

Qu'est-ce que le clustering ?

- analyse de clustering
 - ♦ regroupement des objets en clusters
- un cluster : une collection d'objets
 - ♦ similaires au sein d'un même cluster
 - ♦ dissimilaires aux objets appartenant à d'autres clusters
- classification non supervisée : pas de classes prédéfinies
- Applications typiques
 - ♦ afin de mieux comprendre les données
 - ♦ comme prétraitement avant d'autres analyses

- Une bonne méthode va produire des clusters dont les éléments ont
 - ◆ une forte similarité intra-classe
 - ◆ une faible similarité inter-classe
- La qualité d'un clustering dépend de la mesure de similarité
- La qualité d'une méthode peut aussi être mesurée par sa capacité à trouver quelques ou tous les motifs intéressants

- Mise à l'échelle
- Capacité à gérer différents types d'attributs
- Découverte de clusters avec des formes arbitraires
- Besoin minimum de connaissances du domaine pour déterminer les paramètres
- Capacité à gérer le bruit et les exceptions
- Indifférent à l'ordre des données en entrée
- Nombre de dimensions
- Incorporation de contraintes par l'utilisateur
- Interprétabilité et utilisabilité

- 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 distance
(ou dissimilarité)

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

- Métrique de similarité/dissimilarité : exprimée en termes d'une fonction de distance, typiquement $d(i,j)$
- Fonction de distance dépend du type des données : binaires, nominales, ordinales ou continues
- Pondération des dimensions selon l'application et la sémantique des données
- Difficulté de définir « suffisamment similaires »
 - ♦ la réponse est très subjective

- continue sur un intervalle
 - ◆ ex : poids, taille
- binaire
- nominale
 - ◆ ex : couleur
- ordinale
- à échelle variable
 - ◆ ex : croissance exponentielle des bactéries, durée de la radioactivité
- Mixte

- Normaliser les données : s'affranchir des unités de mesures
- écart absolu à la moyenne

$$s_f = \frac{1}{n} (|x_{1f} - m_f| + |x_{2f} - m_f| + \dots + |x_{nf} - m_f|)$$

- Calculer la mesure normalisée (z-score)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

- L'utilisation de l'écart absolu est plus robuste que celle de l'écart type

- Distance de Minkowski :

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

avec $i = (x_{i_1}, x_{i_2}, \dots, x_{i_p})$ et $j = (x_{j_1}, x_{j_2}, \dots, x_{j_p})$ deux objets à p dimensions, et q un entier positif

- si $q = 1$: distance de Manhattan

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

- si $q = 2$: distance euclidienne

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

- Propriétés

- ♦ $d(i, i) = 0$
- ♦ $d(i, j) \geq 0$ (positive)
- ♦ $d(i, j) = d(j, i)$ (symétrique)
- ♦ $d(i, j) \leq d(i, k) + d(k, j)$ (inégalité triangulaire)

- table de contingence

		Objet <i>j</i>	
		1	0
Objet <i>i</i>	1	<i>a</i>	<i>b</i>
	0	<i>c</i>	<i>d</i>

- coefficient simple d'appariement (invariant, si la variable est symétrique)

$$d(i, j) = \frac{b + c}{a + b + c + d}$$

- coefficient de Jaccard (non invariant, si la variable est asymétrique)

$$d(i, j) = \frac{b + c}{a + b + c}$$

- Exemple

Nom	Sexe	Fièvre	Tousse	Test-1	Test-2	Test-3	Test-4
Jacques	M	O	N	P	N	N	N
Marie	F	O	N	P	N	P	N
Jean	M	O	P	N	N	N	N

- sexe est symétrique
- les autres sont asymétriques
- soit O et P = 1, et N = 0

$$d(\text{jacques}, \text{marie}) = \frac{0 + 1}{2 + 0 + 1} = 0.33$$

$$d(\text{jacques}, \text{jean}) = \frac{1 + 1}{1 + 1 + 1} = 0.67$$

$$d(\text{jean}, \text{marie}) = \frac{1 + 2}{1 + 1 + 2} = 0.75$$

- généralisation des valeurs binaires : plus de 2 états
- méthode 1 : appariement simple
 - ♦ m : nombre d'appariements, p : nombre total de variables

$$d(i, j) = \frac{p - m}{p}$$

- méthode 2 : utiliser un grand nombre de variables binaires
 - ♦ création d'une variable binaire pour chacun des états d'une variable nominale

- l'ordre est important : rang
- peut être traitée comme une variable continue sur un intervalle
 - ♦ remplace x_{if} par son rang $r_{if} \in \{1, \dots, M_f\}$
 - ♦ transforme chaque variable sur $[0,1]$ en remplaçant le i -ième objet de la f -ième variable

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

- ♦ calcule la dissimilarité en utilisant les méthodes de valeurs continues sur un intervalle

- mesure positive sur une échelle non linéaire, échelle exponentielle qui suit approximativement Ae^{BT} ou Ae^{-BT}
- Méthodes
 - ♦ les traiter comme des variables continues sur un intervalles : mauvais choix
 - ♦ appliquer une transformation logarithmique puis les traiter comme des variables continues sur un intervalle

$$y_{if} = \log(x_{if})$$

- ♦ les traiter comme des variables ordinales en traitant leur rang

- Les objets peuvent être décrits avec tous les types de données
 - ♦ binaire symétrique, binaire asymétrique, nominale, ordinale, ...
- Utilisation d'une formule pondérée pour combiner leurs effets

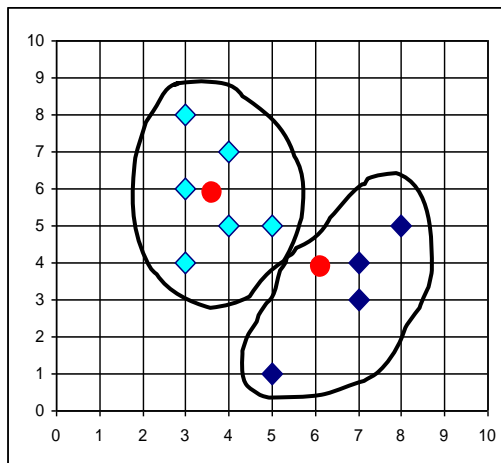
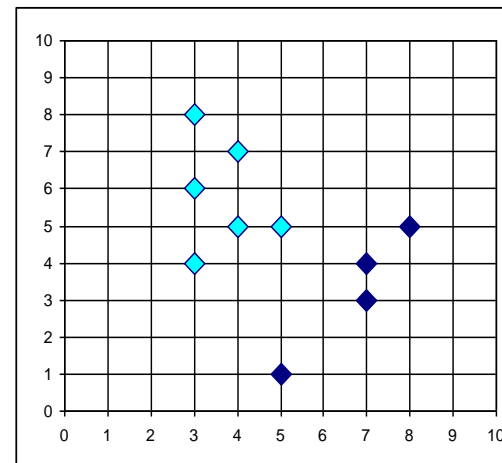
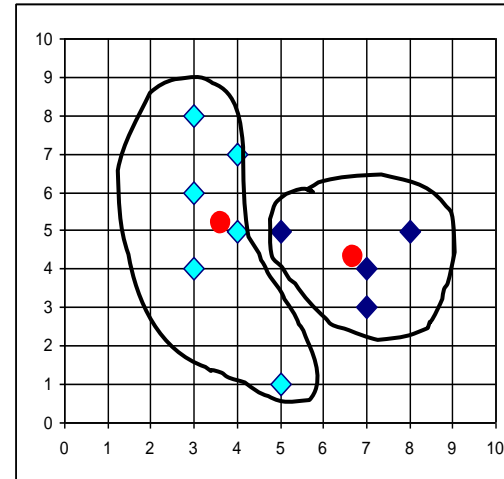
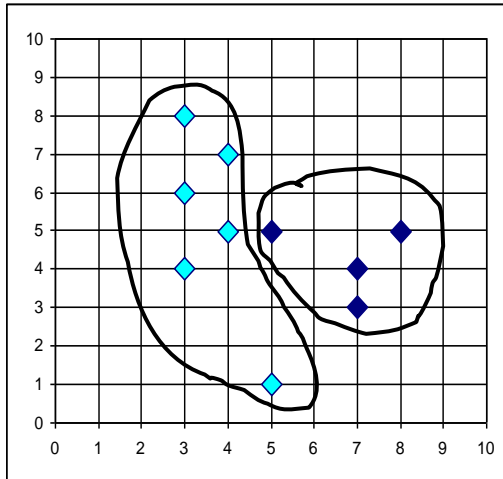
$$d(i, j) = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}}$$

