Cours : K-Means

Apprentissage non supervisé – Clustering

K. Kadri

Plan

Qu'est-ce que le clustering ?

- Objectif : regrouper des points "semblables" sans labels.
- Créer des groupes appelés **clusters**.
- Applications :
  - segmentation client,
  - compression d'images,
  - détection d'anomalies,
  - pré-traitement en ML supervisé.

Apprentissage non supervisé

Pas de vérité terrain $\rightarrow$ on cherche la structure cachée dans les données.

Intuition : regrouper les points par proximité

- On veut former K clusters.
- Chaque cluster a un **centre** : le **centroïde**.
- Les points sont affectés au centroïde le plus proche (distance euclidienne).

Idée clé

Minimiser la distance totale entre les points et leur centroïde.

Étapes de l'algorithme

1. Choisir K centres initiaux (aléatoires ou k-means++).
2. Assignation : chaque point $\rightarrow$ centroïde le plus proche.
3. Mise à jour : recalculer chaque centroïde (moyenne des points assignés).
4. Répéter 2–3 jusqu'à convergence.

Convergence

Quand les centroïdes ne bougent plus ou très peu.

Fonction objectif

Inertie (Within-Cluster Sum of Squares)

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

- x_i : un point
- C_k : cluster k
- μ_k : centroïde du cluster

Objectif

Minimiser $J \rightarrow$ clusters compacts et homogènes.

Méthode du coude (Elbow method)

- On calcule l'inertie pour plusieurs valeurs de K.
- On cherche un point où la diminution ralentit fortement.

Interprétation

Avant le coude : ajouter un cluster améliore beaucoup. Après le coude : amélioration marginale $\rightarrow$ K optimal.

Silhouette Score

Définition

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

- $a(i)$: distance intra-cluster,
- $b(i)$: distance au cluster le plus proche.
- $s(i) \in [-1, 1]$.

Bonne valeur

Plus s est proche de $1 \rightarrow$ meilleur clustering.

Avantages de K-Means

- Simple à comprendre et à implémenter.
- Très rapide (scalable).
- Bon pour les données bien séparées en formes sphériques.

Cas d'usage

Segmentation client, quantification couleur, partitionnement rapide.

Limites de K-Means

- Nécessite de fixer K.
- Suppose des clusters “ronds”.
- Sensible aux outliers.
- Peut converger vers un mauvais optimum (solution locale).

Solution pratique

Utiliser l'init `k-means++` et plusieurs random states.