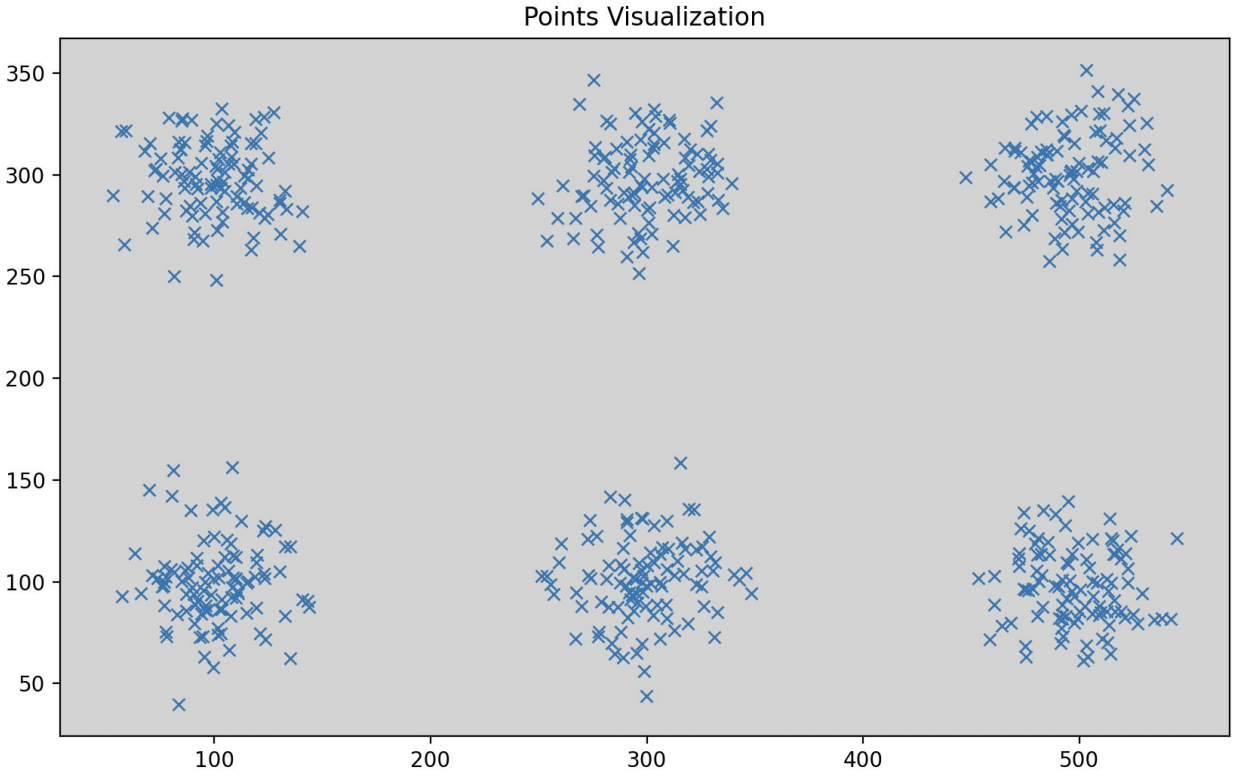
Discussion ECS 111

Week 3 04/15/2025

We will cover:

- Assignment Description
- Clustering Algorithm Demonstrations
  - K-Means(Partitional Clustering)
  - DBSCAN(Density-based Clustering)
- After clustering the trainset
  - How to evaluate the testset

