Chapitrre 4

Classification supervisée

Classe Master 1 : I2A + STIC

Plan

1. **Introduction**
2. **Approches de classification**
3. **Algorithme K-moyennes**
4. **Classification hiérarchique ascendante**

1. Introduction

- La classification est l'un des méthodes les plus importants du *data mining*.
- Elle permet d'analyser de grandes quantités de données pour en tirer du sens et pour prendre des décisions.
- L'idée générale est de *regrouper ou prédire des catégories* à partir d'informations disponibles.
- Il existe deux grandes approches :
  - ✓ La classification supervisée, où l'on apprend à partir d'exemples déjà classés.
  - ✓ La classification non supervisée, où l'on cherche soi-même des groupes naturels sans connaître les les catégories à l'avance.

1. Introduction

- La classification est utilisée dans plusieurs domaines : dans les recommandations, la reconnaissance d'images, la détection de comportements ou la segmentation de clients, cybersécurité, etc.
- La Collision entre le français et l'anglais dans le domaine est la suivante :

Français	Anglais
Classification	Clustering
Classement	Classification

1. Approches de classification

1. Classement : Classification supervisée

- Il permet de prédire si une instance de donnée est membre d'un groupe ou d'une classe prédéfinie.
- Classe : Groupes d'instances avec des profils particuliers
- Apprentissage supervisé : classes connues à l'avance

Donc, Le *classement* consiste à placer chaque individu de la population dans une classe, parmi plusieurs classes prédéfinies, en fonction des caractéristiques de l'individu indiquées comme variables explicatives

1. Introduction

1. Classement : Classification supervisée

- **Applications:**
  - ✓ marketing direct (profils des consommateurs),
  - ✓ grande distribution (classement des clients),
  - ✓ médecine (malades/non malades), etc.
- **Méthodes courantes :**
  - ✓ Classificateur de Bayes
  - ✓ Analyse discriminante et classificateur de Fisher
  - ✓ Kppv
  - ✓ Méthode du centroïde
  - ✓ Arbres de décision
  - ✓ Réseaux de neurones

1. Introduction

2. Classification non supervisée (Clustering)

- Partitionnement logique des objets en clusters.
- **Clusters:** groupes d'instances ayant les mêmes caractéristiques.
- Apprentissage non supervisé: (classes inconnues à l'avance)

La classification *non supervisée ,* appelée aussi *automatique,* consiste donc à regrouper les individus d'une population en un nombre limité de classes qui :

- ✓ ne sont pas prédéfinies mais déterminées au cours de l'opération.
- ✓ regroupent les individus ayant des caractéristiques similaires et séparent les individus ayant des caractéristiques différentes

1. Introduction

2. Classification non supervisée (Clustering)

- **Applications:**
  - ✓ Economie (segmentation de marchés), médecine, etc
- **Méthodes courantes :**
  - ✓ Algorithme K-means (K-moyennes).
  - ✓ Classification hiérarchique ascendante (AHC).

- **Problème:**

✓ Interprétation des clusters identifiés

- **Remarque:**

✓ Clustering= segmentation=partitionnement=regroupement

3. Algorithme K-means (K-moyennes)

Principe de base

- le principe de base de *K-means* est regrouper les objets en se basant à leur *distance* par rapport à un centre (*centroid*).
- Initialement, les *centres* sont choisis aléatoirement, mais ils seront mis à jour par re-calcul.
- Les cluster initiaux seront affinés, voire changés.
- La classification finale correspondra à une stabilisation des clusters : aucun objet ne change d'appartenance à son cluster.
- Remarque : Il existe beaucoup d'algorithmes de clustring, dont le plus ancien est celui des **K-means** avec beaucoup de variations.

3. Algorithme K-means (K-moyennes)

Description de l'algorithme

- Soit les *N objets* à classifier (pour les besoins d'analogie géométrique, on désigne les objets comme étant des *points*):
- Choisir *K centres* aléatoires (*seeds*).
- Calculer la *distance* entre chacun des *points restants* et chaque *centre* (seed).
- Calculer le *centre* de chaque *cluster* **:** La valeur de chaque caractéristique de chaque centre est égale à la moyenne des valeurs de la caractéristique pour tous les points du cluster Correspondant.

