

APPRENTISSAGE NON SUPERVISÉ



EL MALKI
YNOV M2
23-24

OBJECTIFS

Données

- Définies sur un ensemble d'attributs
- Aucune classe ni connaissance à priori sur ces données

But

- On cherche à diviser ces données en catégories
- Trouver descriptions intéressantes pour résumer, comprendre et interpréter les données



APPROCHES NON SUPERVISEES

1. Objectifs
2. Quelques familles de méthodes :
 1. **Clustering**
 1. Introduction et distances
 2. K-moyennes
 3. DBScan
 4. Clustering hiérarchique
 5. Discussion



CLUSTERING

Permet de regrouper des données **similaires** en clusters

Deux catégories d'approches

- Partitionnement
- Hiérarchique

Pré-requis

Mesure de distance ou de similarité

Evaluation de la qualité

- Minimiser distance intra-cluster
- Maximiser distance inter-cluster

La qualité du clustering dépend de la mesure de distance utilisée, de l'algorithme lui même et de l'application (des données)



CLUSTERING

QUELQUES OUVRAGES DE RÉFÉRENCE

Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* (Vol. 344). John Wiley & Sons.

Han, J., Kamber, M., & Pei, J. (2011). *Data mining: concepts and techniques: concepts and techniques*. Elsevier.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc..

Everitt, B. S. (1979). Unresolved problems in cluster analysis. *Biometrics*, 169-181.



DE L'IMPORTANCE DE NORMALISER

- Dans l'espace euclidien, il est fortement recommandé de normaliser les données
- Soient $\mathbf{x}_i=(0.1, 20)$ et $\mathbf{x}_j=(0.9, 720)$

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(0.9 - 0.1)^2 + (720 - 20)^2} = 700.000457,$$

- La distance est dominée par la seconde dimension
- Nécessité de considérer les dimensions sur le même intervalle de valeurs

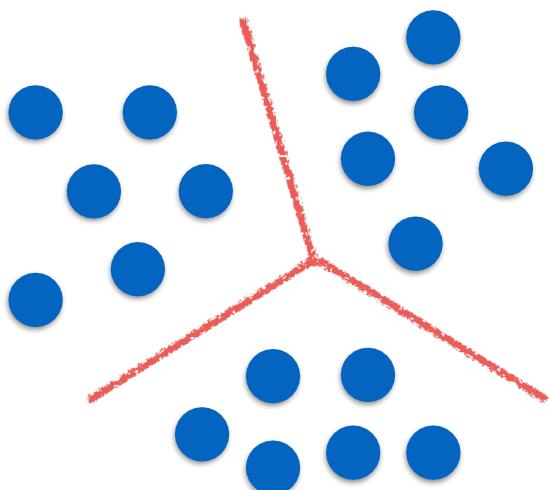


K-MOYENNES

Catégorie

Technique par partitionnement

Principe



- Partitionnement en k groupes
- k est fixé par l'utilisateur



K-MOYENNES

Affectation

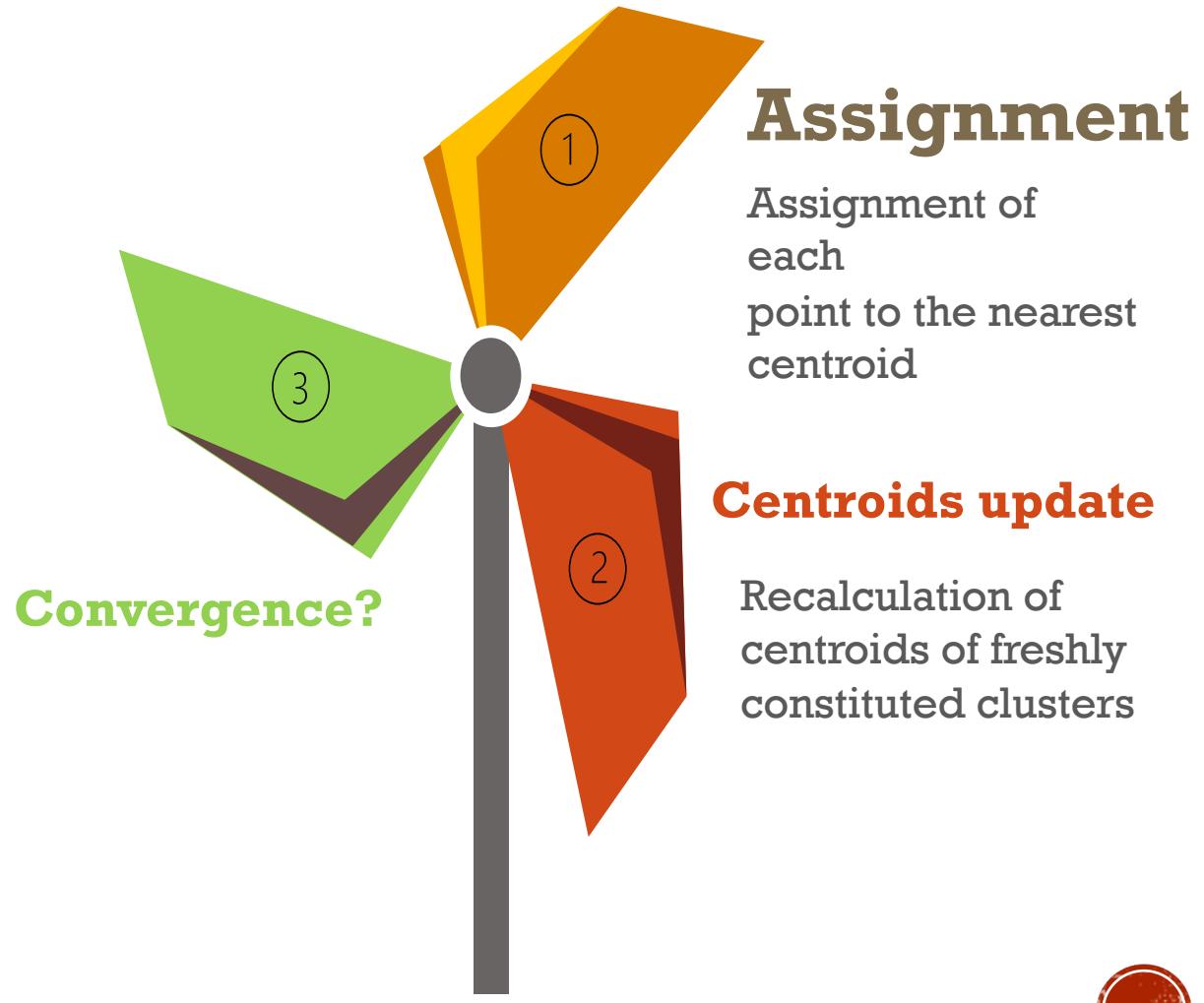
$$z_{ik}^{(t)} = \begin{cases} 1 & \text{si } k = \operatorname{argmin}_{z \in \{1, \dots, K\}} \operatorname{dist}(\mathbf{x}_i, \mu_z) \\ 0 & \text{sinon} \end{cases}$$

