



Zied Elouedi
2018/2019

Chapitre 4

Clustering



Plan

- Types de variables
- Méthodes par partitionnement
 - K-moyenne
 - K-modes
 - K-prototypes
- Méthodes hiérarchiques
 - Par divisions
 - Par agglomérations
- Clustering incrémental

● ● ● | Introduction

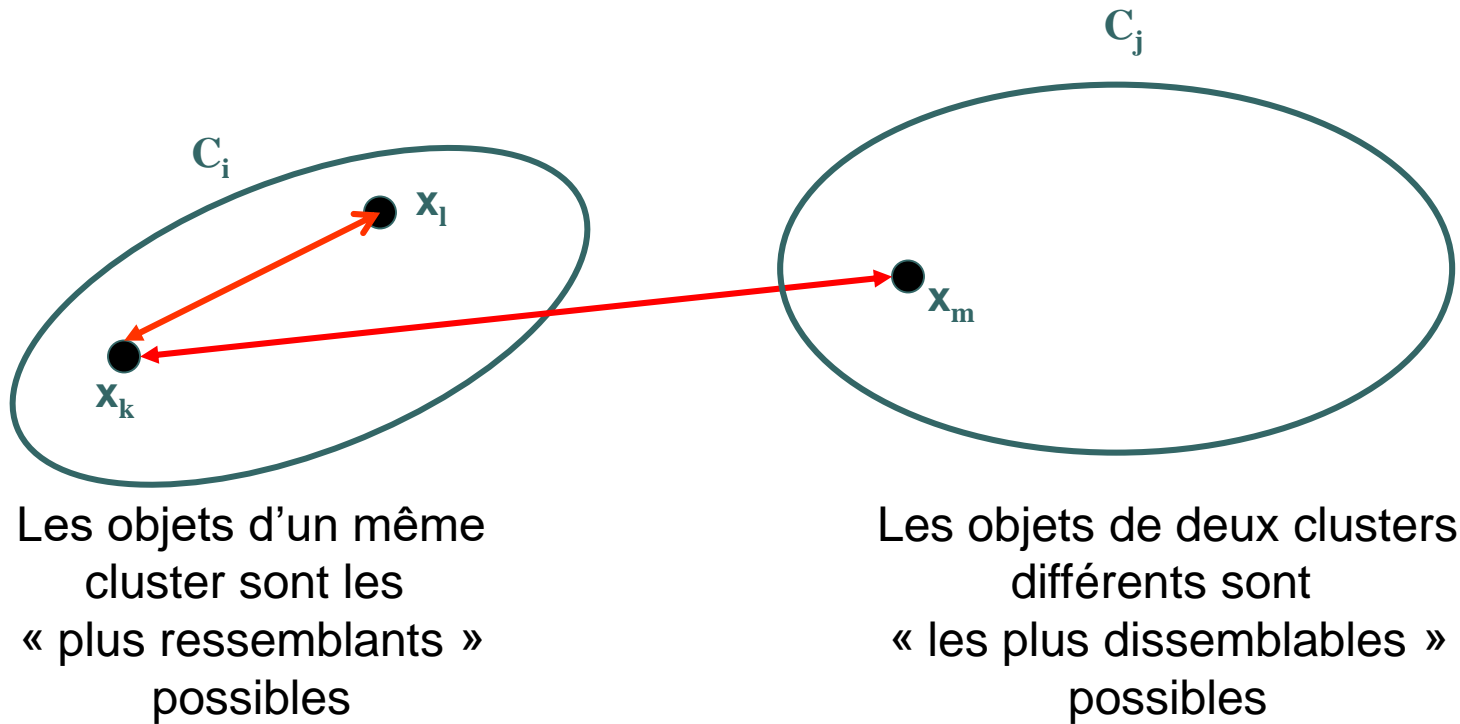
Classification **non supervisée**: Classes non prédéfinies a priori



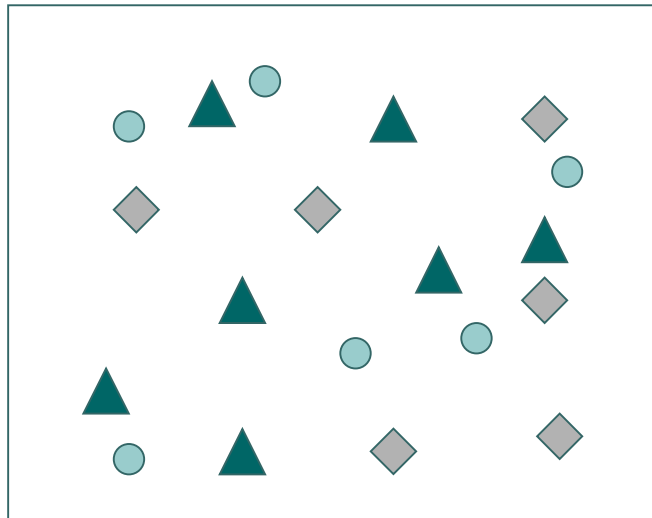
Regroupement des objets en **des clusters** formant **les classes**.

- On ne connaît pas les classes à priori :
 - Elles sont à découvrir automatiquement.
 - Il est parfois possible d'en fixer le nombre.

Introduction

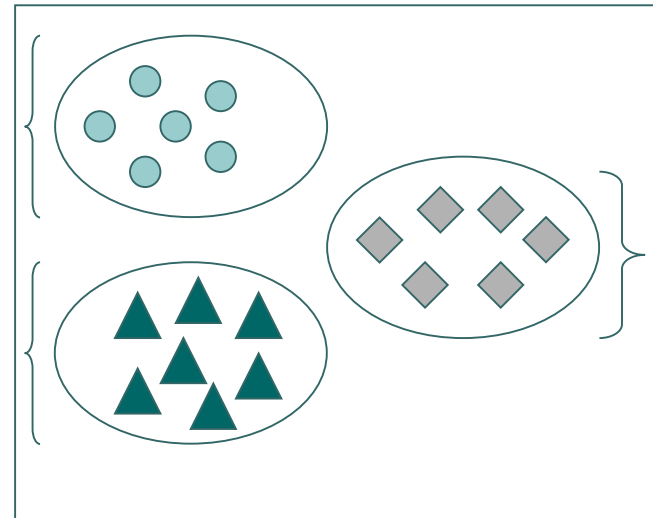


Introduction



Cluster 1

Cluster 2



Cluster 3

Maximiser la similarité au sein du même groupe (intra-cluster).
Minimiser la similarité au sein des groupes différents (inter-cluster).

Applications

- Marketing

Segmentation
des marchés

Divers
domaines

- Reconnaissance des formes

Gènes
semblables

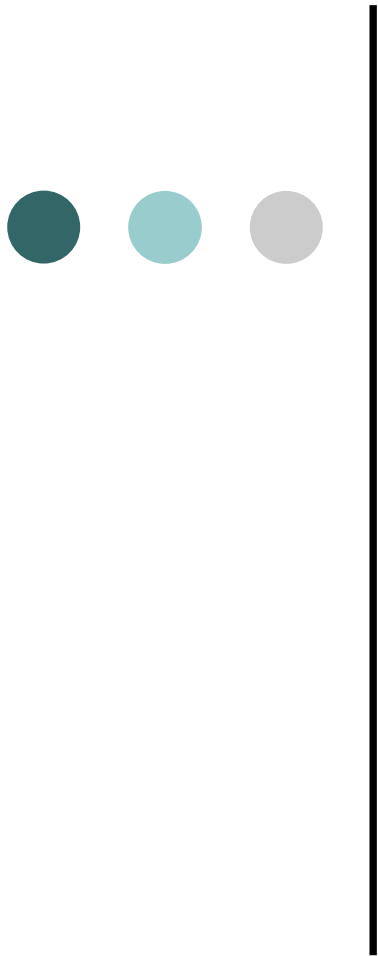
- Bio-informatique

Textes
proches

- Textmining

Groupes d'accès
similaires

- Web mining



Proximité

Proximité

- Mesure de similarité

$$s(x_1, x_2) = s(x_2, x_1) \geq 0,$$
$$s(x_1, x_1) \geq s(x_1, x_2)$$

➡ Plus la mesure est grande, plus les éléments sont similaires.

- Mesure de dissimilarité

$$d(x_1, x_2) = d(x_2, x_1) \geq 0,$$
$$d(x_1, x_2) = 0 \rightarrow x_1 = x_2$$

➡ Plus la mesure est faible, plus les éléments sont similaires.

- Distance est une dissimilarité qui vérifie l'inégalité triangulaire :

$$d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2)$$

Ces mesures sont souvent exprimées en fonction d'une distance qui change selon la nature des variables (continues, catégoriques,...)