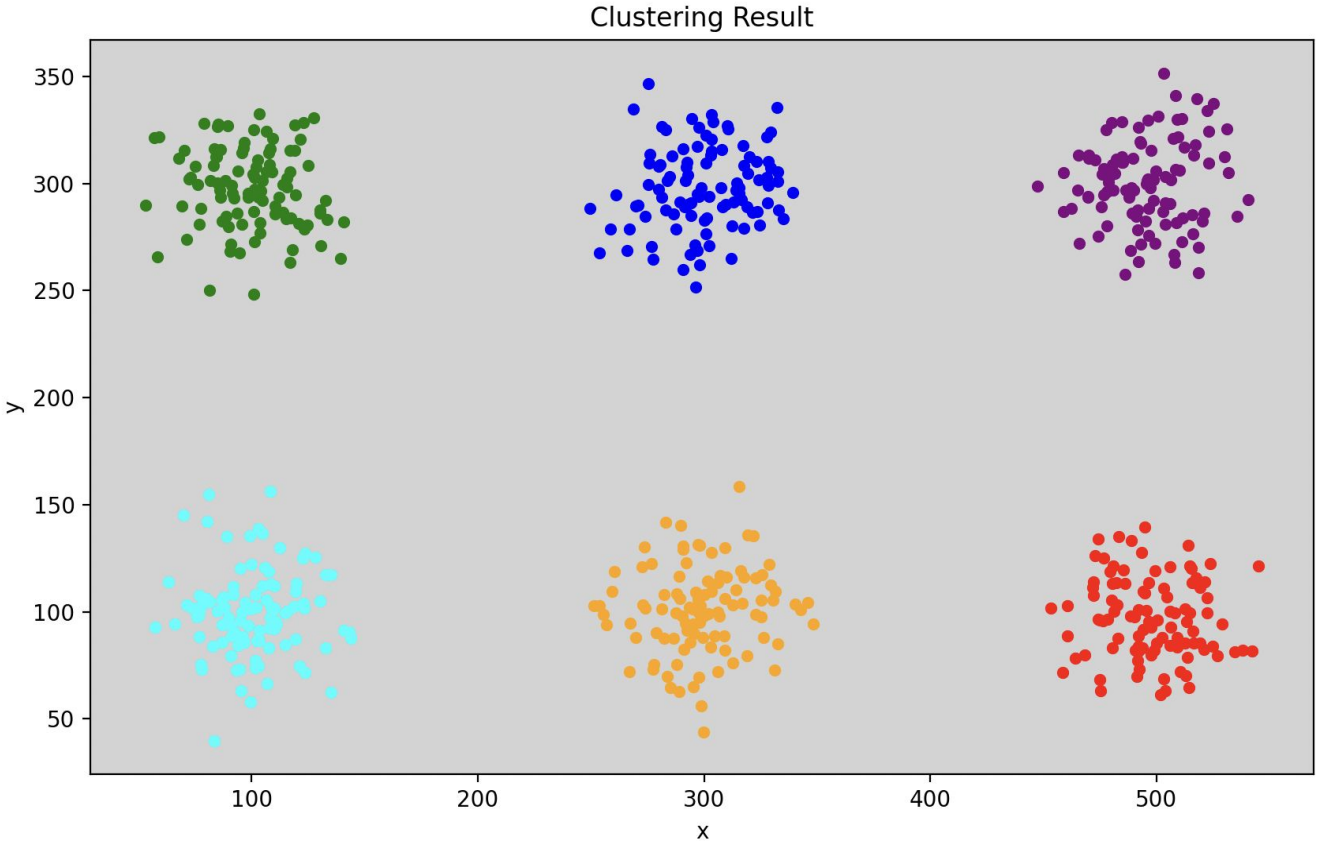
Assignment illustration Example



Group of data points

Need to be clustered

Clustering Results-train



Use Clustering Algorithm

Find these 6 clusters

+ Build algorithm
+ Run algorithm
+ Evaluate the clustering result
+ Tune parameters

K-means Steps

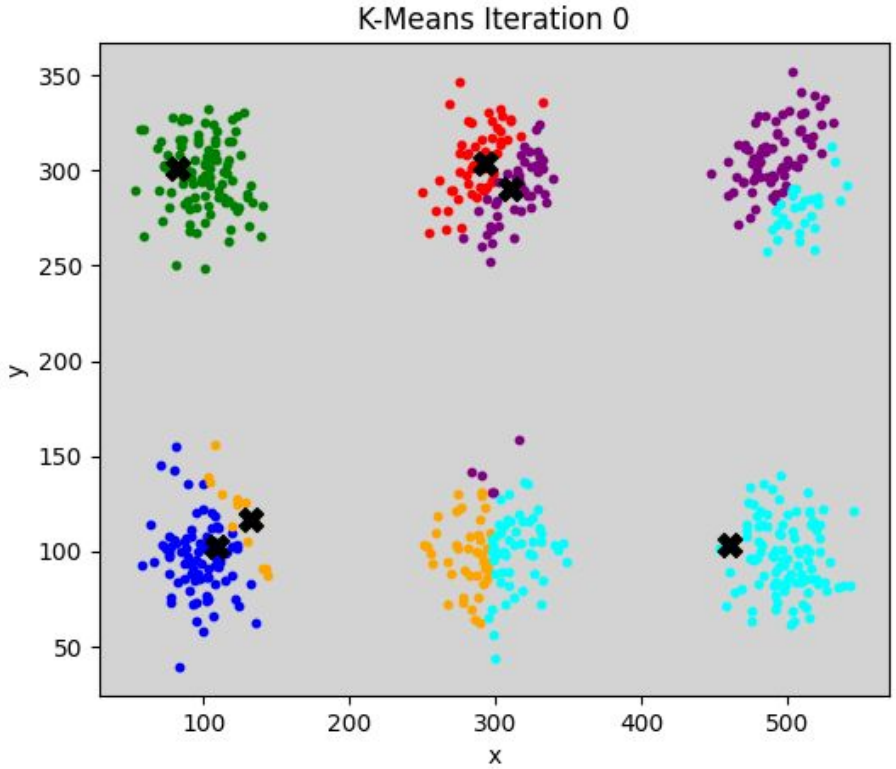
1. Initialization:
   - Randomly select k initial cluster centers (called centroids).

2. Assignment Step:
   - For each data point, compute its distance to all centroids.
   - Assign each point to the cluster with the nearest centroid.

3. Update Step:
   - For each cluster, recalculate the centroid as the mean of all assigned points.

4. Repeat Steps 2 and 3 until:
   - The cluster assignments do not change, or
   - The centroids converge (i.e., movement is below a threshold), or
   - A maximum number of iterations is reached.

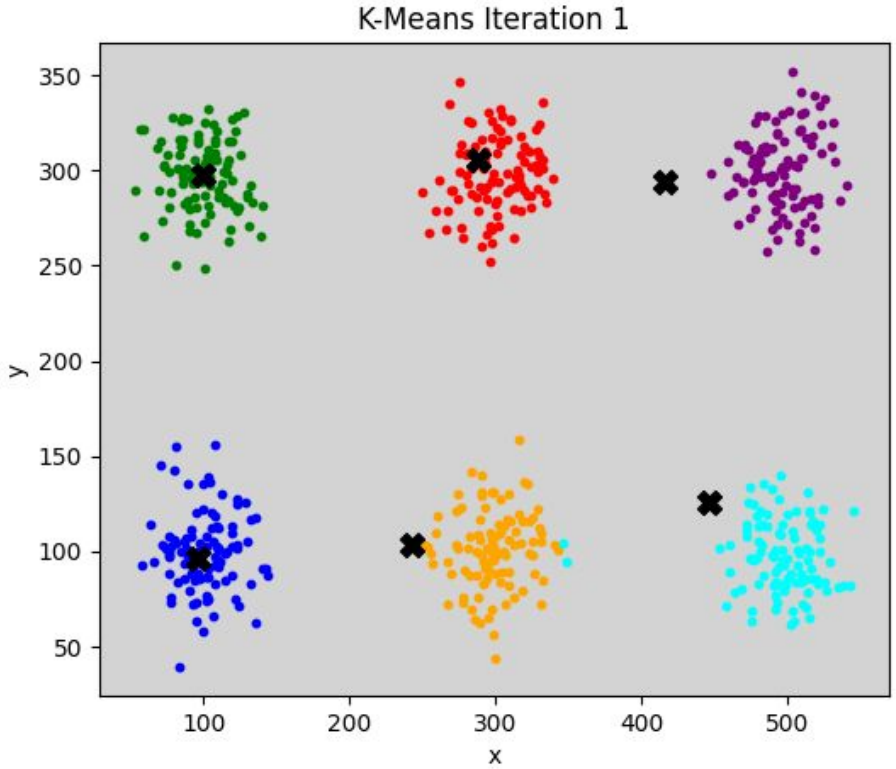
K-means Iteration-Wise Demonstrations



Iteration 0 Initialization

+ Initial cluster centers are randomly selected
+ Each data point is assigned to the nearest center based on Euclidean distance

