

Lecture 5: Introduction to Clustering & k-Means

ECE 2410 – Introduction to Machine Learning

Farzad Farnoud

University of Virginia

Spring 2026

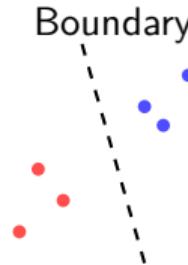
Outline

- 1 Unsupervised Learning
- 2 k-Means Intuition
- 3 The k-Means Algorithm
- 4 Visual Walkthrough
- 5 Summary

Supervised vs. Unsupervised Learning

Supervised Learning (L1-L4)

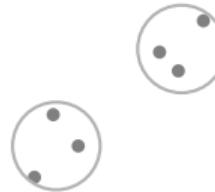
- **Data:** Pairs (x, y)
- **Goal:** Predict y given x
- **Feedback:** Correct labels provided



Unsupervised Learning (L5-L7)

- **Data:** Just x ! No labels.
- **Goal:** Find *structure* or patterns
- **Feedback:** None (exploratory)

Grouping?



Why Clustering?

Definition

Clustering: Grouping data points so that items in the same group are more similar to each other than to those in other groups.

Real-World Applications:

-  **Customer Segmentation:** Group customers by purchasing behavior (e.g., "Big Spenders", "Bargain Hunters")
-  **Image Compression:** Reduce millions of colors to a small palette
-  **Genetics:** Cluster genes with similar expression patterns
-  **Document Analysis:** Organize articles by topic without reading them

Today's Algorithm: k-Means

We'll focus on one of the simplest and most widely used clustering algorithms: **k-Means**.

Why k-Means?

- Simple to understand and implement
- Fast (scales to large datasets)
- Often works surprisingly well!

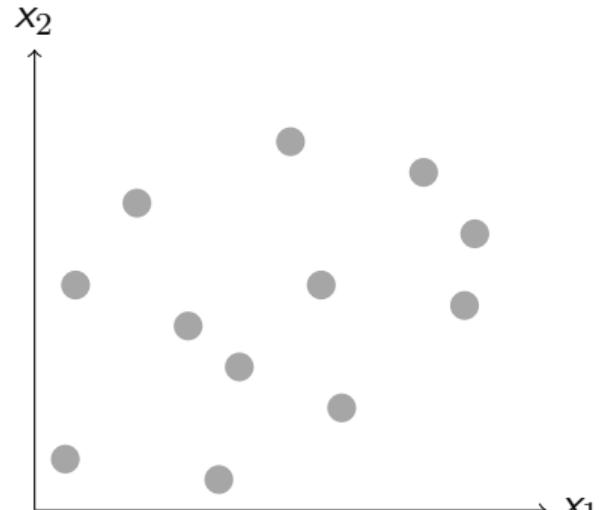
The Clustering Problem

Input:

- A dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ where each $\mathbf{x}_i \in \mathbb{R}^D$
- A number of clusters K (chosen by us)

Output:

- K cluster **centers** (centroids): μ_1, \dots, μ_K
- Cluster **assignments** for each point: c_1, \dots, c_N where $c_i \in \{1, \dots, K\}$



$N = 10$ data points

How Do We Measure a “Good” Clustering?



Key Question

We need a **number** to quantify how good a clustering is.
But how?

Think of it Like This...

The Post Office Problem:

- You have N houses (data points).
- You want to build K post offices (centers).
- Each house is served by its nearest post office.
- The cost of traveling distance d is d^2 .
- **Goal:** Minimize total cost!

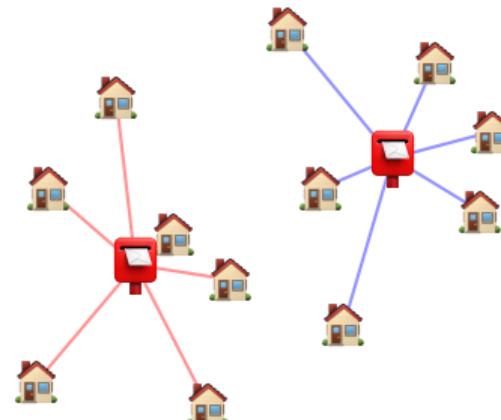


The Objective Function

We want to minimize the **total squared distance** from each point to its assigned center:

$$J = \sum_{i=1}^N \|x_i - \mu_{c_i}\|^2$$

- x_i : Data point i
- c_i : Cluster assignment of point i
- μ_{c_i} : Center of the cluster assigned to point i



Key Insight

This is an **optimization problem**: find assignments $\{c_i\}$ and centers $\{\mu_k\}$ that minimize J .