Recalage des centres

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^n z_{ik}^{(t)} \times \mathbf{x}_i}{\sum_{i=1}^n z_{ik}^{(t)}}$$

Input

- $X \in \mathbb{R}^d$: Dataset
- $k \in \mathbb{N}$: number of clusters
- $G = \{G_i \in \mathbb{R}^d\}_{i=1 \dots k}$: set of initial centroids



K-MOYENNES

ALGORITHME

Algorithme 1 : K-moyennes

Données : Les données $(\mathbf{x}_1, \dots, \mathbf{x}_N)$ et K le nombre de classes

$t = 0$;

$maxIter = 1000$;

$\mu = (\mathbf{x}_1^{(0)}, \dots, \mathbf{x}_N^{(0)})$;

$\Gamma^{(t)} = +\infty$;

$converge = 0$;

tant que $!converge$ et $t < maxIter$ **faire**

pour $i = 1..n$ **faire**

pour $k = 1..K$ **faire**
 Calculer $z_{ik}^{(t)}$

pour $k = 1..K$ **faire**

 Calculer $\mu_k^{(t+1)}$

 Calculer $\Gamma^{(t+1)}$;

si $\frac{|\Gamma^{(t+1)} - \Gamma^{(t)}|}{|\Gamma^{(t)}|} < \epsilon$ **alors**

$converge = 1$;

$t = t + 1$;



K-MOYENNES

ALTERNATIVES À LA CONVERGENCE

Pas ou peu de ré-assignements de points

$$\left| \sum_{i=1}^N \sum_{k=1}^K z_{ik}^{(t+1)} - \sum_{i=1}^N \sum_{k=1}^K z_{ik}^{(t)} \right| \leq \sigma$$

Pas ou peu de changement dans les centroïdes

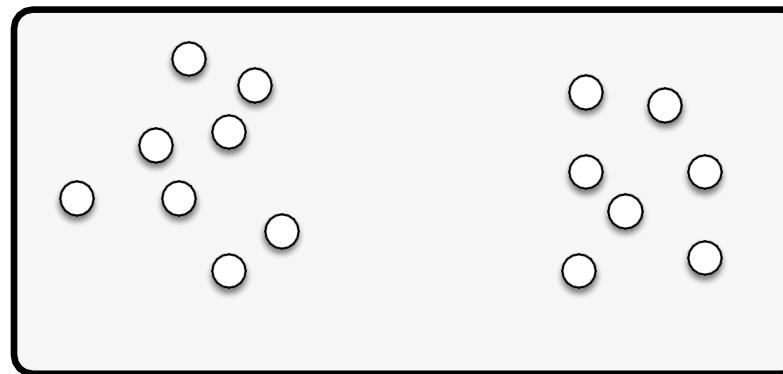
$$\boldsymbol{\mu}_k^{(t)} \approx \boldsymbol{\mu}_k^{(t+1)} \quad \forall k \in 1 \dots K$$