Structure de données

- Matrice de données

$$\begin{bmatrix} x_{11} & \dots & x_{1f} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{if} & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & \dots & x_{nf} & \dots & x_{np} \end{bmatrix}$$

- Matrice de dissimilarité

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$



Types de variables



Types de variables

- Numériques (Poids, Taille, ...)
- Binaires (Service militaire, Option, ...)
- Catégoriques (Couleur, Situation familiale, ...)
- Ordinales (Résultat d'un concours, Qualité d'un produit, ...)

Variables numériques

- Distance Euclidienne

$$d(x_1, x_2) = [(x_{11}-x_{21})^2 + (x_{12}-x_{22})^2 + \dots + (x_{1p}-x_{2p})^2]^{1/2}$$

- Distance de Minkowski

$$d(x_1, x_2) = [(x_{11}-x_{21})^q + (x_{12}-x_{22})^q + \dots + (x_{1p}-x_{2p})^q]^{1/q} \quad q > 0$$

➡ Généralisation de la distance Euclidienne

- Distance de Manhattan

$$d(x_1, x_2) = |x_{11}-x_{21}| + |x_{12}-x_{22}| + \dots + |x_{1p}-x_{2p}|$$

➡ Cas particulier de la distance Minkowski ($q = 1$)

Exemple

	Age	Nombre Enfants	Salaire
P1	40	2	1000
P2	75	4	600
P3	50	3	1100
P4	35	1	550

- Distance Euclidienne

$$d(P1, P2) = ((-35)^2 + (-2)^2 + (400)^2)^{1/2} = 401.53$$

$$d(P1, P3) = ((-10)^2 + (-1)^2 + (-100)^2)^{1/2} = 100.5$$

$$d(P1, P4) = ((5)^2 + (1)^2 + (450)^2)^{1/2} = 450.03$$



P1 est plus proche de P3 que de P2 et P4

- Distance de Manhattan

$$d(P1, P2) = |-35| + |-2| + |400| = 437$$

$$d(P1, P3) = |-10| + |-1| + |-100| = 111$$

$$d(P1, P4) = |5| + |1| + |450| = 456$$

On peut aussi normaliser



P1 est plus proche de P3 que de P2 et P4

Normalisation

Pour tout attribut k, $x'_{ik} = \frac{x_{ik} - \min_i x_{ik}}{\max_i x_{ik} - \min_i x_{ik}}$

	Age	Nombre Enfants	Salaire
P1	40	2	1000
P2	75	4	600
P3	50	3	1100
P4	35	1	550



	Age	Nombre Enfants	Salaire
P1	0.125	0.333	0.818
P2	1	1	0.09
P3	0.375	0.666	1
P4	0	0	0

Distance Euclidienne

$$d(P1, P2) = ((-0.875)^2 + (-0.667)^2 + (0.728)^2)^{1/2} = 1.319$$

$$d(P1, P3) = ((-0.25)^2 + (-0.333)^2 + (-0.182)^2)^{1/2} = 0.454$$

$$d(P1, P4) = ((0.125)^2 + (0.333)^2 + (0.818)^2)^{1/2} = 0.891$$



P1 est plus proche de P3 que de P2 et P4

Distance de Manhattan

$$d(P1, P2) = |-0.875| + |-0.667| + |0.728| = 2.27$$

$$d(P1, P3) = |-0.25| + |-0.333| + |-0.182| = 0.765$$

$$d(P1, P4) = |0.125| + |0.333| + |0.818| = 1.276$$



P1 est plus proche de P3 que de P2 et P4

Variables numériques : autres mesures

- Distance de Sebestyen

$$d^2(x_1, x_2) = (x_1 - y_2)^t W (x_1 - y_2)$$

W : matrice diagonale de pondération

➡ Donner un poids différent aux attributs.

- Distance de Mahalanobis

$$d^2(x_1, x_2) = (x_1 - y_2)^t C^{-1} (x_1 - y_2)$$

C : matrice diagonale de variance-covariance

➡ Si les variables corrélées prennent trop d'importance, on peut normaliser la distance euclidienne par la covariance.

▪

▪

▪

Variables binaires

- Table de contingence

		O_j		
		1	0	Σ
O_i	1	a	b	$a+b$
	0	c	d	$c+d$
Σ		$a+c$	$b+d$	p

a = nombre de positions où i est à 1 et j est à 1

$$O_i = (1, 0, 1, 1, 1) \quad O_j = (1, 0, 1, 0, 0) \quad \Rightarrow \quad a=2, b=2, c=0, d=1$$

- Coefficient de Russel et Rao (Proportion d'occurrences positives):

$$d(O_i, O_j) = \frac{a}{a + b + c + d} = 2/5$$

- Coefficient de Jaccard (Poids des occurrences fausses est neutralisé):

$$d(O_i, O_j) = \frac{a}{a + b + c} = 1/2$$

- Coefficient de Dice (Poids double pour les occurrences vraies):

$$d(O_i, O_j) = \frac{2a}{2a + b + c} = 2/3$$



Variables binaires: autres mesures

$$d(O_i, O_j) = \frac{a}{a + 2(b + c)} = 2/6 = 1/3$$


$$d(O_i, O_j) = \frac{a + d}{a + b + c + d} = 2/5$$

▪

▪

▪

Variables catégoriques

- Généralisation des variables binaires.
 - **Méthode 1:** Utiliser un grand nombre de variables binaires.
 Créer une variable binaire pour chaque modalité
(ex: variable rouge qui prend les valeurs vraie ou fausse)
 - **Méthode 2:** Matching simple

m: Nombre de ressemblances,

p: Nombre total de variables

$$d(O_i, O_j) = \frac{p - m}{p}$$



Variables catégoriques : autres distance

- Distance d'édition (distance de Levenshtein)
- Distance cosinus

-
-
-

Variables ordinales

- Une variable ordinale peut être discrète ou continue.
- L'ordre peut être important.
- L'idée est:
 - Remplacer x_{if} par son rang $r_{if} \in \{1, \dots, M_f\}$
 - Remplacer le rang de chaque variable par une valeur dans $[0, 1]$ en remplaçant la variable n dans l'objet O_i par

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

- Utiliser une distance pour calculer la similarité.



Evaluation de la qualité d'un clustering

Inertie intra-cluster

- Chaque cluster C_k est caractérisé par :

- Centre de gravité : $\mu_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i$

- Inertie mesurant la concentration des points du cluster au tour du centre de gravité :

$$J_k = \sum_{x_i \in C_k} d^2(x_i, \mu_k)$$

➡ Plus l'inertie intra-cluster est faible, plus la dispersion des points autour du centre de gravité est petite.

- Inertie intra-cluster :

$$J_w = \sum_k \sum_{x_i \in C_k} d^2(x_i, \mu_k) = \sum_k J_k$$

- Matrice de variance-covariance : $\sum_k = \sum_{x_i \in C_k} (x_i - \mu_k)(x_i - \mu_k)^t$

Inertie inter-cluster

- Soit μ le centre de gravité du nuage des points :

$$\mu = \frac{1}{N} \sum_i x_i$$

- Les différents centres de gravité des clusters forment aussi un nuage de points.

- Inertie inter-cluster : $J_b = \sum_k |C_k| d^2(\mu, \mu_k)$

➡ Plus l'inertie inter-cluster est grande, plus les clusters sont bien séparés.

- Matrice de variance-covariance inter-cluster :

$$\sum_b = \sum_k (\mu_k - \mu)(\mu_k - \mu)^t$$

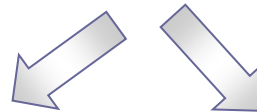
Obtenir une bonne partition : Minimiser l'inertie intra-cluster (groupes homogènes) et maximiser l'inertie inter-cluster (séparation inter-groupe).



Principales approches de clustering

Principales approches

Cluster



**Méthodes par
partitionnement**



Diviser un ensemble de
N objets en K clusters

**Méthodes
hiérarchiques**



Création d'une décomposition
hiérarchique des objets
selon certains critères



Méthodes par partitionnement



Clustering par partitionnement

- Construire une partition des données à partir de N objets, **K clusters** ($K < N$).
- Approche directe:
 - Construire toutes les partitions possibles.
 - Evaluer la qualité de chaque cluster afin de retenir la meilleure partition.
 - ➡ Nombre de partitions possibles augmente de manière exponentielle.
 - ➡ C'est un problème NP difficile.



Comment faire?

- Minimiser l'inertie intra-classe $J_w = \sum_k \sum_{x \in C_k} d^2(x, \mu_k)$
- Ne pas parcourir toutes les partitions possibles.
- Utiliser des heuristiques pour trouver une bonne partition.
- K-means, K-medoids, CLARA, CLARANS, etc.