The k-Means Algorithm

An iterative algorithm that alternates between two steps:

- ① **Initialization:** Pick K random points as initial centers μ_1, \dots, μ_K .
- ② **Repeat until convergence:**
 - **Step A (Assignment):** Assign each point x_i to the *nearest* center. (**If equal, keep last assignment**)
$$c_i = \arg \min_k \|x_i - \mu_k\|^2$$
 - **Step B (Update):** Move each center μ_k to the *mean* of its assigned points.

$$\mu_k = \frac{1}{|C_k|} \sum_{i \in C_k} x_i$$

Convergence: Stop when assignments don't change.

Why Does This Work?

Step A (Assignment):

- Given fixed centers μ , the best assignment for each point is obviously its *nearest* center.
- If any assignment changes, J decreases.

Step B (Update):

- Given fixed assignments, the best center for a cluster is its *mean* (minimizes squared error).
- If any center changes, J decreases.

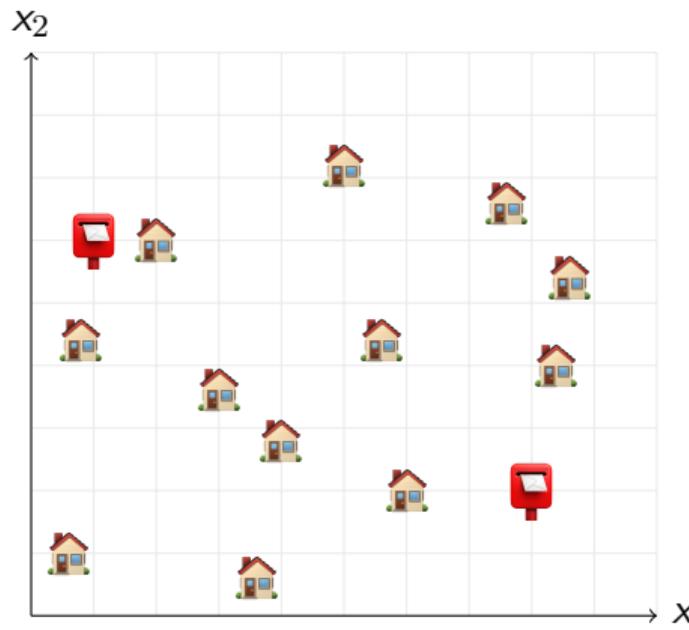
Stopping Criterion

If a step doesn't change assignments or centers, we stop.

Guarantee

J decreases at every step \Rightarrow **Algorithm always converges!**

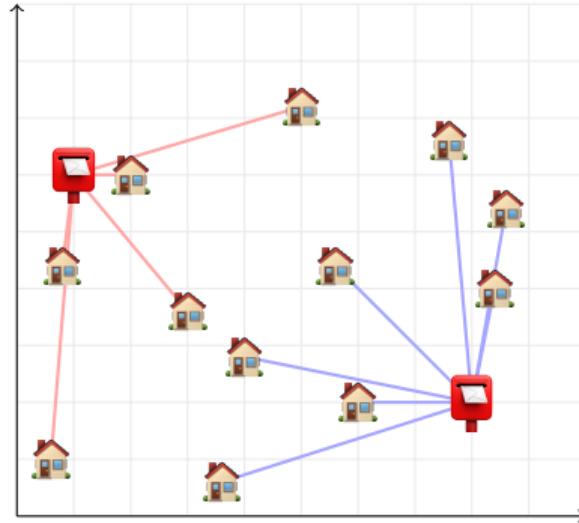
Step 0: The Data + Random Initialization



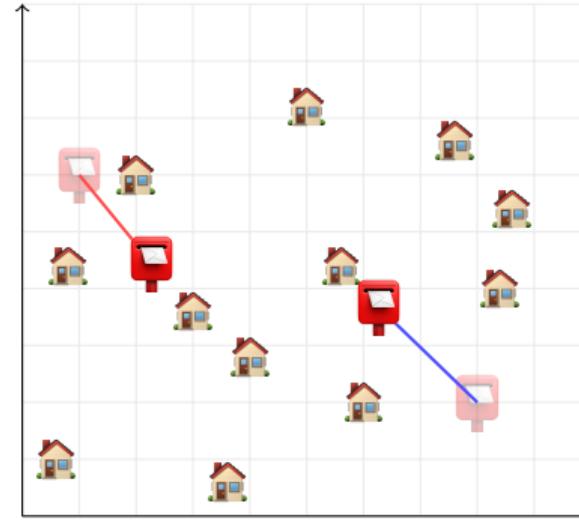
$N = 12$ houses, $K = 2$ post offices (random placement)