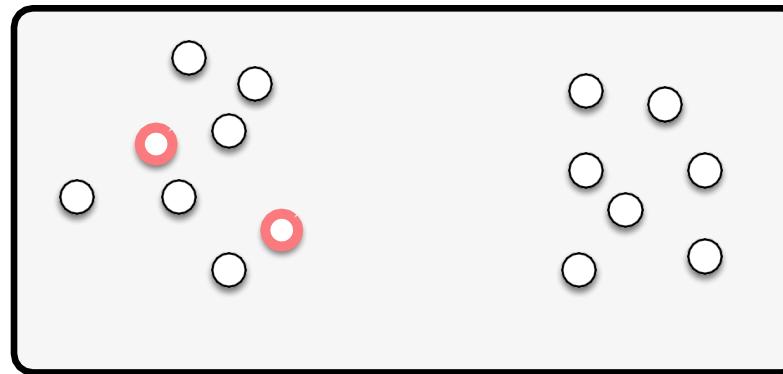
K-MOYENNES

EXEMPLE

Données



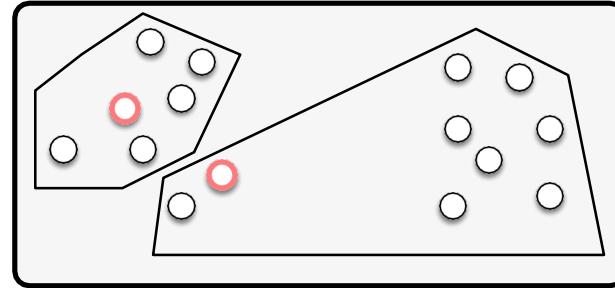
Initialisation



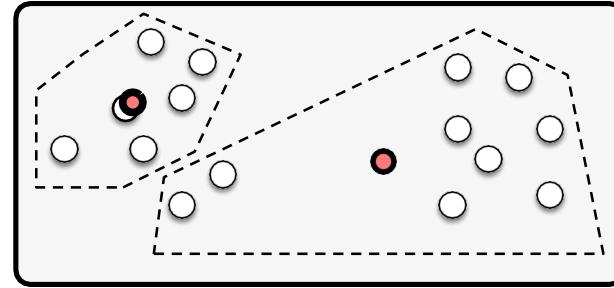
K-MOYENNES

Exemple

Affectation



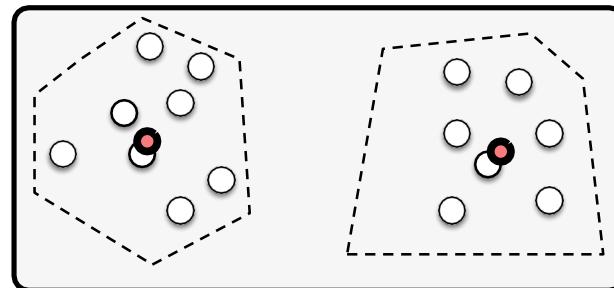
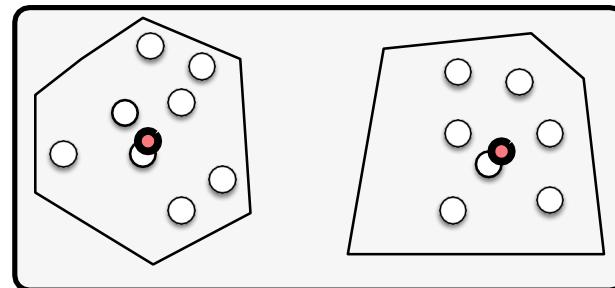
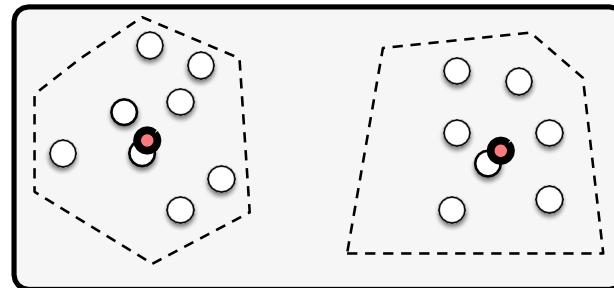
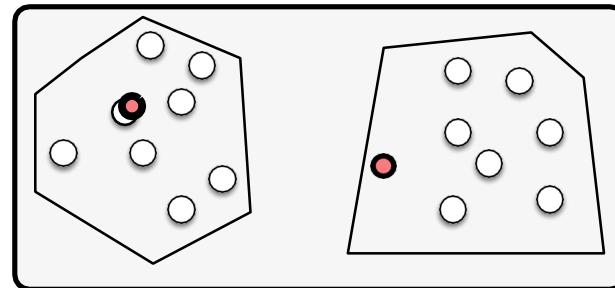
Recalcul



Itération 1

Itération 2

Itération 3



K-MOYENNES

EXERCICE

- $k = 3$
- Premiers centroïdes : A,B et C
- Distance city bloc
- Critère d'arrêt : immobilité des centroïdes

	x	y
A	1	1
B	2	2
C	1	3
D	3	5
E	4	6
F	5	5
G	5	3
H	5	2
I	6	1



K-MOYENNES

FORCES ET FAIBLESSES

AVANTAGES

- Facile à comprendre
- Facile à implémenter
- Efficace (quasi-linéaire)
- Le plus populaires des algorithmes de clustering**