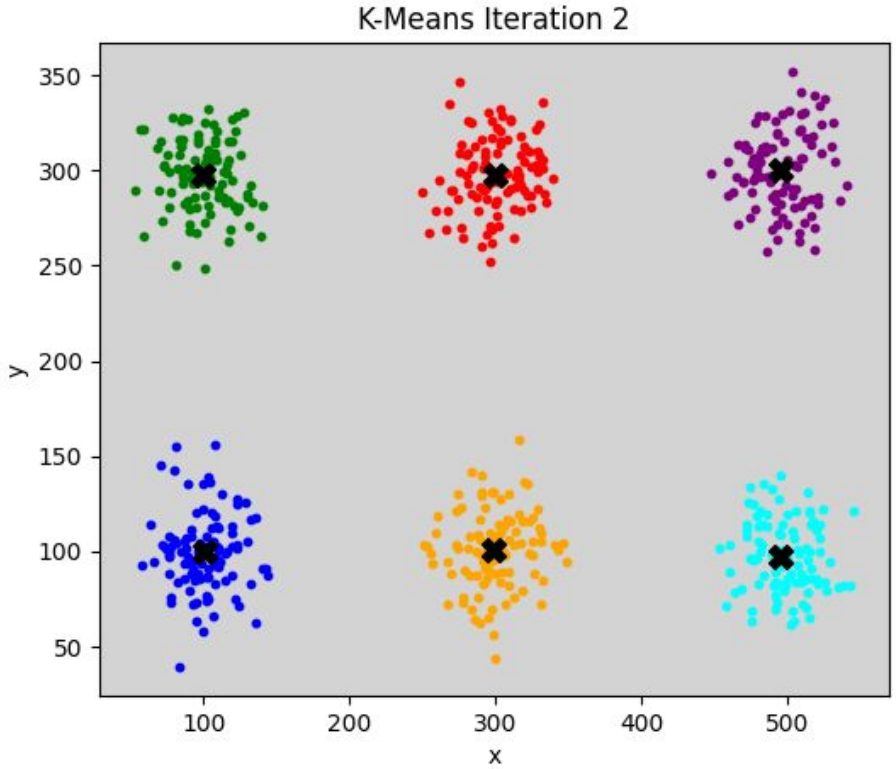
K-means Iteration-Wise Demonstrations



Iteration 1 First Update

+ Cluster centers have moved to the mean of the assigned points
+ Data points are re-assigned based on the updated centers.

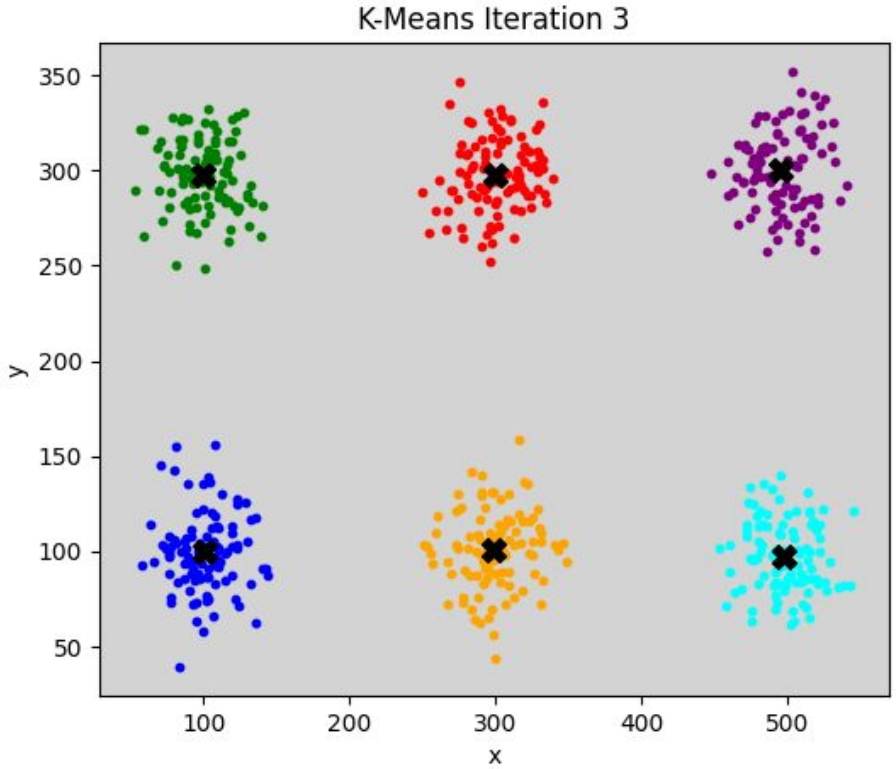
K-means Iteration-Wise Demonstrations



Iteration 2 Refinement

+ Clusters continue to refine as centers move closer to true cluster centers
+ More data points are correctly grouped together

K-means Iteration-Wise Demonstrations



Iteration 3 Convergence

+ Cluster centers have stabilized
+ Assignments no longer change as K-Means has converged.