Iteration 1: Assign & Update

Assignment

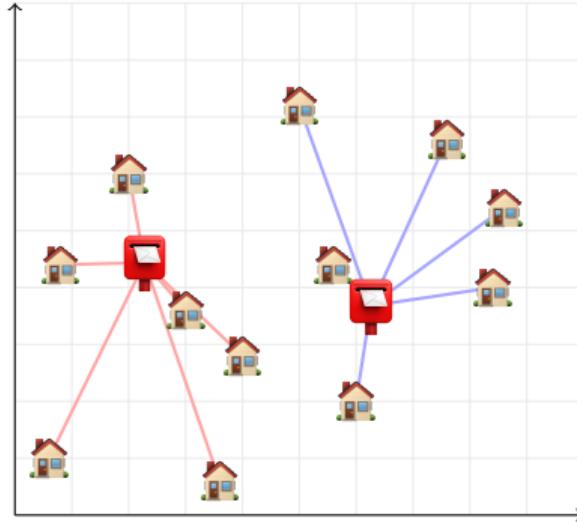


Update Centers

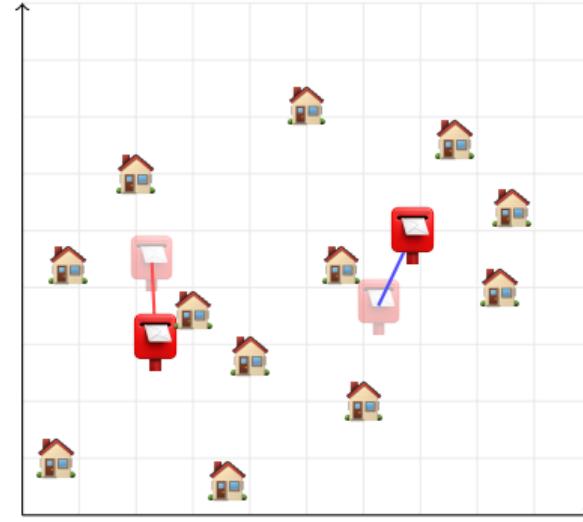


Iteration 2: Assign & Update

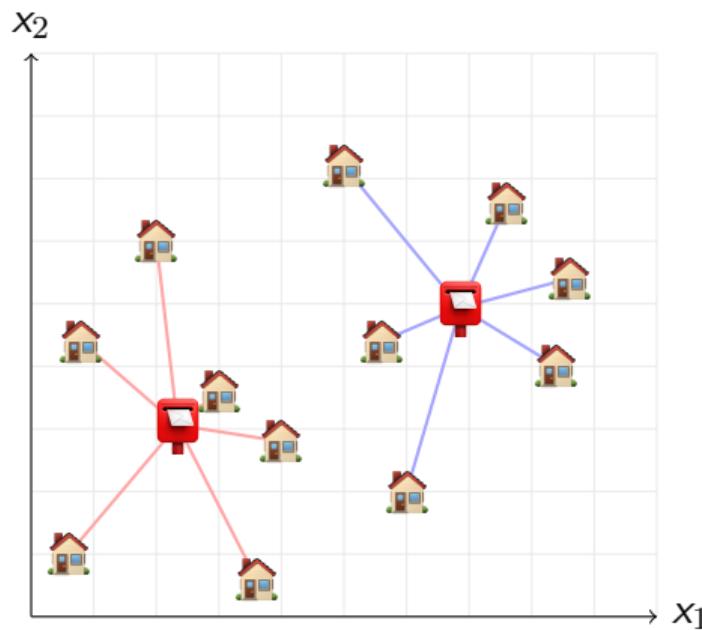
Re-Assignment



Update Centers



Iteration 3: Converged!



Converged! Assignments unchanged from Iteration 2.

Summary

Key Concepts

- ① **Unsupervised Learning:** Finding structure without labels.
- ② **k-Means Algorithm:**
 - **Initialize** K centers randomly.
 - **Assign** each point to its nearest center.
 - **Update** centers to the mean of their points.
 - **Repeat** until convergence.
- ③ **Simple but Powerful:** Used everywhere from image compression to genetics.

Questions for Next Time

- Does it always find the *best* clustering? 🤔
- How do we choose K ?
- What are its limitations?