K-means clustering - 01

tds sheets | clustering

CONTEXT

We consider that we have N data points in a simple d-dimensional Euclidean space

$$\{x_1, x_2, ..., x_N\}$$

We want both to identify clusters among these data and to get the centers of each cluster. For a given clustering with K clusters, we define

$$c : \{1, 2, ..., N\} \rightarrow \{1, 2, ..., K\} \qquad \text{and} \qquad \alpha = (\alpha_1, \alpha_2, ..., \alpha_K)$$

respectively **the mapping that assigns to each point a cluster** and **the centers of the clusters**. The criteria we want to use to assess the quality of our clustering is the **inertia**

$$\underbrace{I(c,\alpha)}_{\text{inertia}} = \underbrace{\sum_{k=1}^{K}}_{\text{sum over the K clusters}} \underbrace{\sum_{n,c(n)=k} ||x_n - \alpha_k||^2}_{\text{sum of squared distances to center within cluster k}} = \underbrace{\sum_{n=1}^{N}}_{\text{sum over the N points}} \underbrace{||x_n - \alpha_{c(n)}||^2}_{\text{squared distance between point n and its cluster's center}}$$

that is the sum of squared distance between points and the center of their respective cluster. We want the inertia to be as small as possible.

INTUITION

As indicated by its name, **K-means assumes there are K clusters in the data**.

For these K unknown clusters, **if we knew the centers we could assign each of our points to a cluster** (choosing the cluster whose center is the closest).

In the other way, **if we knew the points assigned to each cluster we could easily compute the centers** (taking the mean of the points belonging to each cluster).

The idea behind K-means is to use an EM algorithm (Expectation-Maximisation) that will alternatively:

- fix cluster's centers and define new clusters based on it c α
- fix clusters and compute new cluster's centers based on it c α

REMARKS

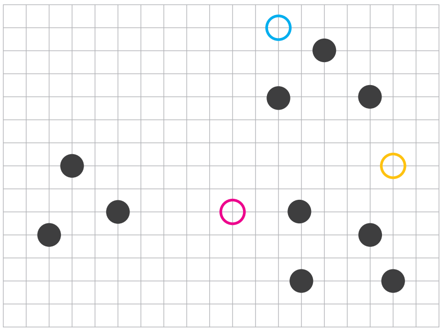
With this algorithm, we are ensured that the **inertia will decrease** at each step. However, **we can be stuck into a local minimum** instead of reaching the global minimum. As for any algorithm that can reach local optimum, the **initialisation has a huge imortance** in the final result.

K-means clustering - 02

tds sheets | clustering

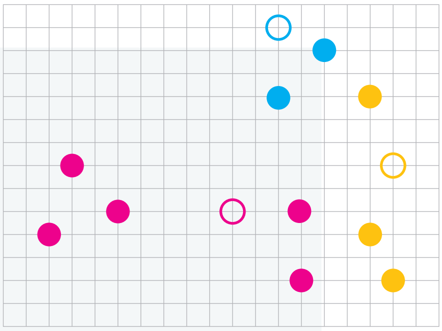
EXAMPLE WITH K=3

Initialisation



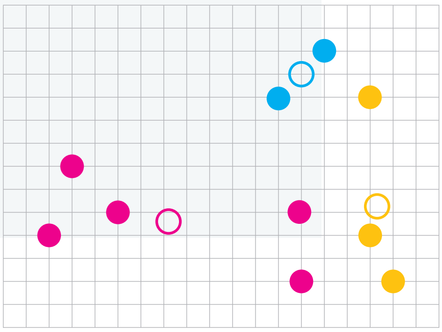
- centers are randomly initialised $\alpha^{(0)}$
- no clusters are defined yet –
- inertia can't be computed for now –

Clusters update



- centers are fixed $\alpha^{(0)}$
- clusters are updated $- \to c^{(1)}$
- inertia can be computed $I(c^{(1)}, \alpha^{(0)})$