DBSCAN Steps

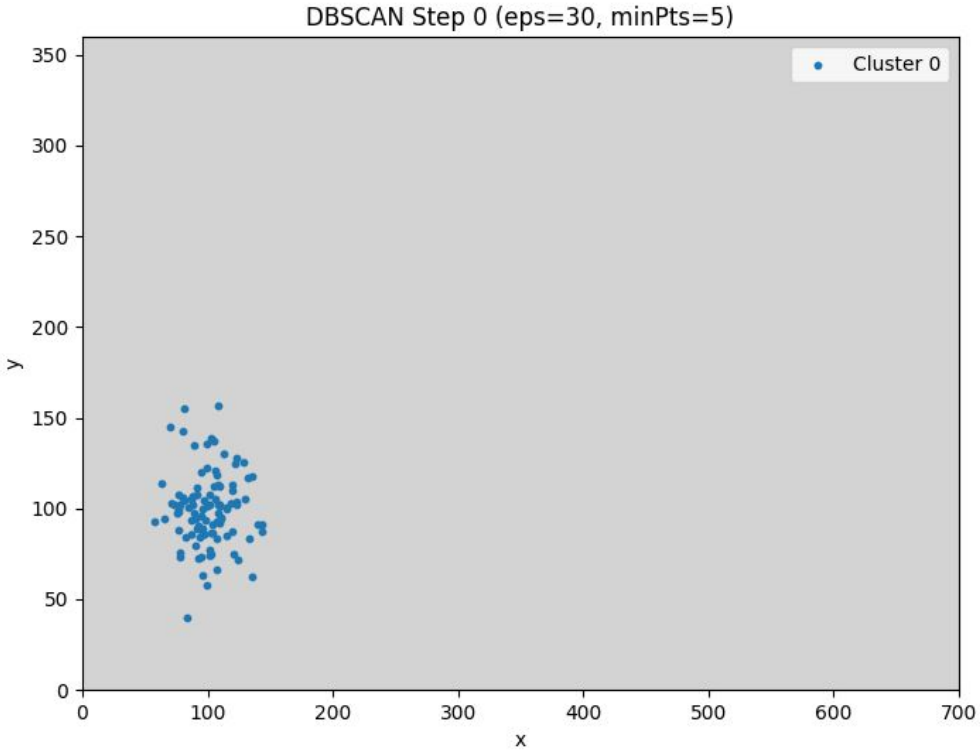
1. Define Parameters:
   - eps (ε): Neighborhood radius.
   - minPts: Minimum number of points to form a dense region (core point).

2. Label Points:
   - For each point:
   - If it has at least minPts neighbors within radius eps:
     → mark it as a core point.
   - If it has too few neighbors and is not part of a cluster:
     → mark as noise.
   - Otherwise, it may later be included as a border point.

3. Cluster Formation:
   - For each unvisited core point:
   - Start a new cluster.
   - Expand it by recursively including all density-reachable points.

4. Continue until all points are visited.

DBSCAN Iteration-Wise Demonstrations



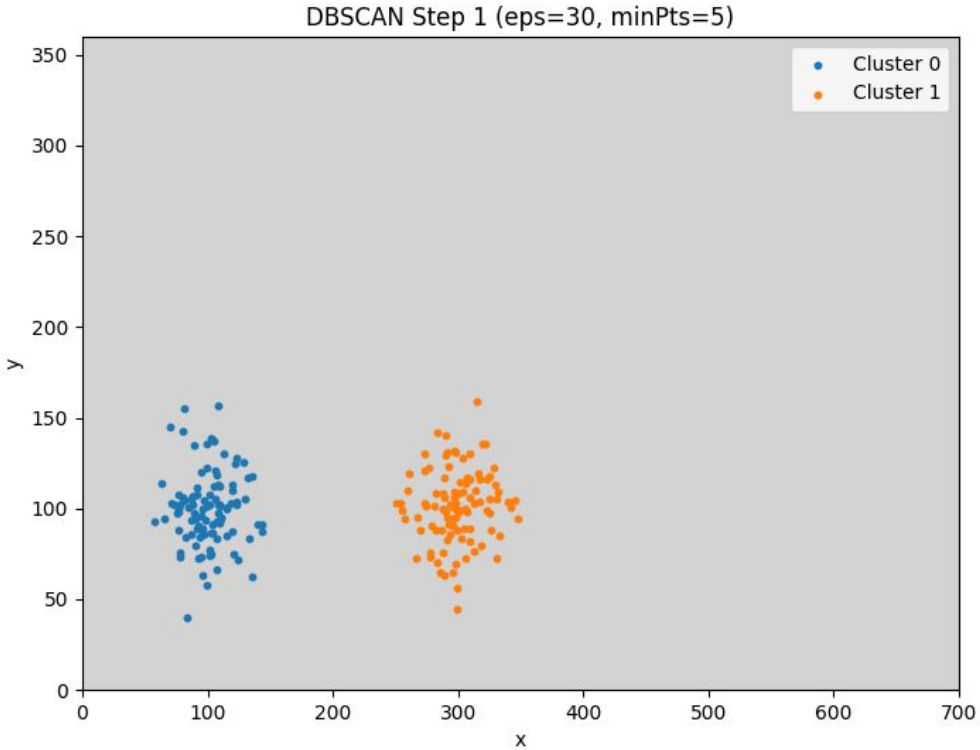
Iteration 0 First Cluster Initialized

We begin with the first unvisited point.

It has enough neighbors (≥ minPts) within the radius (eps), so it becomes a core point and initializes Cluster 0.

All density-reachable points are added to this cluster.

DBSCAN Iteration-Wise Demonstrations

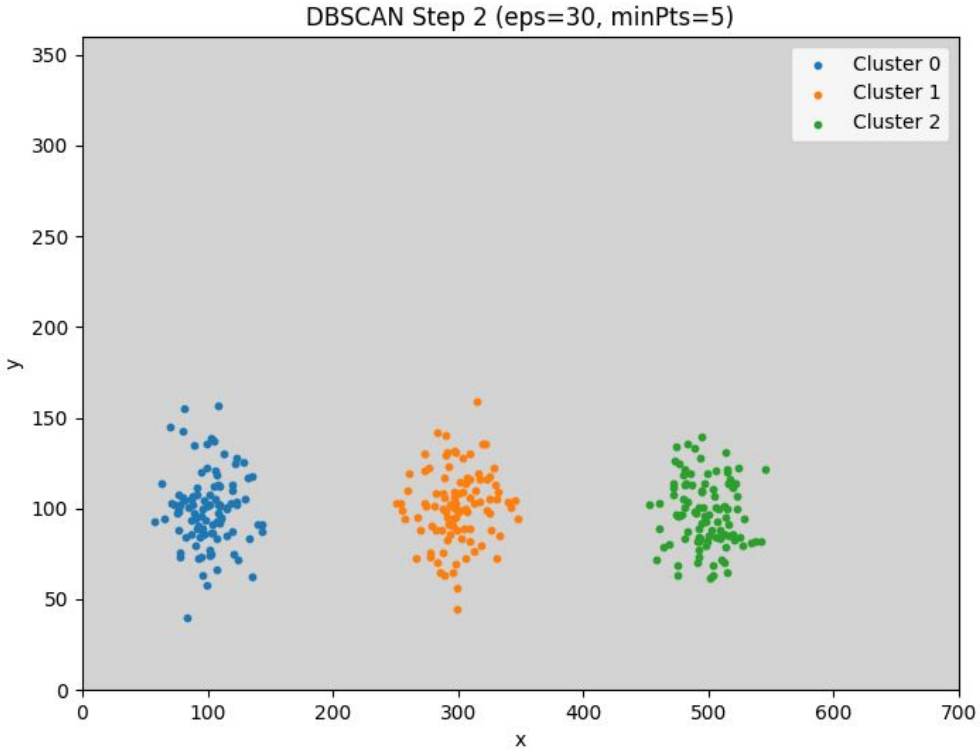


Iteration 1 Second Cluster Grows

The next unvisited point also has enough neighbors to be a core point.