- partitionnement
 - ◆ partitionne les objets et évalue les partitions
- hiérarchique
 - ◆ décomposition hiérarchique d'ensembles d'objets
- densité
 - ◆ basée sur une fonction de densité ou de connectivité
- grille
 - ◆ basée sur une structure de granularité à plusieurs niveaux

- Construire une partition de la base de données D contenant n objets en un ensemble de k clusters
- Etant donné k , trouver une partition en k clusters qui optimisent le critère de partitionnement
 - ♦ Optimum global : traiter toutes les partitions exhaustivement
 - ♦ Heuristique : *k-means* ou *k-médoïdes*
 - *k-means* : chaque cluster est représenté par son centre
 - *k-médoïdes* ou *PAM (partition around medoids)* : chaque cluster est représenté par un des objets du cluster

- 4 étapes
 1. Partitionne les objets en k ensembles non vides
 2. Calcule le centroïde de chaque partition/cluster
 3. Assigne à chaque objet le cluster dont le centroïde est le plus proche
 4. boucle en 2, jusqu'à ce les clusters soient stables.

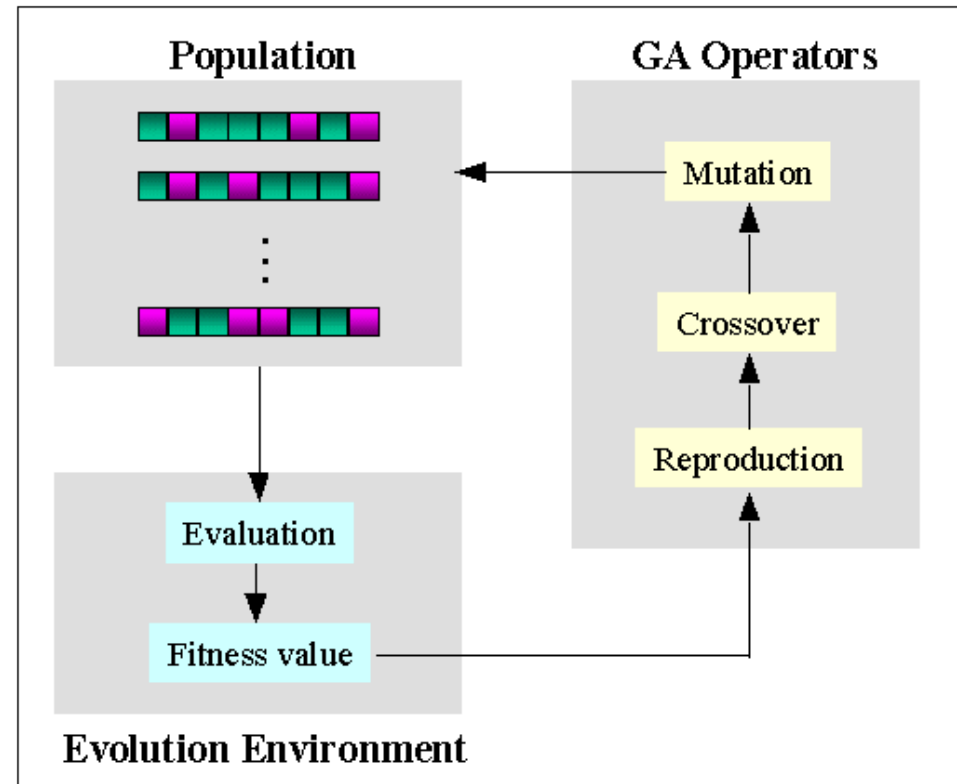
k-means, exemple



- Avantages
 - ◆ Relativement efficace : $O(tkn)$, avec n le nombre d'objets, t le nombre d'itérations et en général t et $k \ll n$
 - ◆ Termine souvent sur un optimum local. L'optimum global peut être atteint en utilisant des techniques telles que les algorithmes génétiques
- Faiblesses
 - ◆ Utilisable seulement lorsque la moyenne est définie. Que faire dans le cas de données nominales ?
 - ◆ Besoin de spécifier k à l'avance
 - ◆ Ne gère pas le bruit et les exceptions
 - ◆ Ne trouve que des clusters de forme convexe

Algorithme génétique

- Individu (0011...100)= solution. Fonction endocage/décodage
- Sélection : score de fitness
- Reproduction. Ex : tirage aléatoire (avec remise) avec probabilité d'être tirée qui dépend du score de fitness
- Opération de crossing over et de mutation
- Paramètres :
 - ♦ nombre d'individus dans la population
 - ♦ taux de mutation et de crossing over
 - ♦ fonction d'évaluation
 - ♦ nombre de générations ou seuil sur le score de fitness
- Variantes :
 - ♦ fonction de fitness qui évolue
- Voir aussi
 - ♦ programmation génétique