● ● ● | Méthode K-moyenne (K-means)

K-moyenne (MacQueen 1967)

- Choisir le nombre de clusters et une mesure de distance.
- Construire une partition aléatoire comportant K clusters non vides.
- Répéter
 - Calculer le centre de gravité de chaque cluster de la partition.
 - Assigner chaque objet au cluster dont le centre gravité est plus proche (distance).

Jusqu'à ce que la partition soit stable (les objets ne changent plus de clusters).

Exemple: K-moyenne (1)

$T = \{2, 4, 6, 7, 8, 11, 13\}$

d = distance Euclidienne

- Choisir 3 clusters au hasard à partir de T :

$C_1 = \{2\}, \mu_1 = 2,$

$C_2 = \{4\}, \mu_2 = 4,$

$C_3 = \{6\}, \mu_3 = 6$

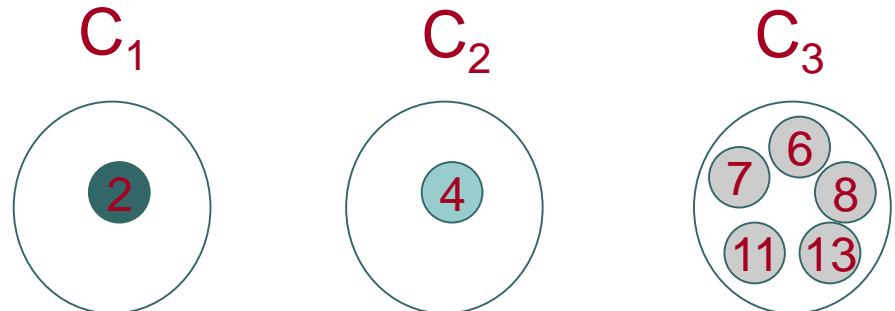
- Les autres objets de T sont affectés au cluster C_3 puisque $d(O, \mu_3)$ est minimale.

- On aura:

$C_1 = \{2\}, \mu_1 = 2,$

$C_2 = \{4\}, \mu_2 = 4,$

$C_3 = \{6, 7, 8, 11, 13\}, \mu_3 = 45/5 = 9$



Exemple: K-moyenne (2)

- $d(6, \mu_2) < d(6, \mu_3)$

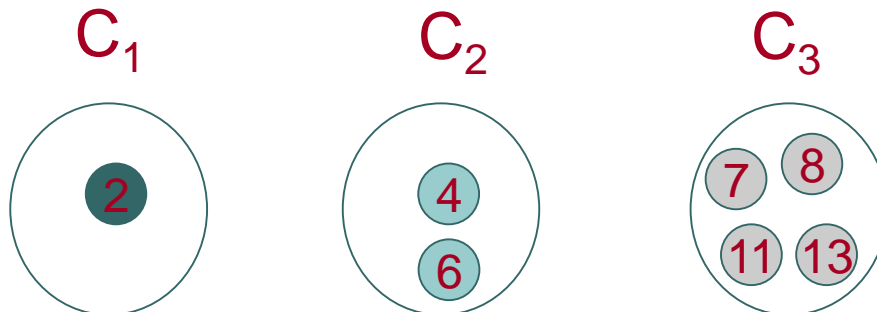
➡ 6 passe au cluster C_2 : les autres objets restent dans leurs clusters.

- On aura:

$$C_1 = \{2\}, \mu_1 = 2,$$

$$C_2 = \{4, 6\}, \mu_2 = 10/2 = 5$$

$$C_3 = \{7, 8, 11, 13\}, \mu_3 = 39/4 = 9.75$$



Exemple: K-moyenne (3)

- $d(7, \mu_2) < d(7, \mu_3)$

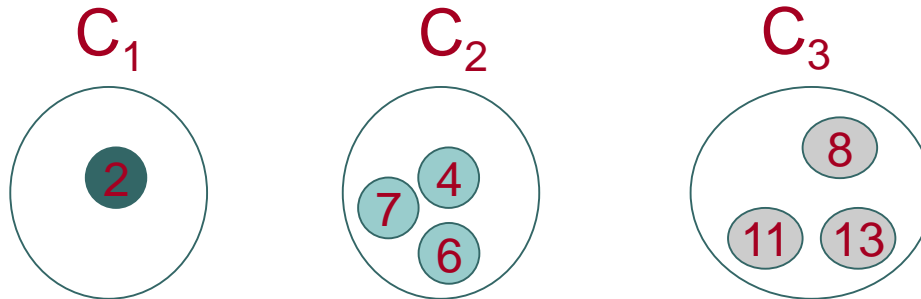
➡ 7 passe au cluster C_2 : les autres objets restent dans leurs clusters.

- On aura:

$$C_1 = \{2\}, \mu_1 = 2,$$

$$C_2 = \{4, 6, 7\}, \mu_2 = 17/3 = 5.66,$$

$$C_3 = \{8, 11, 13\}, \mu_3 = 32/3 = 10.66$$



Exemple: K-moyenne (4)

- $d(8, \mu_2) < d(8, \mu_3)$

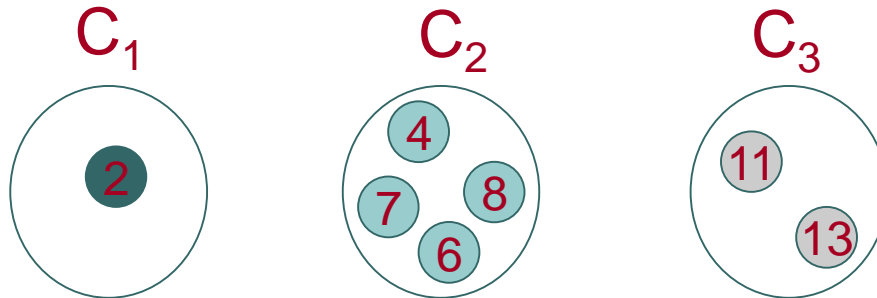
➡ 8 passe au cluster C_2 : les autres objets restent dans leurs clusters.

- On aura:

$$C_1 = \{2\}, \mu_1 = 2,$$

$$C_2 = \{4, 6, 7, 8\}, \mu_2 = 25/4 = 6.25,$$

$$C_3 = \{11, 13\}, \mu_3 = 24/2 = 12$$



Exemple: K-moyenne (5)

- $d(4, \mu_1) < d(4, \mu_2)$

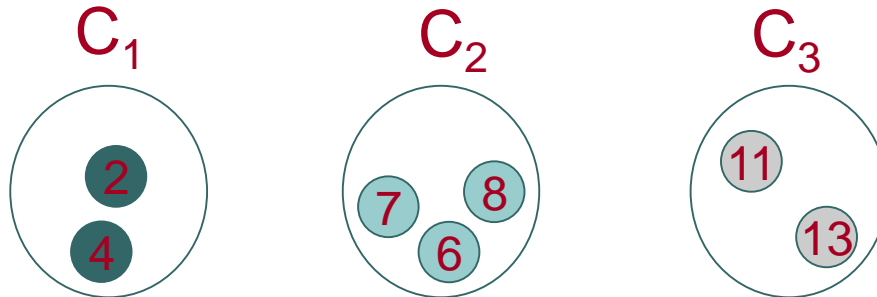
➡ 4 passe au cluster C_1 : les autres objets restent dans leurs clusters.

- On aura:

$$C_1 = \{2, 4\}, \mu_1 = 3,$$

$$C_2 = \{6, 7, 8\}, \mu_2 = 21/3 = 7,$$

$$C_3 = \{11, 13\}, \mu_3 = 24/2 = 12$$



La partition est stable.



Complexité

- La complexité de K-moyenne est $O(NKId)$.
 - N : Nombre d'objets.
 - K : Nombre de clusters.
 - I : Nombre d'itérations.
 - d : Nombre d'attributs.



Attention !!

