Lecture Notes: K-Means Clustering

Luu Minh Sao Khue

1 What is K-Means Clustering?

1.1 The clustering problem

In supervised learning, training data consist of labeled input–output pairs (x_i, y_i), and the goal is to learn a function that predicts y from x. In many real-world scenarios, however, labeled data are not available. Instead, we are given only a collection of observations $\{x_i\}_{i=1}^n$ and wish to discover whether the data exhibit internal structure.

This motivates **clustering**, whose objective is to:

- group similar observations together,
- separate dissimilar observations into different groups,
- uncover structure useful for interpretation or downstream tasks.

Because no ground-truth labels are provided, clustering does not have a unique correct solution. Any clustering method must therefore rely on assumptions about what constitutes a meaningful grouping. K-Means adopts a simple and widely applicable assumption: each cluster can be represented by a single point, called a *centroid*, and observations assigned to the same cluster should be close to that centroid.

1.2 High-level intuition

K-Means represents a dataset using K centroids, where K is specified by the user. Each centroid corresponds to one cluster. The algorithm iteratively alternates between two operations:

1. assigning each data point to the nearest centroid,
2. updating each centroid as the mean of the points assigned to it.

The term *K-Means* reflects this use of the arithmetic mean in the centroid update step.

2 Notation and Data Representation

Let:

- n denote the number of data points,
- d denote the dimensionality of each data point,

- K denote the number of clusters.

Each data point is represented as a vector

$$x_i \in \mathbb{R}^d,$$

with components

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{id}).$$

The algorithm maintains K centroids

$$\mu_1, \mu_2, \ldots, \mu_K, \quad \mu_k \in \mathbb{R}^d.$$

Cluster membership is described using either of the following equivalent representations.

Label representation. Each data point x_i is assigned a cluster label

$$z_i \in \{1, 2, \ldots, K\},$$

where $z_i = k$ indicates that x_i belongs to cluster k.

Set representation. For each cluster k, define the index set

$$C_k = \{i \in \{1, \ldots, n\} : z_i = k\}.$$

The number of points in cluster k is denoted by $|C_k|$.

3 Distance Measure

3.1 Euclidean distance

To quantify similarity between points, K-Means uses Euclidean distance. For two vectors $x, \mu \in \mathbb{R}^d$, the Euclidean distance is defined as

$$\|x - \mu\|_2 = \sqrt{\sum_{j=1}^{d} (x_j - \mu_j)^2}.$$

3.2 Squared Euclidean distance

In practice, K-Means uses the squared Euclidean distance

$$\|x - \mu\|_2^2 = \sum_{j=1}^{d} (x_j - \mu_j)^2.$$

Using squared distance avoids the square root, simplifies computation, and leads to a closed-form solution for centroid updates. All distance comparisons and objective values in K-Means are based on squared Euclidean distance.

4 Objective Function

K-Means is formulated as an optimization problem that seeks to minimize the total within-cluster variance.

4.1 Within-cluster sum of squares

The objective function, also referred to as the **within-cluster sum of squares (WCSS)** or **inertia**, is defined as

$$J = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|_2^2.$$

The outer sum aggregates the contribution from all clusters, while the inner sum accumulates the squared distances of points within cluster k from its centroid μ_k.

4.2 Equivalent formulation using labels

Using the assignment variables z_i, the same objective can be written as

$$J = \sum_{i=1}^{n} \|x_i - \mu_{z_i}\|_2^2.$$

This form emphasizes that each data point contributes exactly one distance term, corresponding to the centroid of its assigned cluster.

5 K-Means Algorithm

5.1 Alternating minimization

The objective function depends on both discrete variables $\{z_i\}$ and continuous variables $\{\mu_k\}$. Optimizing over both simultaneously is intractable. K-Means applies an **alternating minimization** strategy:

1. fix centroids and update assignments,
2. fix assignments and update centroids.

Each step does not increase the objective value J.

5.2 Assignment step

Given centroids $\mu_1, \ldots, \mu_K$, each point is assigned to the nearest centroid:

$$z_i \leftarrow \arg\min_{k \in \{1,\ldots,K\}} \|x_i - \mu_k\|_2^2, \quad i = 1, \ldots, n.$$

This assignment minimizes the contribution of x_i to the objective.

5.3 Update step

Given fixed assignments, the centroid of cluster k is updated as

$$\mu_k \leftarrow \frac{1}{|C_k|} \sum_{i \in C_k} x_i, \quad k = 1, \ldots, K.$$

This update minimizes the objective with respect to μ_k when assignments are fixed.

5.4 Algorithm summary

Input: data points $\{x_i\}_{i=1}^n$, number of clusters K.

1. Initialize centroids $\mu_1, \ldots, \mu_K$.
2. Repeat until convergence:
   (a) assign each point to its nearest centroid,
   (b) update each centroid as the mean of its assigned points.
3. Output the final assignments $\{z_i\}$ and centroids $\{\mu_k\}$.

5.5 Stopping criteria and convergence

The algorithm may terminate when assignments no longer change, when the maximum centroid displacement is below a threshold ε, or when a maximum number of iterations is reached.

Because the objective function J is non-increasing and bounded below by zero, K-Means always converges. The solution is generally a local minimum.

6 Evaluation of K-Means Clustering

6.1 Inertia (WCSS)

The inertia value is the final objective value

$$\text{WCSS} = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|_2^2.$$

Smaller values indicate more compact clusters, but inertia always decreases as K increases and therefore cannot be used alone to select K.

6.2 Elbow method

The elbow method evaluates inertia as a function of K. As K increases, inertia decreases rapidly at first and then more slowly. The value of K at which the rate of decrease changes significantly is chosen as a compromise between compactness and model complexity.

6.3 Silhouette score

For each point x_i, define:

- a_i: the average distance from x_i to all other points in its own cluster,
- b_i: the minimum average distance from x_i to points in any other cluster.

The silhouette score for x_i is defined as

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}.$$

The overall silhouette score is the average of s_i over all points. Values close to 1 indicate well-separated clusters, values near 0 indicate overlapping clusters, and negative values suggest incorrect assignments.

7 Computational Complexity

Let T denote the number of iterations. Each iteration computes distances between n points and K centroids in d dimensions, giving a computational cost of

$$O(nKd)$$

per iteration and

$$O(TnKd)$$

overall.

8 Assumptions and Limitations

K-Means performs best when clusters are approximately spherical, similarly sized, and separable under Euclidean distance. It is sensitive to outliers, feature scaling, and non-Euclidean structure.

9 Summary

K-Means clustering partitions data into K groups by minimizing the within-cluster sum of squared distances. The algorithm alternates between assignment and update steps, converges monotonically, and is efficient in practice. Its simplicity makes it widely used, while its assumptions define both its applicability and its limitations.