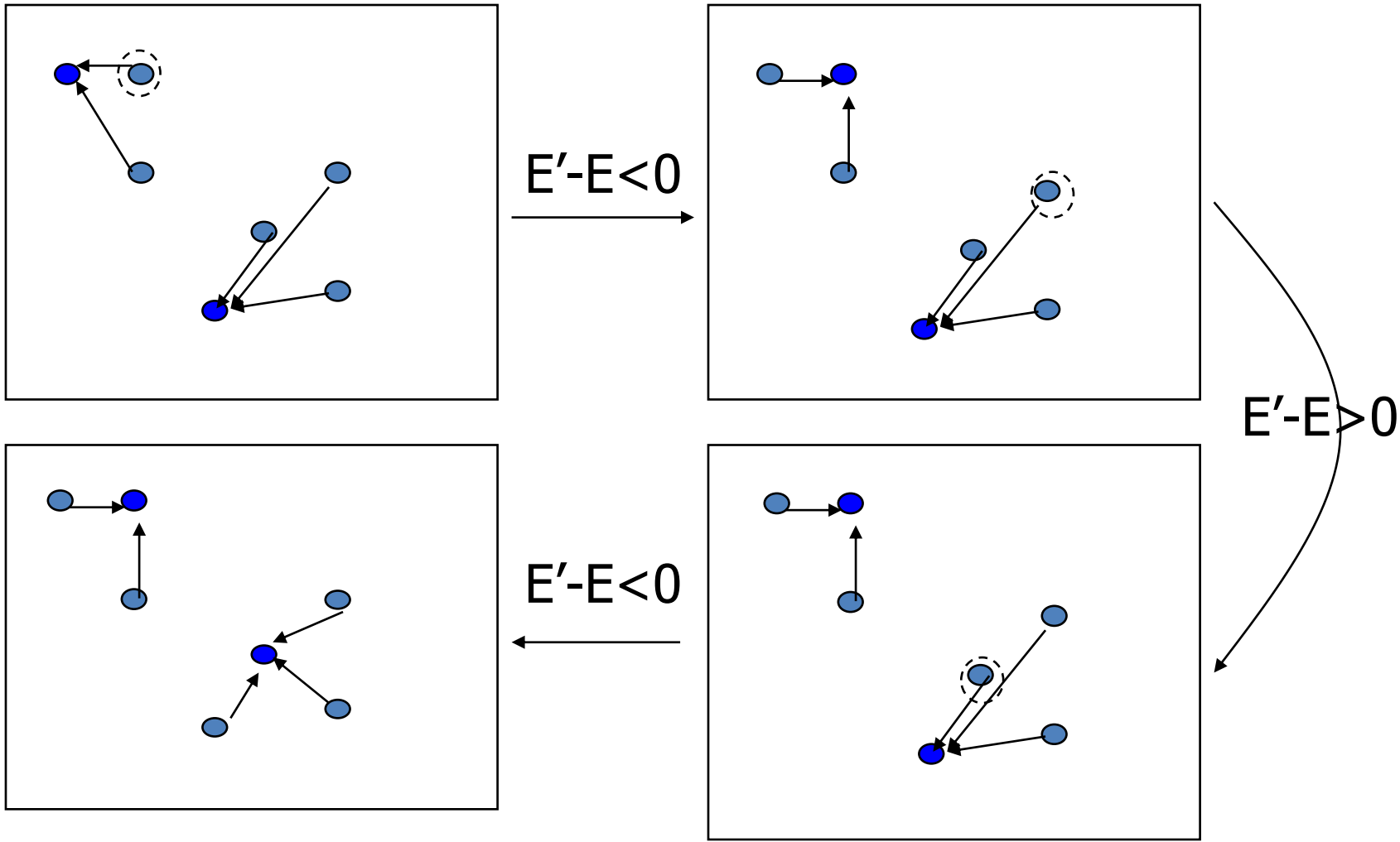


- Trouve des représentants, appelés médoïdes, dans les clusters
- PAM
 - ♦ médoïde : l'objet d'un cluster pour lequel la distance moyenne à tous les autres objets du cluster est minimale
 - ♦ critère d'erreur :
- Algorithmes

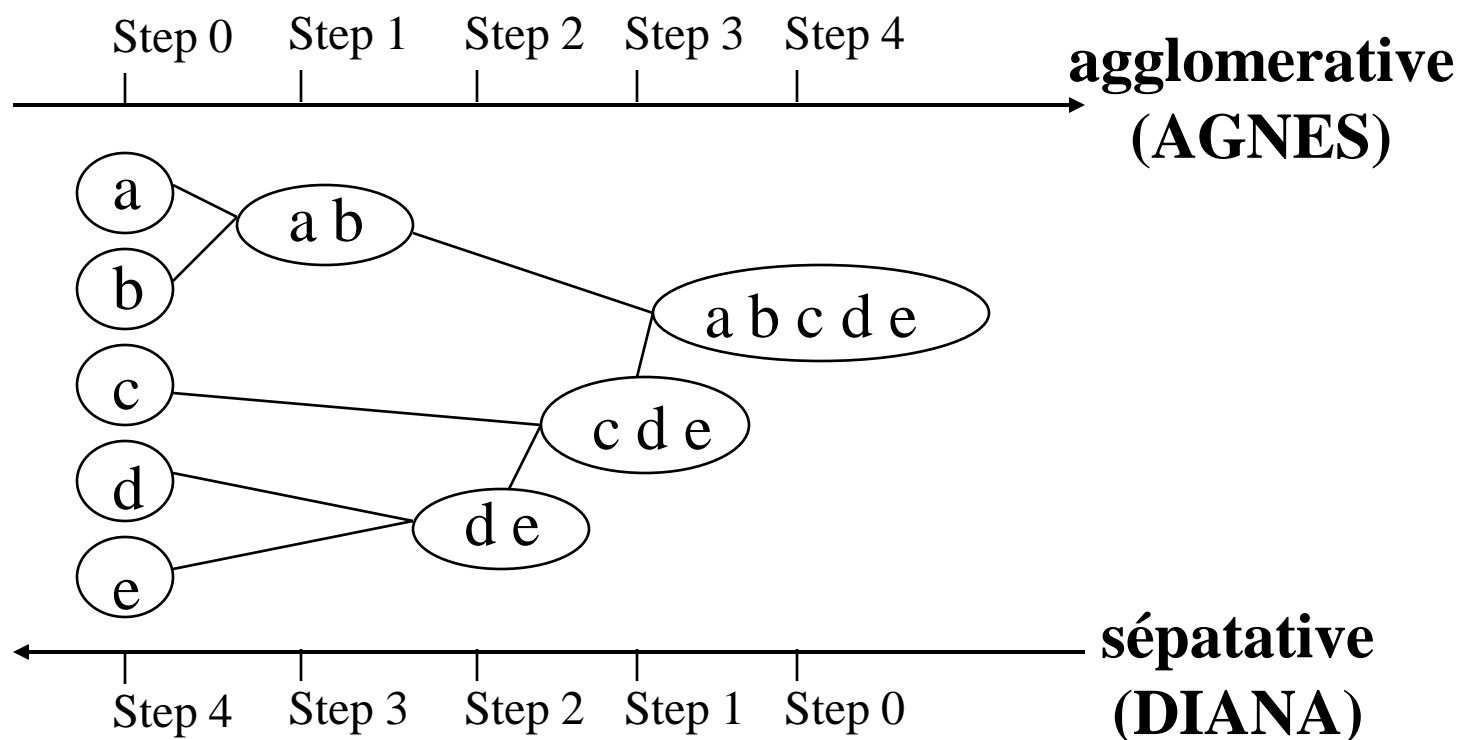
$$E = \sum_{i=1}^k \sum_{p \in C_i} d(p, m_i)^2$$

1. Sélectionner k objets arbitrairement
2. Assigner le reste des objets au médoïde le plus proche
3. Sélectionner un objet non médoïde et échanger si le critère d'erreur peut être réduit
4. Répéter 2 et 3 jusqu'à ne plus pouvoir réduire le critère d'erreur

