Unsupervised Learning: K-Means Clustering & DBSCAN

BY: SUBASH SAH

Introduction to Machine Learning

- Machine Learning (ML) is a subset of Artificial Intelligence (AI).
- It enables computers to learn from data and make decisions without being explicitly programmed.
- ML focuses on patterns, predictions and data-driven decisions
- Example: Email spam detection, product recommendations, fraud detection.

Types of Machine Learning

1. Supervised Learning
   - Uses labeled data (input -> output known)
   - Examples: Regression, Classification
     - Example: Predicting house prices, detecting spam emails
2. Unsupervised Learning
   - Uses unlabeled data (no predefined output)
   - Finds hidden patterns or groups.
     - Example: Customer segmentation, anomaly detection
3. Reinforcement Learning
   - Learns by interacting with an environment and receiving rewards or penalties.
     - Example: Self-driving cars, game-playing AI.

Introduction to Unsupervised Learning

- In unsupervised learning, the model explores data and finds structure without labeled outputs.
- It is mainly used for clustering and association tasks.
- The system learns by observing similarities and differences between data points.

Goal: Discover hidden groups or relationships.

Examples of Unsupervised Learning

- Clustering: K-Means, DBSCAN, Hierarchical Clustering
- Association: Apriori, FP-Growth (used in Market Basket Analysis)
- Dimensionality Reduction: PCA

What is Clustering?

- Clustering groups data points that are similar to each other.
- It helps understand data distribution and structure.
- Common use cases:
  - Customer segmentation
  - Image compression
  - Document grouping
  - Fraud detection

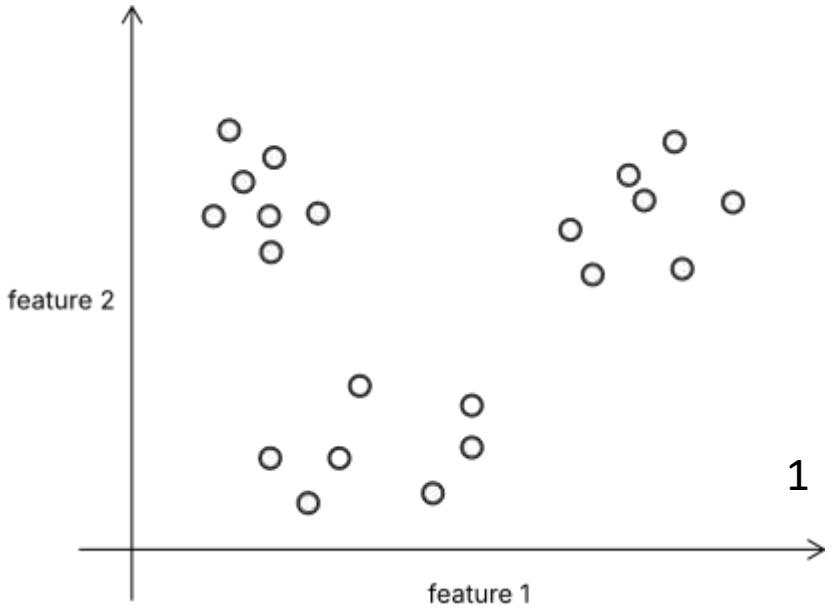
K-Means Clustering

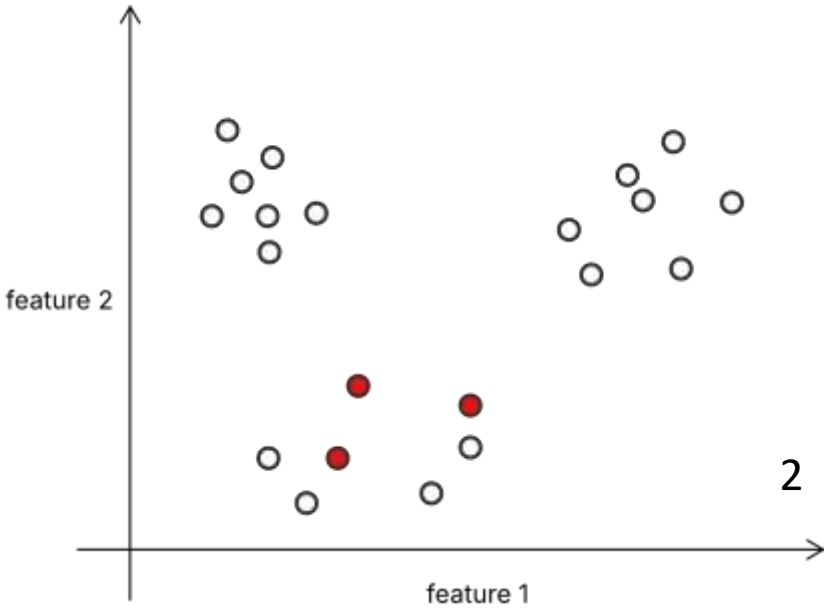
- K-Means is one of the most popular clustering algorithms.
- It divides data into K groups (clusters) based on feature similarity.
- Each cluster has a centroid – the mean position of all points in that cluster.