Centers update



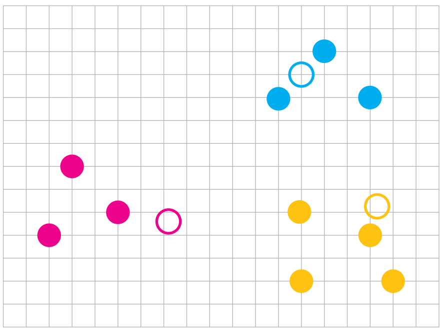
- centers are updated $\alpha^{(0)} \to \alpha^{(1)}$
- clusters are fixed $c^{(1)}$
- inertia decrease $I(c^{(1)}, \alpha^{(1)}) \leq I(c^{(1)}, \alpha^{(0)})$

$$\forall k \quad C_k^{(1)} = \{n, c^{(1)}(n) = k\} \qquad \alpha_k^{(1)} = \frac{1}{C_k^{(1)}} \sum_{n \in C_k^{(1)}} x_n$$

$$\Rightarrow \quad \sum_{n \in C_k^{(1)}} ||x_n - \alpha_k^{(1)}||^2 \leq \sum_{n \in C_k^{(1)}} ||x_n - \alpha_k^{(0)}||^2$$

$$\Rightarrow \quad I(c^{(1)}, \alpha^{(1)}) \leq I(c^{(1)}, \alpha^{(0)})$$

Clusters update



- centers are fixed $\alpha^{(1)}$
- clusters are updated $c^{(1)} \to c^{(2)}$
- inertia decrease $I(c^{(2)}, \alpha^{(1)}) \leq I(c^{(1)}, \alpha^{(1)})$

$$\forall n \quad c^{(2)}(n) = \arg\min_k ||x_n - \alpha_k^{(1)}||^2$$

$$\Rightarrow \quad ||x_n - \alpha_{c^{(2)}(n)}^{(1)}||^2 \leq ||x_n - \alpha_{c^{(1)}(n)}^{(1)}||^2$$

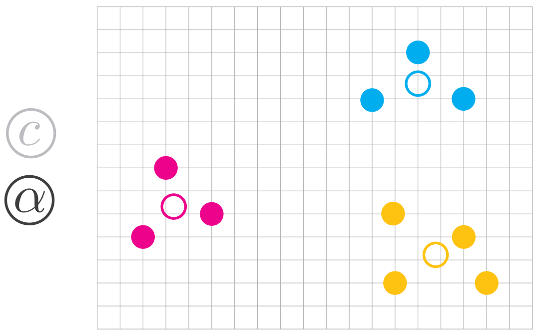
$$\Rightarrow \quad I(c^{(2)}, \alpha^{(1)}) \leq I(c^{(1)}, \alpha^{(1)})$$

K-means clustering - 03

tds sheets | clustering

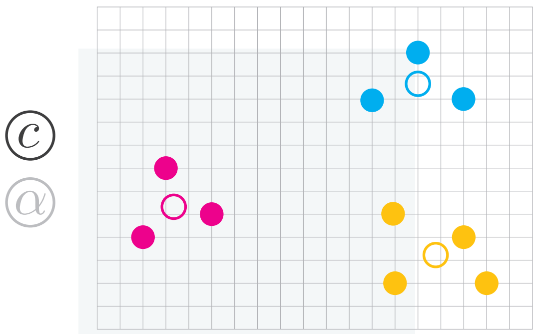
EXAMPLE WITH K=3 (SUITE)

Centers update



- centers are updated $\alpha^{(1)} \rightarrow \alpha^{(2)}$
- clusters are fixed $c^{(2)}$
- inertia decrease $I(c^{(2)}, \alpha^{(2)}) \leq I(c^{(2)}, \alpha^{(1)})$

End



- centers are fixed $\alpha^{(2)}$
- clusters are already up-to-date $c^{(2)}$
- inertia has reached a local minimum $I(c^{(2)}, \alpha^{(2)})$

HOW TO CHOOSE K?

As K-means algorithm is defined for a given K, one main question is **"How to choose the value of K?"**. We can easily verify that **the higher K is, the lower optimal inertia will be**. So we can't just choose the value K that minimise inertia, it won't have any sense.

Instead, a good way to select a value of K is to use the **elbow method**. The idea is to plot **the graph of the best obtained inertia with respect to the value of K**. Then, if there is one, choose the value of K "at the elbow". If such elbow appears, it means indeed that after it, adding more clusters brings less value and so is not interesting.