3. Algorithme K-means (K-moyennes)

Description de l'algorithme

- ***Réaffecter*** les points selon leur distance par rapport au centre de chaque cluster : de nouveaux clusters peuvent être générés.
- Recalculer le centre C de chaque cluster
- S'arrêter lorsqu'aucun point ne change de cluster (stabilisation des clusters). à son cluster.

3. Algorithme K-means (K-moyennes)

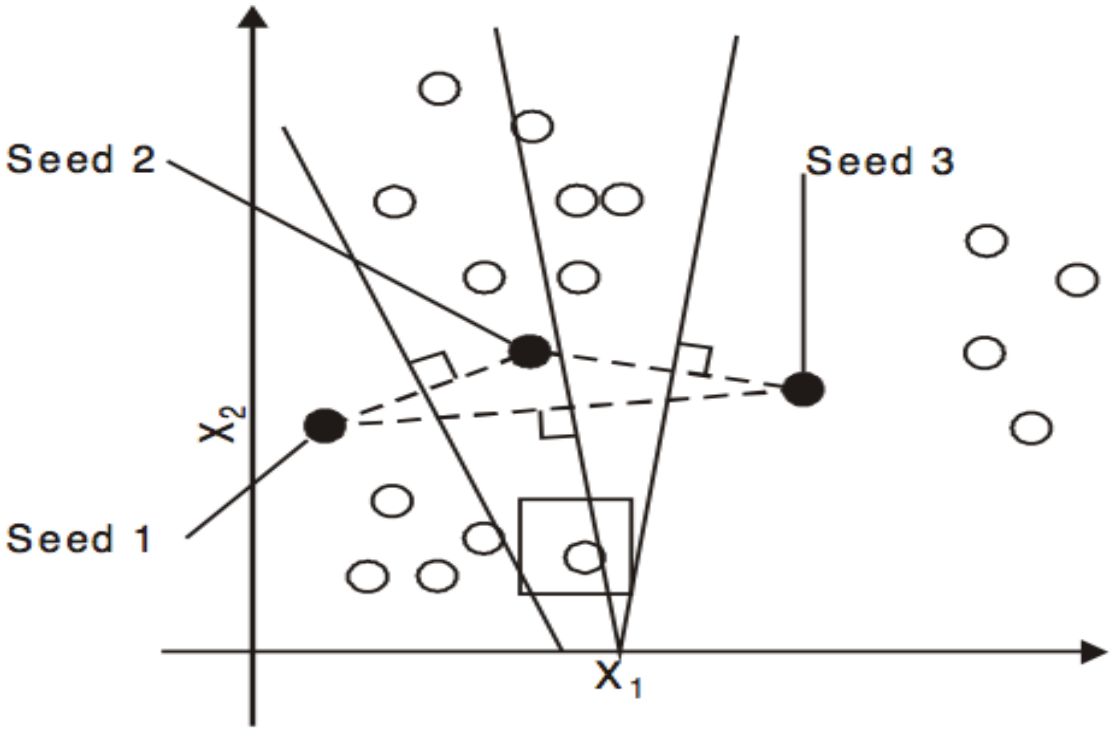
Illustration de l'algorithme

- On suppose que les *objets* sont décrits par deux *caractéristiques X1 et X2*.
- Les trois figures suivantes (de Nagabhushana) montrent les premières itérations de k-means.