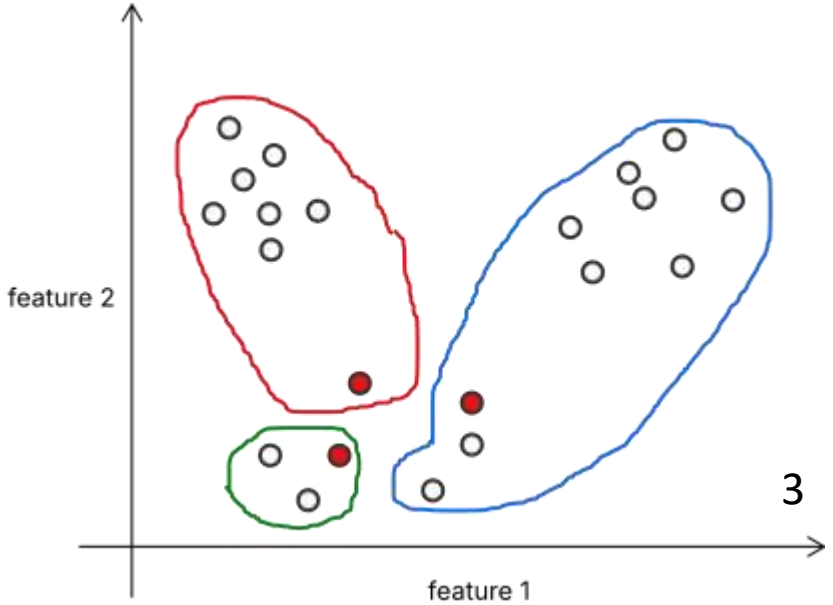
K-Means Algorithm steps

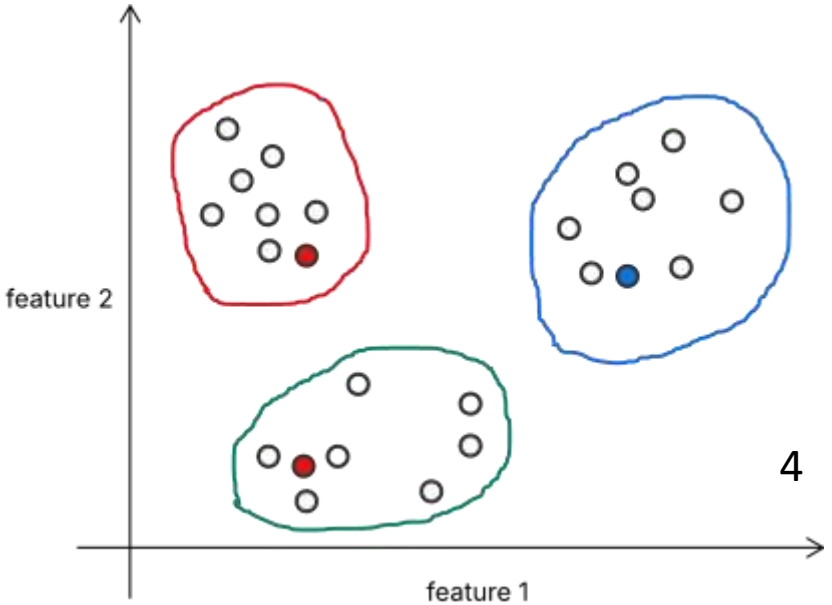
- Choose number of clusters K
- Initialize K centroids randomly.
- Assign each point to the nearest centroid (based on distance).
- Recalculate centroids based on cluster assignments.
- Repeat until centroids stop moving (convergence).