A new Cluster 1 is created and grown similarly by expanding through neighboring core points.

DBSCAN Iteration-Wise Demonstrations

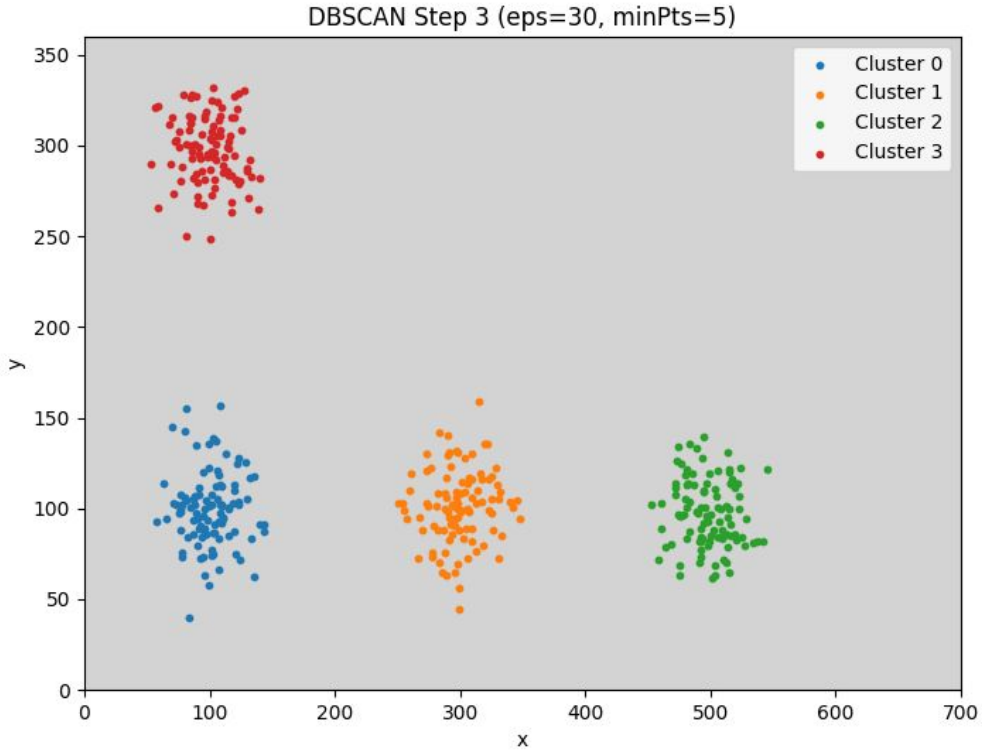


Iteration 2 Third Cluster Grows

Another dense region is identified, forming Cluster 2.

Notice how clusters can form in very different parts of the space, depending only on local density.

DBSCAN Iteration-Wise Demonstrations

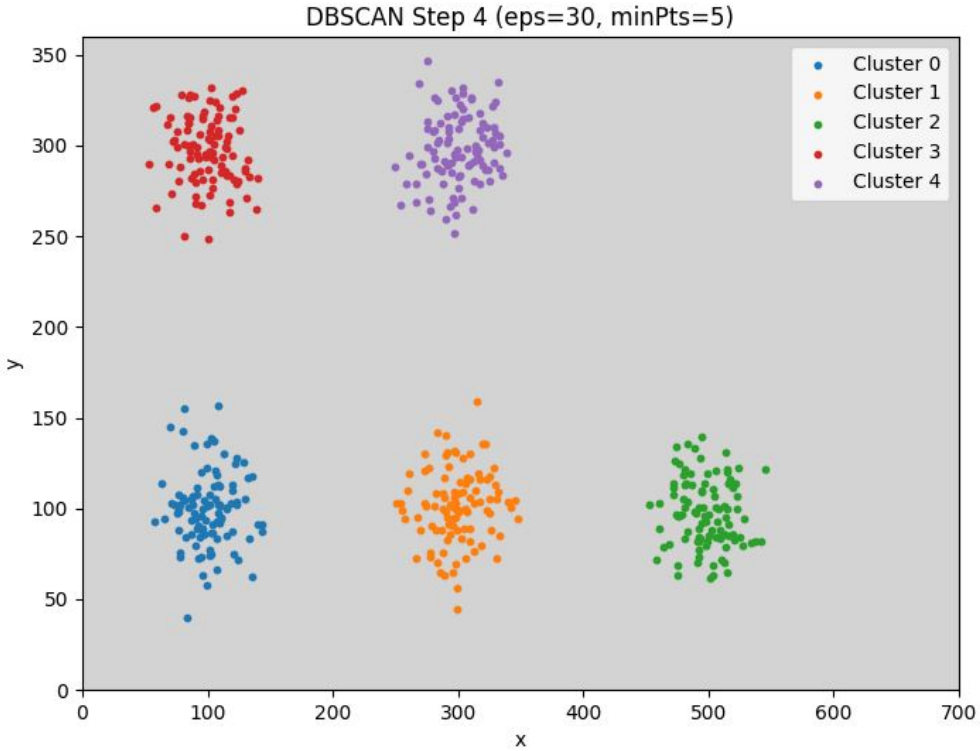


Iteration 3 Cluster 3 Above Others

This cluster is discovered in the upper region.

DBSCAN naturally separates spatially distinct groups, regardless of their position or shape.