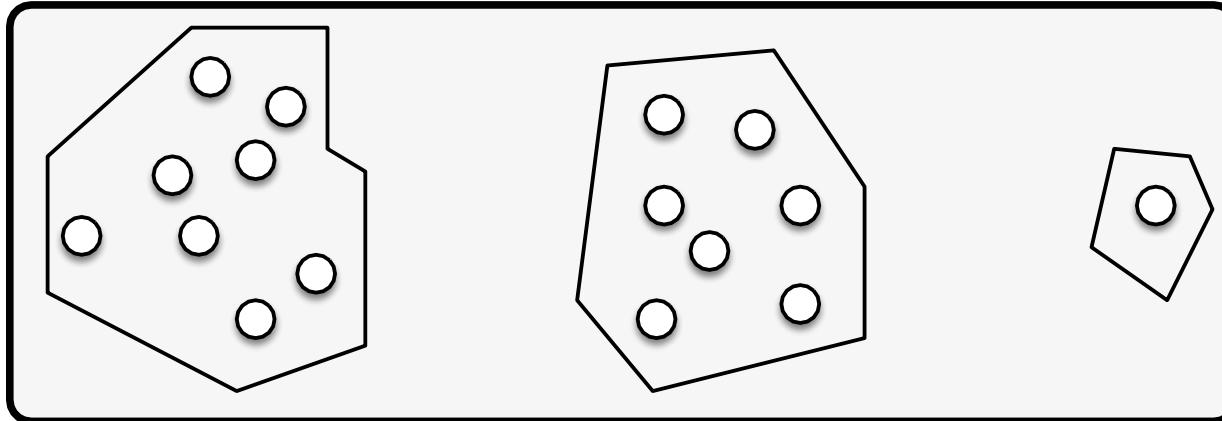
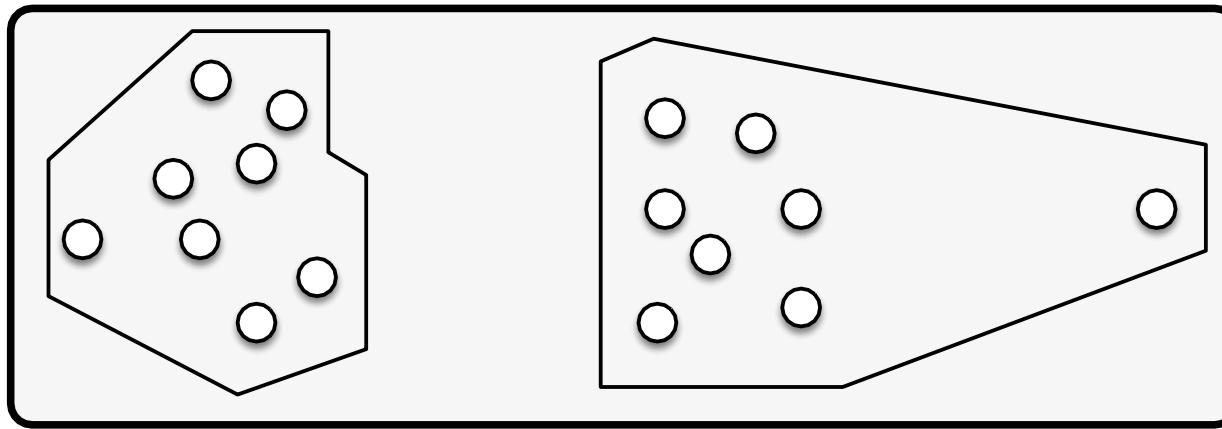
INCONVÉNIENTS

- Nécessite une notion de moyenne / distance
- Le paramètre k
- Sensible aux outliers
- Sensible aux centroïdes initiaux¹⁴
- Non déterministe
- (Hyper-)sphères



K-MOYENNES

SENSIBILITÉ AUX OUTLIERS



K-MOYENNES

SENSIBILITÉ AUX OUTLIERS

Suppression des points qui posent problème

Les points qui sont trop loins des centroïdes

Faire de l'échantillonnage

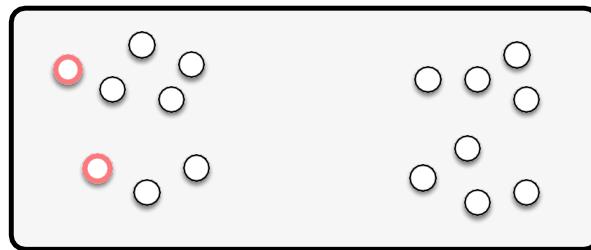
Statistiquement, on a peu de chance de se retrouver avec des outliers



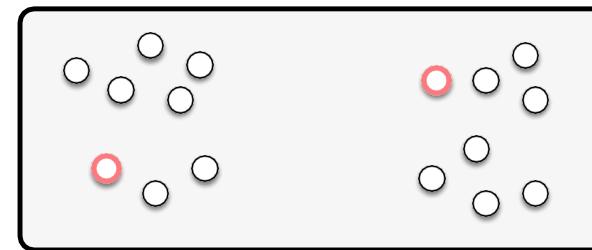
K-MOYENNES

Sensibilité aux centroïdes initiaux

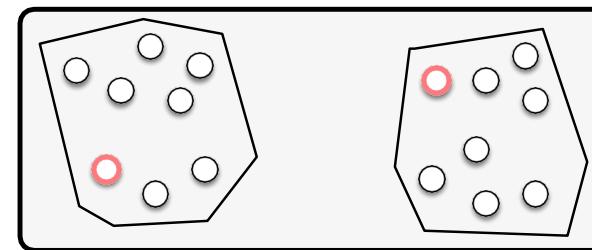
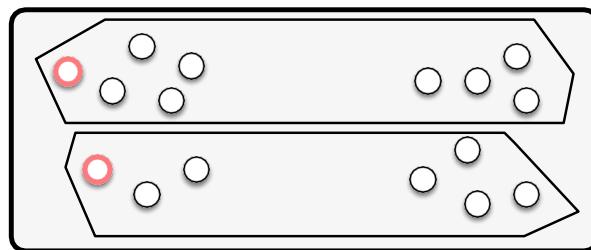
Cas 1