- Besoin de fixer K à l'avance.
- Importance du choix des partitions initiales.
- Problèmes avec les outliers :
 - Points extrêmes en dehors des clusters.
 - Faussent les moyennes et par conséquent les centres.
- K-moyenne a des problèmes quand les clusters ont:
 - différentes tailles.
 - différentes densités.
 - des formes non globulaires



Choix du K

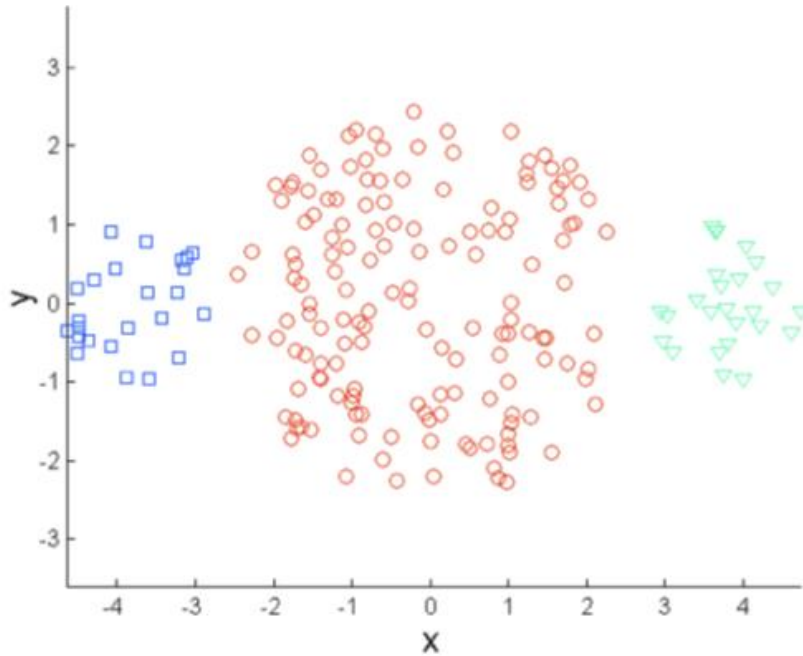
- Choix du nombre de clusters K
 - dépend de l'utilisateur.
 - Des méthodes pour déterminer K, chercher une meilleure partition, imposer de contraintes sur la densité des clusters, etc).
 - ...



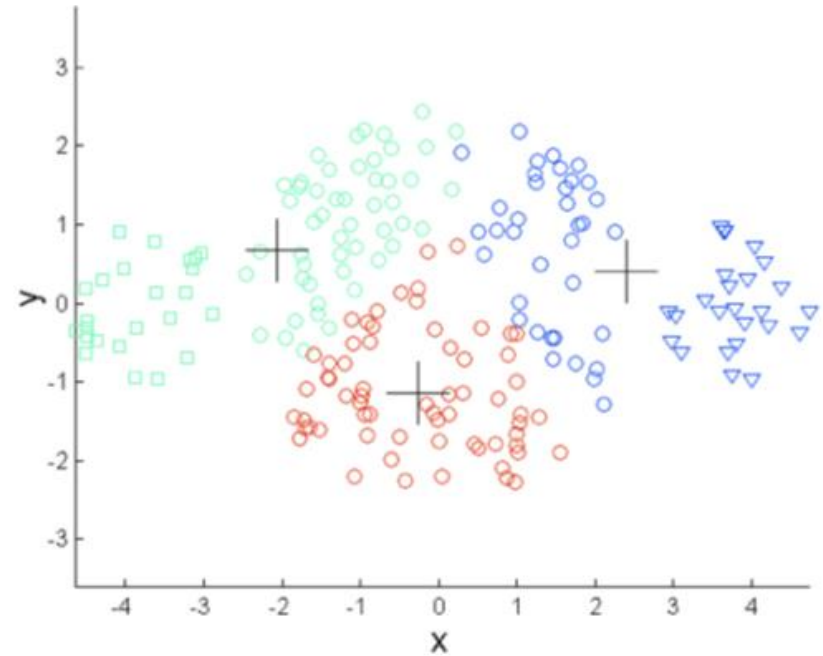
Problème d'initialisation

- Faire plusieurs expérimentations avec différents clusters initiaux et choisir de la meilleure configuration..
- Proposer des méthodes de sélection de partitions.
- Utilisation du clustering hiérarchique.
- Post-traitement
- ...

Clusters de tailles différentes

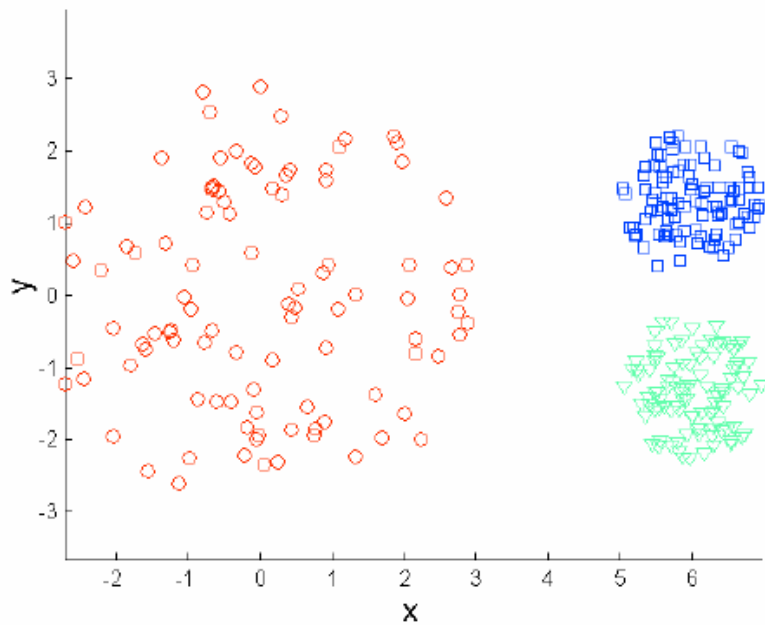


Points originaux

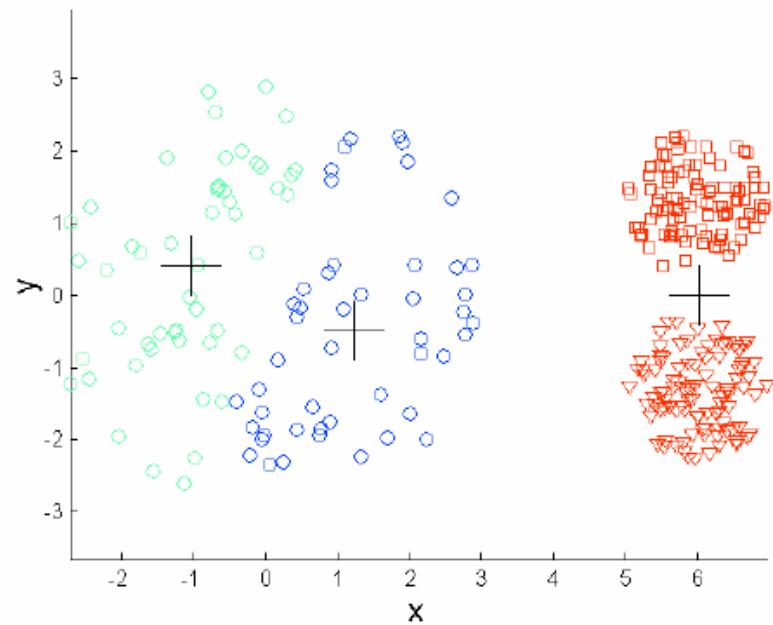


K-means (3 clusters)