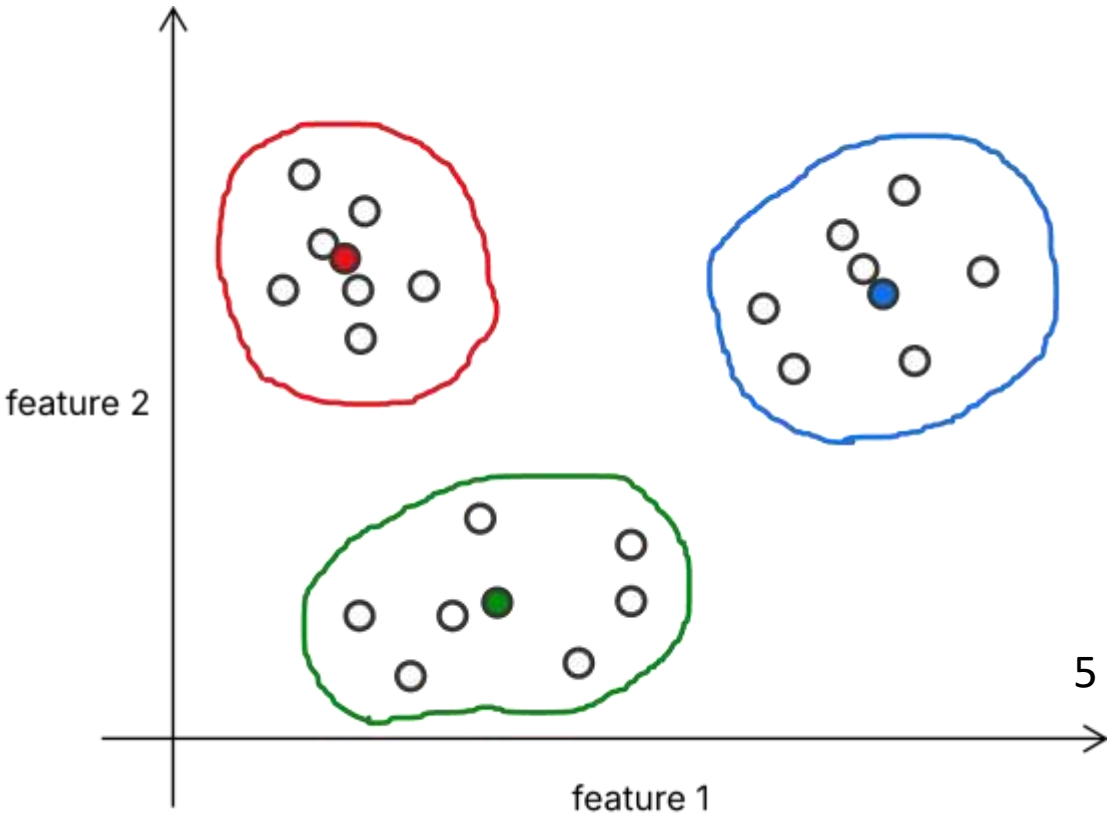
Choosing K: Elbow Method or Silhouette Score to find the optimal number of clusters.


