
- Density means the **concentration** of data points in a given region.
- Clusters are regions of **high density** separated by regions of **low density**.
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan only
  - No need to define the number of clusters in advance

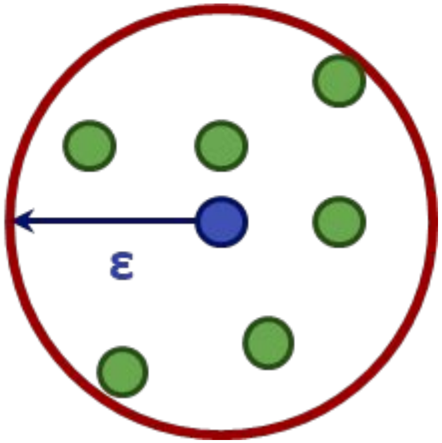


DBSCAN Clustering

- DBSCAN: **D**ensity**-B**ased **S**patial **C**lustering of **A**pplications with **N**oise
- It defines a cluster as a **maximal set of density-connected points**.
- **Density** is the number of data points within a certain **radius**.
- The parameters of this algorithm are:
  - **ε** the maximum radius of the neighborhood
  - **MinPts** the minimal number of point in a dense neighborhood.

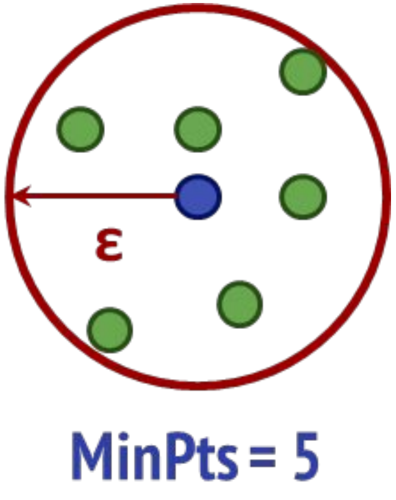
ε-neighborhood

- The neighborhood within a radius **ε** of a given object is called the **ε-neighborhood** of the object.



