

Machine Learning

K-Means Clustering

Unsupervised Learning

Class Objectives

By the end of this class, students will be able to:

- Understand what K-Means clustering is
- Explain when to use K-Means
- Perform the K-Means algorithm step-by-step
- Calculate distances and update centroids manually
- Interpret clustering results

What Is K-Means?

K-Means is a **centroid-based clustering** algorithm that:

- Automatically groups data into **K clusters**
- Works on **unlabeled data** (unsupervised)
- Uses **distance measures** to assign points to clusters
- Iteratively updates cluster centers (centroids) until convergence

Distance Measure: Euclidean Distance

$$d(\mathbf{x}, \mathbf{a}) = \sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2}$$

When to use it

Use K-Means when:

- You want **grouping/segmentation** (customers, documents, locations, behaviors)
- Features are numeric and “distance” is meaningful
- You expect clusters to be roughly **compact** (not weird shapes)

Avoid or be careful when:

- Clusters are **non-spherical / varying density**
- Many **outliers** exist
- K is unknown and hard to pick

K-Means Algorithm

Input: Dataset **D**, Number of clusters **K**

Algorithm Steps:

1. Initialize **K** centroids (random points or K-means++)
2. **Assignment Step:** Assign each point to the nearest centroid
3. **Update Step:** Recompute each centroid as the mean of assigned points
4. Repeat steps 2–3 until centroids/assignments stop changing (convergence)

Numerical Example (K = 3)

Data Points:

Point	x	y
A1	2	10
A2	2	5
A3	8	4
B1	5	8
B2	7	5
B3	6	4
C1	1	2
C2	4	9

Initial Centroids (given):

$$C_1 = (2, 10) \quad C_2 = (5, 8) \quad C_3 = (1, 2)$$

Epoch 1 — Assignment Step

Distance table (C1=(2,10), C2=(5,8), C3=(1,2))

Point	d(C1)	d(C2)	d(C3)	Cluster
A1	0.00	3.61	8.06	1
A2	5.00	4.24	3.16	3
A3	8.49	5.00	7.28	2
B1	3.61	0.00	7.21	2
B2	7.07	3.61	6.71	2
B3	7.21	4.12	5.39	2
C1	8.06	7.21	0.00	3
C2	2.24	1.41	7.62	2

Updated Centroids:

- Cluster 1: $\{A1\} \rightarrow C_1 = (2, 10)$
- Cluster 2: $\{A3, B1, B2, B3, C2\} \rightarrow C_2 = (6, 6)$
- Cluster 3: $\{A2, C1\} \rightarrow C_3 = (1.5, 3.5)$

Epoch 2 — reassign using updated centroids

Updated Centroids:

$$C_1 = (2, 10), \quad C_2 = (6, 6), \quad C_3 = (1.5, 3.5)$$

Distance table (Epoch 2):

Point	d(C1)	d(C2)	d(C3)	New Cluster
A1	0.00	5.66	6.52	1
A2	5.00	4.12	1.58	3
A3	8.49	2.83	6.52	2
B1	3.61	2.24	5.70	2
B2	7.07	1.41	5.70	2
B3	7.21	2.00	4.53	2
C1	8.06	6.40	1.58	3
C2	2.24	3.61	6.04	1

Update Centroids Again:

$$C_1 = (3, 9.5) \quad C_2 = (6.5, 5.25) \quad C_3 = (1.5, 3.5)$$

Epoch 3 — reassign using updated centroids

Updated Centroids:

$$C_1 = (3, 9.5), \quad C_2 = (6.5, 5.25), \quad C_3 = (1.5, 3.5)$$

Distance table (Epoch 3):

Point	d(C1)	d(C2)	d(C3)	New Cluster
A1	1.12	6.54	6.52	1
A2	4.61	4.51	1.58	3
A3	7.43	1.95	6.52	2
B1	2.50	3.13	5.70	1
B2	6.02	0.56	5.70	2
B3	6.26	1.35	4.53	2
C1	7.76	6.39	1.58	3
C2	1.12	4.51	6.04	1

Updated Centroids:

$$C_1 = (3.67, 9) \quad C_2 = (7, 4.33) \quad C_3 = (1.5, 3.5)$$

Epoch 4 — reassign and check convergence

Updated Centroids:

$$C_1 = (3.67, 9), \quad C_2 = (7, 4.33), \quad C_3 = (1.5, 3.5)$$

Distance table (Epoch 4):

Point	d(C1)	d(C2)	d(C3)	New Cluster
A1	1.94	7.56	6.52	1
A2	4.33	5.04	1.58	3
A3	6.62	1.05	6.52	2
B1	1.67	4.18	5.70	1
B2	5.21	0.67	5.70	2
B3	5.52	1.05	4.53	2
C1	7.49	6.44	1.58	3
C2	0.33	5.55	6.04	1

Cluster assignments are unchanged → **Convergence reached**