feature 2
feature 1
1

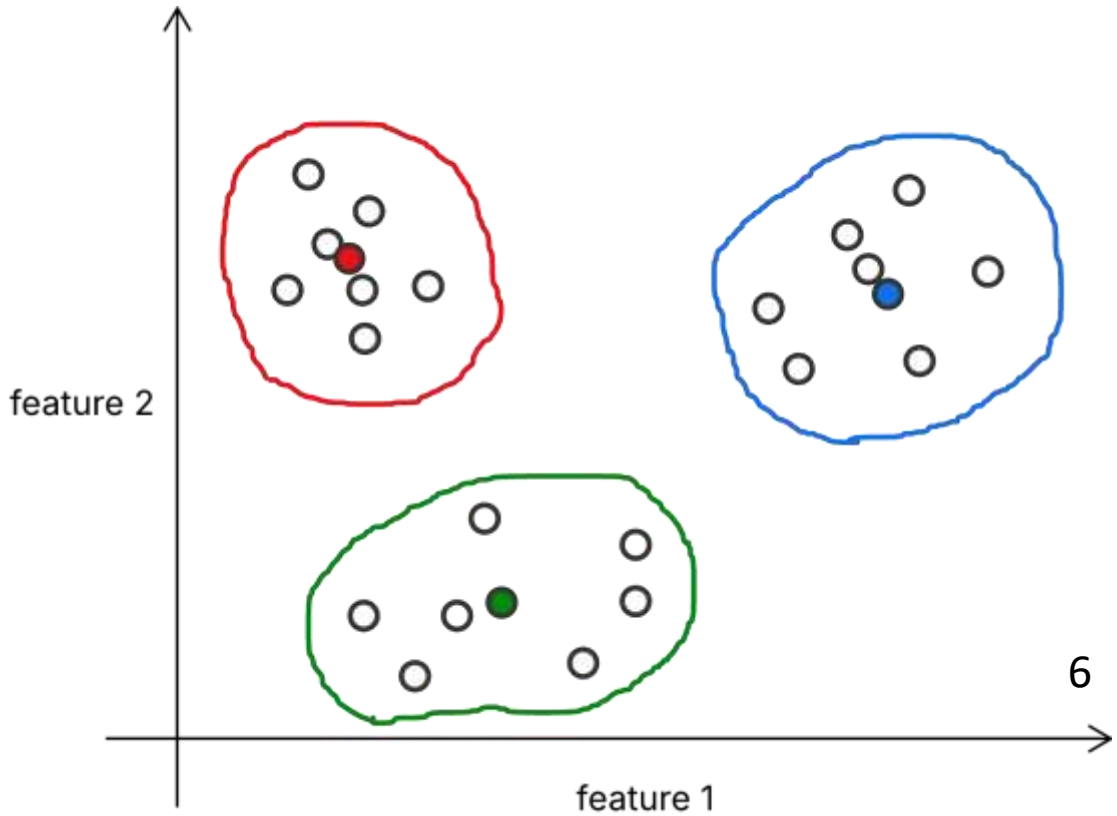

feature 2
feature 1
2


feature 2
feature 1
3


feature 2
feature 1
4



5



6

Advantages and Limitations of K-Means

Advantages:

- Simple and easy to implement.
- Works well with large datasets.
- Fast convergence.

Limitations:

- Must specify K beforehand.
- Sensitive to outliers.
- Works best for spherical-shaped clusters.
- Struggles with irregular or noisy data.

DBSCAN

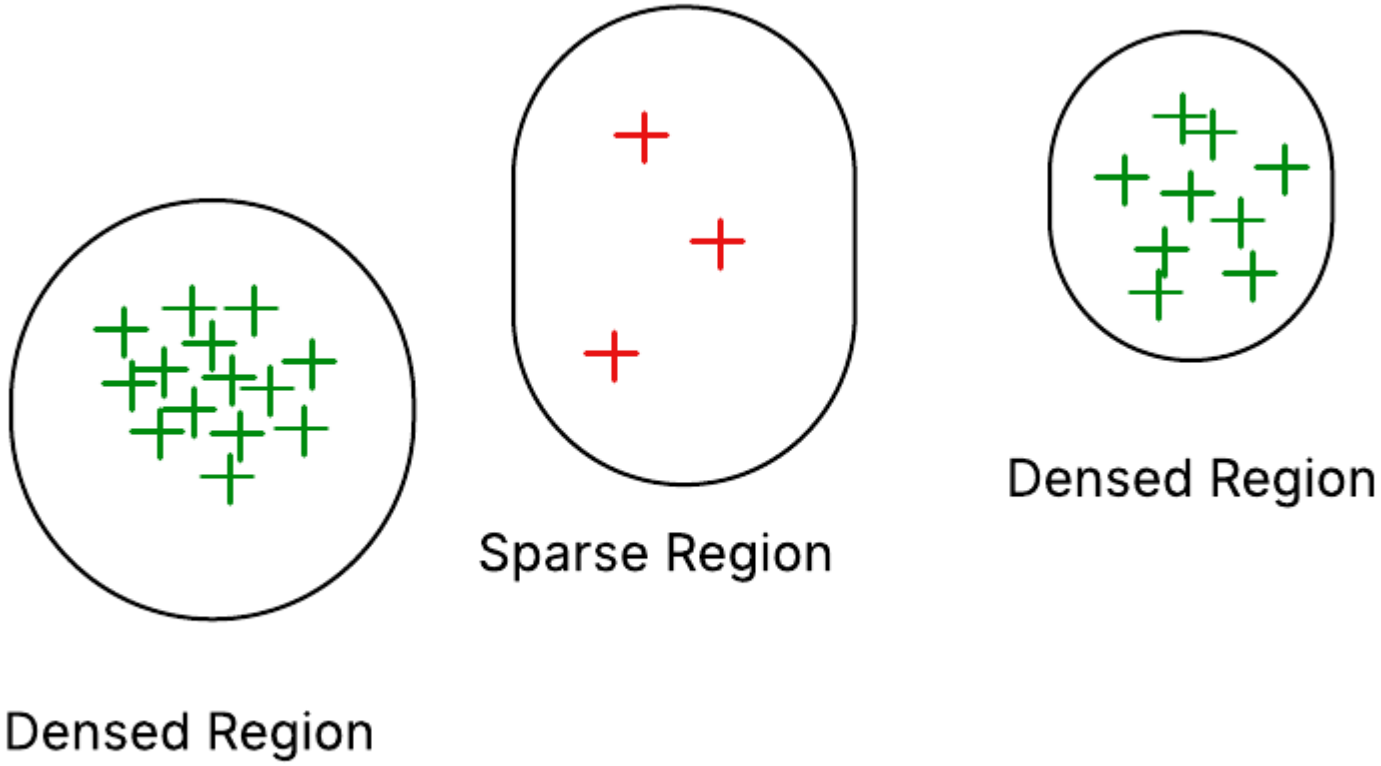
Density-Based clustering algorithm.

