k-means

Rebecca C. Steorts

K-means clustering

Assume observations $(x_1, \ldots, x_n)$, where each $x_i \in \mathbb{R}^d$.

Goal

Partition n observations into K sets ($K \leq n$), $S = \{S_1, \ldots, S_k\}$ such that the sets minimize the within-cluster sum of squares

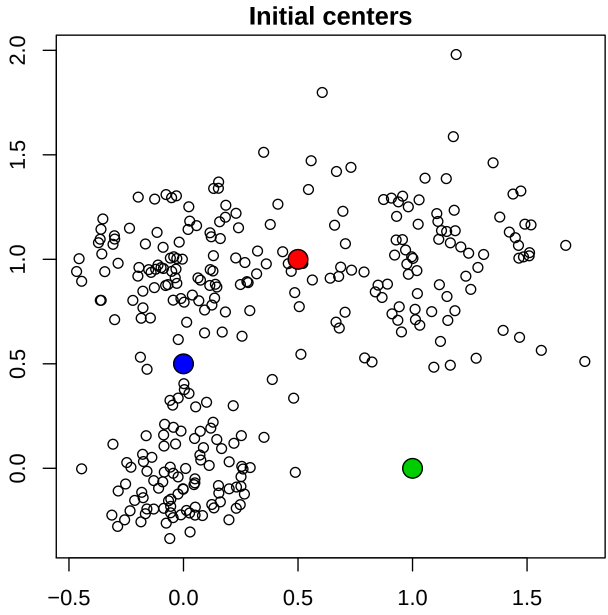
$$\text{argmin}_S \sum_{i=1}^{K} \sum_{x_j \in S_i} (x_j - \mu_i)^2, \tag{1}$$

where μ_i is the mean of the points in S_i.