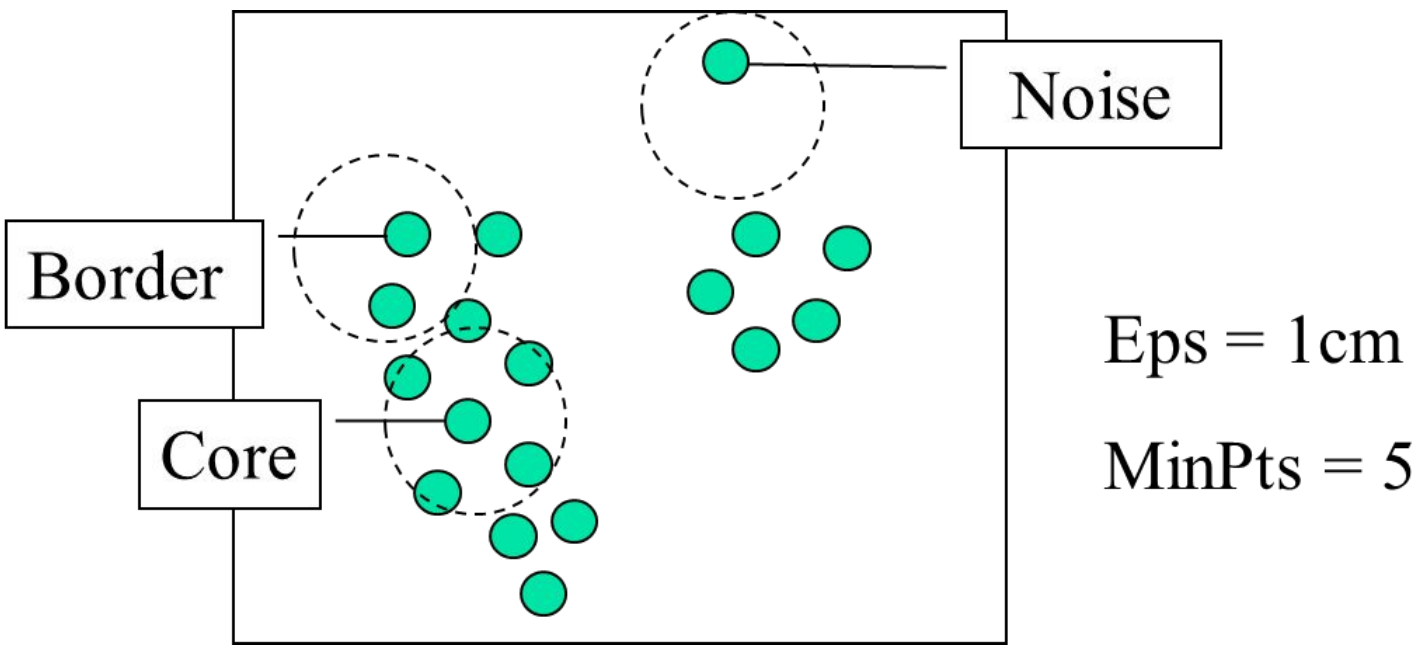
Core objects

- If the **ε-neighborhood** of an object contains at least a minimum number of objects **MinPts**, then the object is called a **core object**.



DBSCAN: Core, Border, and Noise Points

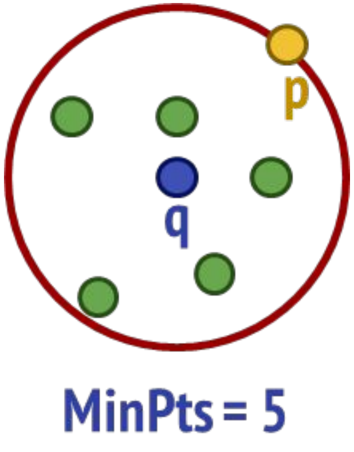
- A core point has at least a specified number of points (**MinPts**) within **ε**
- A border point is not a core point, but is in the neighborhood of a core point
- A noise point is any point that is not a core point or a border point



Density Reachability (1 / 2)

- Given a set of objects **D**, we say that an object ***p*** is **directly density-reachable** from an object ***q*** if:

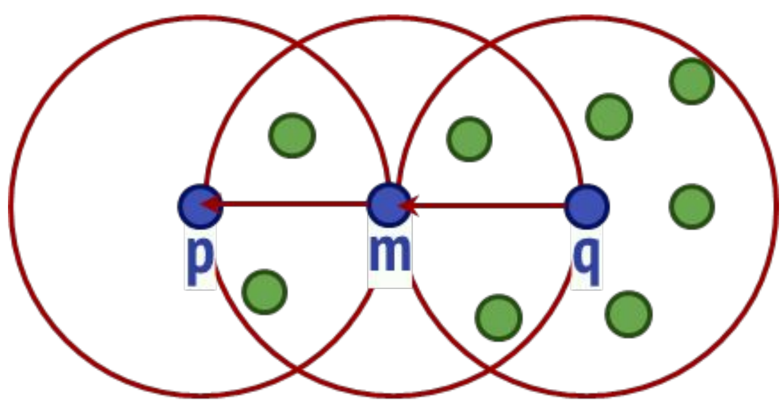
  *"**p*** is within the **ε-neighborhood** of ***q***, and ***q*** is a **core object**."



Density Reachability (2 / 2)

- Given a set of objects **D**, an object ***p*** is **density-reachable** from an object ***q*** with respect to **ε** and **MinPts** if:

  "There is a chain of objects $\mathbf{p_1, \ldots, p_n}$, where $\mathbf{p_1 = q}$ and $\mathbf{p_n = p}$ such that $\mathbf{p_{i+1}}$ is **directly density-reachable** from $\mathbf{p_i}$ with respect to **ε** and **MinPts**, for $\mathbf{1 \leq i \leq n}$ and $\mathbf{p_i} \in \mathbf{D}$."