Best suited for handling noises or outliers.

Mostly preferred as other clustering algorithms are affected by outliers.

Clustering is done on the basis of density.

Key Idea: Dense regions = clusters; sparse regions = noise.



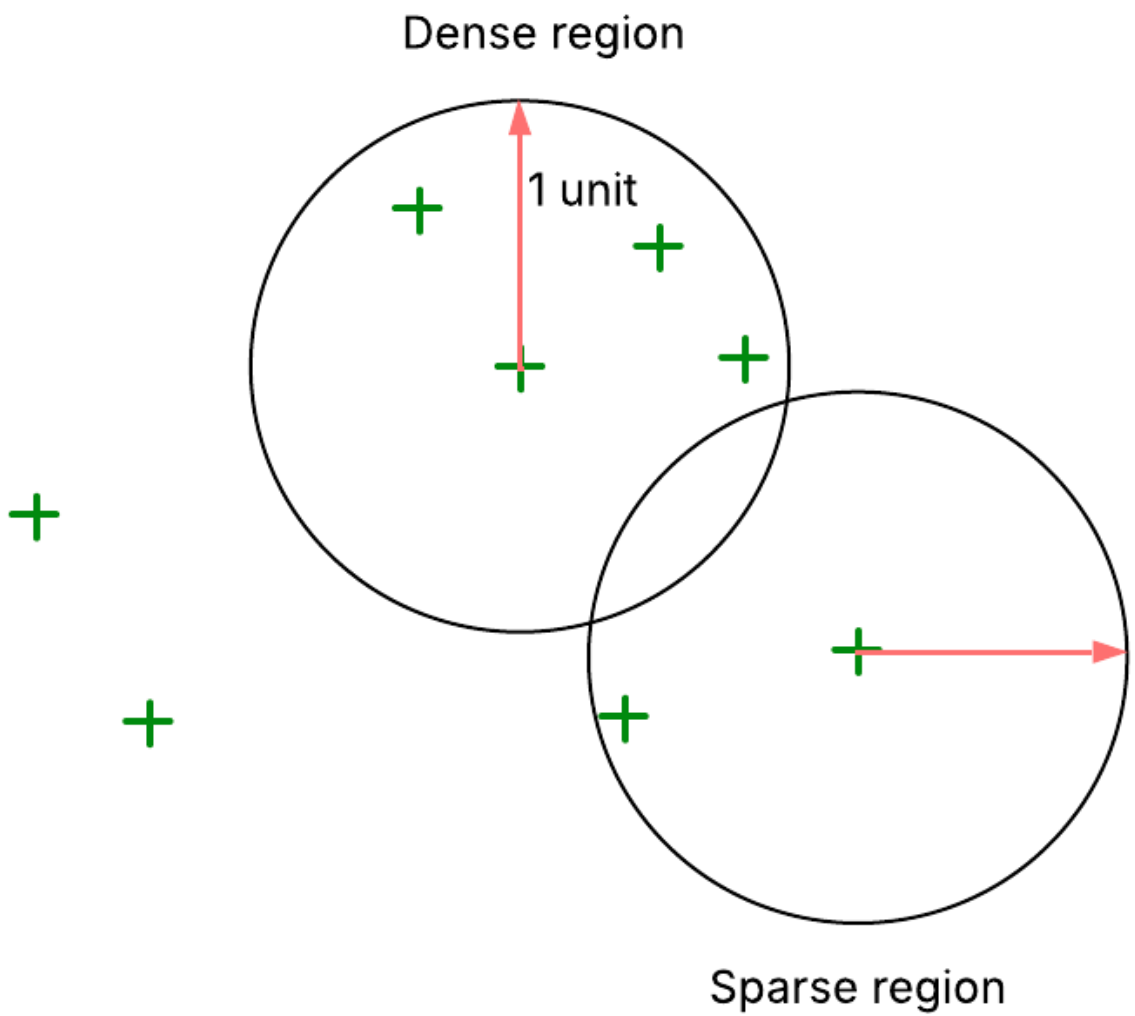
On the basis of that sparse region, the two dense regions are separated as different clusters.

How to know dense or sparse region?

Two important hyperparameters:

- MinPts: Maximum distance between two samples to be considered neighbors.
- Epsilon: Minimum number of points required to form a dense region (cluster).

