Sciences des données

Un voyage initiatique

Cécile Capponi, Rémi Eyraud, Hachem Kadri

LIS, Aix-Marseille Université, CNRS
Equipe QARMA

M1 Informatique – S5 2020-2021

Plan

Plan

Plan

Plan

Plan

Outline

Classification supervisée vs. Classification non-supervisée

Classification supervisée : rappels

Apprendre un modèle de classification à partir de données étiquetées

- $\mathcal{X}$ espace d'entrée, $\mathcal{Y}$ espace des cibles (classes)
- avec $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ où chaque $\mathcal{X}_i$ est le domaine d'un attribut a_i, symbolique ou numérique.
- On suppose l'existence d'une variable aléatoire $Z = (X, Y)$ à valeurs dans $(\mathcal{X} \times \mathcal{Y}$
- Les exemples (données observées) sont des couples (x, y) de $(\mathcal{X} \times \mathcal{Y}$ tirés selon la distribution jointe $P(Z = (x, y)) = P(X = x)P(Y = y|X = x)$
- Un échantillon de données observées (exemples) est un ensemble fini $S = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ identiquement et indépendamment distribués selon la loi de P
- Apprendre $h : \mathcal{X} \to \mathcal{Y}$ à partir de S qui minimise le risque d'erreurs en généralisation

Classification supervisée vs. Classification non-supervisée

Classification non-supervisée

Apprendre un modèle de classification à partir de données sans connaissance des classes/cibles/étiquettes

- $S = \{x_1, \cdots, x_n\}$
- Apprendre $\mathcal{Y}$ et $h : \mathcal{X} \to \mathcal{Y}$ à partir de S qui minimise le risque d'erreurs en généralisation
- On ne peut ême plus évaluer le risque sur l'échantillon d'apprentissage

Regroupement (*Clustering*

S peut être :

- un échantillon de données (des exemples) : dans ce cas, le problème de clustering est un problème de classification non-supervisée
- toutes les données : dans ce cas, le problème de clustering est un problème de partitionnement.

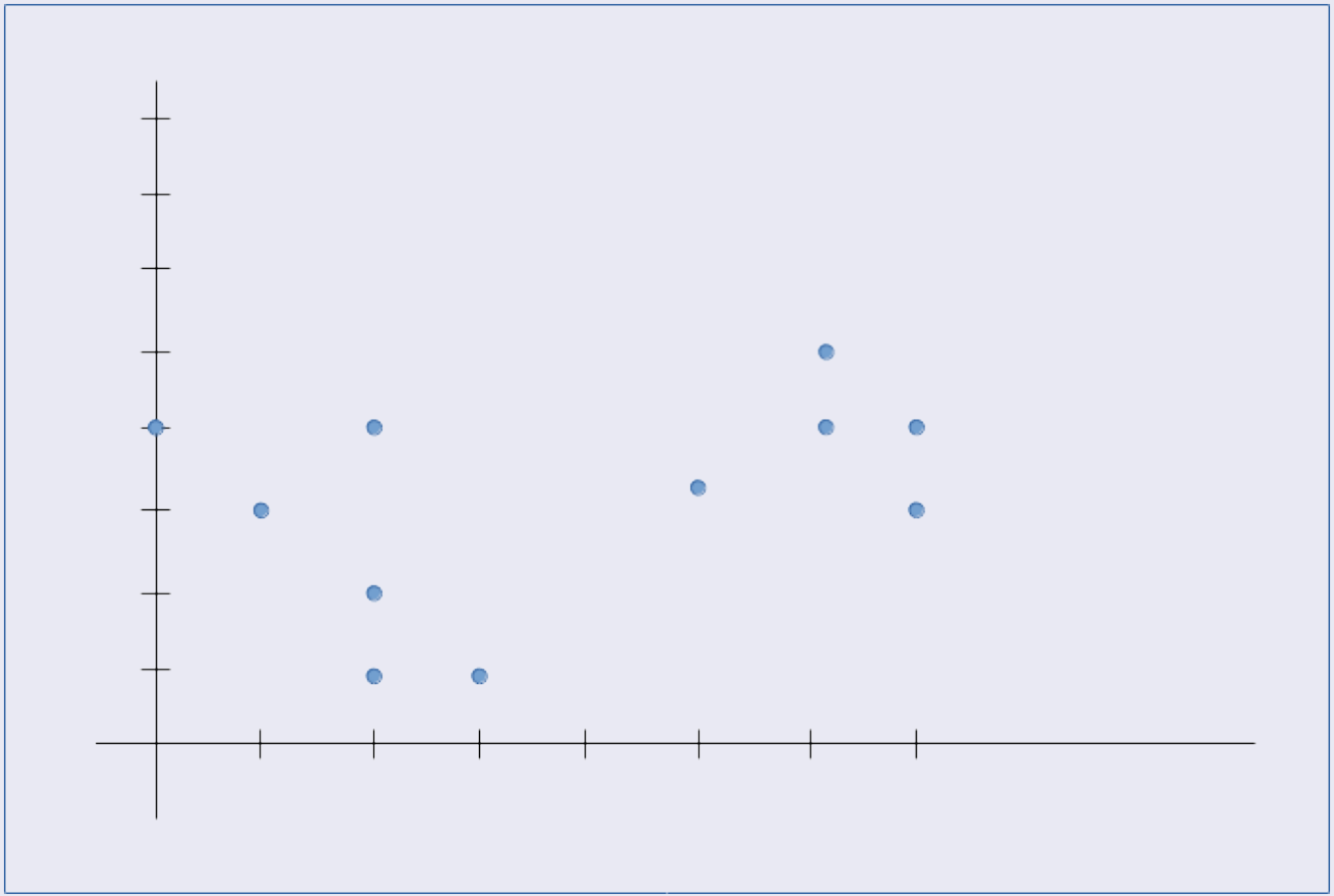
Algorithme de regroupement (*clustering*) : se servir d'autres informations pour segmenter les données en classes (ex. information sur le nombre de classes, topologie de l'espace des attributs, etc.)

