K-means clustering

K-Means Clustering Explained

K-means clustering is a popular **unsupervised learning algorithm** used to group similar data points into k clusters. It works by minimizing the distance between data points and their assigned cluster centers.

Approach for K-Means Clustering (Step-by-Step Explanation)

K-Means clustering is an **iterative algorithm** used to group data points into kk clusters by minimizing the distance between points and cluster centroids.

Step 1: Select Initial Cluster Centroids

- Choose k random data points from the dataset.
- These points serve as the **initial cluster centroids** $\mu_1^{(1)}, \mu_2^{(1)}, \ldots, \mu_k^{(1)}$.
- The choice of centroids affects the final clustering. Using **K-Means++** improves centroid selection.

Step 2: Assign Each Data Point to the Nearest Centroid

- Compute the **Euclidean distance** between each data point xpx_p and all centroids $\mu_j^{(t)}$:

  $C_i^{(t)} = \left\{ x_p : \|x_p - \mu_i^{(t)}\|^2 \leq \|x_p - \mu_j^{(t)}\|^2, \quad \forall j, 1 \leq j \leq k \right\}$
- Assign each data point to the **closest centroid** based on distance.
- This step creates **k clusters**, each containing a group of points.

Step 3: Update Cluster Centroids

- Compute the **mean** of all points assigned to each cluster to find the new centroid:

  $\mu_i^{(t+1)} = \frac{1}{|C_i^{(t)}|} \sum_{x_j \in C_i^{(t)}} x_j$
- This shifts the centroid towards the actual center of the cluster.
- If a centroid remains unchanged, the algorithm starts converging.

Step 4: Repeat Until Convergence

- Repeat **Steps 2 and 3** until the centroids **no longer change significantly**.
- This ensures clusters **stabilize** and data points are correctly grouped.

Key Notes

✅ **K-Means minimizes intra-cluster distance** (variance).

✅ **Sensitive to centroid initialization** → Different starting points can lead to different results.

✅ **Works well for spherical clusters but struggles with complex shapes.**

Example of K-Means in Action

Iteration	Action
1️⃣	Randomly place k centroids
2️⃣	Assign points to the nearest centroid
3️⃣	Compute new centroid positions
4️⃣	Repeat until centroids stop moving

📌 **End Result:** k clusters with optimized centroids.

How to Choose the Best k?

Since **k is user-defined**, we use methods like:

✅ **Elbow Method** → Plot inertia (SSE) vs. k, pick the "elbow" point.

✅ **Silhouette Score** → Measures cluster separation and cohesion.

✅ **Gap Statistic** → Compares clustering performance with random data.

Advantages of K-Means

✅ **Simple & Fast** → Works well for large datasets.

✅ **Scalable** → Efficient for high-dimensional data.

✅ **Works Well for Well-Separated Clusters** → If clusters are spherical and distinct.

Limitations of K-Means

❌ **Requires Predefined kk** → Wrong k leads to poor clustering.

❌ **Sensitive to Outliers** → Outliers can shift centroids.

❌ **Assumes Spherical Clusters** → Doesn't work well for irregular shapes.

When to Use K-Means?

✔️ Customer segmentation

✔️ Image compression

✔️ Document classification

✔️ Market analysis