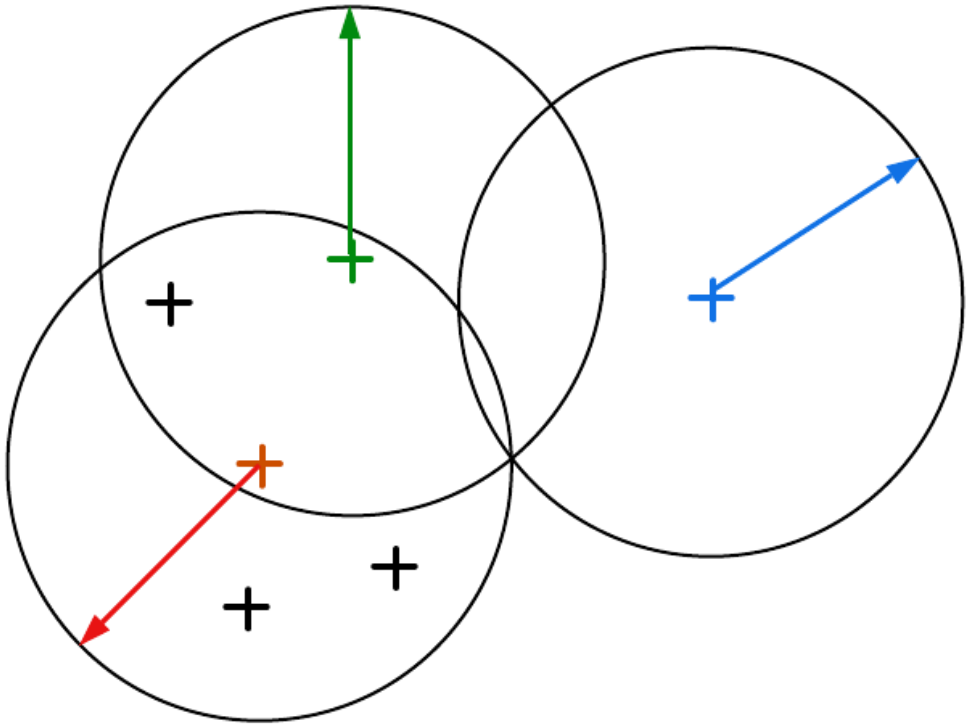
Lets say epsilon = 1 unit and MinPts = 4, then a region is considered dense if a circle is drawn from a point with radius = 1 unit and that circle contains at least 4 datapoints.



MinPts = 4
Epsilon = 1

Three important points

- **Core Point**: If a circle drawn from a point is dense, then that point is a core point.
- **Border Point**: If a circle drawn from a point is sparse, but it has at least one core point.
- **Noise**: A circle drawn from a point is sparse and it doesn’t contain any core points inside it.