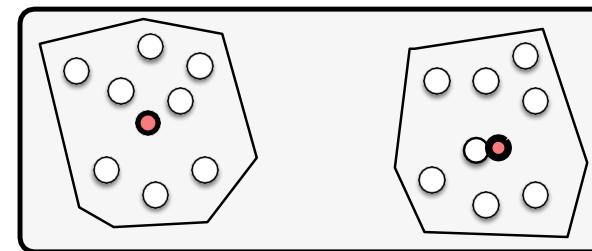
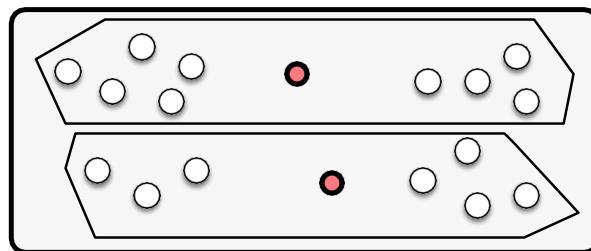
Cas 2



Itération 1



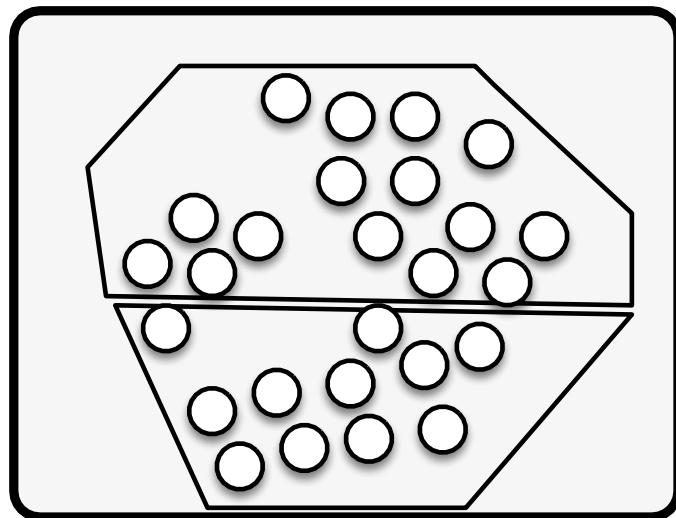
Fin



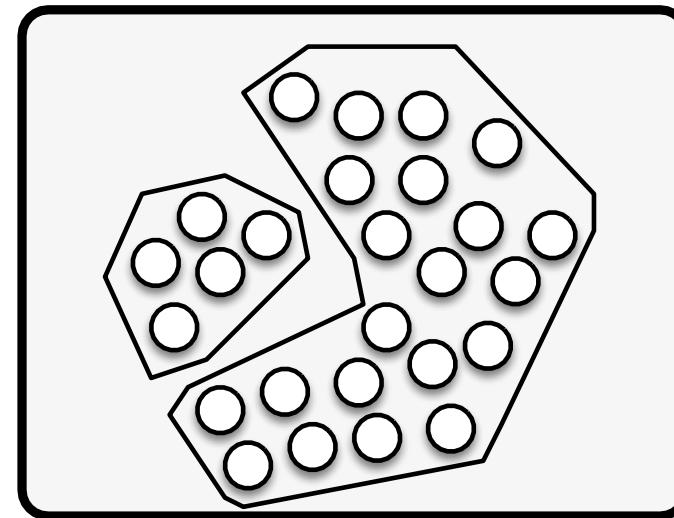
K-MOYENNES

(HYPER-)SPHÈRES

Résultat
k-moyennes



Clustering
idéal



K-MOYENNES

RÉSUMÉ

- Des faiblesses mais très utilisé
- Il n'existe pas de garantie qu'un algorithme soit plus efficace qu'un autre (dépend de l'application ou du type de données)
- La comparaison entre résultats issus de différents algorithmes de clustering est difficile (personne connaît la bonne réponse)



CLUSTERING BASÉ SUR LA DENSITÉ

K-moyennes



Idéalement



20

CLUSTERING BASÉ SUR LA DENSITÉ

Caractéristiques

Clusters de forme arbitraire

Utile pour l'analyse d'image

Gère le bruit

Nécessite la définition de paramètres associés à la densité

Densité : Nombre de points dans un cercle d'un certain rayon

Quelques approches existantes

DBSCAN par Ester, et al. (KDD'96)

DENCLUE par Hinneburg & D. Keim (KDD'98/2006)

OPTICS par Ankerst, et al (SIGMOD'99).

