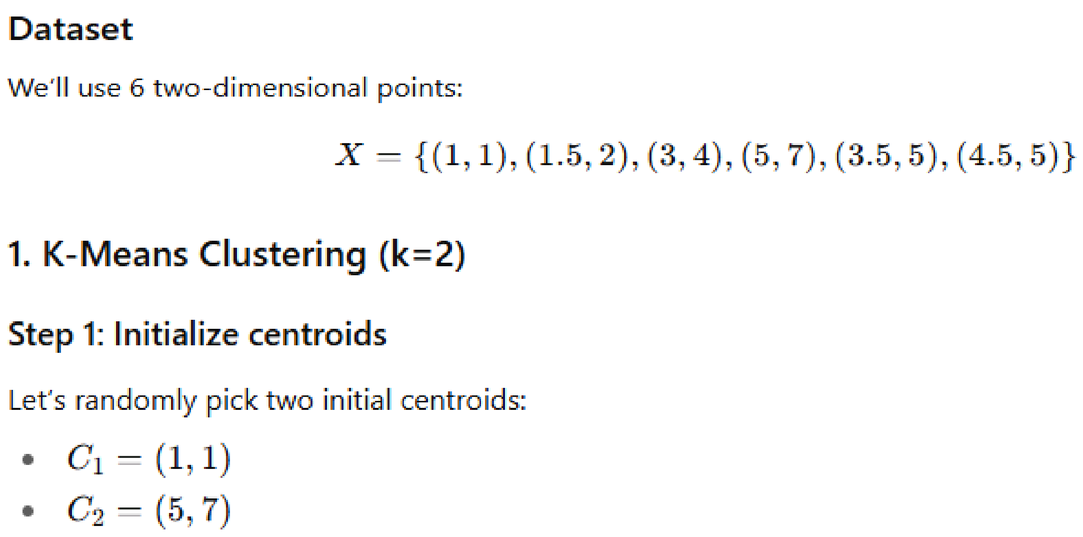
Clustering



Dataset

We'll use 6 two-dimensional points:

$$X = \{(1,1),(1.5,2),(3,4),(5,7),(3.5,5),(4.5,5)\}$$

1. K-Means Clustering (k=2)

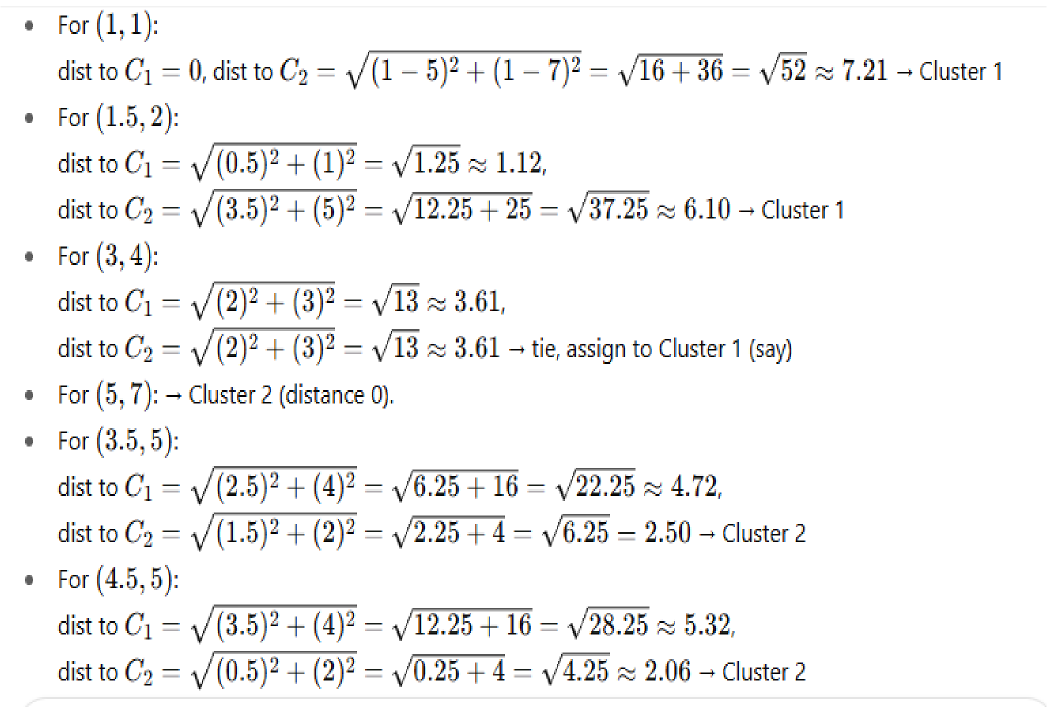
Step 1: Initialize centroids

Let's randomly pick two initial centroids:

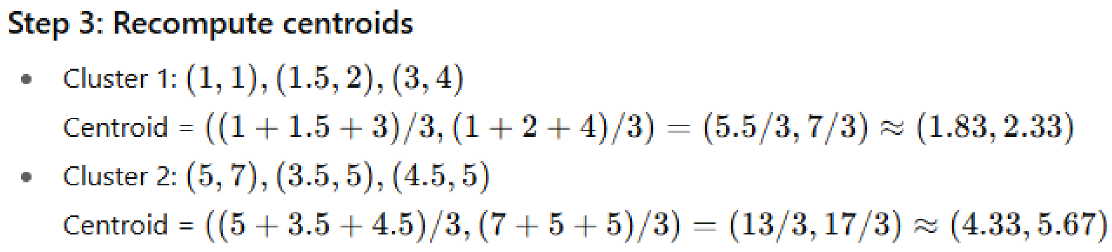
- $C_1 = (1,1)$
- $C_2 = (5,7)$

Step 2: Assign points to nearest centroid

Compute **Euclidean distances**:



- For $(1,1)$:
  dist to $C_1 = 0$, dist to $C_2 = \sqrt{(1-5)^2+(1-7)^2} = \sqrt{16+36} = \sqrt{52} \approx 7.21$ → Cluster 1
- For $(1.5,2)$:
  dist to $C_1 = \sqrt{(0.5)^2+(1)^2} = \sqrt{1.25} \approx 1.12$,
  dist to $C_2 = \sqrt{(3.5)^2+(5)^2} = \sqrt{12.25+25} = \sqrt{37.25} \approx 6.10$ → Cluster 1
- For $(3,4)$:
  dist to $C_1 = \sqrt{(2)^2+(3)^2} = \sqrt{13} \approx 3.61$,
  dist to $C_2 = \sqrt{(2)^2+(3)^2} = \sqrt{13} \approx 3.61$ → tie, assign to Cluster 1 (say)
- For $(5,7)$: → Cluster 2 (distance 0).
- For $(3.5,5)$:
  dist to $C_1 = \sqrt{(2.5)^2+(4)^2} = \sqrt{6.25+16} = \sqrt{22.25} \approx 4.72$,
  dist to $C_2 = \sqrt{(1.5)^2+(2)^2} = \sqrt{2.25+4} = \sqrt{6.25} = 2.50$ → Cluster 2
- For $(4.5,5)$:
  dist to $C_1 = \sqrt{(3.5)^2+(4)^2} = \sqrt{12.25+16} = \sqrt{28.25} \approx 5.32$,
  dist to $C_2 = \sqrt{(0.5)^2+(2)^2} = \sqrt{0.25+4} = \sqrt{4.25} \approx 2.06$ → Cluster 2



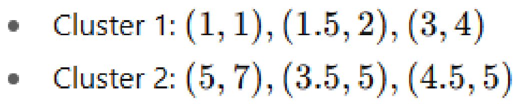
Step 3: Recompute centroids

- Cluster 1: $(1, 1), (1.5, 2), (3, 4)$
  Centroid = $((1 + 1.5 + 3)/3, (1 + 2 + 4)/3) = (5.5/3, 7/3) \approx (1.83, 2.33)$
- Cluster 2: $(5, 7), (3.5, 5), (4.5, 5)$
  Centroid = $((5 + 3.5 + 4.5)/3, (7 + 5 + 5)/3) = (13/3, 17/3) \approx (4.33, 5.67)$

Step 4: Reassign points

(If we recheck distances, assignment remains same → convergence.)

Final Clusters:



- Cluster 1: $(1, 1), (1.5, 2), (3, 4)$
- Cluster 2: $(5, 7), (3.5, 5), (4.5, 5)$

2. Hierarchical Agglomerative Clustering (HAC)

We'll use **single linkage (minimum distance)**.

Step 1: Compute pairwise distances

- dist((1,1),(1.5,2))=1.12
- dist((1,1),(3,4))=3.61
- dist((1,1),(3.5,5))=5.32
- dist((1,1),(4.5,5))=5.70
- dist((1,1),(5,7))=7.21
- dist((1.5,2),(3,4))=2.50
- dist((1.5,2),(3.5,5))=3.61
- dist((1.5,2),(4.5,5))=4.30
- dist((1.5,2),(5,7))=6.10
- dist((3,4),(3.5,5))=1.12
- dist((3,4),(4.5,5))=1.80
- dist((3,4),(5,7))=3.61
- dist((3.5,5),(4.5,5))=1.00
- dist((3.5,5),(5,7))=2.50
- dist((4.5,5),(5,7))=2.24

Step 2: Merge closest pair

- Smallest distance = 1.00 between (3.5,5) & (4.5,5).
  New cluster: { (3.5,5), (4.5,5) }.

Step 3: Next merge

- Smallest remaining = 1.12 between (1,1) & (1.5,2), and also (3,4) & (3.5,5).

Merge (1,1) & (1.5,2).
Merge (3,4) with cluster {(3.5,5),(4.5,5)}.