Classification non-supervisée/ clustering / regroupement

Exemple

$$\mathcal{X} = \mathcal{N}^2$$

$S = \{(0,4),(1,3),(2,1),(2,2),(2,4),(3,1),(3,2),(3,4),(5,3),(6,4),(6,5),(7,3),(7$

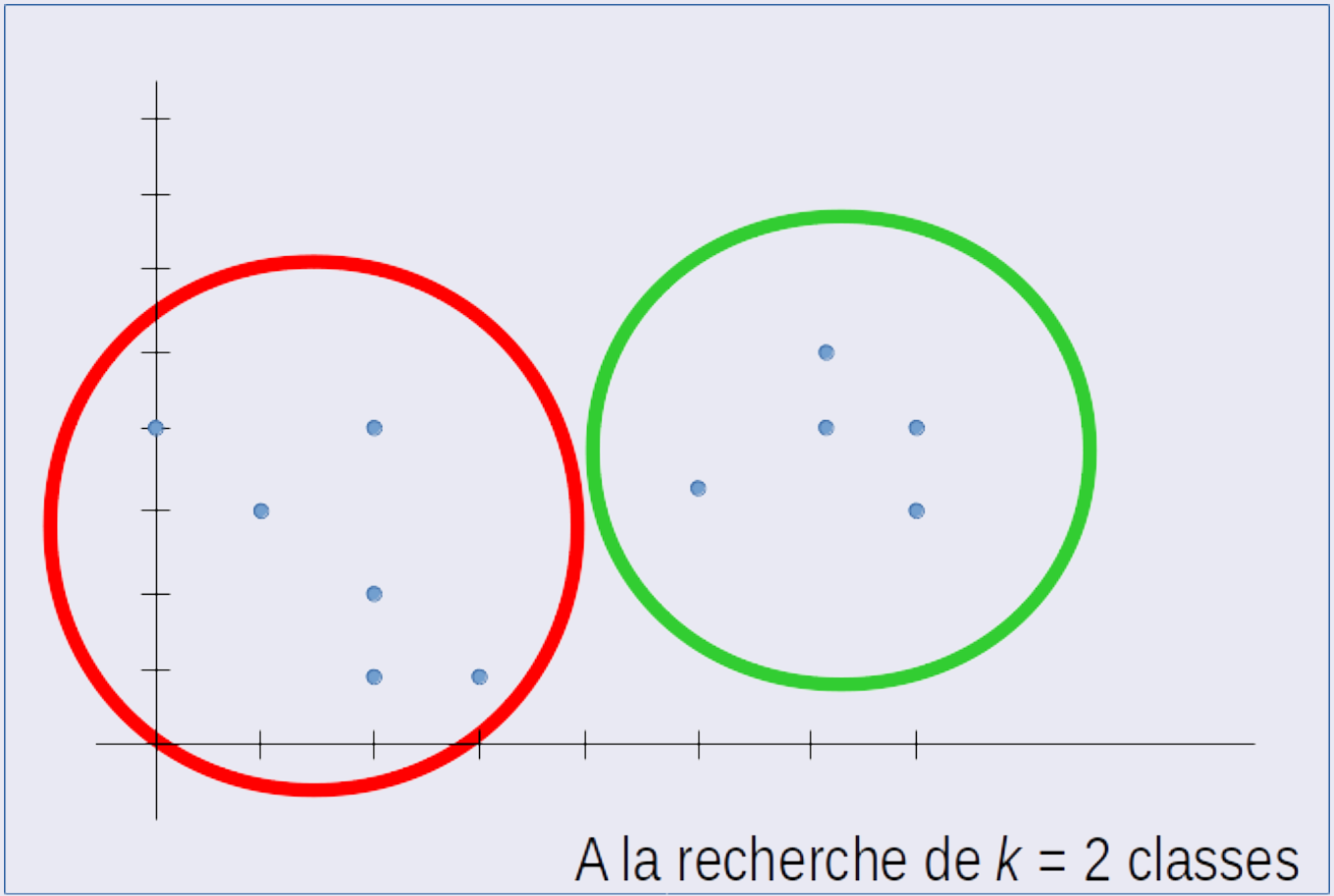


Classification non-supervisée/ clustering / regroupement

Exemple

$$\mathcal{X} = \mathcal{N}^2$$

$$S = \{(0,4),(1,3),(2,1),(2,2),(2,4),(3,1),(3,2),(3,4),(5,3),(6,4),(6,5),(7,3),(7$$

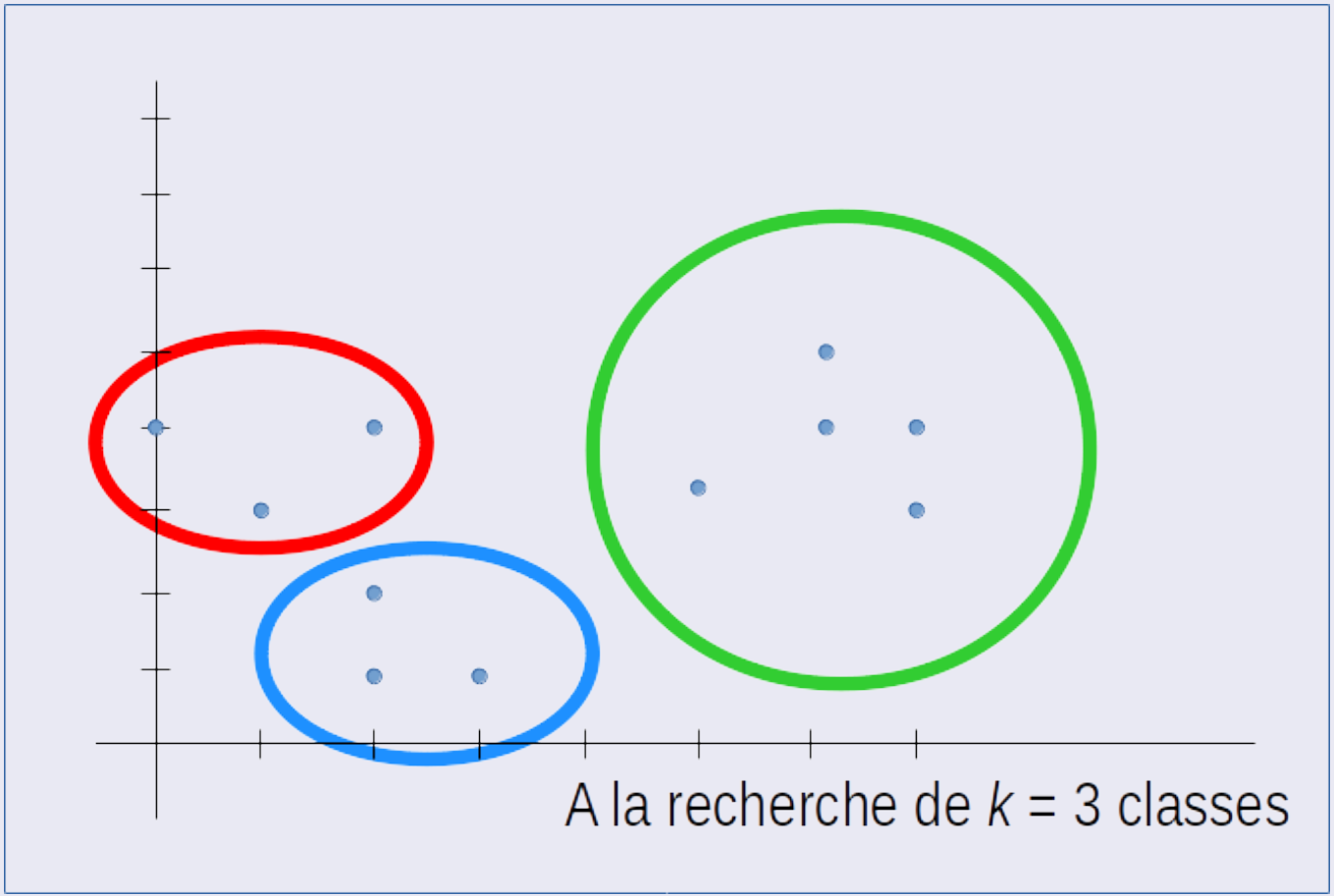


Classification non-supervisée/ clustering / regroupement

Exemple

$$\mathcal{X} = \mathcal{N}^2$$

$$S = \{(0,4),(1,3),(2,1),(2,2),(2,4),(3,1),(3,2),(3,4),(5,3),(6,4),(6,5),(7,3),(7$$



Principe du clustering (regroupement)

Construire une collection de groupes d'objets tels que

(groupe/classe/catégorie/cible/étiquette/*cluster*)

1. les objets d'un même groupe soient similaires
2. les objets de deux groupes différents soient dissimilaires

Quelques cas d'usage

- Marketing : groupes distincts de clients pour ciblage promotionnel, recommandation de produits
- Traitement d'image : regroupement de zones similaires (segmentation)
- Internet : analyse de réseaux sociaux, algorithme de gestion de fil, etc.
- Détection de fraude / anomalies : identification d'outliers (hors groupes majeurs)
-

Qu'est-ce qu'un bon regroupement ?

Principe majeur

Une bonne méthode de regroupement permet de ganrantir

- Une grande similarité inter-groupe
- Une faible similarité intra-groupes

Qualité du regroupement définie en fonction de la similarité utilisée par la méthode pour comparer les objets

Mesure la similarité entre objets

Une mesure de distance (positivité, séparation, symétrie, inégalité triangulaire)

Mesure de la qualité d'un regroupement

Prise en compte des similarités intra- (à maximiser) et extra-groupes (à minimiser) : plusieurs métriques, par exemple indice de Davies-Bouldin.

Outline

Les différentes approches de regroupement

Algorithmes de partitionnement

Construire plusieurs partitions de S puis évaluer leur qualité selon certains critères, et garder celle de meilleure qualité.

Algorithmes hiérarchiques

Créer une décomposition hiérarchique (arbre) selon certains critères, de façon *top-down* ou *bottom-up* ou hybride.

Algorithmes basés sur la densité des groupes

Optimiser des critères de densité de groupes et/ou de connectivité

Algorithmes de grille

Construire des partitions en se basant sur une structure $\neq$ niveaux de granularité

Algorithmes à modèle