CLIQUE par Agrawal, et al. (SIGMOD'98)



DBSCAN

[Ester, et al., 1996]
Terminologie

Point central

Un point est **central** s'il a au moins **MinPts** points dans son voisinage formé par un cercle de rayon de **Eps** unités

Point bordure

Un point est **bordure** s'il a moins de **MinPts** points dans son voisinage mais est dans le voisinage d'un point central

Point bruit

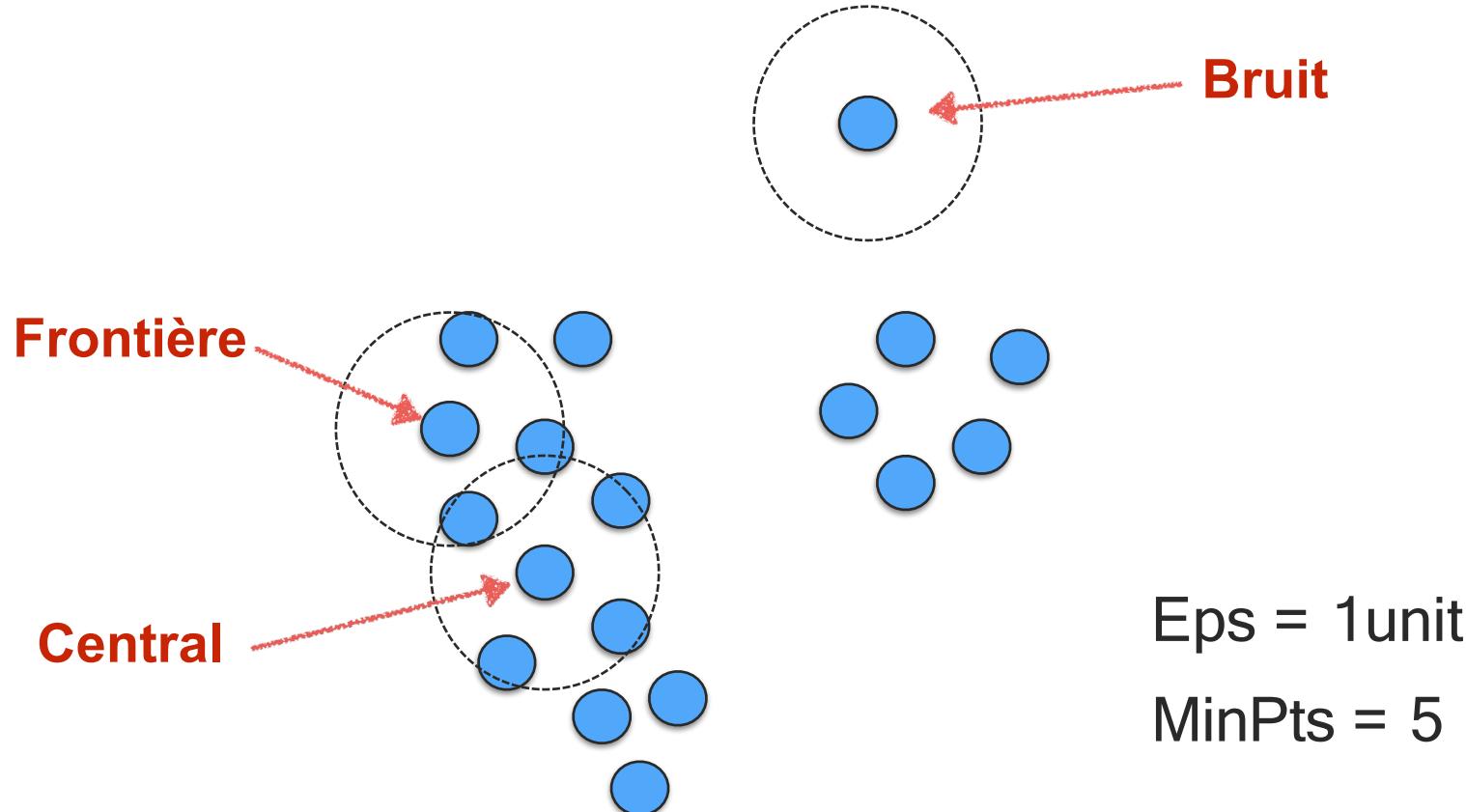
Un point est **du bruit** s'il n'est ni central ni bordure

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231).



DBSCAN

ILLUSTRATION DE LA TERMINOLOGIE



DBSCAN

PRINCIPES

Points centraux

Deux points centraux qui sont à moins de Eps unités l'un de l'autre sont dans le même cluster

Points bordures

Tout point central qui est dans le voisinage d'un point central appartient au même cluster que lui

Points bruits

Le bruit n'est pas considéré



DBSCAN

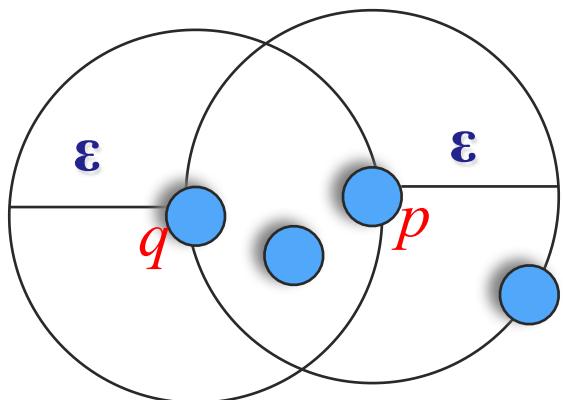
CONCEPT : ϵ -VOISINAGE

ϵ -voisinage de p

Les points qui sont dans un voisinage de ϵ

Point central

Le point p est central si la taille de son voisinage est au minimum MinPts



Avec MinPts = 4 :