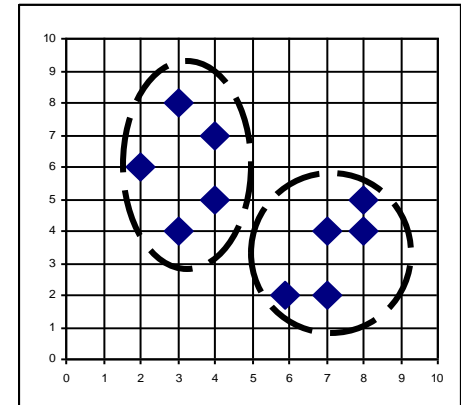
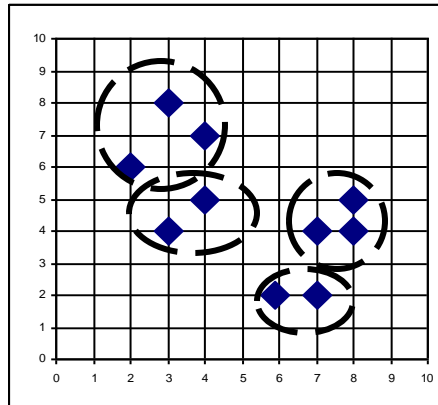
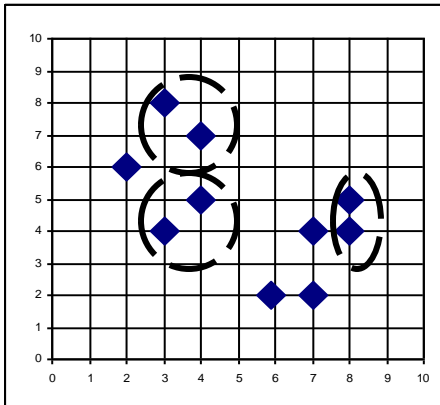
PAM. Exemple



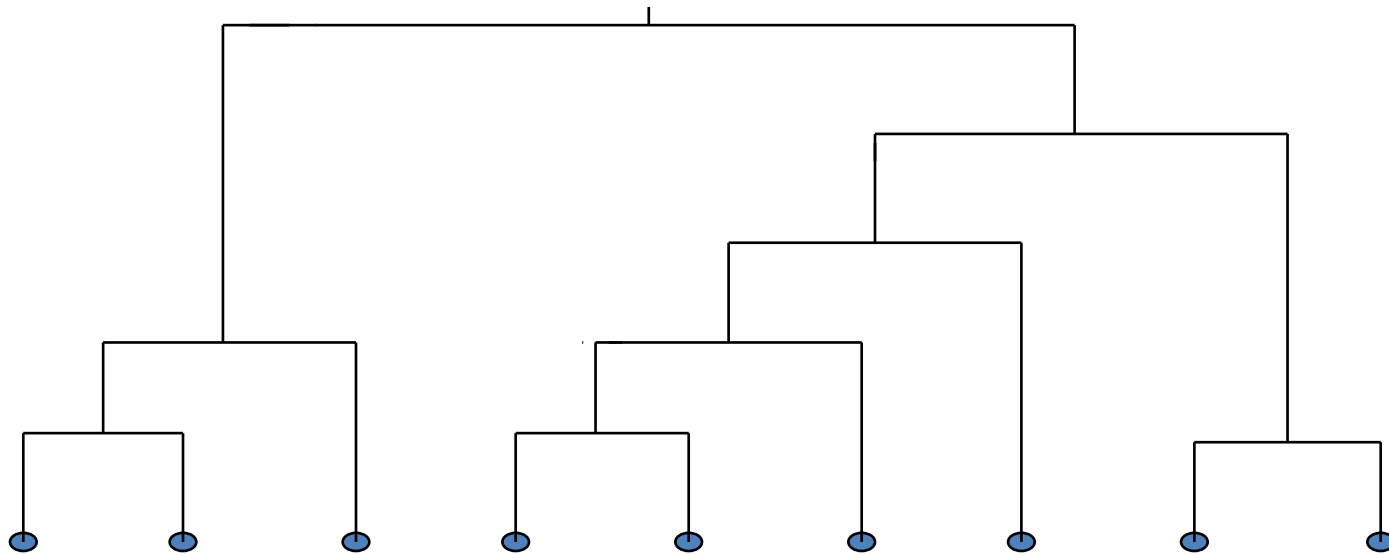
- Utilisation d'une matrice de distance : ne nécessite pas de spécifier le nombre de clusters



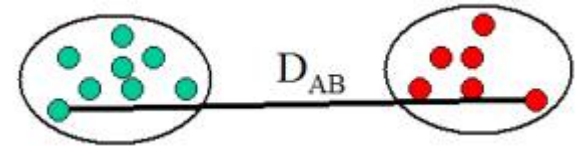
- Utilise une matrice de dissimilarité
- Fusionne les nœuds les moins dissimilaires



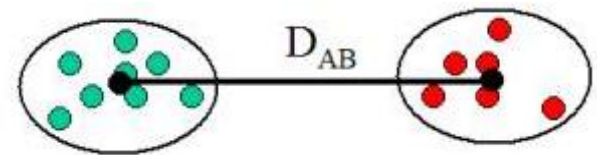
- Décompose les données en plusieurs niveaux imbriqués de partitionnement
- Un clustering est obtenu en coupant le dendrogramme au niveau choisi



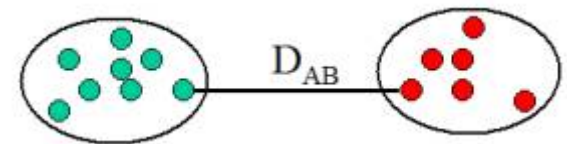
- complete linkage
 - ◆ plus petite similarité/plus grande distance entre toutes les paires de gènes entre 2 clusters



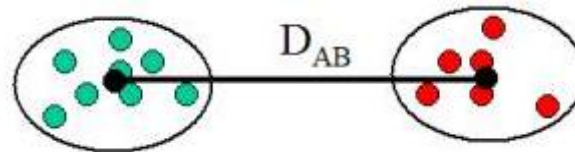
- average linkage
 - ◆ similarité moyenne entre les paires de gènes



- single linkage
 - ◆ plus grande similarité/plus petite distance entre 2 gènes de 2 clusters