Un modèle est supposé existant pour chaque groupe (cluster) : vérifier chaque modèle sur chaque groupe pour choisir le meilleur selon certains

Algorithmes à partitionnement

Construire une partition à k clusters à partir d'une base S de n données (points dans $\mathcal{X}$).

Les k clusters doivent optimiser le critère choisi

- *Global Optimal* considère l'ensemble des k partitions
- *Heuristic Methods* : Algorithmes k-means et k-medoids, où chaque cluster est représenté par une et une seule donnée (un point x dans l'espace $\mathcal{X}$).
  - Les k-means (k-moyennes, (Mac Queen'67) : chaque cluster est représenté par son centre de gravité
  - Les k-medoïdes (ou PAM, Partition Around Medoids, Kaufman and Rousseeuw'87) : chaque cluster est représenté par un de ses points membres, lequel ?

Algorithme des k-means (k-moyenne)

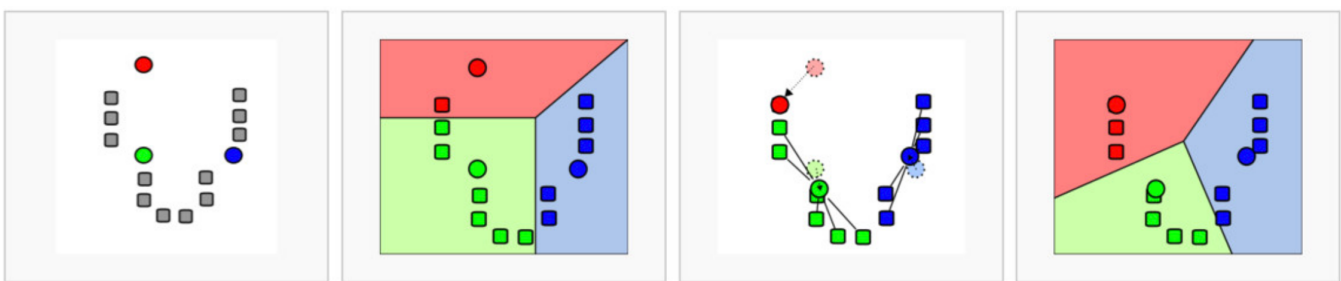
Version itérative et intuitive

Entrée : k le nombre de parties, $S = \{x_i\}_{i=1..n}$ l'ensemble des données (points) observées, avec $\forall i,\ x_i \in \mathcal{X}$

1. Choisir k points $m_j \in S$, centres de k clusters C_j, $j \in 1..k$
2. Itérer jusqu'à stabilité des clusters :
   1. Affecter chaque point $x \in S$ au cluster C_j de centre m_j tel que distance(x, m_j) est minimale
   2. Recalculer le centre m_j de chaque cluster C_j pour qu'il soit son barycentre
   3. **Si** $\exists j \in 1..k$ un cluster pour lequel le centre m_j a été recalculé ligne précédente, **alors** itérer, **sinon** sortir de l'itération

Sortie : $\{m_j\}_{j=1..k}$

Algorithme des *k*-moyennes (*k*-means) : illustrations en 1D et 2D



Algorithme des k-moyennes : algorithme

Construction des clusters par recentrages successifs

K-MEANS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
1 $(\vec{s}_1, \vec{s}_2, \ldots, \vec{s}_K) \leftarrow$ SELECTRANDOMSEEDS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
2 **for** $k \leftarrow 1$ **to** K
3 **do** $\vec{\mu}_k \leftarrow \vec{s}_k$
4 **while** stopping criterion has not been met
5 **do for** $k \leftarrow 1$ **to** K
6 **do** $\omega_k \leftarrow \{\}$
7 **for** $n \leftarrow 1$ **to** N
8 **do** $j \leftarrow \arg\min_{j'} |\vec{\mu}_{j'} - \vec{x}_n|$
9 $\omega_j \leftarrow \omega_j \cup \{\vec{x}_n\}$ *(reassignment of vectors)*
10 **for** $k \leftarrow 1$ **to** K
11 **do** $\vec{\mu}_k \leftarrow \frac{1}{|\omega_k|} \sum_{\vec{x} \in \omega_k} \vec{x}$ *(recomputation of centroids)*
12 **return** $\{\vec{\mu}_1, \ldots, \vec{\mu}_K\}$