- p central (MinPts = 4)
- q ne l'est pas



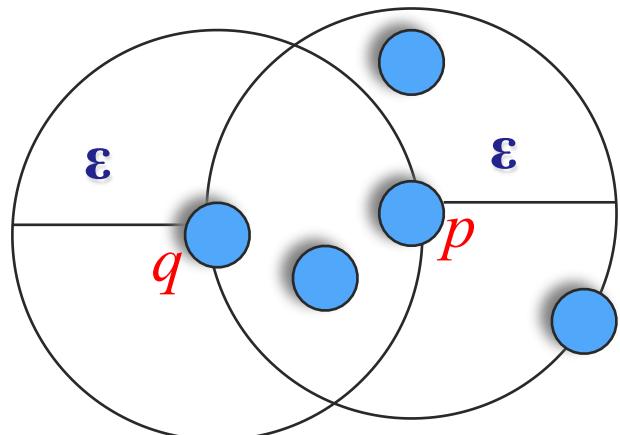
DBSCAN

CONCEPT : ACCESSIBILITÉ

Directement densité-accessible

Un point q est *directement densité-accessible* depuis un point p si

q est dans l' ε -voisinage de p et p est un point central



- q est *directement densité-accessible* depuis p
- p n'est pas *directement densité-accessible* depuis p

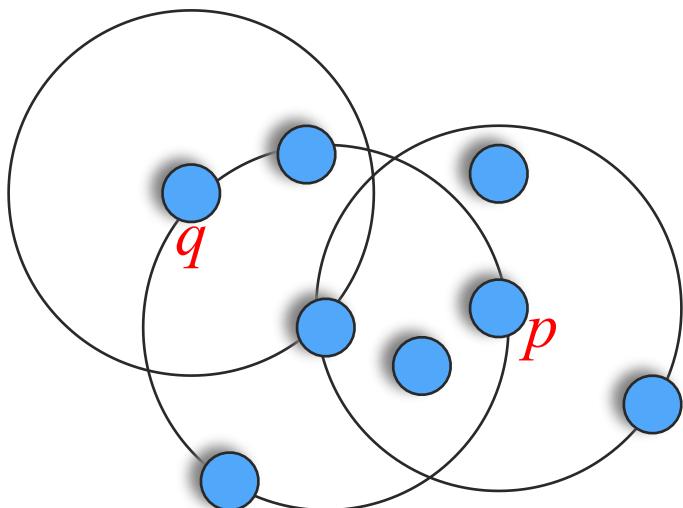


DBSCAN

CONCEPT : ACCESSIBILITÉ

Densité-accessible

Un point **p** est *densité-accessible* depuis un point **q** s'il existe une chaîne p_1, \dots, p_n , avec $p_1=q$ et $p_n=p$, tel que p_{i+1} est directement accessible depuis p_i pour tout i dans $[2, n]$



- q est *densité-accessible* depuis p
- p n'est pas *densité-accessible* depuis p
- Fermeture transitive de la relation directement densité-accessible
- Asymétrique

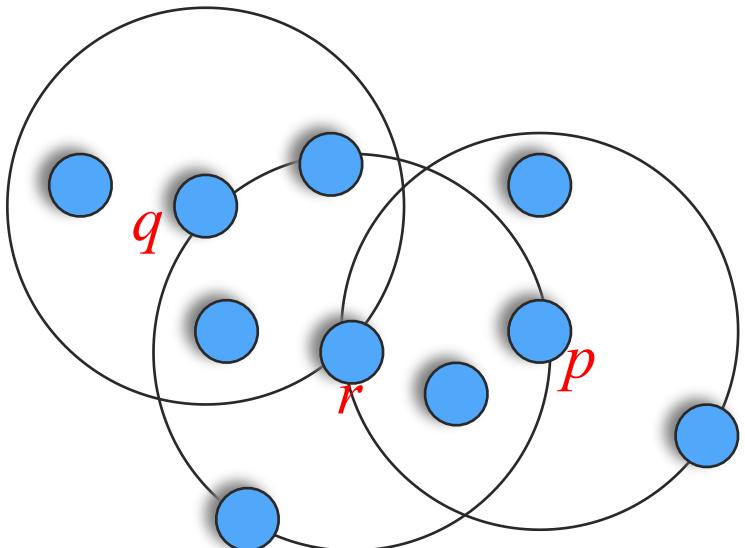


DBSCAN

CONCEPT : ACCESSIBILITÉ

Densité-connectivité

Un point **p** est densité-connecté à **q** s'il existe un point **o** qui soit *densité-accessible* à **p** et **q**



- **p** et **q** sont densité-connectés par **r**
- Symétrique



DBSCAN

CLUSTER ET BRUIT

Cluster

Un cluster **C** est un ensemble non vide de points **D** satisfaisant les conditions suivantes :

