IA et science des données

Cours 9 – mardi 25 mars 2025

Clustering: algorithme des k-moyennes

Christophe Marsala

Sorbonne Université

LU3IN026 - 2024-2025

Plan du cours

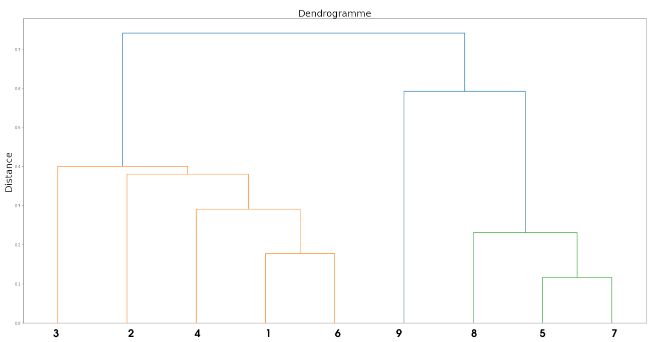
Rappels

- Classification : trouver des classes de descriptions
- Un ensemble de données sans classe connue
  - on recherche à faire des regroupements de descriptions similaires
  - on souhaite mettre en évidence des classes, des catégories
- But : former des groupes de données qui se ressemblent
  - clustering : faire des groupes parmi les données
  - cluster : ensemble de données regroupées ensemble
- Exemple :
  - le clustering hiérarchique
  - l'algorithme des k-moyennes

Conclusion sur le clustering hiérarchique

- Algorithme très efficace sur des jeux de données assez réduit, sinon ça devient vite peu lisible
- Le nombre de classes à trouver n'est pas défini : il est estimé par l'étude du dendrogramme
- Les calculs sont très coûteux ! ($\geq o(n^2)$)

Exemple de dendrogramme final



Algorithme des k-moyennes (ou k-means)

- Un des algorithmes de clustering le plus courant
- Idée : ceux qui se ressemblent, s'assemblent
  - trouver des clusters qui séparent les données de façon équitable
    - les clusters seront repérés par leur centre
- Mise en œuvre
  - choix du nombre de clusters à trouver : $k > 1$, entier naturel
  - mesure de la proximité entre données : mesure de distance
    - par exemple, distance euclidienne entre leurs descriptions
  - choisir k centres de clusters et affecter les données au cluster qui leur est le plus proche
  - modifier les k centres en fonction des données qui sont dans leur cluster

APPRENTISSAGE NON-SUPERVISÉ (CLUSTERING)

BUT: CONSTRUIRE DES GROUPES D'EXEMPLES HOMOGÈNES ET DISTANTS

DEUX TYPES D'APPROCHES :

1) AGGLOMÉRATIVE : CLUSTERING HIÉRARCHIQUE

2) PAR PARTITIONNEMENT : $\rightarrow$ TROUVER UNE PARTITION DES EXEMPLES QUI OPTIMISE UN CRITÈRE

CHOISIR UN REPRÉSENTANT D'UN CLUSTER

K-MOYENNE = CENTRE DE GRAVITÉ DU CLUSTER

K-MÉDOÏDE = UN EXEMPLE DU CLUSTER

1

NOTATIONS :

- $\mathbb{X} = \{x_1, \dots, x_n\}$ EXEMPLES DE $\mathbb{R}^d$
- ON CHERCHE K CLUSTERS $\mathcal{C}_1, \dots, \mathcal{C}_K$ HOMOGÈNES ET DISTANTS LES UNS DES AUTRES
- $x_i = (x_{i1}, \dots, x_{id}) \quad \forall i = 1 \dots n$
- $\mathcal{C}_k$ = CLUSTER = SOUS-ENSEMBLE DE $\mathbb{X}$