Précisions

- Autres possibilités d'initialisations, autres distances
- Normalisation conseillée des vecteurs
- Critère classique de convergence = (presque) plus aucun vecteur ne change de cluster
- Décision pour x : le cluster dont la moyenne est la plus proche de x

Illustration en 1D ($\mathcal{X} = \mathcal{R}$)

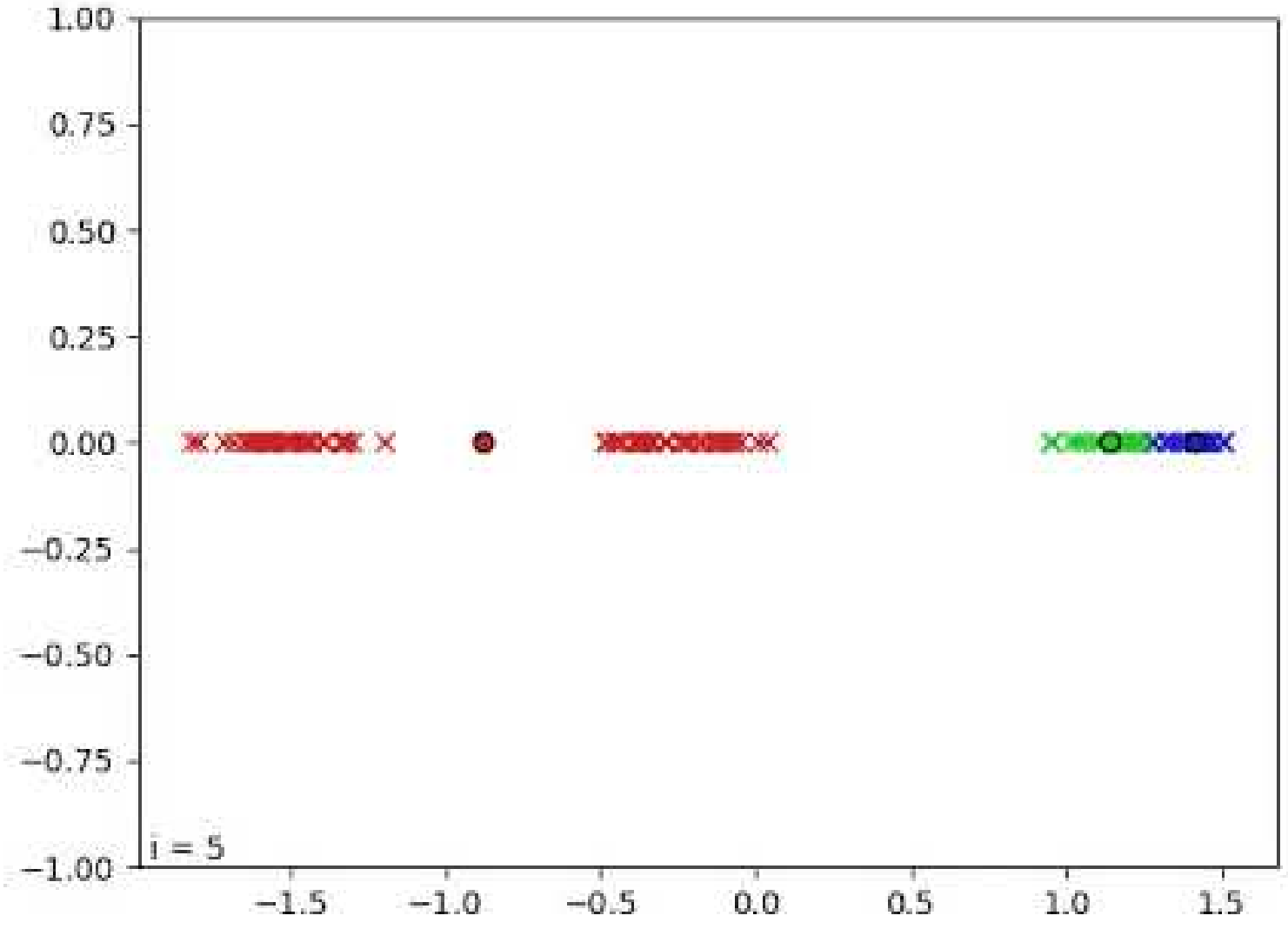
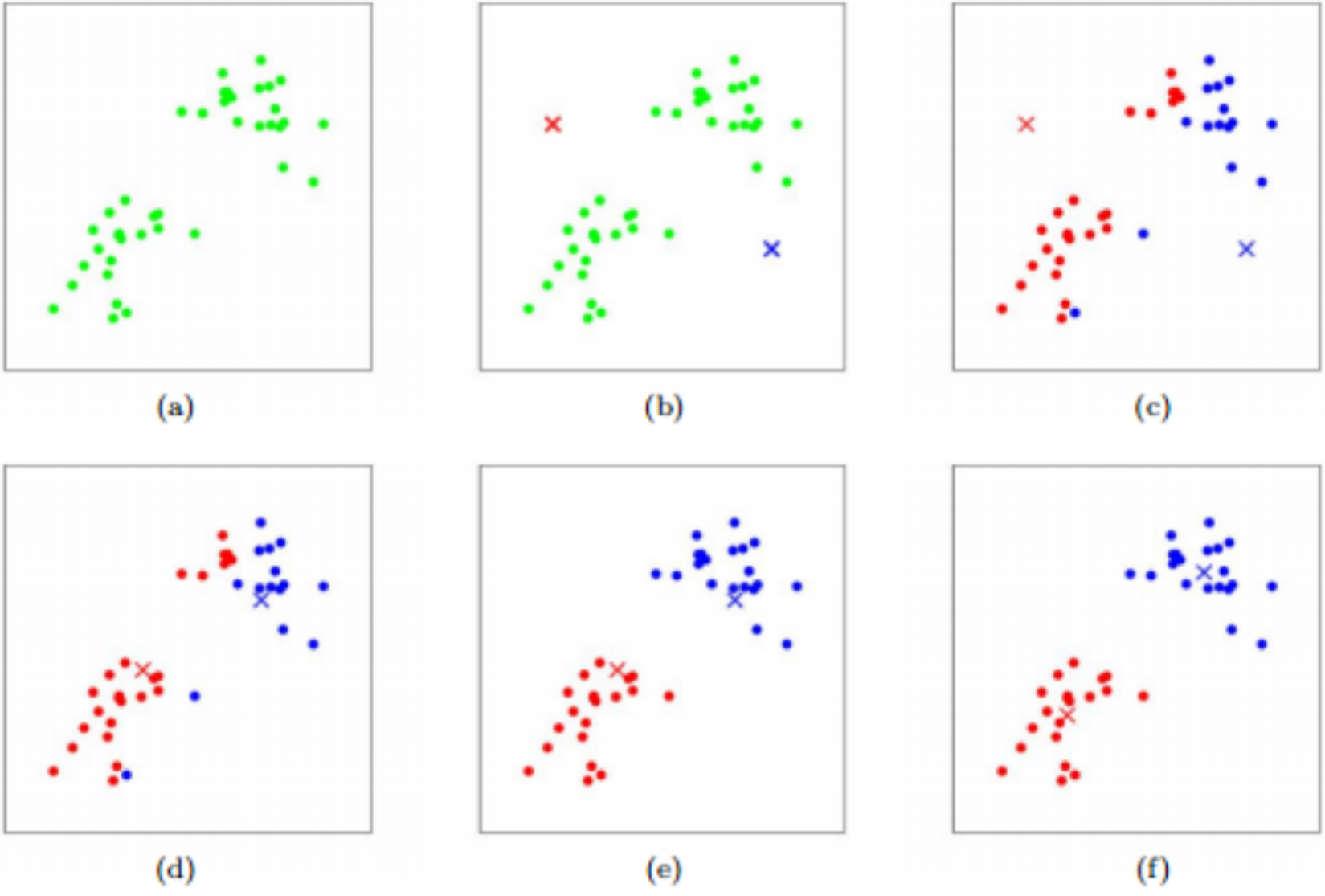