3. Algorithme K-means (K-moyennes)

Illustration de l'algorithme

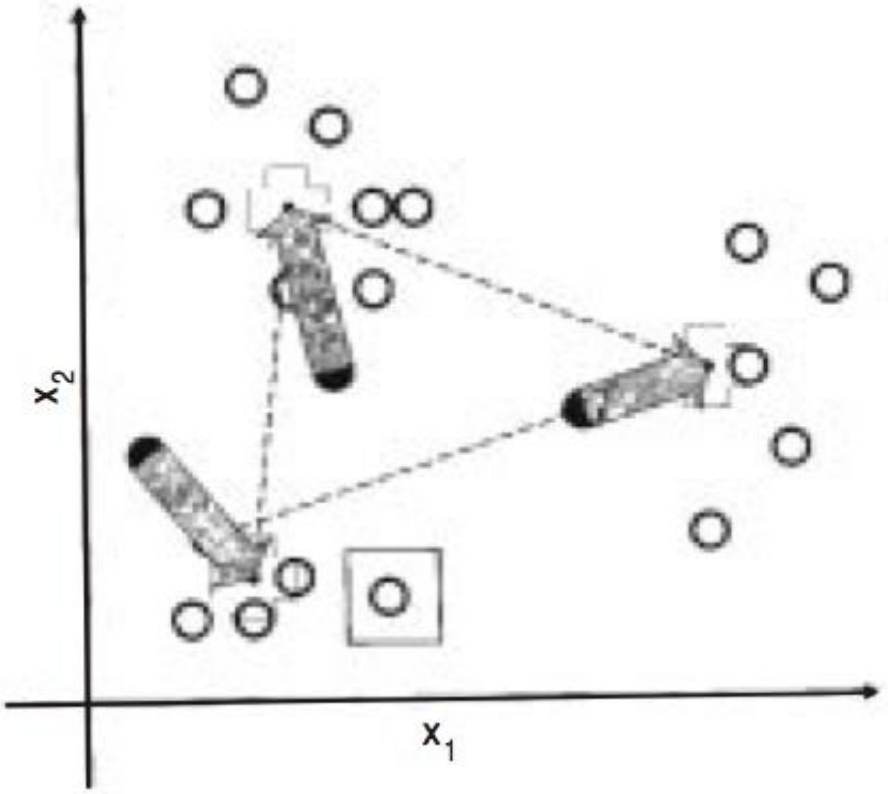
- Cette figure montre le choix des *seeds initiaux* (cercles noirs) et la formation des premiers clusters (trois).



3. Algorithme K-means (K-moyennes)

Illustration de l'algorithme

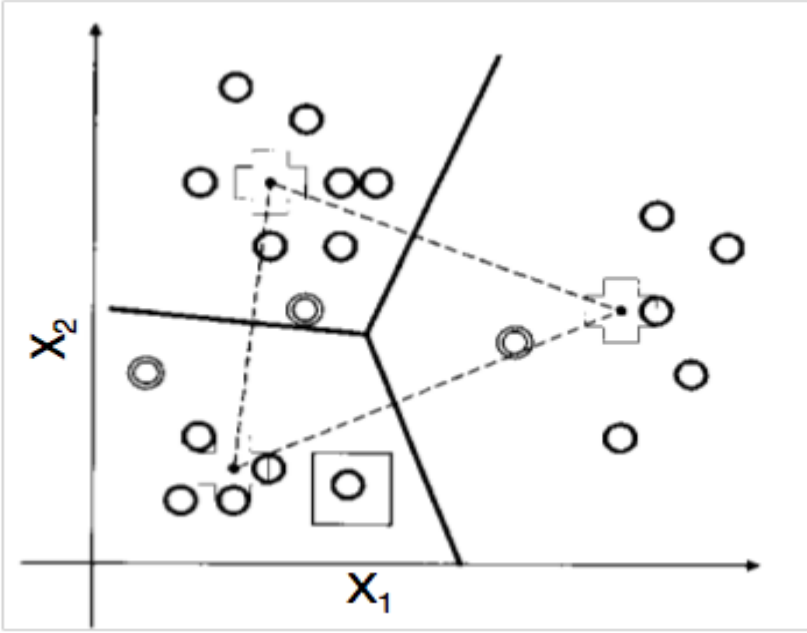
- Cette dexieme figure montre les centroids (symbole +)