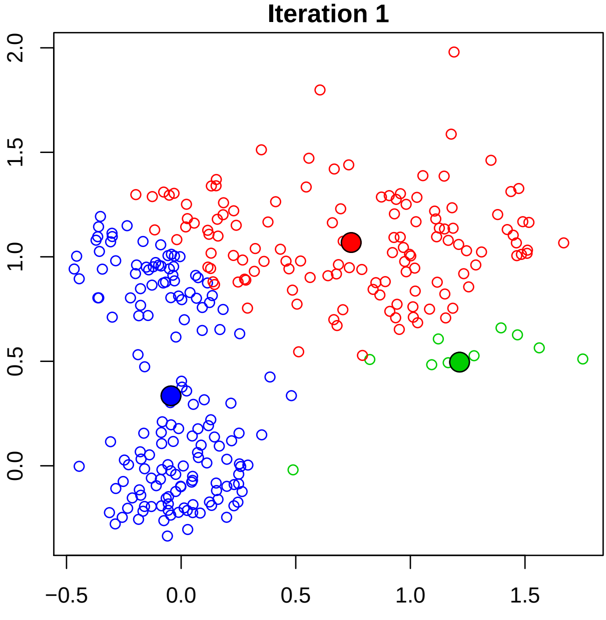
Example

Here $X_i \in \mathbb{R}^2$, $n = 300$, and $K = 3$

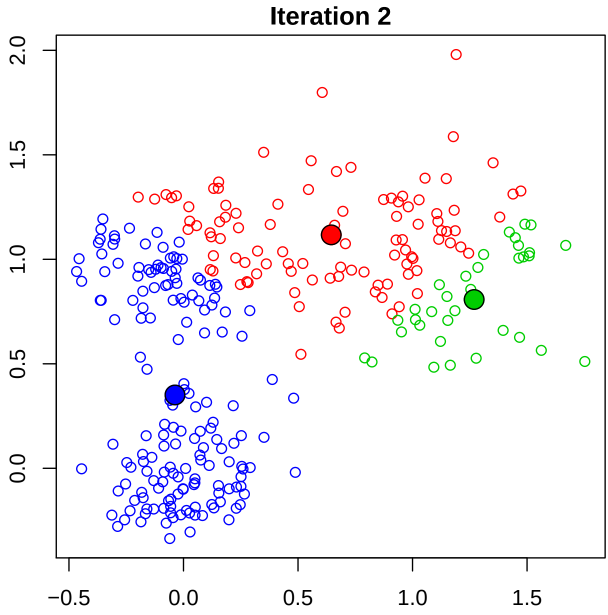
Example



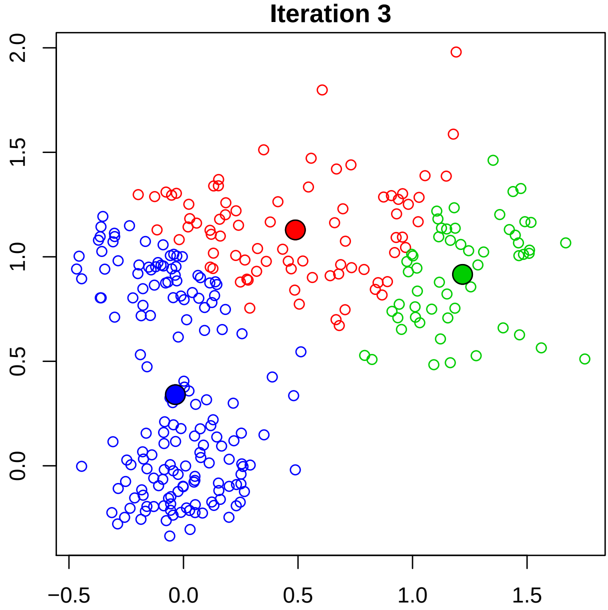
Example



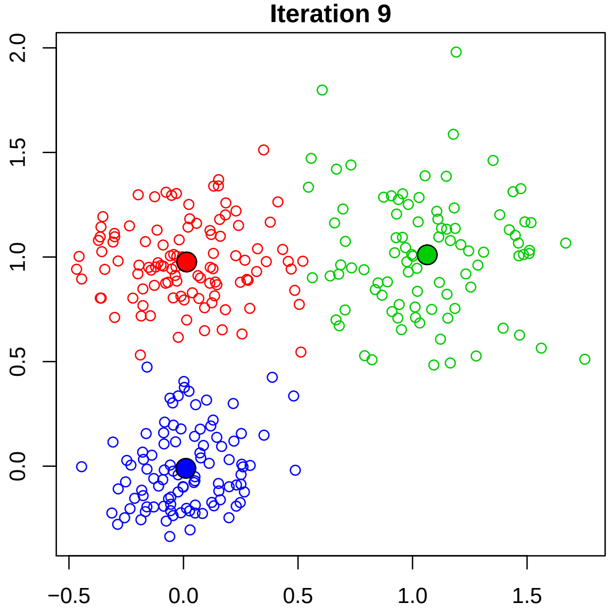
Example



Example



Example



Algorithm

- Input: data and number of clusters (K)
- Initialize the K cluster centers (can be random if needed)

Iterate

1. Assignment: Decide the class membership of the n data points by assigning them to the nearest cluster centers
2. Update: Re-estimate the K cluster centers (mean or centroid) by assuming the memberships found in step one are correct.

Terminate. If none of the data points changed membership in the last iteration, exit. Otherwise, go back to step one.

Exercise

Can you prove or explain why the algorithm is guaranteed to terminate?

Solution

Recall our function we want to minimize:

$$\operatorname{argmin}_S \sum_{i=1}^{K} \sum_{x_j \in S_i} (x_j - \mu_i)^2, \tag{2}$$

The function above is always non-negative.

1. Assignment: data points are assigned to the nearest centroid, which either keeps the function the same or decreases it.
2. Update: The centroids are recalculated as the mean of assigned points, which also results in the function either remaining the same or decreases it.

Because we a non-increasing function that is bounded below (it cannot go below 0), it must eventually reach a point where it cannot decrease any further. This means that the assignments of points to clusters will no longer change after a finite number of iterations.

Seed choice

- Some seeds can result in poor convergence or a sub-optimal clustering.
- K-means is known to easily get stuck in a local minima.
- Important to look at multiple starting points.
- Recommended to initialize with the results of another method.

k-means, more formally

0. Randomly initialize the K centers

$$\mu^0 = (\mu_1^0, \ldots, \mu_K^0)$$

1. Classify. At iteration t, assign each point ($j \in \{1, \ldots, n\}$) to the nearest center.

$$C^t(j) \leftarrow \text{argmin}_i (\mu_i^t - x_j)^2$$

2. Re-center. Now, μ_i is the centroid of the new sets.

$$\mu_i^{(t+1)} \leftarrow \text{argmin}_\mu \sum_{j:C^t(j)=i} (\mu - x_j)^2$$

What is k-means optimizing?

Define the following function F of centers μ and point allocation C:

$$\mu = (\mu_1, \ldots, \mu_K) \tag{3}$$

$$C = (C(1), \ldots, C(n)) \tag{4}$$

$$F(\mu, C) = \sum_{j=1}^{n} (\mu_{C(j)} - x_j)^2 \tag{5}$$

$$= \sum_{i=1}^{K} \sum_{j:C(j)=i} (\mu_i - x_j)^2 \tag{6}$$

Optimal solution of k-means is the $\min_{\mu,C} F(\mu, C)$.

k-means algorithm

$$\min_{\mu, C} F(\mu, C) = \min_{\mu, C} \sum_{j=1}^{n} (\mu_{C(j)} - x_j)^2 = \min_{\mu, C} \sum_{i=1}^{K} \sum_{j: C(j)=i} (\mu_i - x_j)^2.$$

1. Fix μ, Optimize C.

$$\min_{C(1),\dots,C(n)} \sum_{j=1}^{n} (\mu_{C(j)} - x_j)^2 = \sum_{j=1}^{n} \min_{C(j)} (\mu_{C(j)} - x_j)^2.$$

assigns each point to the nearest cluster center

2. Fix C, Optimize μ.

$$\min_{\mu_1,\dots\mu_K} \sum_{i=1}^{K} \sum_{j: C(j)=i} (\mu_i - x_j)^2 = \sum_{i=1}^{K} \min_{\mu_i} \sum_{j: C(j)=i} (\mu_i - x_j)^2.$$

re-centers the mean or centroid

k-means algorithm

Optimize the function

$$\min_{\mu, C} F(\mu, C) = \min_{\mu, C} \sum_{j=1}^{n} (\mu_{C(j)} - x_j)^2$$

Algorithm:

1. Fix μ, Optimize C. This is an expectation step.
2. Fix C, Optimize μ. This is a maximiation step.

This is a special case of the EM algorithm.