Illustration en 2D ($\mathcal{X} = \mathcal{R}^2$)

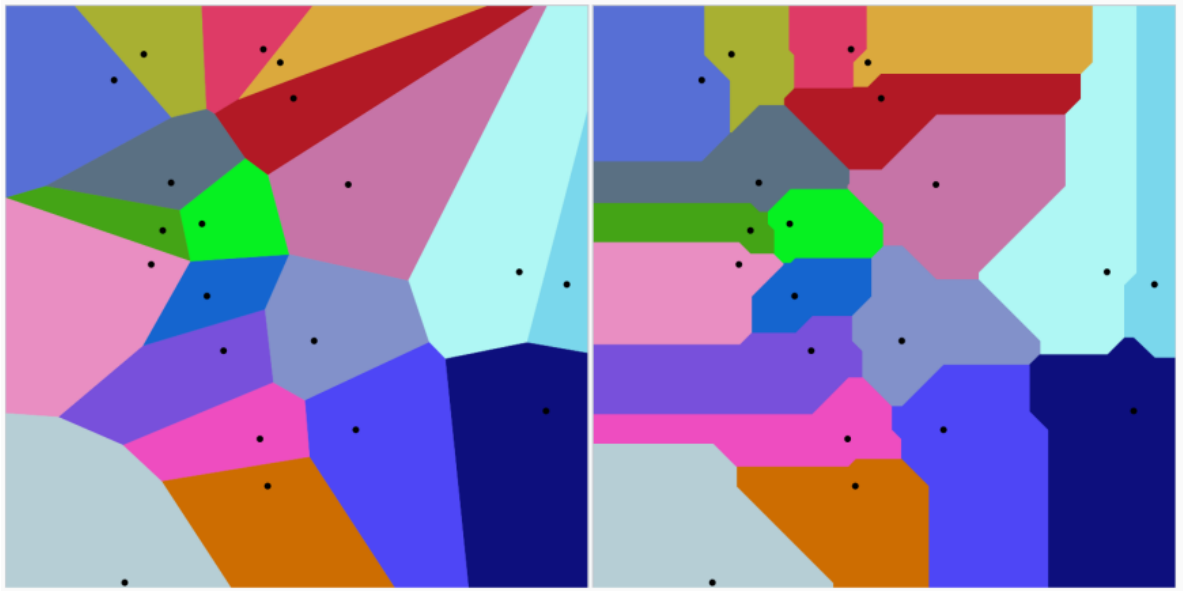


[Standford, 2018]

Commentaires sur l'algorithme des k-moyennes

Ses forces

- Relativement efficace : $O(tkn)$ avec k le nombre de clusters, n le nombre d'exemples (points) dans l'échantillon S, et t le nombre d'itérations avant convergence.
- Souvent, $k, t << n$
- Passage à l'echelle
- Tend à réduire la variance inter-cluster (théorème)

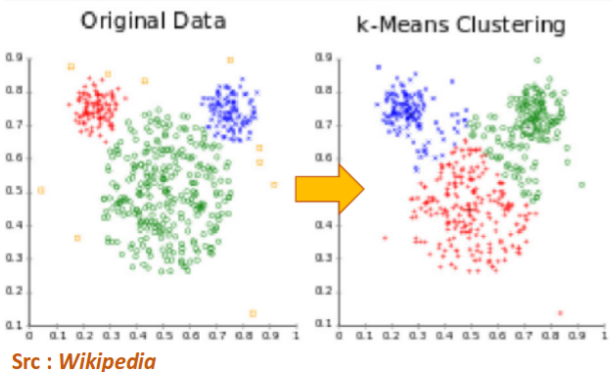


Distance de Manhattan Distance euclidienne

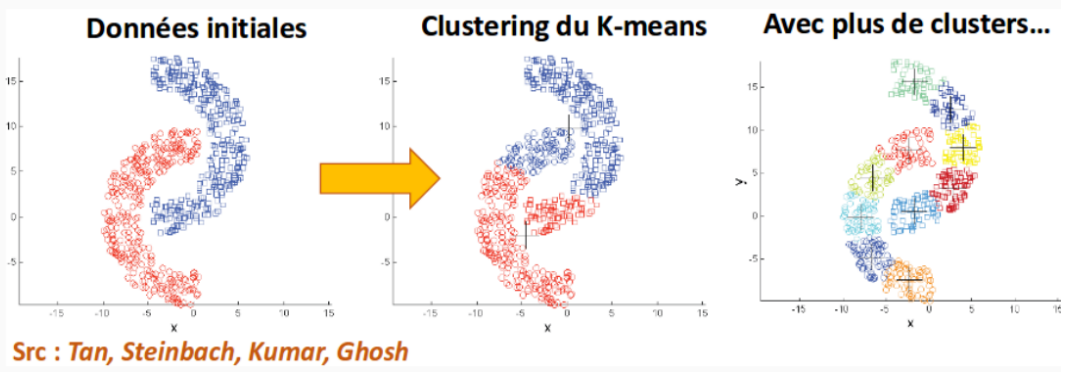
Commentaires sur l'algorithme des k-moyennes