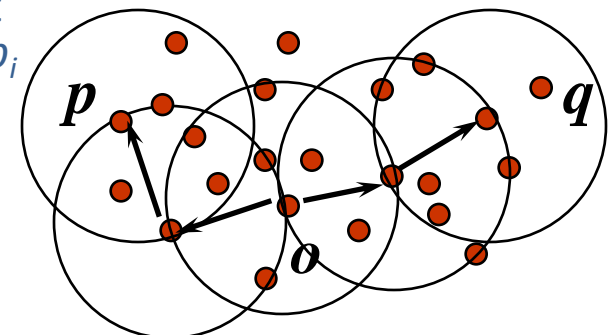
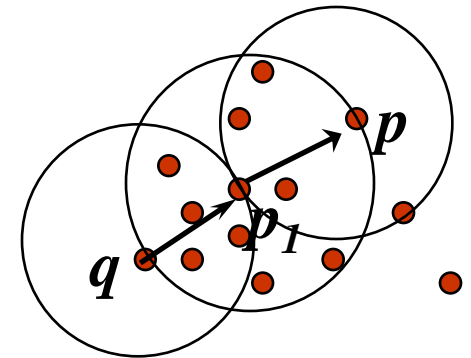
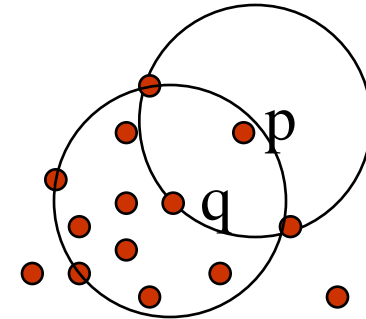
- centroïde
 - ◆ distance entre les centroïde des clusters



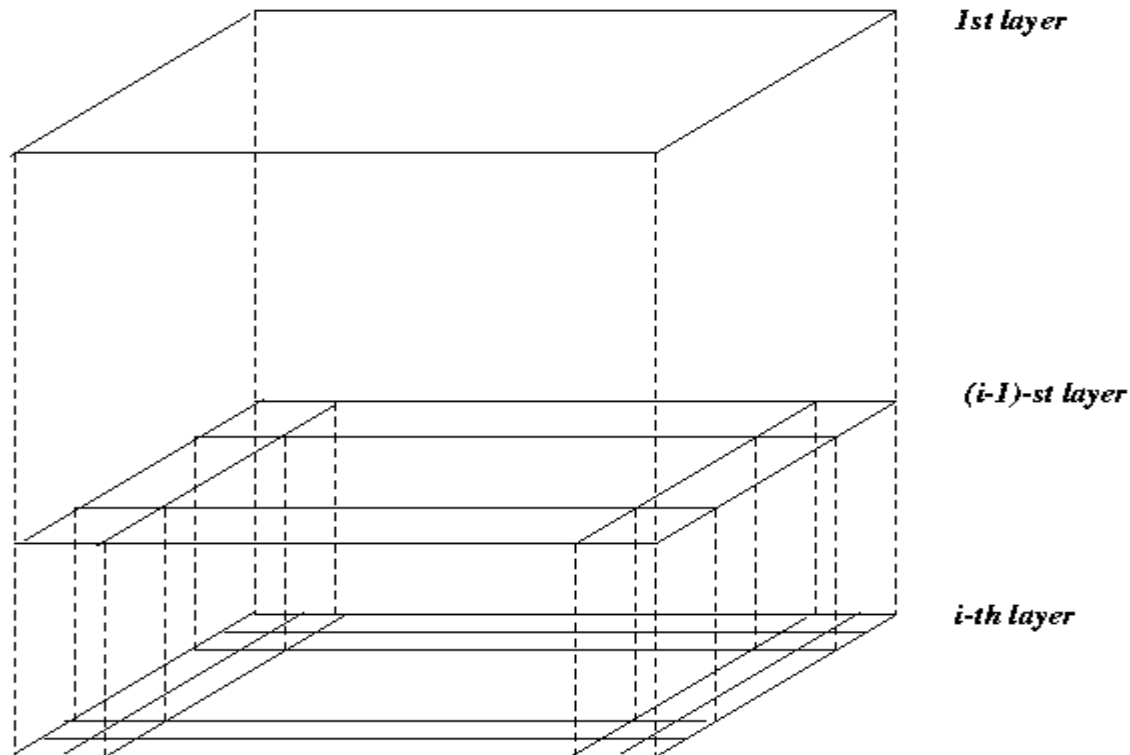
- Ward
 - ◆ distance = augmentation de la distance au carré au centroïde en fusionnant 2 clusters

Méthodes basées sur la densité

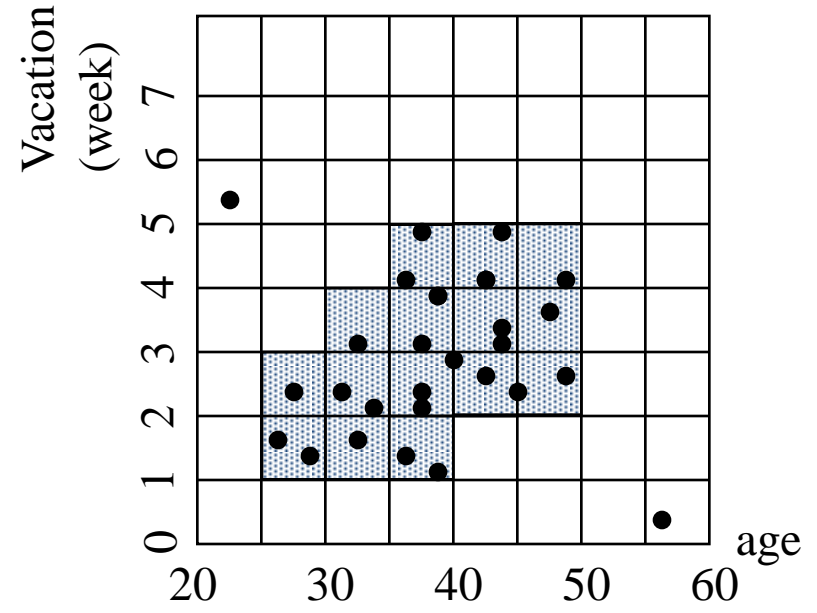
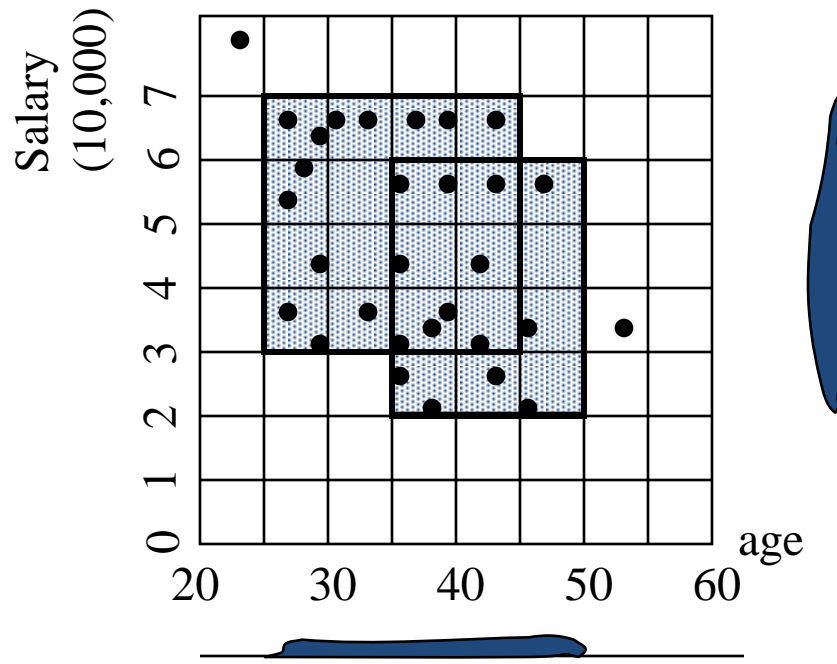
- Principales caractéristiques
 - ♦ Cluster de forme arbitraire
 - ♦ Gestion du bruit
 - ♦ Besoin d'un paramètre de densité comme critère d'arrêt
- 2 paramètres
 - ♦ Eps : rayon maximal de voisinage
 - ♦ MinPts : nombre minimal de points dans le voisinage défini par Eps
- $N_{Eps}(p) : \{ q \in D \mid \text{dist}(p,q) \leq Eps \}$
- un point p est **directement atteignable** d'un point q si
 - ♦ p appartient à $N_{Eps}(q)$
 - ♦ $|N_{Eps}(q)| \geq \text{MinPts}$
- un point p est **atteignable** d'un point q si
 - ♦ il existe une chaîne de points p_1, \dots, p_n telle que $p_1=q$ et $p_n=p$ et que les p_{i+1} sont directement atteignables des p_i
- un point p est **connecté** à un point q si
 - ♦ il existe un point o tel que p et q sont atteignables depuis o