$\rightarrow$ CONTIENT $|\mathcal{C}_k|$ EXEMPLES $(= n_k)$

K-MOYENNES $\rightarrow$ REPRÉSENTÉ PAR SON CENTROÏDE c_k

- $c_k = (c_{k,1}, \dots, c_{k,d})$ = CENTRE DE GRAVITÉ

$$\forall j = 1..d, \quad \boxed{c_{k,j} = \frac{1}{n_k} \sum_{x_i \in \mathcal{C}_k} x_{i,j}}$$

2

REMARQUE :

FONCTION CARACTÉRISTIQUE D'UN ENSEMBLE $\mathcal{C}_k \subseteq \mathbb{X}$:

$$\chi_k : \mathbb{X} \longrightarrow \{0,1\}$$
$$x \longmapsto \chi_k(x) = \begin{cases} 1 & \text{si } x \in \mathcal{C}_k \\ 0 & \text{sinon} \end{cases}$$

ON PEUT ÉCRIRE :

$$\forall k, \forall j : \quad c_{k,j} = \frac{1}{n_k} \sum_{i=1}^{n} \chi_k(x_i)\, x_{ij}$$

3

ALGORITHME DES K-MOYENNES

ÉTANT DONNÉ : $\mathbb{X}$, $K > 1$ (ET $\varepsilon > 0$ PETIT)

$\boxed{\text{ÉTAPE } 0}$ SÉLECTIONNER K EXEMPLES DE $\mathbb{X}$

SOIT $\tilde{x}_1, \dots, \tilde{x}_K$ CES EXEMPLES

- AFFECTER CHAQUE $x \in \mathbb{X}$ AU GROUPE DE L'EXEMPLE $\tilde{x}_i$ DONT IL EST LE + PROCHE

$\rightarrow$ PARTITION : $\boxed{\mathcal{P}^{(0)} = \{\mathcal{C}_1^{(0)}, \dots, \mathcal{C}_K^{(0)}\}}$

- DÉTERMINER LES CENTRES DE CHAQUE GROUPE : $c_1^{(0)}, \dots, c_K^{(0)}$ AVEC LES EXEMPLES DU GROUPE

4

$\boxed{\text{ÉTAPE } 1}$ RÉ-AFFECTER CHAQUE $x \in \mathbb{X}$ AU GROUPE DONT IL EST LE PLUS PROCHE DU CENTRE

$\rightarrow$ NOUVELLE PARTITION : $\boxed{\mathcal{P}^{(1)} = \{\mathcal{C}_1^{(1)}, \dots, \mathcal{C}_K^{(1)}\}}$

- RE-CALCULER LES NOUVEAUX CENTRES DES GROUPES : $c_1^{(1)}, \dots, c_K^{(1)}$

[Q] L'OBJECTIF EST-IL ATTEINT ?

$\Rightarrow$ CRITÈRE D'ARRÊT = CONVERGENCE

- SI OUI : LA PARTITION OBTENUE EST $\mathcal{P}$
- SI NON : NOUVELLE $\boxed{\text{ÉTAPE}}$

5

OBJECTIF DES K-MOYENNES

TROUVER UNE PARTITION $P = \{\mathcal{C}_1, \dots, \mathcal{C}_K\}$ de $\mathbb{X}$ t.q.:

1) $\forall k = 1 \dots K, \quad |\mathcal{C}_k| > 0$

2) $\mathcal{C}_1 \cup \dots \cup \mathcal{C}_K = \bigcup_{k=1}^{K} \mathcal{C}_k = \mathbb{X}$

3) $\forall k, k', \quad k \neq k' \Rightarrow \mathcal{C}_k \cap \mathcal{C}_{k'} = \emptyset$

REM : ETANT DONNÉ $\mathbb{X}$ et $K > 1$

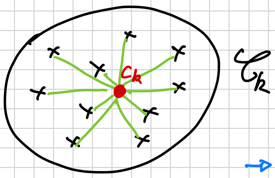
[Q] QU'EST-CE QU'UNE BONNE PARTITION ?