Ses faiblesses

- Très dépendant de l'initialisation
- Dépendant de la distance utilisée
- k doit être connu
- Les clusters sont construits à partir de points inexistants dans S ; solution = les k-médoïdes
- Ne permet pas de décoouvrir des groupes non-convexes



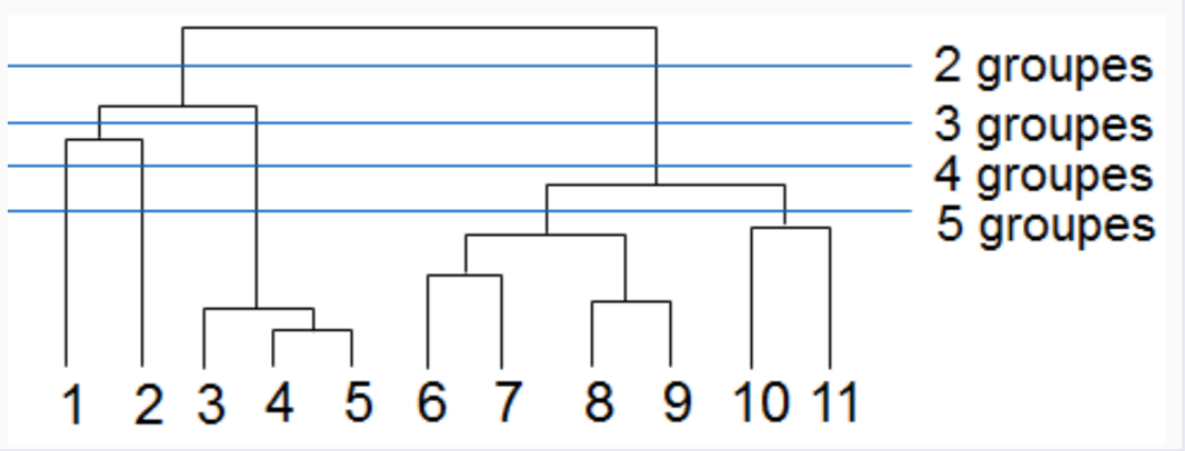
Src : *Wikipedia*



Src : *Tan, Steinbach, Kumar, Ghosh*

Autres algorithmes de clustering

Regroupements hiérarchiques



Nombreuses autres approches

- Clustering spectral (pour groupes non-convexes)
- Algorithme par estimation de densités (mélanges de gaussiennes, *expectation-maximisation* (EM), etc.)
- Approches par échantillonage (CLARA, CURE, etc.)

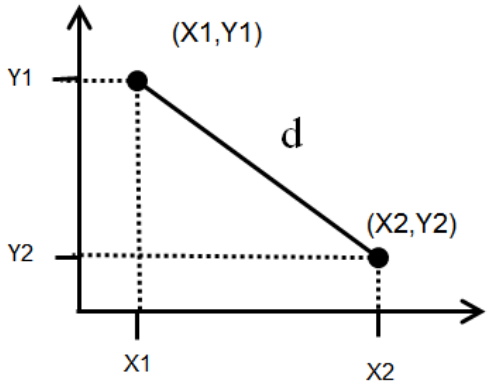
Outline

Les distances usuelles définies dans $\mathbb{R}^d$

Soient $x = (x_1, \cdots, x_d)$ et $x' = (x'_1, \cdots, x'_d)$ deux points de $\mathbb{R}^d$.

La distance euclidienne

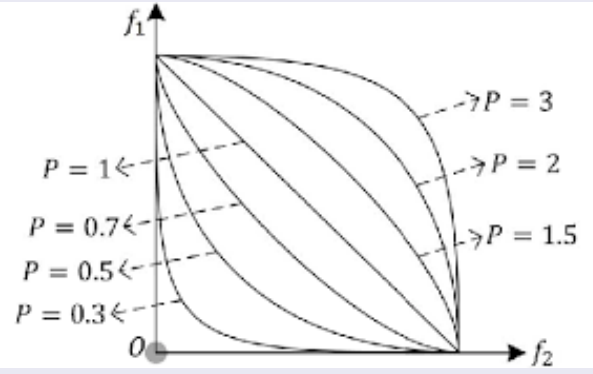
$$d_2(x, x') = \sqrt{\sum_{i=1}^{d} (x_i - x'_i)^2}$$



$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

La distance de Minkowski (et déclinaison infinie = Tchebychev)

$$d^p(x, x') = \left(\sum_{i=1}^{d} |x_i - x'_i|^p \right)^{\frac{1}{p}}$$

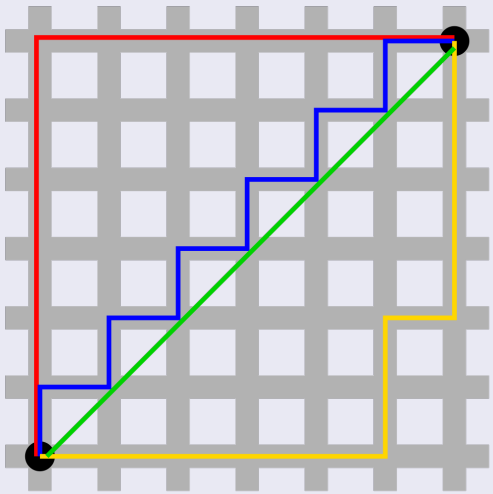


Les distances usuelles définies dans $\mathbb{R}^d$

Soient $x = (x_1, \cdots, x_d)$ et $x' = (x'_1, \cdots, x'_d)$ deux points de $\mathbb{R}^d$.

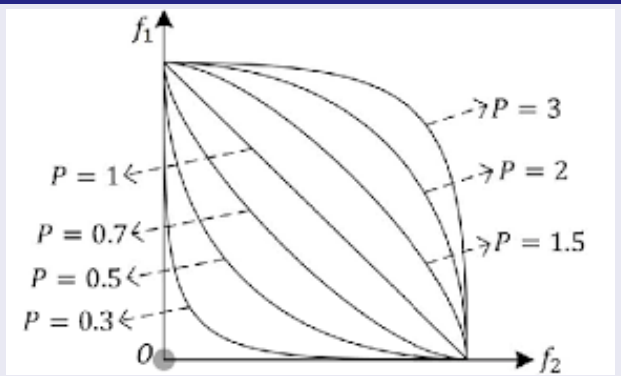
La distance de Manhattan (distance taxi)

$$d_1(x, x') = \sum_{i=1}^{d} |x_i - x'_i|$$



La distance de Minkowski (et déclinaison infinie = Tchebychev)

$$d^p(x, x') = \left(\sum_{i=1}^{d} |x_i - x'_i|^p \right)^{\frac{1}{p}}$$



Pré-traitement des données si $\mathcal{X} = \mathbb{R}^d$

Pour éviter de trop gros écarts de grandeur entre différentes composantes d'un même point.