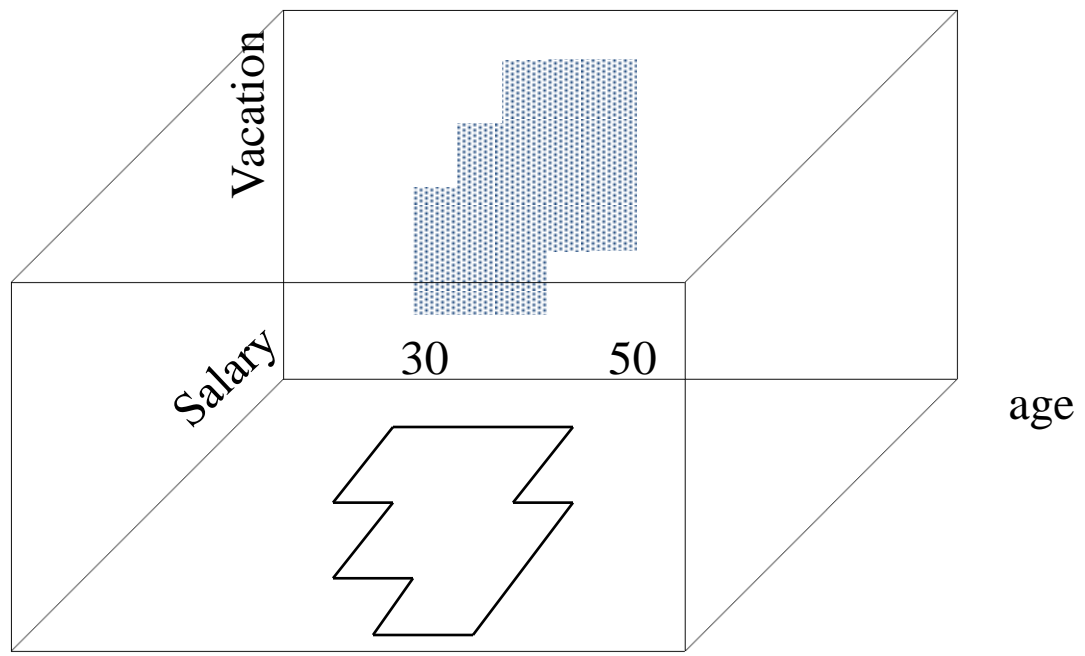
- Utilisation d'une grille à des résolutions multiples comme structure de données
- L'espace est divisé en cellules rectangulaires



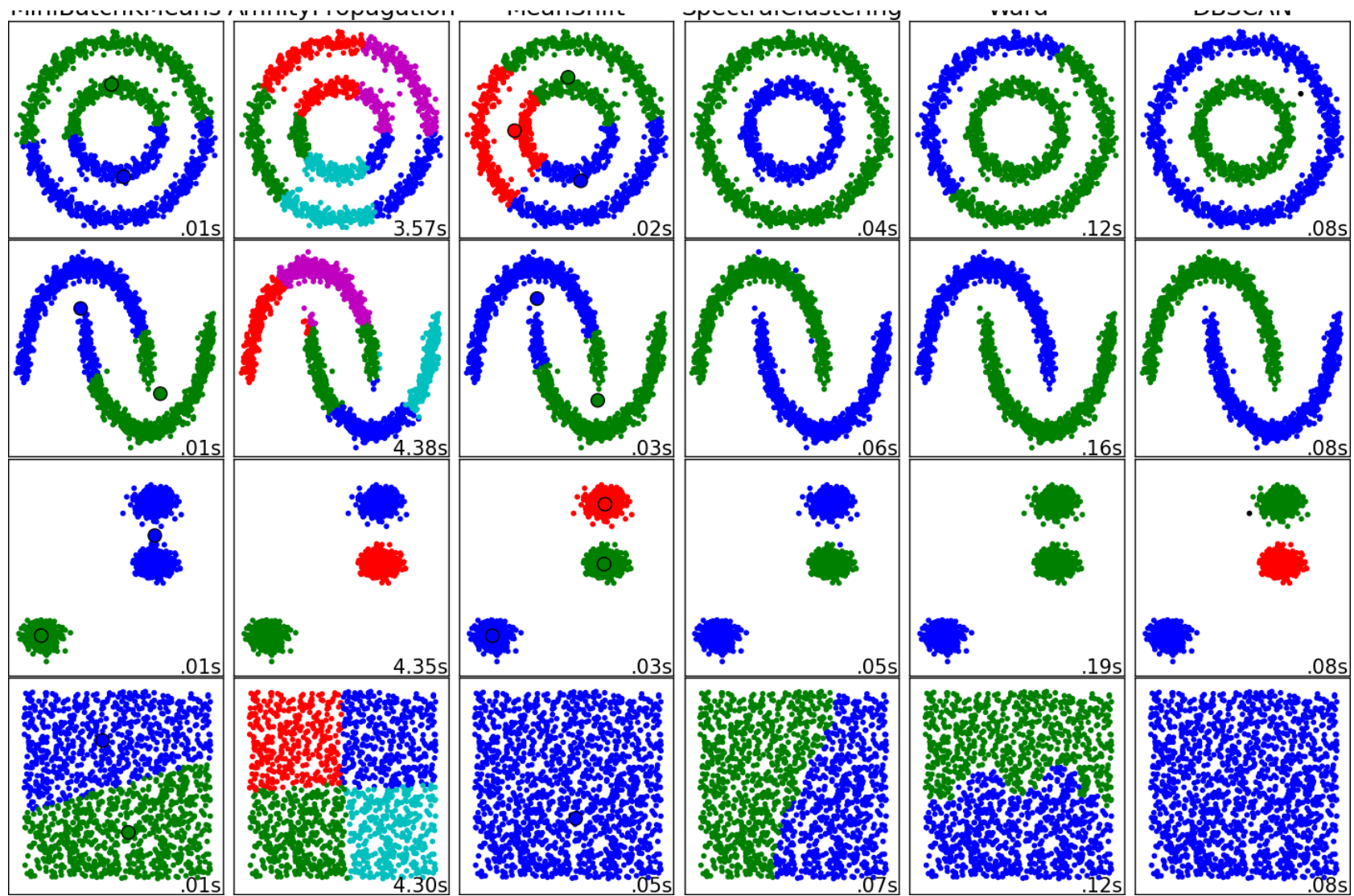
- Chaque cellule de niveau i est divisée en un certain nombre de cellules plus petites au niveau $i+1$
- Informations statistiques calculées et stockées à chaque niveau
- Approche descendante
- Suppression des cellules non pertinentes pour les itérations suivantes
- Répéter le processus jusqu'à atteindre le niveau le plus bas
- Avantages
 - ♦ parallélisable, mise à jour incrémentale
 - ♦ $O(k)$, où k est le nombre de cellule au plus bas niveau
- Faiblesse
 - ♦ les bords des clusters sont soit horizontaux soit verticaux, pas de diagonale !