DBSCAN Iteration-Wise Demonstrations

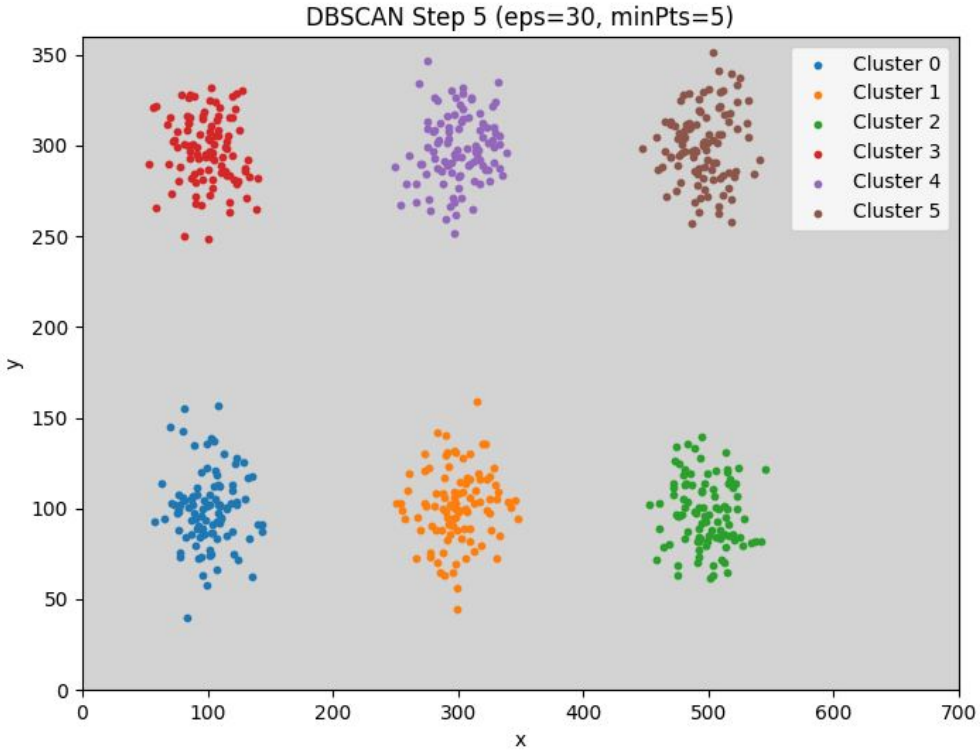


Iteration 4 Fifth Cluster Emerges

More dense regions are detected.

DBSCAN continues to scan through all unvisited points, assigning them to clusters or marking them as noise.

DBSCAN Iteration-Wise Demonstrations

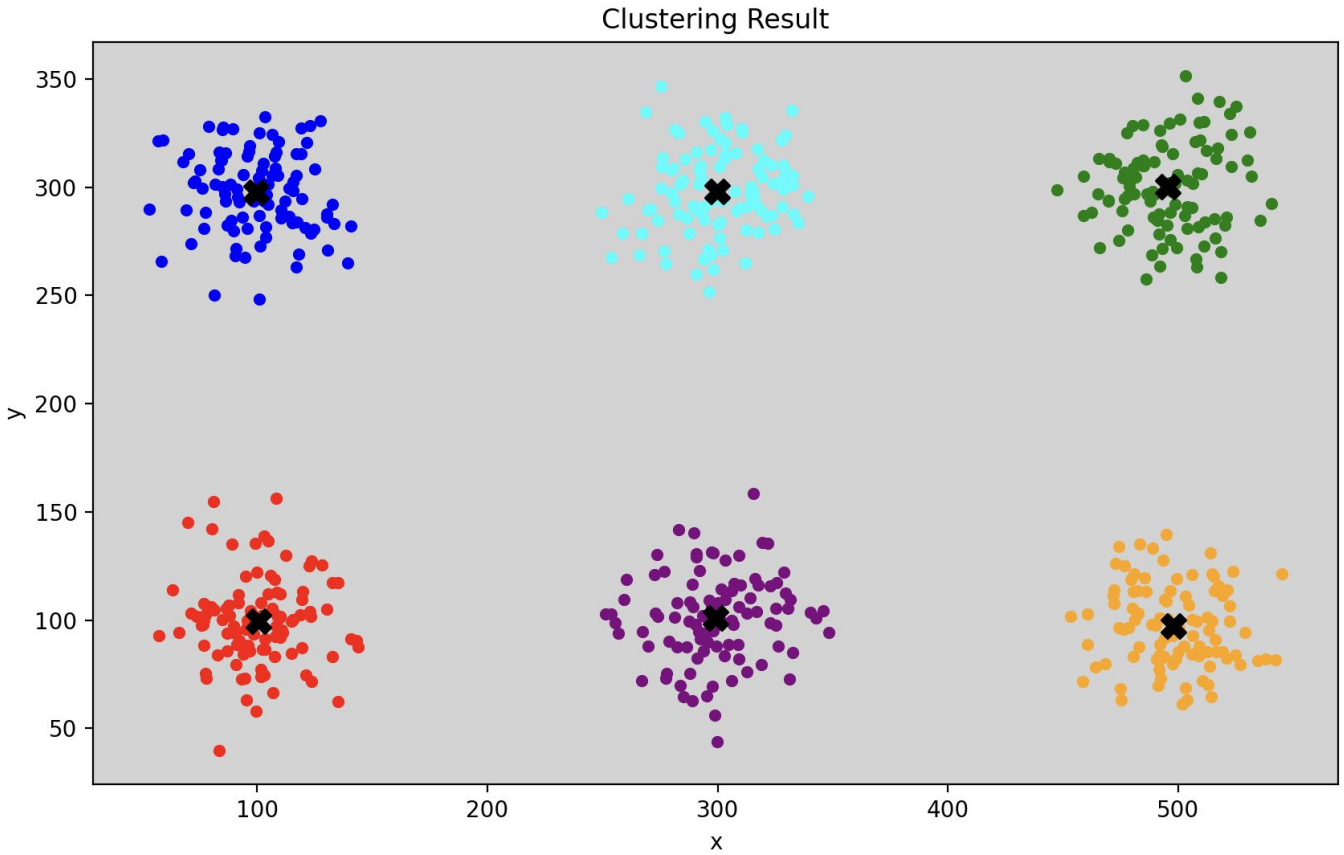


Iteration 5 Final Cluster Added

The sixth and final cluster is added.