6

UNE BONNE PARTITION MINIMISE LES DISTANCES INTRA-CLUSTER

⇒ INERTIE D'UN CLUSTER : MESURE INTRA-CLUSTER

$$\boxed{J_k = \sum_{x_i \in \mathcal{C}_k} d_E^2\left(x_i, c_k\right)}$$



MESURE DE LA DENSITÉ AUTOUR DE c_k

→ COMPACITÉ DU CLUSTER

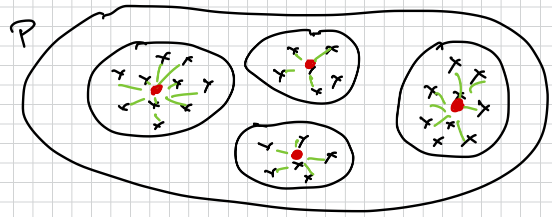
7

INERTIE D'UNE PARTITION (CO-INERTIE)

$$P = \{\mathcal{C}_1, \dots, \mathcal{C}_K\}$$

$$\boxed{J(P) = \sum_{k=1}^{K} J_k}$$

MESURE GLOBALE DE LA COMPACITÉ DES CLUSTERS



8

ALGORITHME DES K-MOYENNES : ARRÊT

→ CRITÈRE DE CONVERGENCE

$P^{(t)}$: PARTITION À L'ÉTAPE t

$P^{(t+1)}$: PARTITION À L'ÉTAPE $t+1$

$$\boxed{\left| J\left(P^{(t+1)}\right) - J\left(P^{(t)}\right) \right| < \varepsilon}$$

$\varepsilon > 0$, petit

$\varepsilon \in \mathbb{R}^{+*}$

9

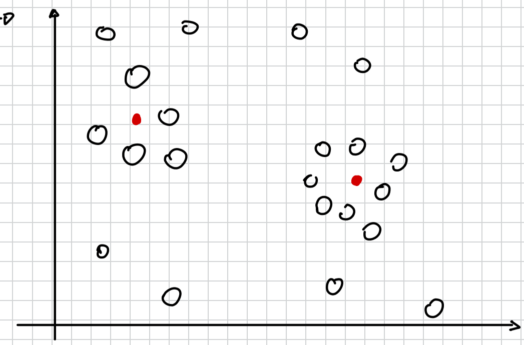
BILAN : ALGORITHME DES K-MOYENNES

- HYPER-PARAMÈTRES : . $K > 1$ ENTIER
  . $\varepsilon > 0$ RÉEL (PETIT)
  + CHOIX DE L'INITIALISATION
- LIMITES DE L'ALGORITHME :
  . CHOIX DE L'INITIALISATION ?
  . CHOIX DE K ET AUSSI DE ε ?
- RISQUES :
  → CONVERGENCE VERS UN MINIMUM LOCAL
  → CONVERGENCE PEUT ÊTRE LONGUE

10

EXEMPLE EN 2 DIMENSIONS :