Clusters de densités différentes

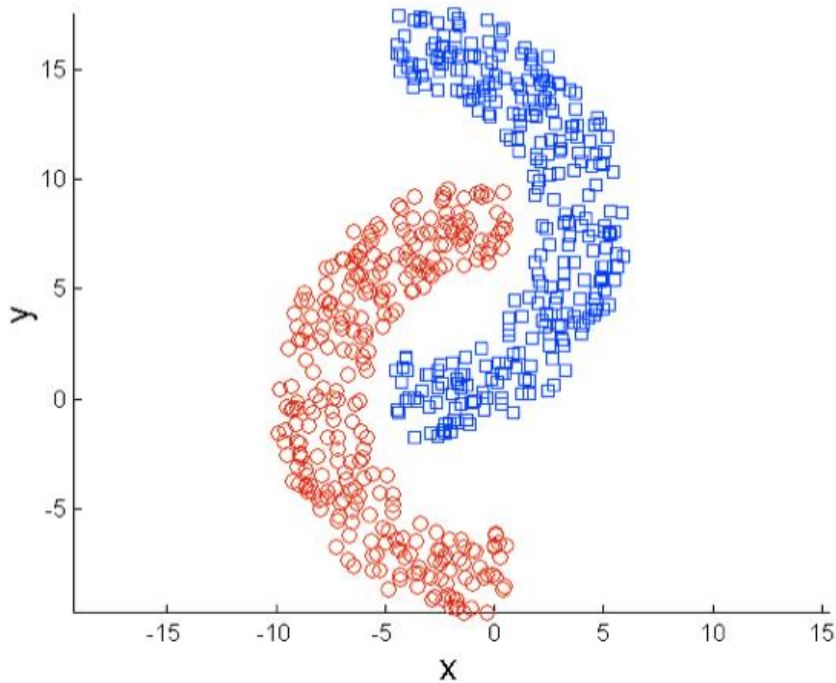


Points originaux

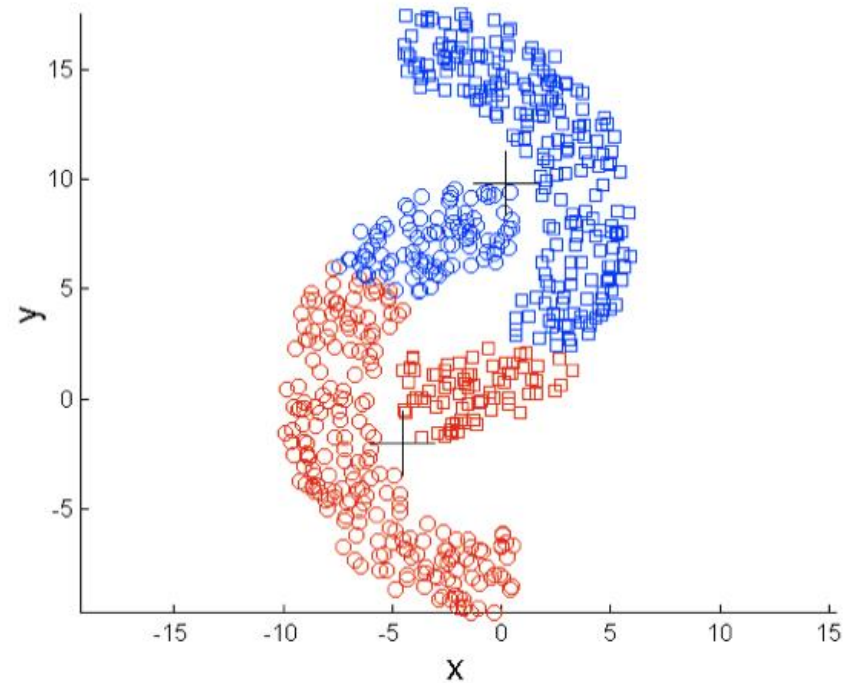


K-means (3 clusters)

Clusters de formes non globulaires



Points originaux



K-means (2 clusters)



A noter

- K-moyenne est appelée aussi **méthode des centres mobiles**.
- Si on a plusieurs attributs
⇒ Nécessité de normaliser les échelles des différents attributs.
- Plusieurs variantes de K-moyenne :
 - Sélection des k clusters initiaux.
 - Mesure de la distance utilisée.
 - Calcul de la moyenne des clusters.



K-modes

K-modes (Huang 1997)

- Traitement des données catégoriques.
- Au lieu de la moyenne, on va utiliser le mode.
- Utilisation plutôt du simple matching.
- Utilisation d'une méthode basée sur les fréquences pour mettre à jour les modes.

Exemple: K-modes (1)

K=3,

Initialisation

$C1=\{O1\}$; $C2=\{O2\}$; and $C3=\{O3\}$

Les trois clusters sont:

$C1=(\text{Finance}, \text{Élevé}, A)$;

$C2=(\text{Finance}, \text{Faible}, B)$;

$C3=(\text{Marketing}, \text{Moyen}, C)$.

	Department	Revenu	Catégorie
O1	Finance	Elevé	A
O2	Finance	Faible	B
O3	Marketing	Moyen	C
O4	Comptabilité	Moyen	C
O5	Marketing	Faible	B
O6	Finance	Elevé	A
O7	Comptabilité	Faible	B
O8	Marketing	Moyen	C

Exemple: K-modes (2)

c'est le simple
matching

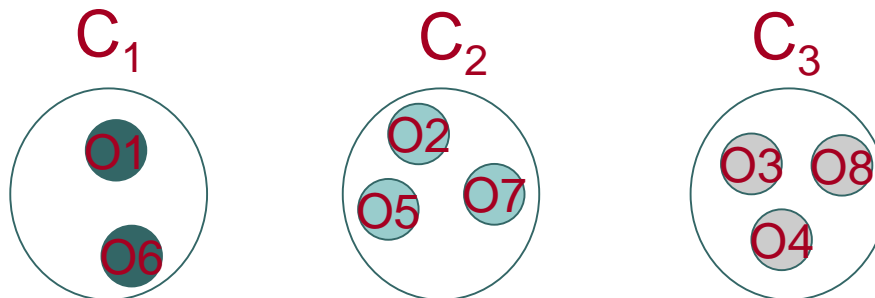
- Pour chaque objet, calculer $d(O_i, C_l)$, $l=1,2,3$

O4 est affecté à C3 puisque $d(O4, C_3)$ est minimale.

O5 est affecté à C2 puisque $d(O5, C_2)$ est minimale.

.

.



Exemple: K-modes (3)

- Mise à jour des modes

$C_1 = (\text{Finance}, \text{Élevé}, A);$

$C_2 = (\text{Finance}, \text{Faible}, B)$ ou $C_2 = (\text{Marketing}, \text{Faible}, B)$ ou
 $C_2 = (\text{Comptabilité}, \text{Faible}, B);$

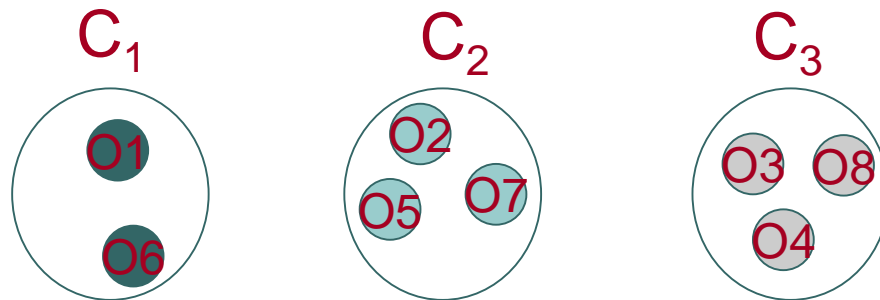
$C_3 = (\text{Marketing}, \text{Moyen}, C).$

Choix arbitraire



Recalcul des dissimilarités des objets \Rightarrow Les objets n'ont pas changé de clusters.

La partition est stable.





K-prototypes

K-prototypes (Huang 1997)

- Traitement des données mixtes (catégoriques et numériques).

- Un vecteur représentatif (prototype) pour un cluster

$C_i = \{C_{i1}, C_{i2}, \dots, C_{im}\}$; m : nombre d'attributs.

$$C_{ij} = \begin{cases} \text{Moyenne si l'attribut est numérique} \\ \text{La plus grande fréquence si l'attribut est catégorique} \end{cases}$$

- La distance des données mixtes

$$\text{dist} = \gamma \cdot \text{dist}_N + (1-\gamma) \cdot \text{dist}_C$$

dist_N : distance Euclidienne

dist_C : Matching simple

γ : coefficient de pondération



Travail à faire

Soit un ensemble des 8 points suivants (A, B,..., et H). Ces points (objets) sont dans un espace à deux dimensions, on veut constituer deux groupes de points.

Appliquer la méthode K-moyenne (K-means) sur l'ensemble des points ci-joint, et ce en détaillant les différentes étapes.

Il est à noter que la distance à utiliser est la distance euclidienne et que la partition initiale des clusters est la suivante :

$C1 = \{B\}$; $C2 = \{D\}$

	Att1	Att2
A	1	3
B	2	2
C	2	3
D	2	4
E	4	2
F	5	2
G	6	2
H	7	3 ₄₈



Solution (1)

Etape 1

$C1 = \{B\}, C2 = \{D\}$

$$d(A, C1) = ((1-2)^2 + (3-2)^2)^{1/2} = 1,41$$

$$d(A, C2) = ((1-2)^2 + (3-4)^2)^{1/2} = 1,41$$

$$d(C, C1) = ((2-2)^2 + (3-2)^2)^{1/2} = 1$$

$$d(C, C2) = ((2-2)^2 + (3-4)^2)^{1/2} = 1$$

$$d(E, C1) = ((4-2)^2 + (2-2)^2)^{1/2} = 2$$

$$d(E, C2) = ((4-2)^2 + (2-4)^2)^{1/2} = 2,82$$

$$d(F, C1) = ((5-2)^2 + (2-2)^2)^{1/2} = 3$$

$$d(F, C2) = ((5-2)^2 + (2-4)^2)^{1/2} = 3,6$$

$$d(G, C1) = ((6-2)^2 + (2-2)^2)^{1/2} = 4$$

$$d(G, C2) = ((6-2)^2 + (2-4)^2)^{1/2} = 4,47$$

$$d(H, C1) = ((7-2)^2 + (3-2)^2)^{1/2} = 5,1$$

$$d(H, C2) = ((7-2)^2 + (3-4)^2)^{1/2} = 5,1$$



Solution (2)