Now, all reachable dense regions have been clustered, and any remaining unassigned points will be marked as noise (gray if shown).

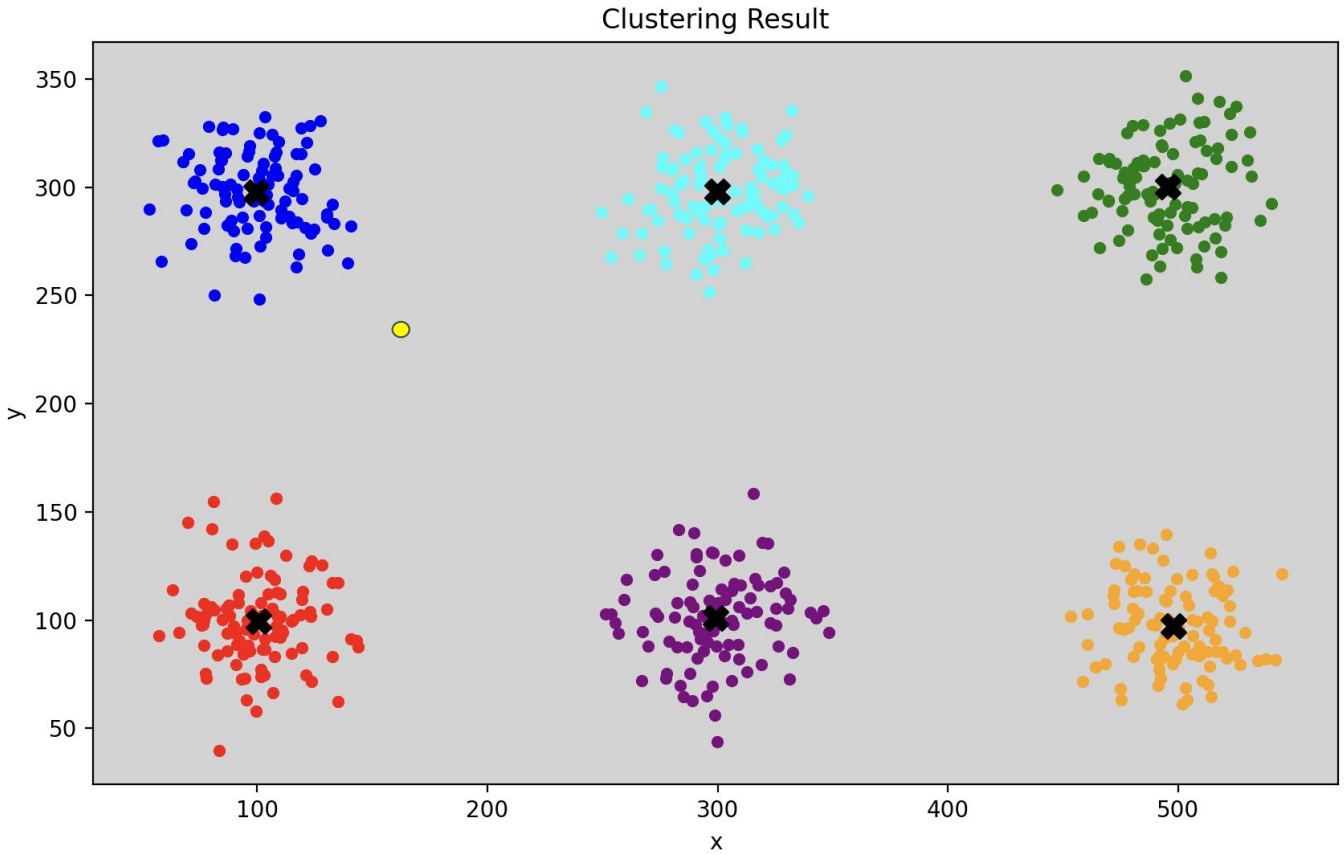
Clustering Results-After training



Find the cluster centers

- Easy to find in
  - K-means
  - Birch
- Hard to find in
  - Hierarchical Clustering
  - Density-based Clustering

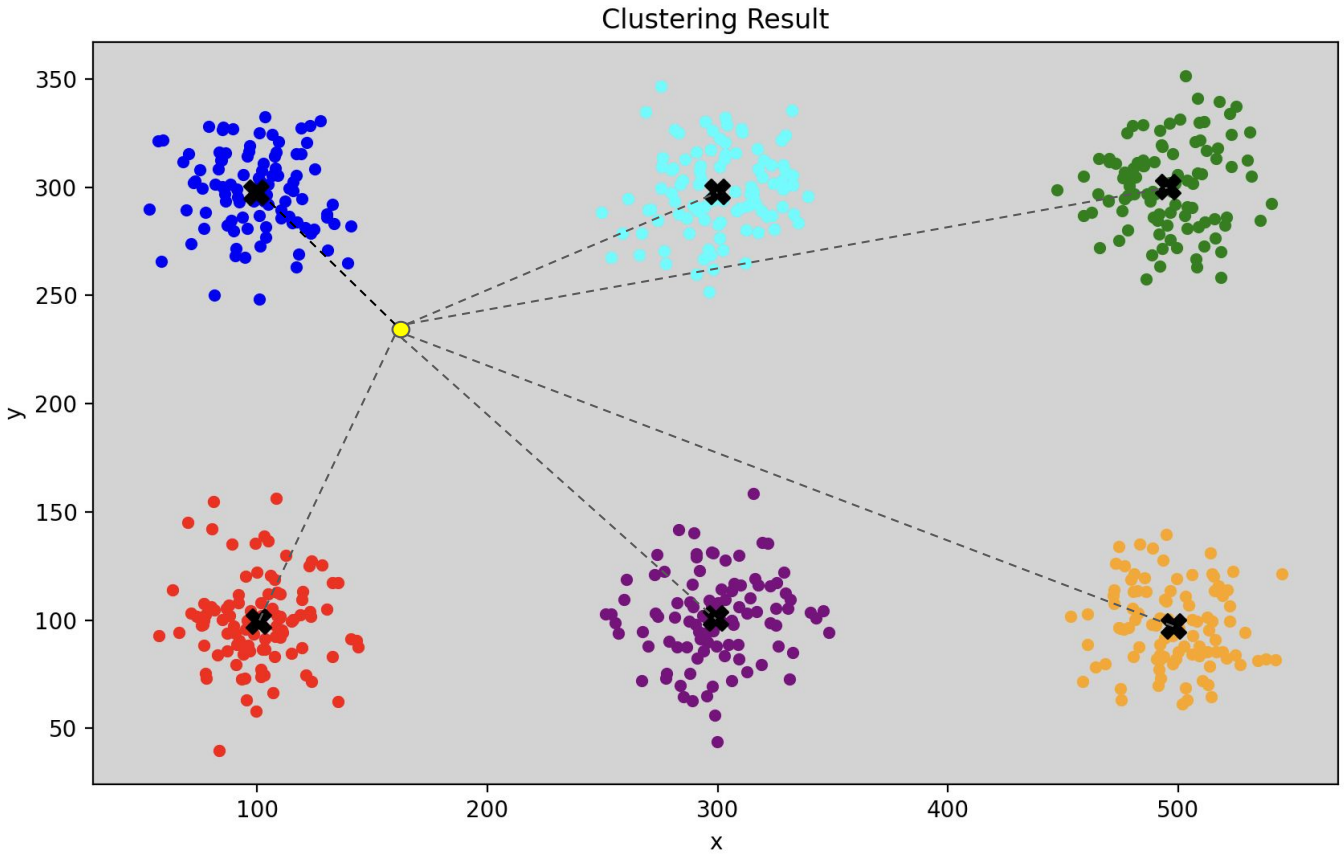
Clustering Results-Evaluation in Testset