3. Algorithme K-means (K-moyennes)

Illustration de l'algorithme

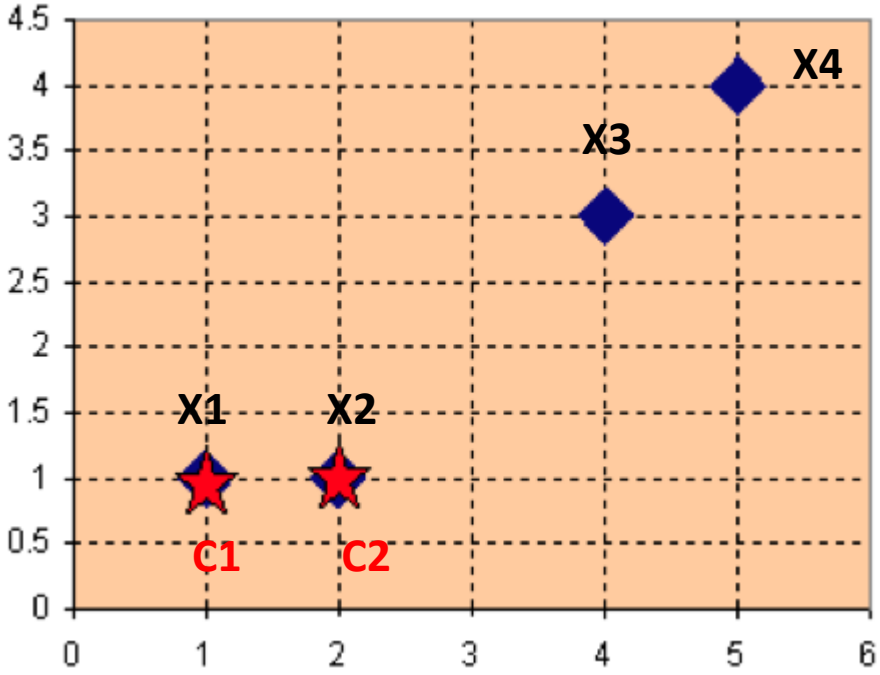
- La troisième figure montre les nouveaux clusters.
- Le point *entouré* d'un *carré* est le point ayant changé de cluster après le calcul des centres.



3. Algorithme K-means (K-moyennes)

Exemple

- -Soit les quatre ***objets*** décrits par deux ***caractéristiques*** avec les valeurs suivantes : X1(1, 1), X2(2, 1), X3(4, 3), X4(5, 4):



- On choisit les X1 et X2 comme *seeds* (étoiles) et on les désigne par C1 et C2

3. Algorithme K-means (K-moyennes)

Exemple

- On calcule la ***distance*** entre chaque ***point*** (X1,X2,X3,X4) et chaque ***centroid*** (C1,C2).
- On utilise la ***distance euclidienne*** (voir plus loin).
- On obtient la matrice suivante :

X_1	X_2	X_3	X_4	
0	1	3,61	5	C_1
1	0	2,83	4,24	C_2

- On remarque que les points X2, X3 et X4 sont proches à C2 et X1 à C1.

On obtient ***deux clusters*** composés de (X1) et (X2, X3, X4).

3. Algorithme K-means (K-moyennes)

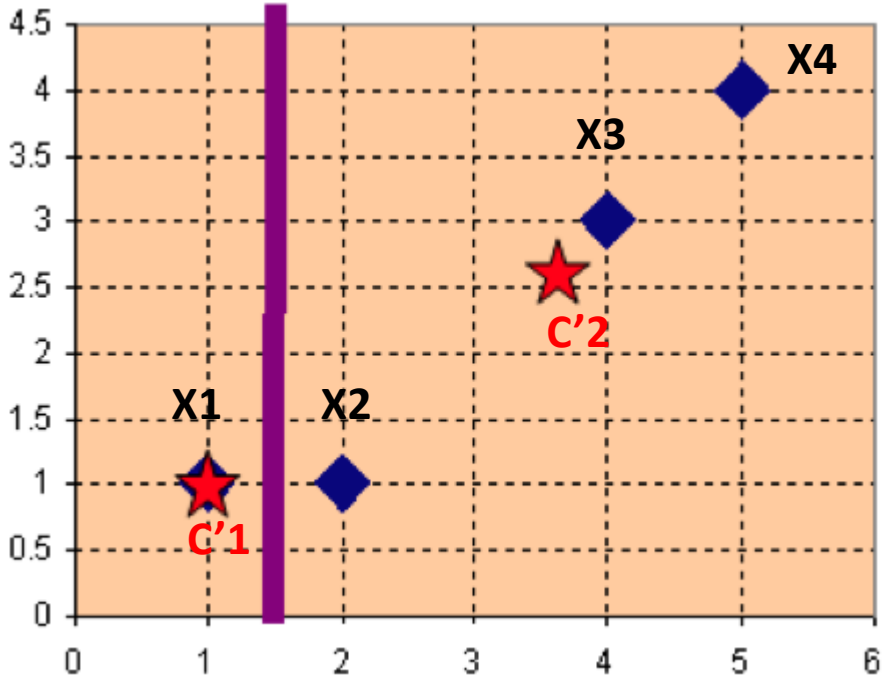
Exemple

On obtient ***deux clusters*** composés de (**X1**) et (**X2**, **X3**, **X4**).