$\tau = 3$



Illustration



- Existe-t-il une structure en clusters des données ?
- Quel est le nombre correct de clusters ?
- Mesure de qualité du partitionnement
- Comparaison du partitionnement à une classification existante
- Comparaison de 2 partitionnements

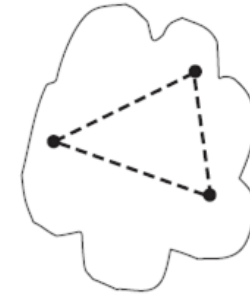
- Non supervisée
 - ◆ À partir des données
 - ◆ Cohésion
 - ◆ Séparation
- Supervisée
 - ◆ Par rapport à des classes connues
- Relative
 - ◆ Comparaison des résultats obtenus
 - Avec différentes méthodes
 - Avec différents paramètres

- Généralement de la forme :

$$\text{overall validity} = \sum_{i=1}^K w_i \text{validity}(C_i).$$

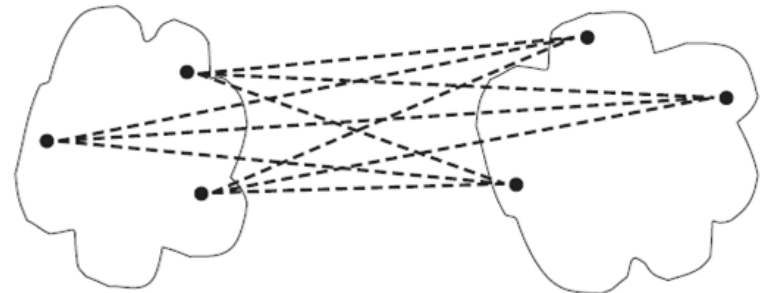
- Cohésion

$$\text{cohesion}(C_i) = \sum_{\substack{\mathbf{x} \in C_i \\ \mathbf{y} \in C_i}} \text{proximity}(\mathbf{x}, \mathbf{y})$$

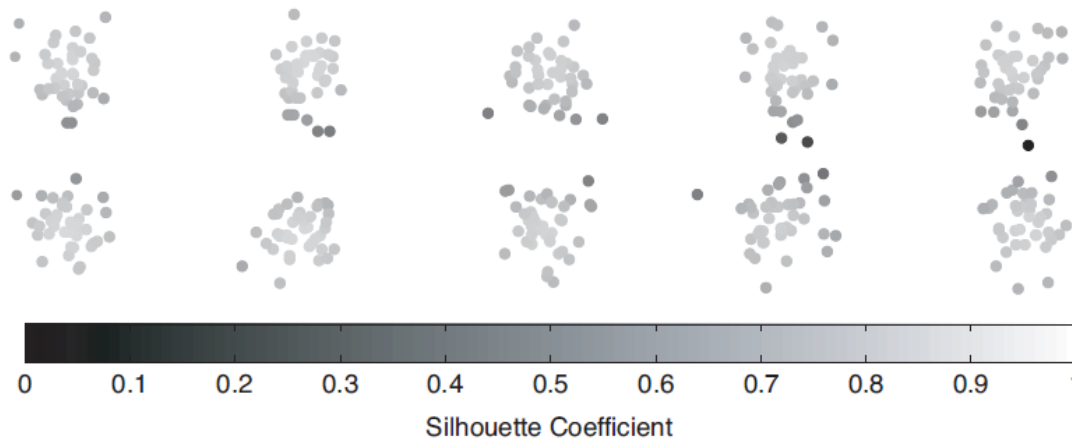


- Séparation

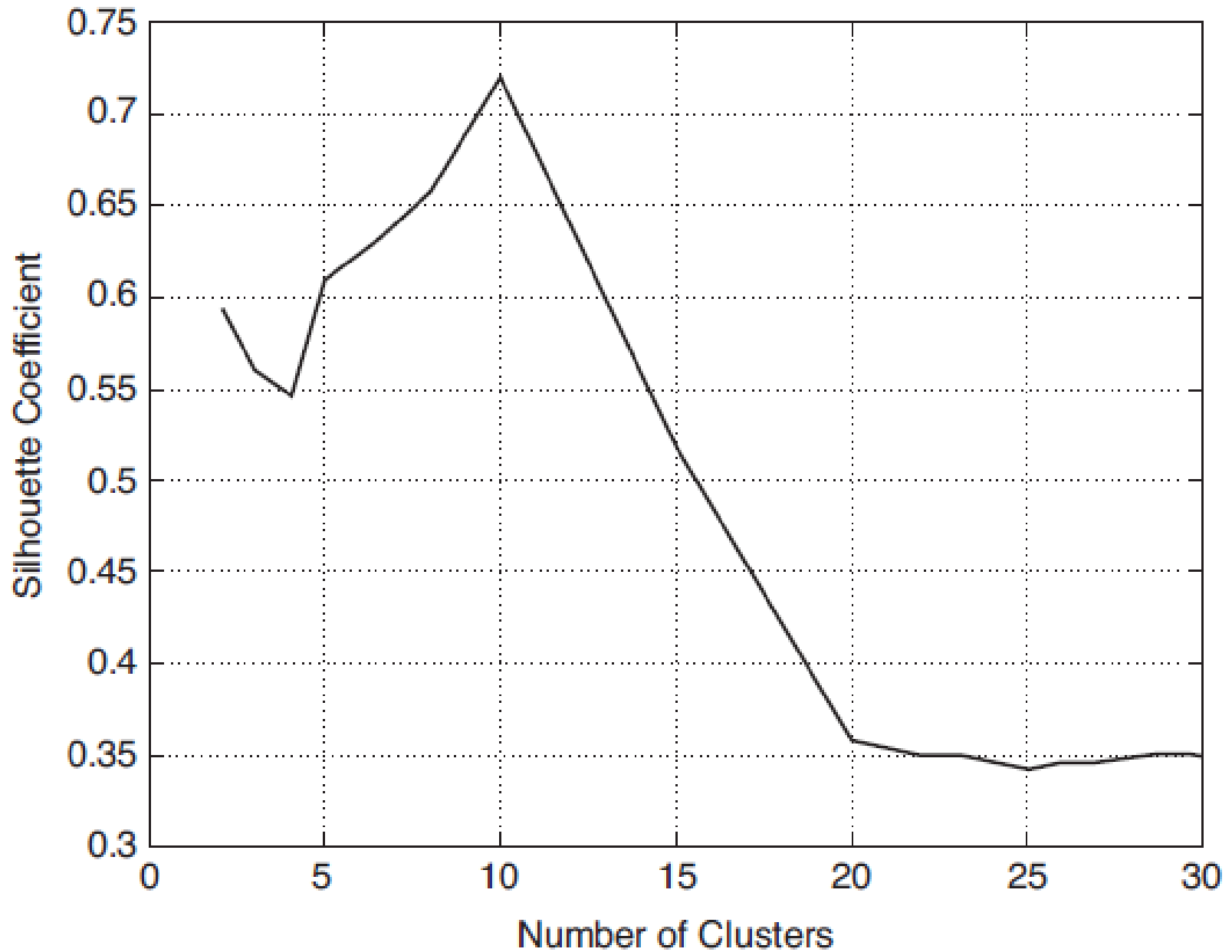
$$\text{separation}(C_i, C_j) = \sum_{\substack{\mathbf{x} \in C_i \\ \mathbf{y} \in C_j}} \text{proximity}(\mathbf{x}, \mathbf{y})$$