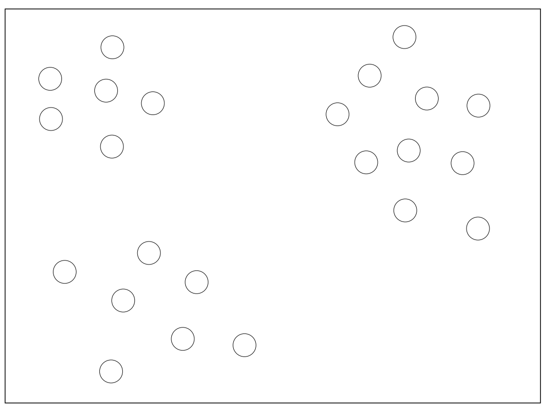
→



11

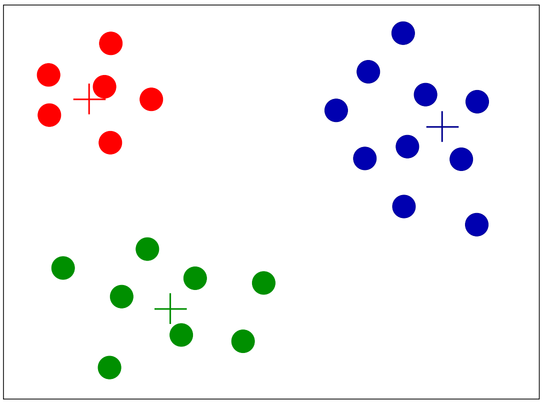
Un petit exemple

- Un ensemble de données quelconque



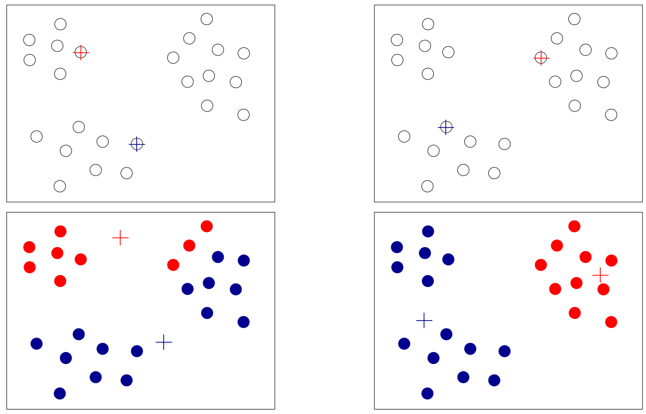
Un petit exemple

- Convergence de l'algorithme : les centres ne changent pas



Clusters différents au final

- Le choix initial des centres est important ! (ici avec $K = 2$)



Algorithme K-moyennes (2)

- Prérequis
  - X : un ensemble de données (base d'apprentissage)
  - un entier naturel $K > 0$ (le nombre de clusters à trouver)
  - une mesure de distance d entre deux exemples x et y : $d(x, y)$
- Algorithme :
  1. choisir aléatoirement K exemples dans X comme premiers centres de clusters c_1, c_2,..., c_K
     - chaque centre c_k définit un cluster C_k
  2. affecter chaque x de X au cluster dont il est le plus proche
     - calculer $d(x, c_1)$,..., $d(x, c_K)$
     - affecter x au cluster C_k pour lequel $d(x, c_k)$ est la plus petite
  3. mettre à jour les centres des clusters
     - c_k est la moyenne des descriptions du cluster C_k
  4. retourner à l'étape 2 jusqu'à ce que l'inertie globale ne change plus beaucoup
- Résultat
  - un ensemble de clusters C_1,..., C_K

Évaluation du résultat d'un clustering

- Évaluer la partition obtenue : mesurer sa qualité
  - différentes approches
  - utilisation des caractéristiques des clusters
- Compacité d'un cluster
  - évaluer combien les exemples sont proches les uns des autres
  - compacité intra-cluster
- Séparabilité des clusters
  - évalue combien les clusters sont éloignés les uns des autres
  - distance inter-clusters
- Mesure globale : index d'une partition
  - index de Dunn
  - index de Xie-Beni
  - ...

Les K-moyennes en pratique

- Algorithme très simple à mettre en œuvre
  - algo ancien : James McQueen 1967
  - mais encore très utilisé !
  - nombreuses variantes...
- Quelques problèmes
  - quelle valeur pour K ?
  - convergence : la stabilisation peut être très longue à venir
  - sensible au choix initial des centres
  - quelle mesure de distance ?

K-moyenne : ce que l'on a vu

- Objectif : trouver une partition en K groupes (ou clusters)
  - bonne partition : minimise l'inertie globale intra-cluster
  - mesure de l'inertie d'un cluster : densité autour de son centre
- Algorithme : itérations successives jusqu'à convergence
  - affectation des exemples aux clusters
  - mise à jour des centres
  - arrêt si convergence ou itérations max