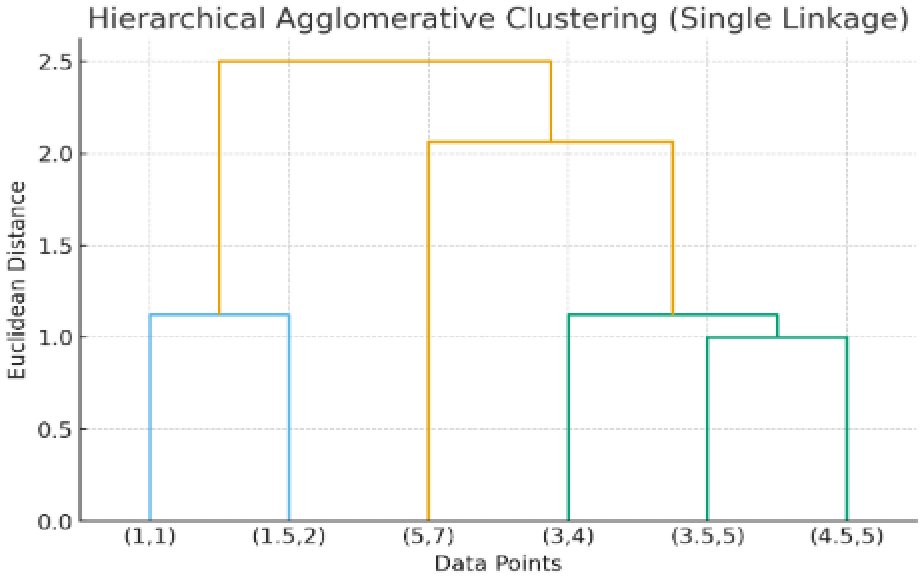
Step 4: Continue merging

Eventually, all points merge.

If we cut the dendrogram at 2 clusters:

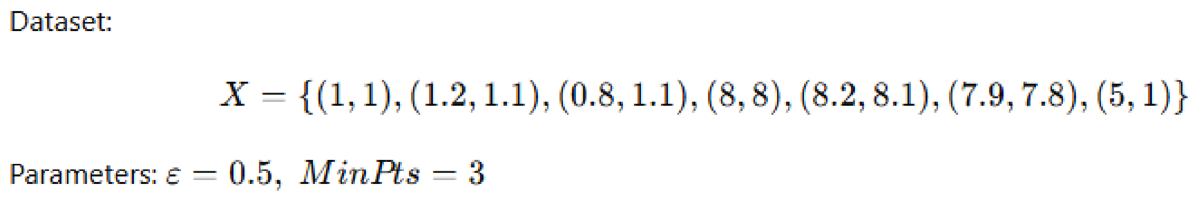
- Cluster 1: (1,1), (1.5,2), (3,4)
- Cluster 2: (3.5,5), (4.5,5), (5,7)

Observation: Both **k-means** and **HAC** gave the same final 2-cluster partition for this dataset.



Here's the dendrogram for the hierarchical agglomerative clustering. The vertical axis shows the distance at which clusters merge, and you can see that cutting the tree around distance ≈ 3 gives two clusters — the same grouping as in k-means.

3. Density based clustering (DBSCAN):

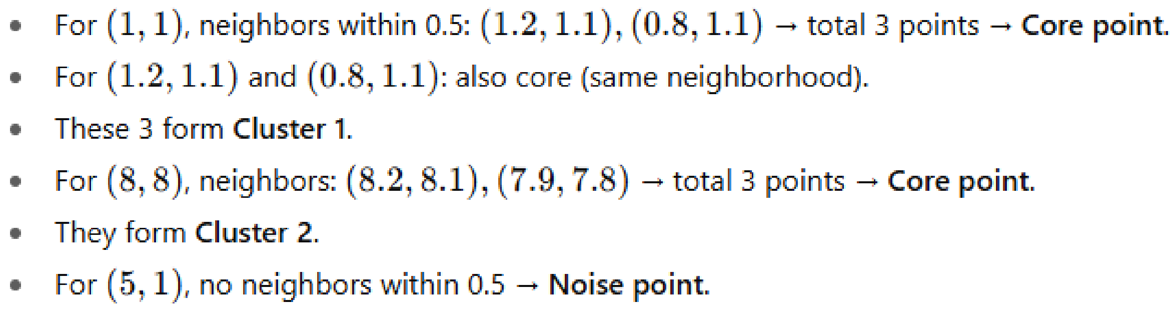


Dataset:

$$X = \{(1,1),(1.2,1.1),(0.8,1.1),(8,8),(8.2,8.1),(7.9,7.8),(5,1)\}$$

Parameters: $\varepsilon = 0.5,\ MinPts = 3$

Step 1: Compute neighborhoods



- For $(1,1)$, neighbors within 0.5: $(1.2,1.1),(0.8,1.1)$ → total 3 points → **Core point.**
- For $(1.2,1.1)$ and $(0.8,1.1)$: also core (same neighborhood).
- These 3 form **Cluster 1.**
- For $(8,8)$, neighbors: $(8.2,8.1),(7.9,7.8)$ → total 3 points → **Core point.**
- They form **Cluster 2.**
- For $(5,1)$, no neighbors within 0.5 → **Noise point.**

Result

- Cluster 1: (1,1),(1.2,1.1),(0.8,1.1)
- Cluster 2: (8,8),(8.2,8.1),(7.9,7.8)
- Noise: (5,1)