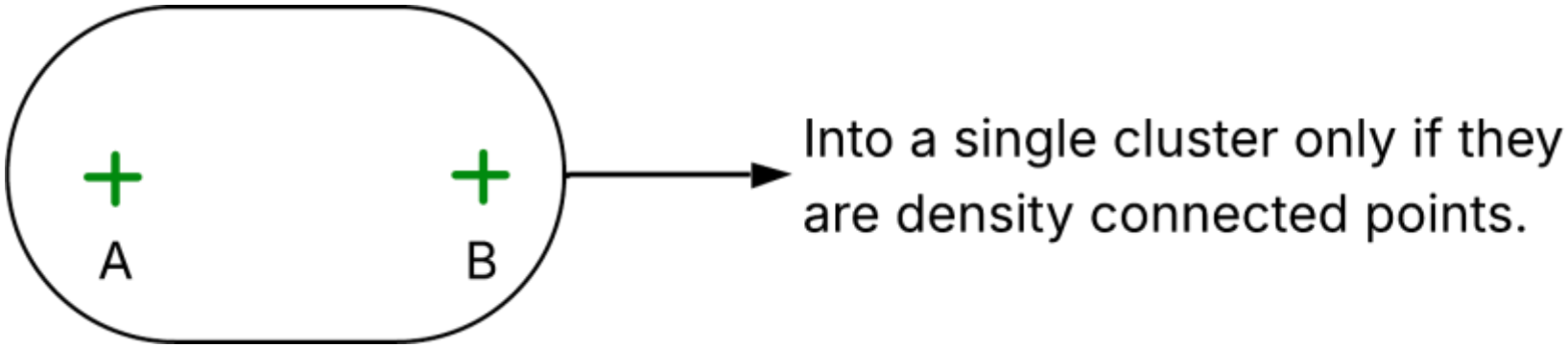
Border point

Core point

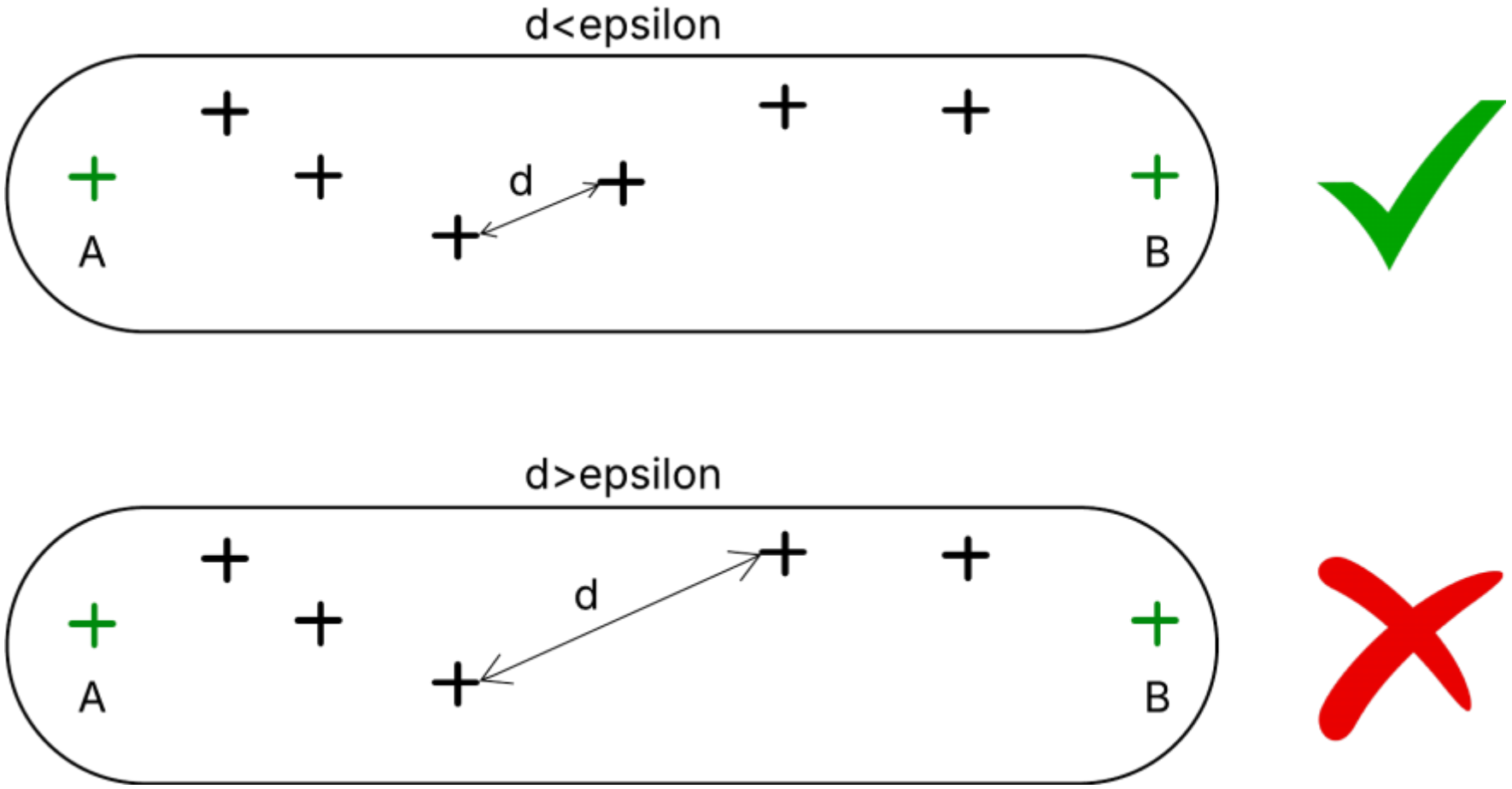
Noise

Density connected points

If two points are **density connected points** then they are put into a single cluster.