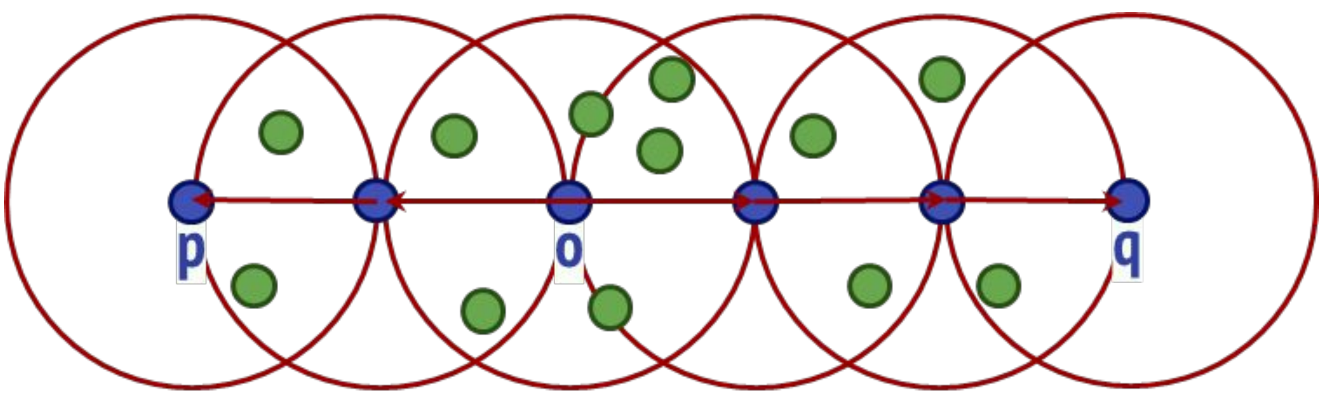


MinPts = 5

Density Connectivity

- Given a set of objects **D**, an object ***p*** is **density-connected** to an object ***q*** with respect to **ε** and **MinPts** if:

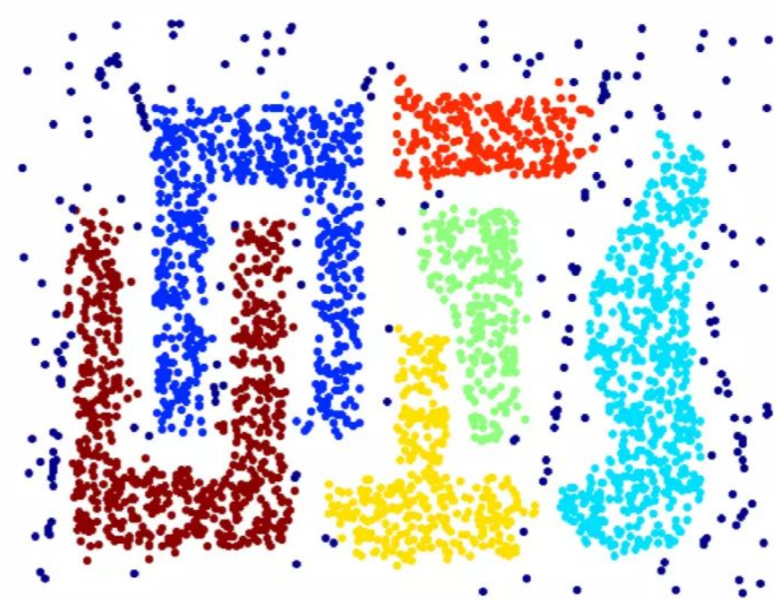
  "There is an object **o** ∈ **D** such that both **p** and **q** are **density-reachable** from **o** with respect to **ε** and **MinPts**"



MinPts = 5

Density-based Clusters

- A **density-based cluster** is a set of **density-connected** points that is maximal with respect to **density-reachability**.
- Every object not contained in any cluster is considered **noise.**