Quid si les composantes ont des ordres de grandeurs très différents ?

Exemple avec la distance de Manhattan

	Age	Salaire
Personne 1	50	11 000
Personne 2	70	11 000
Personne 3	60	11 122
Personne 4	60	11074

$d_1(p_1, p_2) = 20$
$d_1(p_1, p_3) = 132$
p_1 ressemble plus à p_2 qu'à p_3

Standardisation préalable des données

La standardisation des données pour éviter ce phénomène

- Calcul de l'écart absolu moyen pour chaque colonne i

$$s_i = \frac{1}{n} \sum_{j=1}^{n} |x_{j,i} - \mu_j|$$

où μ_j est la moyenne des valeurs de la colonne j, et n est le nombre de points dans S

- Calcul de la valeur standardisée pour chaque composante de chaque point

$$\bar{x}_{j,i} = \frac{x_{j,i} - \mu_j}{s_j}$$

Sur l'exemple

Calcul des valeurs et écarts absolus pour chaque colonne

	Age	Salaire
Personne 1	50	11 000
Personne 2	70	11 000
Personne 3	60	11 122
Personne 4	60	11074

$\mu_{age} = 60, s_{age} = 5$
$\mu_{sal} = 11049, s_{sal} = 49$

	Age	Salaire
Personne 1	-2	-0,5
Personne 2	2	0.175
Personne 3	0	0,324
Personne 4	0	0

$d_1(p_1, p_2) = 4$
$d_1(p_1, p_3) = 2$
p_1 ressemble maintenant plus à p_3 qu'à p_2

Mesures de distance dans le cas d'attributs binaires

Table de contingence des attributs à valeurs dans {0,1}

		Point x	
		1	0
Point x'	1	a	b
	0	c	d

Par exemple, avec $x = (1,1,0,1,0)$ et $x' = (1,0,0,0,1)$, on a $a = 1, b = 2, c = 1, d = 1$

Coefficient d'appariemment simple

$$A(x,x') = \frac{b+c}{a+b+c+d}$$

Sur l'exemple, $A(x,x') = 3/5$

Coefficient de Jaccard (cas asymétrique)

$$J(x,x') = \frac{b+c}{a+b+c}$$

Sur l'exemple, $J(x,x') = 3/4$

(A)symétrie des attributs binaires

Attribut symétrique

Quand la fréquence des 0 est à peu près la même que celle des 1 dans la population observée ($p(x_i = 0) \sim p(x_i = 1)$)
Exemple : si même proportion de féminin et masculin, coder masculin par 0 et féminin par 1 équivaut au codage inverse. Dans une filière type Master Informatique, le genre devient attribut asymétrique...

Attribut asymétrique

- Quand la fréquence des 0 est très différente celle des 1 dans la population observée ($p(x_i = 0) >> p(x_i = 1)$)
  Exemple : un test HIV peut être positif ou négatif, mais une valeur est plus présente que l'autre.
- Une règle de *data scientist* (qui explique Jaccard) : codage par 1 de la modalité la moins fréquente.
  Exemple : deux personnes ayant la valeur 1 pour le test HIV sont plus similaires entre elles que le sont deux autres personnes ayant 0 pour le test. Idem sous Covid !

Distance dans le cas d'attributs nominaux

Généralisation des attributs binaires aux *n*-aires

Lorsque les valeurs sont prises dans un domaine fini, par exemple un attribut `couleur` de domaine {`rouge, vert, jaune, magenta, noir`}.

Calcul du matching simple entre données x et x'

Soit m le nombre d'appariemments, q le nombre d'attributs nominaux

$$A_N(x, x') = \frac{q - m}{q}$$

Ou transformation en attributs binaires par génération d'attributs (un par valeur possible)

Sur exemple des couleurs, l'attribut `couleur` est remplacé par 5 attributs binaires, de noms `rouge, vert, jaune, magenta, noir`.

Distance dans le cas d'attributs ordinaux

Existence d'un ordre total entre les valeurs de l'attribut

Réels, entiers, caractères, etc. : les valeurs peuvent être discrètes (ex. un classement de course), ou continues (le temps de course).

Génération d'intervalles

Pour chaque attribut a :

- Pour chaque donnée x, remplacer x_a par son rang r_a dans l'intervalle $[1, \max(a)]$
- pour chaque donnée x, normaliser son rang dans $[0, 1]$

$$x_a \leftarrow \frac{r_a - 1}{\max(a) - 1}$$

- Utiliser une distance de Minkowski pour calculer les similarités

Distance dans le cas de chaînes de caractères

Quelle similarité entre deux séquences de caractères ?

`Marie` vs `Mary`, ACGGCTAA vs. AGGGCTA, etc.

Distance de Levenshtein (distance d'édition)

On calcule le nombre d'opérations élémentaires (insérer, supprimer, remplacer) pour passer d'une chaîne à l'autre
Distance de Levenhstein entre deux mots = coût minimal pour transformer le premier vers le second en utilisant seulement les opérations élémentaires.
d_L(ACGGCTAA, AGGGCTA-) $= 2$ (un remplacement, une suppression)
Possibilité d'associer un coût à chaque type d'opération élémentaire

Variantes pour d'autres types de données

Damereau-Levenhstein, distance d'édition sur arbres, Jaro-Winkler, Hamming, etc.