Etape 2

$C1 = \{A, B, C, E, F, G, H\} = (27/7, 17/7)$ $C2 = \{D\} = (2, 4)$

$d(A, C1) = 2,91$

$d(A, C2) = 1,41$

$d(B, C1) = 1,9$

$d(B, C2) = 2$

$d(C, C1) = 1,94$

$d(C, C2) = 1$

$d(D, C1) = 2,43$

$d(D, C2) = 0$

$d(E, C1) = 0,45$

$d(E, C2) = 2,83$

$d(F, C1) = 1,22$

$d(F, C2) = 3,6$

$d(G, C1) = 2,18$

$D(G, C2) = 4,47$

$d(H, C1) = 3,19$

$D(H, C2) = 5,1$



Solution (3)

Etape 3

$C1 = \{B, E, F, G, H\} = (24/5, 11/5)$ $C2 = \{A, C, D\} = (5/3, 10/3)$

$d(A, C1) = 3,88$

$d(A, C2) = 0,75$

$d(B, C1) = 2,8$

$d(B, C2) = 1,37$

$d(C, C1) = 2,91$

$d(C, C2) = 0,47$

$d(D, C1) = 3,33$

$d(D, C2) = 0,75$

$d(E, C1) = 0,82$

$d(E, C2) = 2,69$

$d(F, C1) = 0,28$

$d(F, C2) = 3,59$

$d(G, C1) = 1,22$

$d(G, C2) = 4,53$

$d(H, C1) = 2,34$

$d(H, C2) = 5,34$



Solution (4)

Etape 4

$C1 = \{E, F, G, H\} = (22/4, 9/4)$ $C2 = \{A, B, C, D\} = (7/4, 3)$

$d(A, C1) = 4,56$

$d(A, C2) = 0,75$

$d(B, C1) = 3,51$

$d(B, C2) = 1,03$

$d(C, C1) = 3,58$

$d(C, C2) = 0,25$

$d(D, C1) = 3,91$

$d(D, C2) = 1,03$

$d(E, C1) = 1,52$

$d(E, C2) = 2,46$

$d(F, C1) = 0,56$

$d(F, C2) = 3,4$

$d(G, C1) = 0,56$

$d(G, C2) = 4,37$

$d(H, C1) = 1,68$

$d(H, C2) = 5,25$

Partition stable $C1 = \{E, F, G, H\}$, $C2 = \{A, B, C, D\}$



Méthodes hiérarchiques

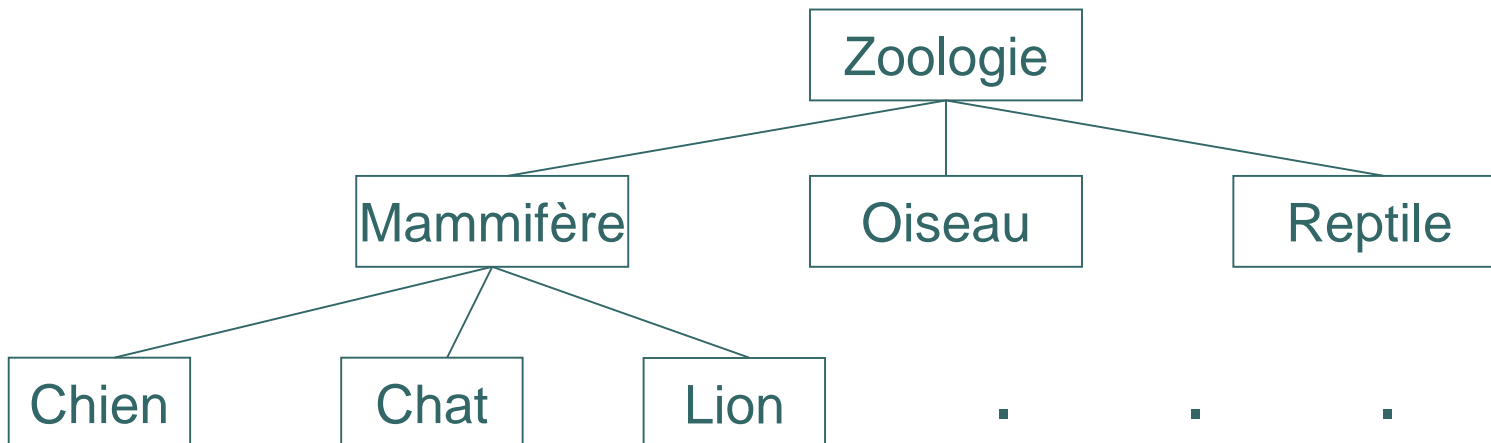
Clustering hiérarchique

- Certains objets peuvent avoir naturellement plusieurs niveaux de regroupement.



- Réaliser plusieurs niveaux de fonctionnement.
- Puis utiliser le niveau nécessaire à l'application.

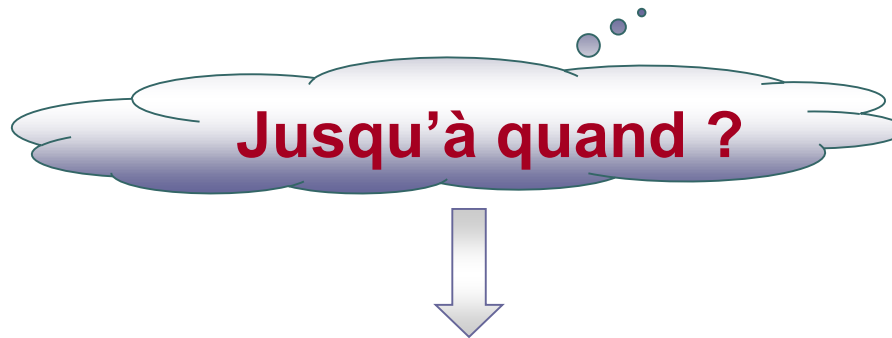
Exploitation de ces niveaux
dans la classification



Méthode hiérarchique: Par divisions

Par divisions = top-down

- Tous les objets constituent un unique cluster.
- Séparer les objets (clusters) les plus dissimilaires (grande distance).

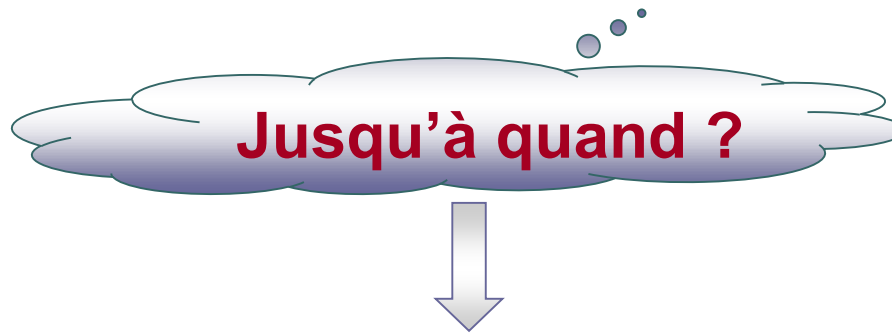


- Tous les objets sont des concepts feuilles.
- Ou une condition est vérifiée (par exemple obtenir K clusters).

Méthode hiérarchique: Par agglomérations

Par agglomérations (CHA) = Bottom-up

- Chaque objet constitue un cluster.
- Regrouper les clusters les plus proches (distance) en un nouveau cluster.



- Arriver à un concept sommet.
- Ou une condition est vérifiée (par exemple on arrive au nombre K de clusters).