- Coefficient de silhouette
 - ◆ Pour le i -ème objet
 - a_i = distance moyenne aux objets du cluster
 - b_i = min des distances moyennes de l'objet aux objets d'un autre cluster
 - $s_i = (b_i - a_i) / \max(a_i, b_i)$
 - ◆ Pour un cluster : moyenne des coefficients des objets du cluster
 - ◆ Pour le partitionnement : moyenne des coefficient de tous les objets



Nombre de clusters



- Motivation : k -means trouvera toujours k clusters
- Statistique de Hopkins
 - ◆ Principe:
 - génération de p objets aléatoirement
 - échantillon de p objets
 - u_i et w_i les distances au plus proche voisin

$$H = \frac{\sum_{i=1}^p w_i}{\sum_{i=1}^p u_i + \sum_{i=1}^p w_i}$$

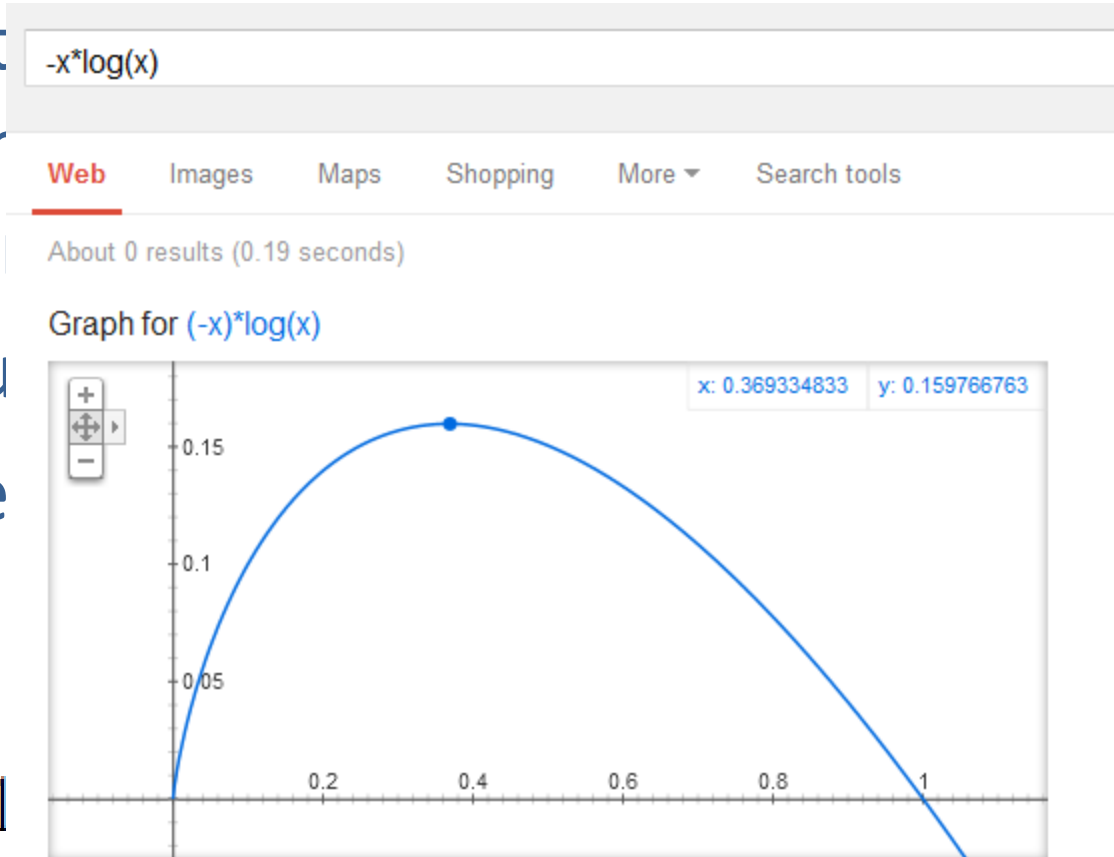
- $H = 0$: $u_i \gg w_i$: structure en clusters
- $H > 0.5$: $u_i \sim w_i$ ou $u_i \ll w_i$: distribution régulière des objets : pas de clusters

- Entropie : chaque cluster contient des objets de la même classe

- ♦ $P_{ij} = m_{ij}/m_i$: probabilité qu'un membre du cluster i appartienne à la classe j dans le cluster

- ♦ Entropie du cluster i
- ♦ Entropie totale des clusters

$$e = \sum_{i=1}^K$$



- Pureté : les clusters contiennent des objets d'une seule classe

$$p_i = \max_j p_{ij}, \quad \text{purity} = \sum_{i=1}^K \frac{m_i}{m} p_i.$$

- Précision : fraction d'un cluster consistant à des objets d'une classe spécifiée
- Recall : propension d'un cluster à contenir tous les objets d'une classe spécifiée
- Mesure F : combinaison des 2 précédentes = propension d'un cluster à contenir à la fois tous les objets d'une classe et seulement les objets de cette classe

$$F(i, j) = (2 \times \text{precision}(i, j) \times \text{recall}(i, j)) / (\text{precision}(i, j) + \text{recall}(i, j))$$

Point	p1	p2	p3	p4	p5
p1	1	1	1	0	0
p2	1	1	1	0	0
p3	1	1	1	0	0
p4	0	0	0	1	1
p5	0	0	0	1	1

Point	p1	p2	p3	p4	p5
p1	1	1	0	0	0
p2	1	1	0	0	0
p3	0	0	1	1	1
p4	0	0	1	1	1
p5	0	0	1	1	1

- Mesure de distance sur des variables binaires
 - ◆ coefficient simple d'appariement

$$d(i, j) = \frac{b + c}{a + b + c + d}$$

- ◆ coefficient de Jaccard

$$d(i, j) = \frac{b + c}{a + b + c}$$

		Objet <i>j</i>	
		1	0
Objet <i>i</i>	1	<i>a</i>	<i>b</i>
	0	<i>c</i>	<i>d</i>