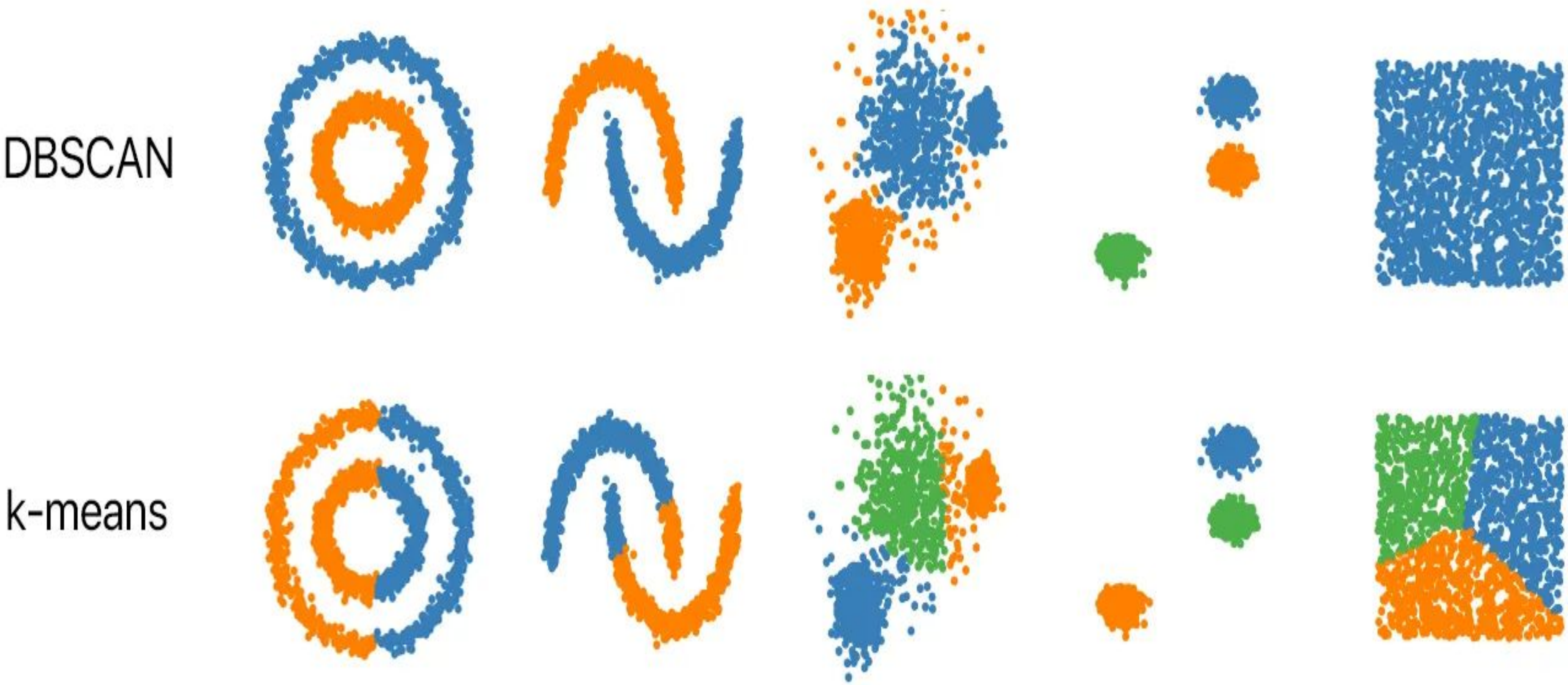
DBSCAN Algorithm: Intuition

“By recursively exploring the neighborhood of **core points** within the **ε**-distance threshold and incorporating **reachable points** into clusters, the DBSCAN algorithm identifies dense regions in the dataset while also detecting **outliers** (noise points) that do not fit within these dense regions.”

DBSCAN Algorithm

1. Choose **ε** (a positive number) and **MinPoints** (a natural number).
2. Select an arbitrary point **P** from the dataset.
3. Check if point **P** is a core point. If yes, form a cluster including **P**.
4. Recursively add core points within the ε-neighborhood of the already added points to the cluster.
5. After fully expanding a cluster, select a new unvisited point and repeat the process (from point **2** to **4**).
6. **Handling Border Points:** Assign each border point to one of the clusters of its ε-neighborhood core points.
7. **Noise Identification:** Label points that are neither core points nor border points as noise.