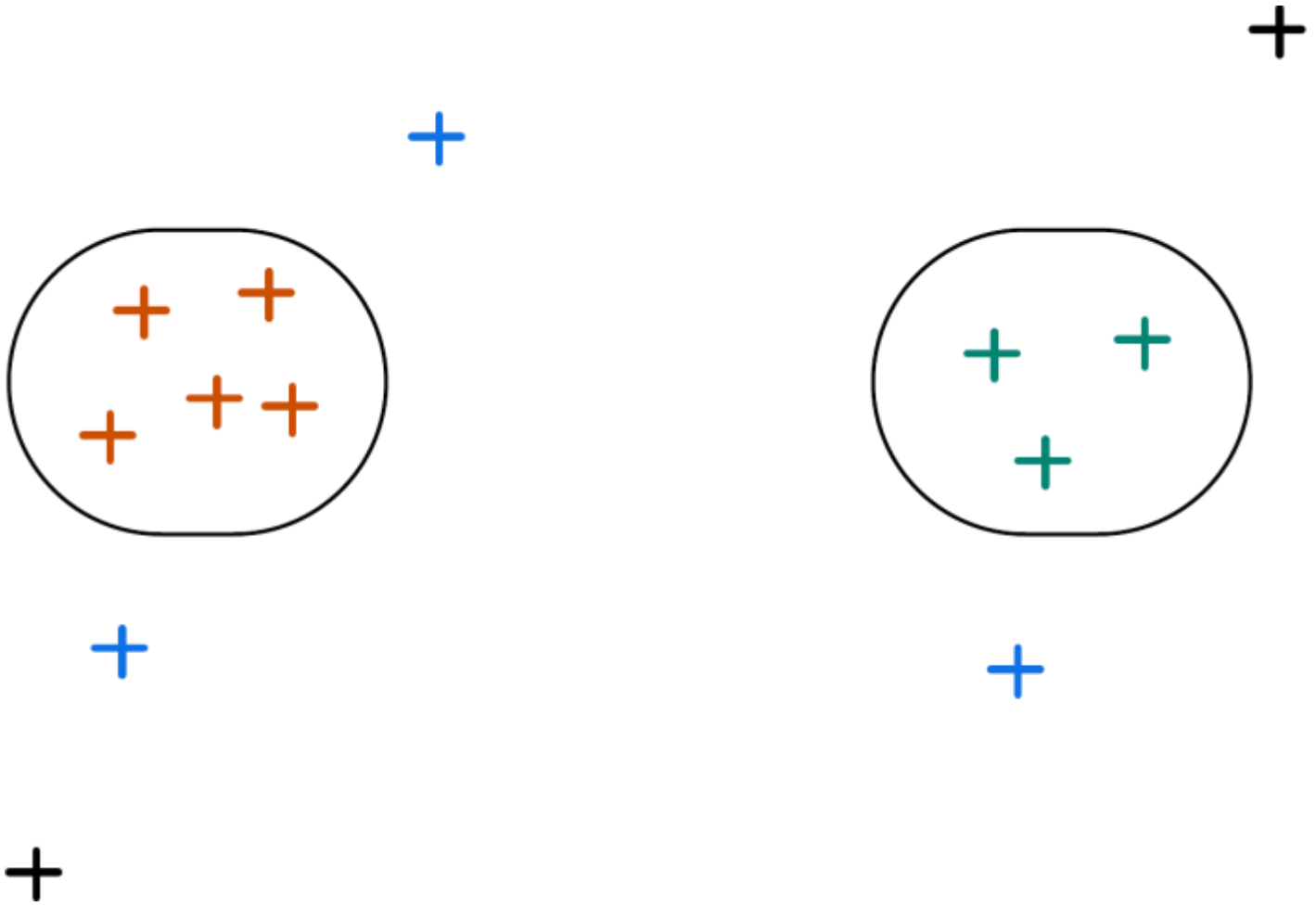


So what are density connected points?


d<epsilon
d
A
B
d>epsilon
d
A
B

DBSCAN Algorithm

- Initialize MinPts and Epsilon
- Identify all the points as either core points, border points or noise.
- For all non clustered core points:
  - Create a new cluster
  - Add all the points that are non clustered and density connected to the current point into this cluster.
- For each non clustered border points, assign it to the nearest cluster of nearest core point.
- Leave all the noise points as it is.



Advantages & Limitations of DBSCAN

Advantages:

- No need to specify K.
- Detects clusters of arbitrary shapes.
- Handles noise and outliers effectively.

Limitations:

- Struggles with varying densities.
- Sensitive to choice of epsilon and MinPts.
- Computationally heavy for very large datasets.
- Cannot predict for new datapoint.

K-Means vs DBSCAN (Comparison Table)

Feature	K-Means	DBSCAN
Need K?	Yes	No
Handles Noise	Poorly	Well
Cluster Shape	Spherical	Arbitrary
Outlier Sensitivity	High	Low
Speed	Fast	Slower (density-based)
Suitable for	Large, well-separated data	Data with noise or irregular clusters
Predict new data?	Yes Has centroids-distance based assignments	No Non-parametric(It doesn't learn fix parameters like centroids, weights)

Thank You