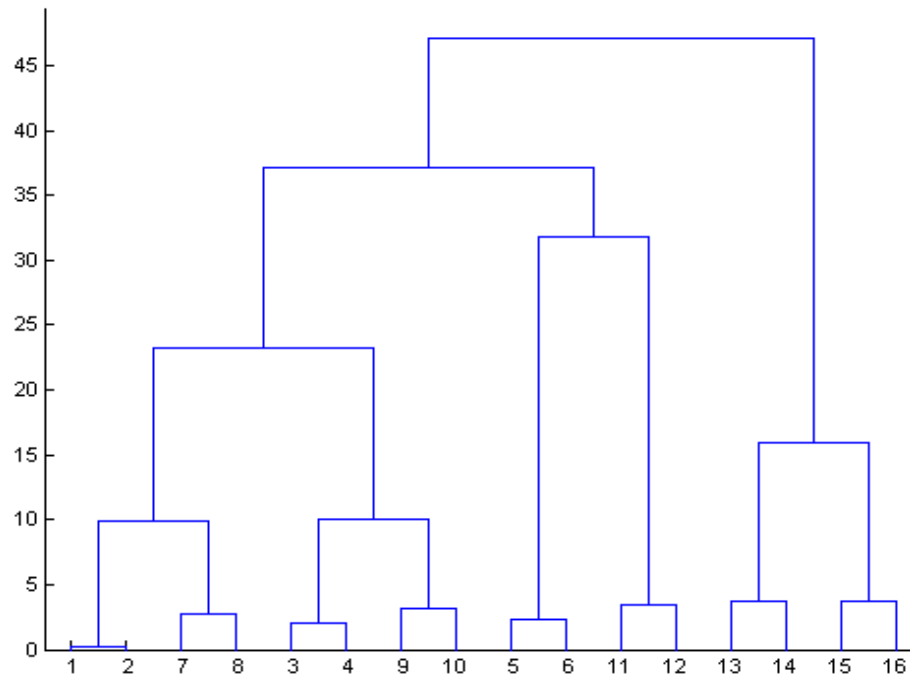


CHA - principe

- Chaque point ou cluster est progressivement "absorbé" par le cluster le plus proche.
- Algorithme
 - Initialisation
 - Chaque objet est placé dans son propre cluster.
 - Calcul de la matrice de ressemblance M entre chaque couple de clusters (ici les points).
 - Répéter
 - Sélection dans M des deux clusters les plus proches C_i et C_j .
 - Fusion de C_i et C_j par un cluster C_G plus général.
 - Mise à jour de M en calculant la ressemblance entre C_G et les clusters existants à l'aide de la mesure de distance entre clusters.
 - Jusqu'à la fusion des deux derniers clusters
- Il peut y avoir d'autres conditions d'arrêt de la fusion.

Dendrogramme

- Dendrogramme : représentation (arbre) des fusions successives du clustering hiérarchique.
- Hauteur d'un cluster dans le dendrogramme est généralement la similarité (ça peut être distance) entre les deux clusters avant la fusion.



Distances entre les clusters

Soient deux clusters C_1 et C_2

- Distance minimale (plus proche voisin) :

$$D_{\min}(C_1, C_2) = \min\{d(x_i, x_j), x_i \in C_1, x_j \in C_2\}.$$

- Distance maximale (diamètre maximale) :

$$D_{\max}(C_1, C_2) = \max\{d(x_i, x_j), x_i \in C_1, x_j \in C_2\}.$$

- Distance moyenne : $D_{\text{moy}}(C_1, C_2) = \frac{\sum_{x_i \in C_1} \sum_{x_j \in C_2} d(x_i, x_j)}{|C_1||C_2|}$

- Distance de centre de gravité : $D_{\text{cg}}(C_1, C_2) = d(\mu_1, \mu_2).$

Avec μ_1 et μ_2 les centres de gravité respectivement des clusters C_1 et C_2 .

- Distance de Ward : $D_W(C_1, C_2) = \sqrt{\frac{|C_1||C_2|}{|C_1| + |C_2|}} d(\mu_1, \mu_2)$

CHA: Métrique

- Trouver la métrique entre les clusters la plus proche de la métrique utilisée entre les individus : min, max, moyenne,
- Saut minimal (Single linkage) : se base sur $D_{\min}(C_1, C_2)$, distance entre deux éléments les plus proches de chaque cluster.
 - ➡
 - Tendance à produire des classes générales (par effet de chaînage).
 - Sensibilité aux individus bruités.
- Saut maximal (complete linkage): se base sur $D_{\max}(C_1, C_2)$, distance entre les deux points les plus éloignés des deux clusters.
 - ➡
 - Tendance à produire des classes spécifiques (on ne regroupe que des classes très proches).
 - Sensibilité aux individus brutes.
- Distance moyenne (average linkage) : se base sur $D_{\text{moy}}(C_1, C_2)$.
 - ➡
 - Tendance à produire des classes de variance proche.
- Barycentre : se base sur $D_{\text{cg}}(C_1, C_2)$.
 - ➡
 - Bonne résistance au bruit.



Exemple

- Regrouper ensemble tous les objets dont la distance est inférieure ou égale à 4.

	P1	P2	P3	P4
P1	0			
P2	1	0		
P3	7	5	0	
P4	2	3	6	0

- Le saut minimal va être utilisé dans cet exemple.

Étape 1

- On a un ensemble constitué de 4 objets:
 $\{P1, P2, P3, P4\}$
- On va essayer de constituer le premier cluster regroupant les objets les plus proches.

	P1	P2	P3	P4
P1	0			
P2	1	0		
P3	7	5	0	
P4	2	3	6	0



	$\{P1, P2\}$	P3	P4
$\{P1, P2\}$	0		
P3	5	0	
P4	2	6	0

Premier cluster $\{P1, P2\}$
 $\{\{P1, P2\}, P3, P4\}$

Étape 2

- On va chercher le deuxième cluster.
- Les objets les plus proches sont $\{P1, P2\}$ et $P4$.
- On recalcule la matrice de distances:

	$\{P1, P2\}$	P3	P4
$\{P1, P2\}$	0		
P3	5	0	
P4	2	6	0

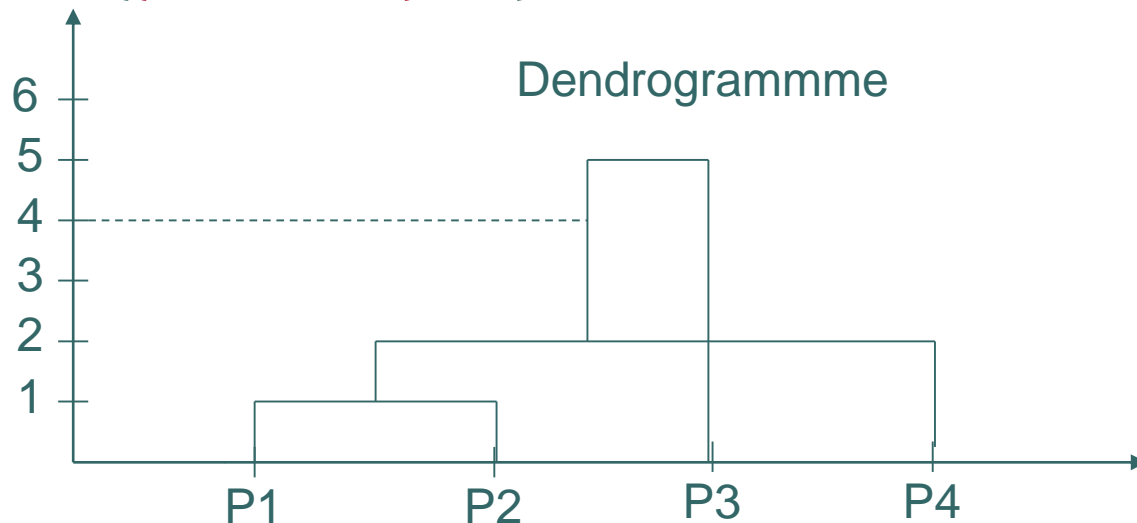


	$\{P1, P2, P4\}$	P3
$\{P1, P2, P4\}$	0	
P3	5	0

Deuxième cluster $\{P1, P2, P4\}$
 $\{\{P1, P2, P4\}, P3\}$

Exemple

- $D(\{P1, P2, P4\}, P3) = 5 > 4$
- On a différents niveaux de clusters:
 1. On a $\{P1, P2, P3, P4\}$
 2. Puis $\{\{P1, P2\}, P3, P4\}$
 3. Puis $\{\{P1, P2, P4\}, P3\}$





Travail à faire

On suppose qu'on a six villes (V1, V2, V3, V4, V5, V6) telles que la matrice suivante représente les distances en kilomètres par paire de villes.

	V1	V2	V3	V4	V5	V6
V1	0	662	877	255	412	996
V2	662	0	295	468	268	400
V3	877	295	0	754	564	138
V4	255	468	754	0	219	869
V5	412	268	564	219	0	669
V6	996	400	138	869	669	0

Appliquer l'algorithme de clustering hiérarchique par agglomérations sur cet ensemble de villes, en se basant sur la matrice des distances ci-dessus. Il est à noter que la métrique utilisée entre les clusters est le saut minimal. La condition d'arrêt est quand toutes les distances restantes entre les villes sont strictement supérieures à 300 ou quand on arrive à un seul cluster.

Solution (1)

	V1	V2	V3	V4	V5	V6
V1	0	662	877	255	412	996
V2	662	0	295	468	268	400
V3	877	295	0	754	564	138
V4	255	468	754	0	219	869
V5	412	268	564	219	0	669
V6	996	400	138	869	669	0

	V1	V2	V3/V6	V4	V5
V1	0	662	877	255	412
V2	662	0	295	468	268
V3/V6	877	295	0	754	564
V4	255	468	754	0	219
V5	412	268	564	219	0

Solution (2)

	V1	V2	V3/V6	V4/V5
V1	0	662	877	255
V2	662	0	295	268
V3/V6	877	295	0	564
V4/V5	255	268	564	0

	V1/V4/V5	V2	V3/V6
V1/V4/V5	0	268	564
V2	268	0	295
V3/V6	564	295	0

	V1/V4/V5/V2	V3/V6
V1/V4/V5/V2	0	295
V3/V6	295	0

Voilà à la fin on aura un seul cluster V1/V4/V5/V2/V3/V6.