Le calcul des nouveaux centres donne :

C'1(1/1, 1/1) donc: **C'1**(1, 1)

C'2 (2+4+5/3, 1+3+4/3) donc: **C'2**(11/3, 8/3).



3. Algorithme K-means (K-moyennes)

Exemple

On recalcule les ***nouvelles distances***. On obtient le tableau suivant :

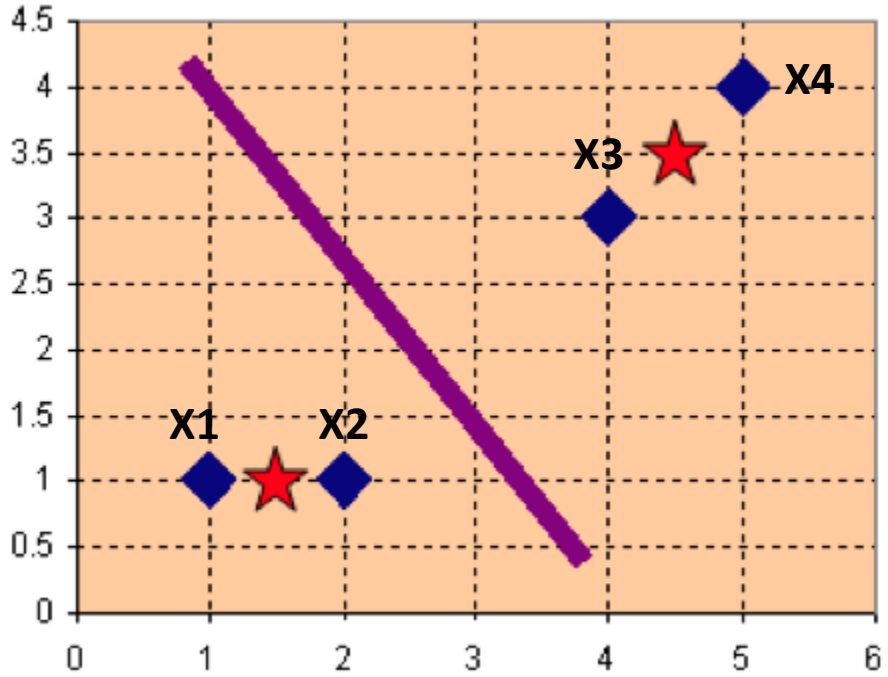
X_1	X_2	X_3	X_4	
0	1	3,61	5	C'_1
3,14	2,36	0,47	1,89	C'_2

Selon cette matrice, on déplace **X2** au cluster à gauche **C'1** et on obtient les clusters de la figure 3.

3. Algorithme K-means (K-moyennes)

Exemple

Selon cette matrice, on déplace X2 au cluster à gauche et on obtient les clusters de la figure 3 :