DBSCAN vs K-Means (1 / 2)



DBSCAN vs K-Means (2 / 2)

- K-Means algorithm with different values of ***K*** and shapes of data:

  https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

- DBSCAN algorithm with different shapes of data:

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

DBSCAN: Determining MinPts

General guidelines for setting ***MinPts***:

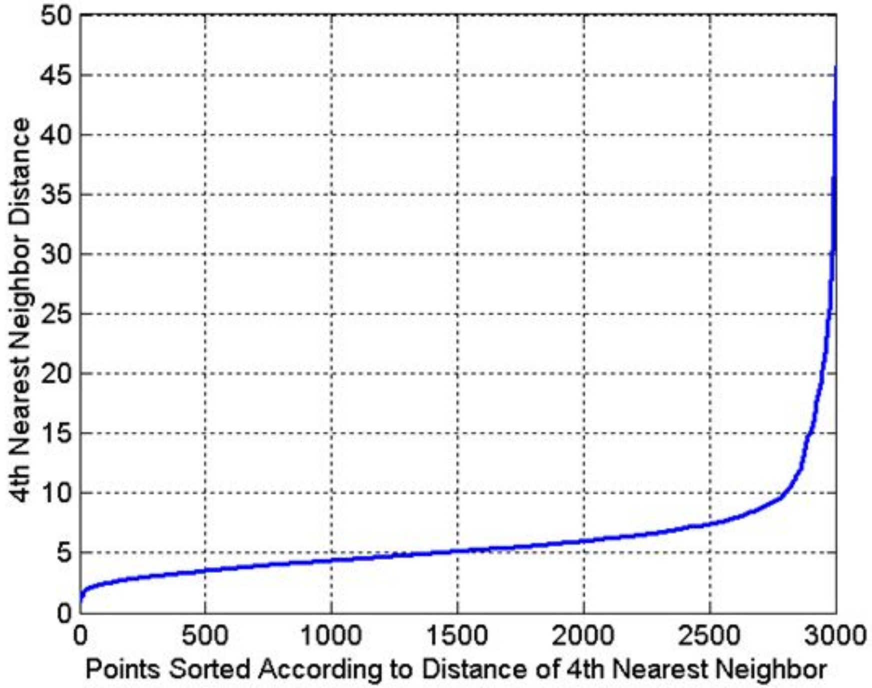
- Larger datasets require a larger ***MinPts*** value.
- ***MinPts*** must be chosen at least 3.
- In noisier datasets, choose a larger ***MinPts*** value.
- ***MinPts*** should generally be ≥ the dimensionality of the dataset.
  - **Example:** MinPts = 2 * number of dimensions
- Domain knowledge is crucial to select an appropriate ***MinPts*** value.

DBSCAN: Determining ε using K-Distance Plot

- Compute the average distance of each point to its ***K*** (***K*** = ***MinPts*** - 1) nearest neighbors.
- Plot these ***K***-distances in ascending order.
- The 'knee' in the plot represents a threshold where a sharp change in distance occurs.
- This point is indicative of the optimal **ε** value.
- Helps to distinguish between core, border, and noise points in the data

DBSCAN: Determining ε and MinPts

- **Idea:** for points in a cluster, their ***K***th nearest neighbors are at close 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



DBSCAN: Advantages and Disadvantages

- **Advantages:**
  - Can handle clusters of different shapes and sizes
  - Resistant to noise and outliers
  - Doesn’t require predefined number of clusters.
- **Disadvantages:**
  - Sensitivity to the two parameters **ε** and **MinPts**
  - Difficulty with varying density clusters
  - Not suitable for high-dimensional data due to the curse of dimensionality

Outline

- Overview of Clustering
- Major Clustering Approaches
  - K-means Clustering
  - Hierarchical Clustering
  - DBSCAN Clustering
- Cluster Evaluation