Clustering incrémental



Clustering incrémental

- Les bases de données sont dynamiques:

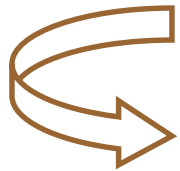
L'ensemble des objets

L'ensemble des attributs

.

.

} Evoluent dans le temps



« Incremental Clustering » pour résoudre le problème de la maintenance des clusters.



K-means séquentiels

- Adaptation de la méthode K-means lorsque les objets arrivent au fur et à mesure.
- Initialiser $\mu_1, \mu_2, \dots, \mu_k$.
- Initialiser n_1, n_2, \dots, n_k à 0.
- Répéter
 - Acquérir x
 - Affectation de chaque point à son cluster le plus proche
 $C_t \leftarrow x$, tel que $t = \arg \min_k d(x, \mu_k)$
 - Incrémenter n_t .
 - Recalculer le centre μ_t de ce cluster.

$$\mu_t = \mu_t + \frac{1}{n_t} (x - \mu_t)$$



K-means Incrémentale (CBIC) (1)

G. Serran, A. Campan. Core Based Incremental Clustering. 2005

L'ensemble des attributs est dynamique et évolue dans le temps

La partition initiale : résultat de K-means standard ou CBIC avant l'extension.




A l'arrivée d'un nouvel attribut.



Recalculer les clusters après extension partant de cette partition initiale.

K-means Incrémentale (CBIC) (2)

- m attributs \rightarrow m+1 attributs
 - $O_i \rightarrow O_i'$, $1 \leq i \leq n$ (n objets)
 - $K_j \rightarrow K_j'$, $1 \leq j \leq p$ (p clusters)

 Dans quelles conditions un objet O_i' (après extension) est correctement classé dans son cluster courant K_j' ?



Définition

- $CORE = \{ Core_j, 1 \leq j \leq p \}$
- $Core_j$: l'ensemble des objets du cluster K_j qui sont plus proches de son centre que des autres centres des différents clusters (après extension).

K-means Incrémentale (CBIC) (3)

Partition Initiale

les objets sont
correctement classés

→ Distances minimales

A vérifier après extension

Pour chaque objet, la valeur du nouvel attribut ($m+1$)

✓ Supérieure ou égale à celle du centre de son cluster → l'objet est correctement classé, appartient au core de ce cluster.

✓ Sinon comparer ses différentes distances par rapport aux différents centres de tous les clusters.



K-means Incrémentale (CBIC) (4)

Algorithme

- Entrées:

n objets avec m attributs

n objets avec m+1 attributs

d_E la distance euclidienne

p nombre des clusters à former

F la partition antérieure (initiale)

Nombre maximum d'itérations

- Sorties:

F' la partition des n objets après extension (m+1 attributs)

K-means Incrémentale (CBIC) (5)

Début

pour chaque cluster F_j de F

calculer Core_j

$F'_j \leftarrow \text{Core}_j$

mettre à jour les centres f'_j des cores

fin pour

tant que F' change et max itérations n'est pas atteint faire

pour tout F'_j faire

$F'_j \leftarrow \{O_i', \text{ les plus proches} \}$ c.à.d $\{O_i' | \forall f'_r d(O_i', f'_j) \leq d(O_i', f'_r)\}$

fin pour

pour tout F'_j faire

calculer le centre

fin pour

fin tant que

Exemple (1)

○ K-means standard ($K = 2$) →

✓ **Modes initiaux :**

O_1 et O_2

✓ **Partition initiale (CBIC):**

	Age	Nombre Enfants	Salaire
O_1	40	2	1000
O_2	75	4	600
O_3	50	3	1100
O_4	35	1	550

$F_1 = (O_1, O_3, O_4); f_1 = (41,67; 2)$

$F_2 = (O_2); f_2 = (75; 4)$

□ **Avec l'arrivée du nouvel attribut :**

$f_1' = (41,67; 2; 883,34)$

$f_2' = (75; 4; 600)$

Exemple (2)

- Comparer les valeurs de cet attribut pour définir les cores :

Pour F_1 :

$$1100 \geq 883,34$$

$$1000 \geq 883,34$$

Par contre $550 < 883,34$

$$\text{Core}_1 = (O_1, O_3)$$

Pour F_2 :

$$600 \geq 600$$

$$\text{Core}_2 = (O_2)$$

$$F_1' = \text{Core}_1 = (O_1, O_3) ; f_1' = (45; 2,5; 1050)$$

$$F_2' = \text{Core}_2 = (O_2) ; f_2' = (75; 4; 600)$$

Exemple (3)

- Calculer les distances de chaque objet des deux centres f_1' et f_2' pour les affecter :



$O_1, O_3 \rightarrow F_1'$

$O_2, O_4 \rightarrow F_2'$



$f_1' = (45; 2,5; 1050)$

$f_2' = (55; 2,5; 575)$

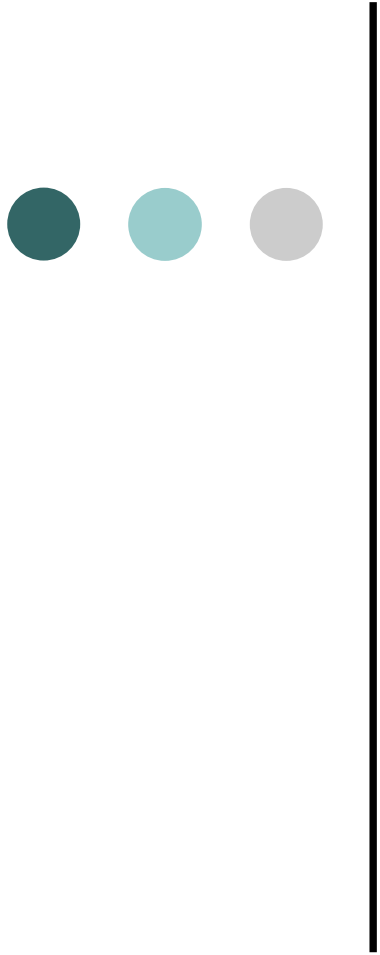
La partition est stable F' ne change plus

Il vaut mieux normaliser.



Conclusion

- Utilisables dans plusieurs applications.
- Possibilité de vérification des résultats de clustering par d'autres techniques de classification supervisée.
- Plusieurs extensions de k-means et k-modes:
 - Soft K-means.
 - Fuzzy K-means.
 - Belief K-modes.
 - Possibilistic K-modes.



TD



Exercice 1

Soit un ensemble des 8 points suivants (A, B,..., et H). Ces points (objets) sont dans un espace à deux dimensions, on veut constituer deux groupes de points.

Appliquer la méthode K-moyenne (K-means) sur l'ensemble des points ci-joint, et ce en détaillant les différentes étapes.

Il est à noter que la distance à utiliser est la distance euclidienne et que la partition initiale des clusters est la suivante :

$C1 = \{B\}$; $C2 = \{D\}$

	Att1	Att2
A	1	3
B	2	2
C	2	3
D	2	4
E	4	2
F	5	2
G	6	2
H	7	3



Exercice 2

Choisir les bases Iris et Unbalanced de Weka

Pour chaque base, tester la méthode K-means, en faisant varier:

- la mesure de distance (distance euclidienne, distance de Manhattan)
- Nombre de clusters K (nombre de classes de la base, puis $K=2$ et une autre valeur)
- Cluster mode (use training set, classes to cluster evaluation).

Pour chaque base, analyser les résultats trouvés avec ces différentes variations.



Exercice 3

On suppose qu'on a six villes (V1, V2, V3, V4, V5, V6) telles que la matrice suivante représente les distances en kilomètres par paire de villes.

	V1	V2	V3	V4	V5	V6
V1	0	662	877	255	412	996
V2	662	0	295	468	268	400
V3	877	295	0	754	564	138
V4	255	468	754	0	219	869
V5	412	268	564	219	0	669
V6	996	400	138	869	669	0

Appliquer l'algorithme de clustering hiérarchique par agglomérations sur cet ensemble de villes, en se basant sur la matrice des distances ci-dessus. Il est à noter que la métrique utilisée entre les clusters est le saut minimal. La condition d'arrêt est quand toutes les distances restantes entre les villes sont strictement supérieures à 300 ou quand on arrive à un seul cluster.