Suppose a new datapoint

Which of the clusters should it belong to?

Clustering Results-Evaluation in Testset



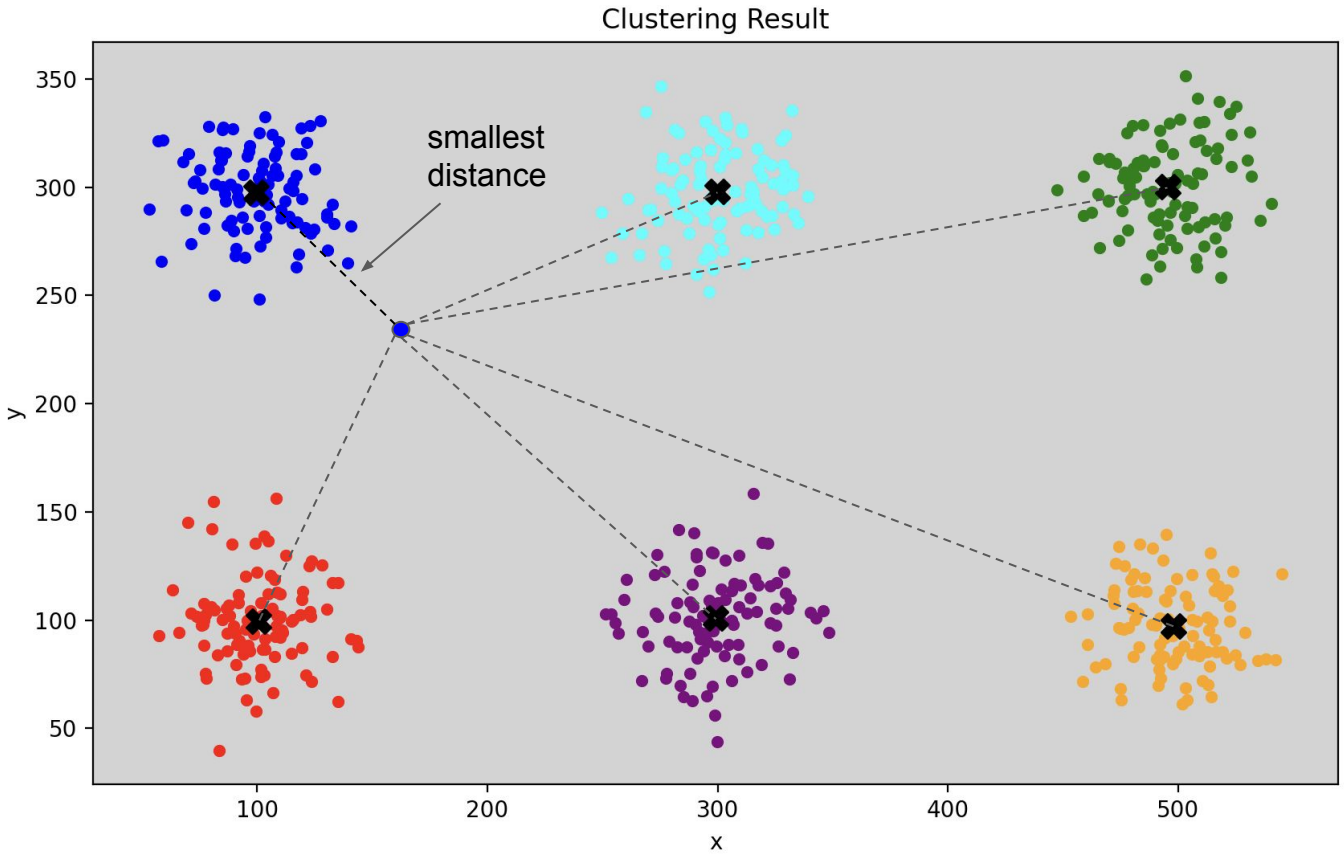
Suppose a new datapoint

Which of the clusters should it belong to?

Find the smallest distance between this point to each cluster center

Assign this point to a cluster

Clustering Results-Evaluation in Testset



Suppose a new datapoint

Which of the clusters should it belong to?

Find the smallest distance between this point to each cluster center

Assign this point to a cluster

Submission of your Assignment

Report:

- Your Code
- Your visualization of train and test datasets
- Rationale of your code

Labeled file

- A .csv file

  X_axis, Y_axis, label

  1.0123, 2.1343, 1