Distance dans le cas d'attributs de tous types

Méthode ad-hoc

Soit $x = (x_1, \cdots, x_d)$ et $x' = (x'_1, \cdots, x'_d)$ Réaliser les pré-traitements nécessaires (encodage binaires, standardisation, etc., puis utiliser les distances adaptées à chaque groupe d'attributs de mêmes spécificité $\rightarrow$ on obtient $z \leq d$ groupes d'attributs, donc distances $g_1, \cdots, g_z$
Distance finale = combinaison linéaire (ou autre) des g

$$d(x, x') = \frac{1}{z} d_z(x_{g_z}, x'_{g_z})$$

Outline

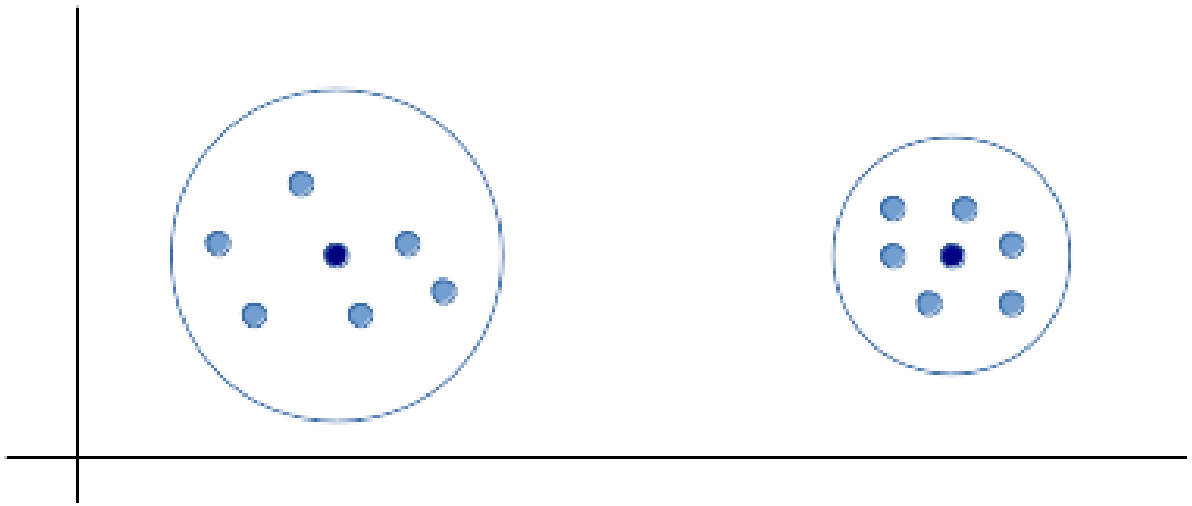
Evaluation de la qualité d'un clustering

Pas d'étiquettes !

Formes attendues des clusters

- Resserrés sur eux-mêmes : deux points qui sont proches doivent appartenir au même cluster

  homogénéité de chaque cluster
- Eloignés les uns des autres : deux points qui sont éloignés doivent appartenir à des clusters différents.



Homogénéité d'un cluster : mesure intra-cluster)

Homogénéité d'un cluster

Notons d la distance que nous allons utiliser, par exemple la distance euclidienne. Centre de gravité (barycentre) d'un cluster C_j = le point moyen :

$$\mu_j = \frac{1}{|C_j|} \sum_{x \in C_j} x$$

Homogénéité d'un cluster = moyenne des distances de chacun des points de ce cluster, au barycentre (centroïde)

$$H_j = \frac{1}{|C_j|} \sum_{x \in C_j} d(x, \mu_j)$$

Bon cluster = mesure d'homogénéité faible

Moyenne des dispersions de k clusters d'un ensemble de points S

$$H = \frac{1}{k} \sum_{j=1}^{k} H_j$$

Mesure de séparation des cluster (extra-cluster)

Eloignement de deux clusters j et l

= distance entre leurs barycentres

$$E_{j,l} = d(\mu_j, \mu_l)$$

Moyenne de l'éloignement entre les k clusters d'un ensemble de points S

$$E = \frac{2}{k(k-1)} \sum_{j=1}^{k} \sum_{l=j+1}^{k} E_{j,l}$$

$\rightarrow$ Combinaison dispersion et éloignement

Qualité d'un partitionnement par Davies-Bouldin

Indice de Davies-Bouldin propre à un cluster j

$$I_{DB,j} = \max_{l \neq j} \frac{H_j + H_l}{E_{j,l}}$$

Indice de Davies-Bouldin d'un clustering

$$I_{DB} = \frac{1}{k} \sum_{j=1}^{k} I_{DB,j}$$

$$I_{DB} = \frac{1}{k} \sum_{j=1}^{k} \max_{j' \neq j} \left(\frac{H_j + H_{j'}}{d(\mu_j, \mu_{j'})} \right)$$

- Bon clustering : indice de Davies Bouldin = 0
- Pire clustering : indice de Davies Bouldin infini

Qualité d'un partitionnement par la mesure de silhouette

Coefficient de silhouette d'un point x

La silhouette $s(x)$ permet d'estimer si x appartient au bon cluster j

- Proche des points de son cluster ?

$$a(x) = \frac{1}{|C_j| - 1} \sum_{x' \neq x \in C_j} d(x, x')$$

- Loin des points hors-cluster ?

$$b(x) = \min_{j' \neq j} \frac{1}{|C_{j'}|} \sum_{x' \in C_{j'}} d(x, x')$$

Si x a été correctement assigné, alors $a(x) < b(x)$

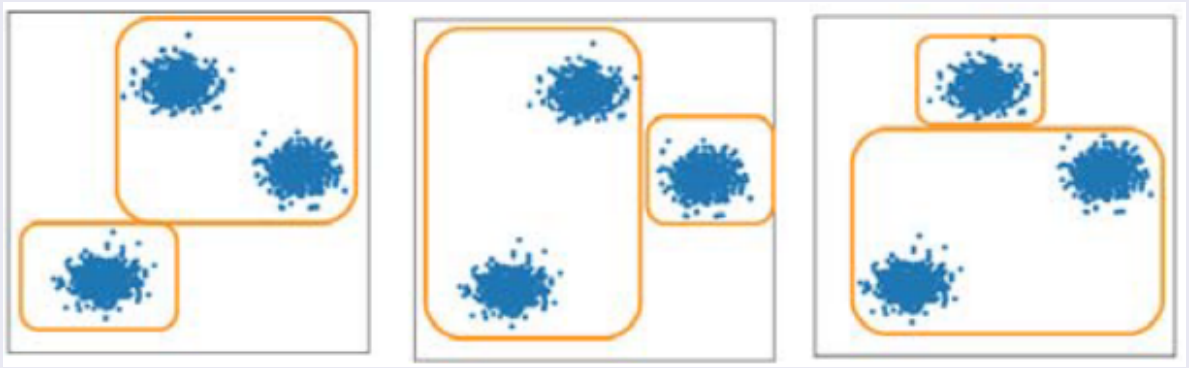
$$s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \in [-1, 1]$$

Moyenne sur tous les clusters

Stabilité du résultat vs. choix de k

Stabilité = résistance à plusieurs éxécutions non-déterministes

- Rendre le problème non-déterministe : enlever quelques points de S au hasard, ou modifier initialisation
- Observer la similarité des résultats de plusieurs exécutions pour les valeurs à tester de k : le choix du bon k peut en être influencé.
- Exemple d'instabilité d'un k-means sur un ensemble dans $\mathbb{R}^2$ (credits M.J. Huguet)



- Indice de Rand (comparer deux solutions en ignorant les permutations : proportion des paires d'éléments qui sont conjointement groupées ou conjointement séparées)