K-Means Clustering - Full Explanation

1. What is K-Means?

- Unsupervised ML algorithm for clustering.
- Groups data into K clusters by minimizing distance to centroids.

2. Mathematical Intuition

Objective Function:

J = Σ || x_i - μ_C(i) ||^2

Where:

- x_i = data point
- μ_C(i) = centroid of assigned cluster

Steps:

- Initialize K centroids randomly.
- Assignment step: assign each point to nearest centroid.
- Update step: recompute centroid as mean of cluster points.
- Repeat until centroids stabilize.

3. Python Code (From Scratch)

- Implemented a custom KMeansScratch class using numpy.
- Includes fit() and predict() methods.

4. Scikit-learn KMeans

- Much simpler using sklearn.cluster.KMeans
- Key attributes: labels_, cluster_centers_, inertia_, n_iter_.

5. Evaluating Clustering

- Unlike supervised ML, no accuracy score.
- Metrics used:

* Inertia (WCSS)
* Silhouette Score
* Adjusted Rand Index (ARI)
* Normalized Mutual Information (NMI)
* Hungarian Algorithm for matching labels.

6. Applications of Clustering

- Customer segmentation (marketing)
- Fraud detection (finance)
- Patient grouping (healthcare)
- Image compression (pixels → clusters of colors)
- Document clustering (text mining)

- Anomaly detection (IoT, fraud)

7. Mixed Data Types in Clustering

- Numerical → scale (StandardScaler/MinMaxScaler)
- Categorical → one-hot encoding or K-Prototypes
- Binary → direct but scale
- Text → TF-IDF, embeddings
- Date/Time → extract features (day, month, duration)

8. Convergence of K-Means

- Cost function J decreases at each step.
- Assignment step → points go to nearest centroid (cannot increase J).
- Update step → centroids = mean of cluster points (minimizes J).
- Always converges in finite steps but may reach local minimum.
- sklearn uses n_init restarts to improve results.

Key Takeaway:

- KMeans is unsupervised → used when no labels exist.
- Useful for pattern discovery, segmentation, anomaly detection.
- Requires careful preprocessing when features are mixed types.