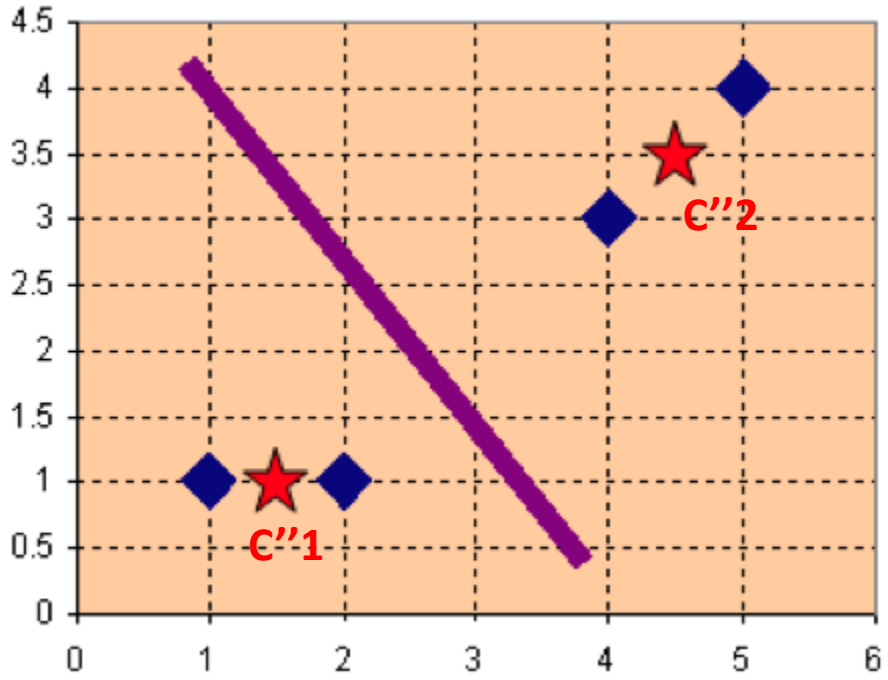


3. Algorithme K-means (K-moyennes)

Exemple

- On recalcule les centres (étoiles de la figure 3). On trouve les centres : **C''1**(1.5, 1)

C''2(4.5, 3.5).



3. Algorithme K-means (K-moyennes)

Exemple

- On calcule les distances et on obtient les distances suivantes :

X_1	X_2	X_3	X_4	
0,5	0,5	3,20	4,61	C''_1
4,30	3,54	0,71	0,71	C''_2

-La matrice montre qu'aucun point ne change de cluster.

- On arrête les itérations.

3. Algorithme K-means (K-moyennes)

Types de variables

- les *types de données* pour le *clustering* ne sont pas tous adéquats.
- On les classe par ordre croissant d'adéquation comme suit :

  - ✓ Variables symboliques (peu adaptées)(couleurs, noms, etc.)
  - ✓ Rangs (un peu mieux) (1, 2, 3, etc.)
  - ✓ Les intervalles (bien adaptées) (degrés de températures, etc.)
  - ✓ Les vraies mesures (idéal) (âge, longueurs, volumes, etc.)

3. Algorithme K-means (K-moyennes)

Types de variables

- Les intervalles et les vraies mesures sont les plus adéquats au clustering.
- Les variables symboliques et les rangs doivent être transformées pour être utilisées.
- Cependant, les transformations ne donnent pas toujours des intervalles ou mesures ayant un sens et comparables.

3. Algorithme K-means (K-moyennes)

Mesures de similarité

- pour trouver les objets associés, on utilise différentes mesures de similarité.

a. Distance : le calcul de distance dépend de la nature des données.

b. Angle entre les vecteurs : ce calcul ne tient pas compte des grandeurs des mesures.

c. Nombre de caractéristiques communes : on calcul le nombre d'appariements

3. Algorithme K-means (K-moyennes)

Mesures de similarité

a. *Distance* : Une fois les mesures standardisées, nous calculons les distances entre deux points ***i*** $(x_{i1}, x_{i2, \ldots,} x_{ip})$ et ***j*** $(x_{j1}, x_{j2, \ldots,} x_{jp})$. Les formules sont :

- Formule de Minkowski:

$$d(i,j)=\sqrt[q]{(|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+\ldots+|x_{i_p}-x_{j_p}|^q)}$$

- **distance de Manhattan:** q=1

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+\ldots+|x_{i_p}-x_{j_p}|$$

- **distance euclidienne :** q=2

$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+\ldots+|x_{i_p}-x_{j_p}|^2)}$$

3. Algorithme K-means (K-moyennes)

Choix du nombre K

- il n'existe pas de meilleurs K ou de mauvais K pour le clustering,
- puisque ce nombre est fourni par l'utilisateur pour spécifier le nombre de clusters souhaités.
- Cependant, les clusters obtenus peuvent être évalués.
- Les bons clusters seront le résultat d'un bon choix de *k et inversement.*

4. classification hiérarchique ascendante

- Elle est souvent appelée “Hierarchical Agglomerative Clustering” ou HAC,
- La classification hiérarchique ascendante est une méthode de clustering qui construit des groupes petit à petit, en partant des éléments individuels pour arriver à des groupes de plus en plus grands.