1. **Maximalité** : pour tous points **p** et **q**, si **p** est dans **C** et si **q** est densité-accessible depuis **p**, alors **q** est aussi dans **C**
2. **Connectivité** : pour tous points **p** et **q** dans **C**, **p** et **q** sont densité-connectés

C contient aussi bien des points centraux que bordures

Bruit

Tous les points qui ne sont pas directement densité-accessible depuis au moins un point central



DBSCAN

ALGORITHME

Choisir un point **p**

Retrouver tous les points densité-accessibles depuis **p** par rapport aux paramètres utilisateurs ϵ et **MinPts**

Si **p** est central, un cluster est créé avec tous ces points

Si **p** est bordure, aucun point n'est densité-accessible depuis lui et DBSCAN passe à un autre point non traité de la base de données

On continue le processus jusqu'à ce que tous les points aient été traités

Le résultat est indépendant de l'ordre de traitement des points

Complexité

- Spatiale : $O(n)$
- Temporelle : $O(n^2) \rightarrow$ vérification de la centralité d'un point

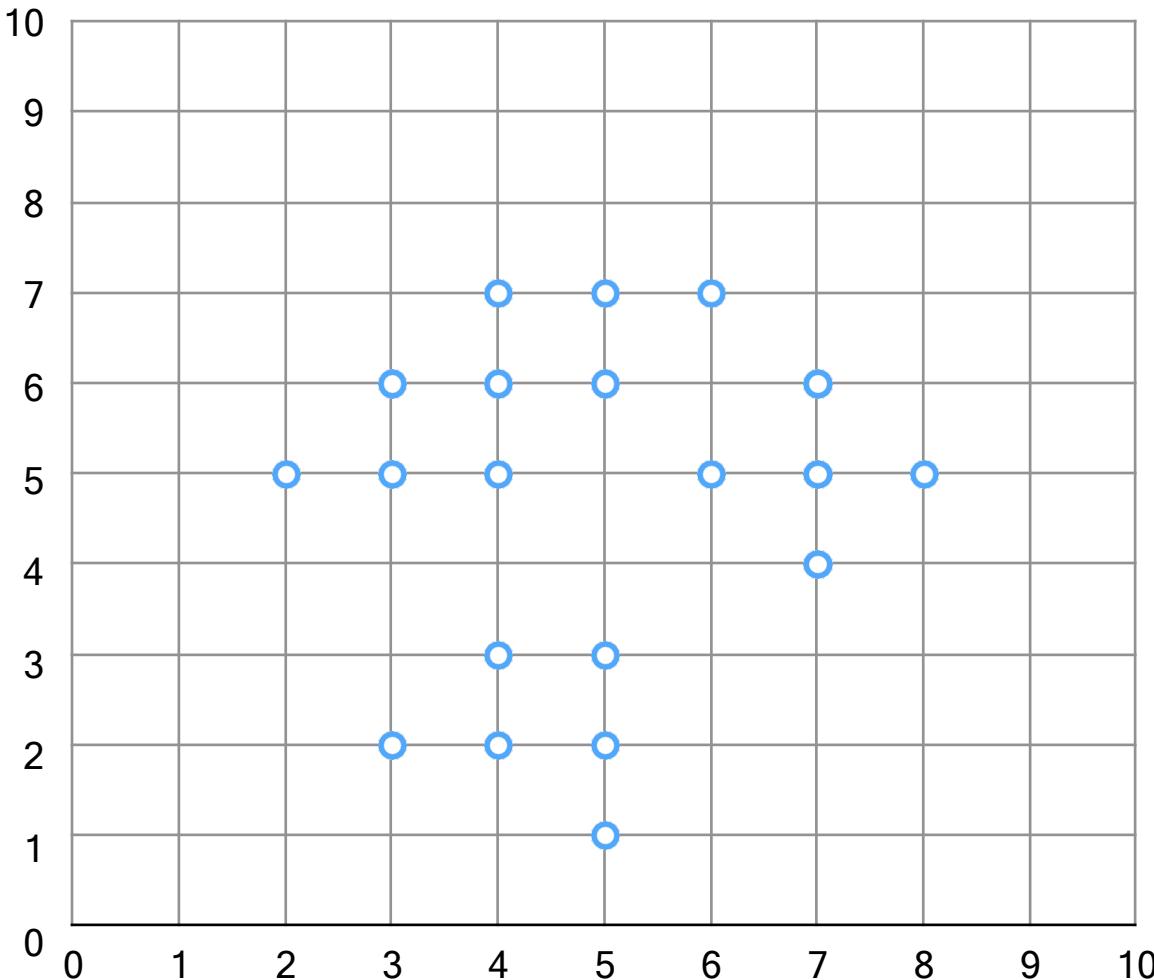


DBSCAN

EXAMPLE

$\epsilon = 1$

MinPts = 4



DBSCAN

FORCES ET FAIBLESSES

AVANTAGES

- Forme des clusters libre
- Insensible au bruit
- Détection d'outlier
- Complexité temporelle plus faible que les K-moyennes

INCONVÉNIENTS

- Fonctionne mal en haute dimensions
- Paramètres utilisateurs difficiles à définir
- Sensible quand les densités sont inégales

32



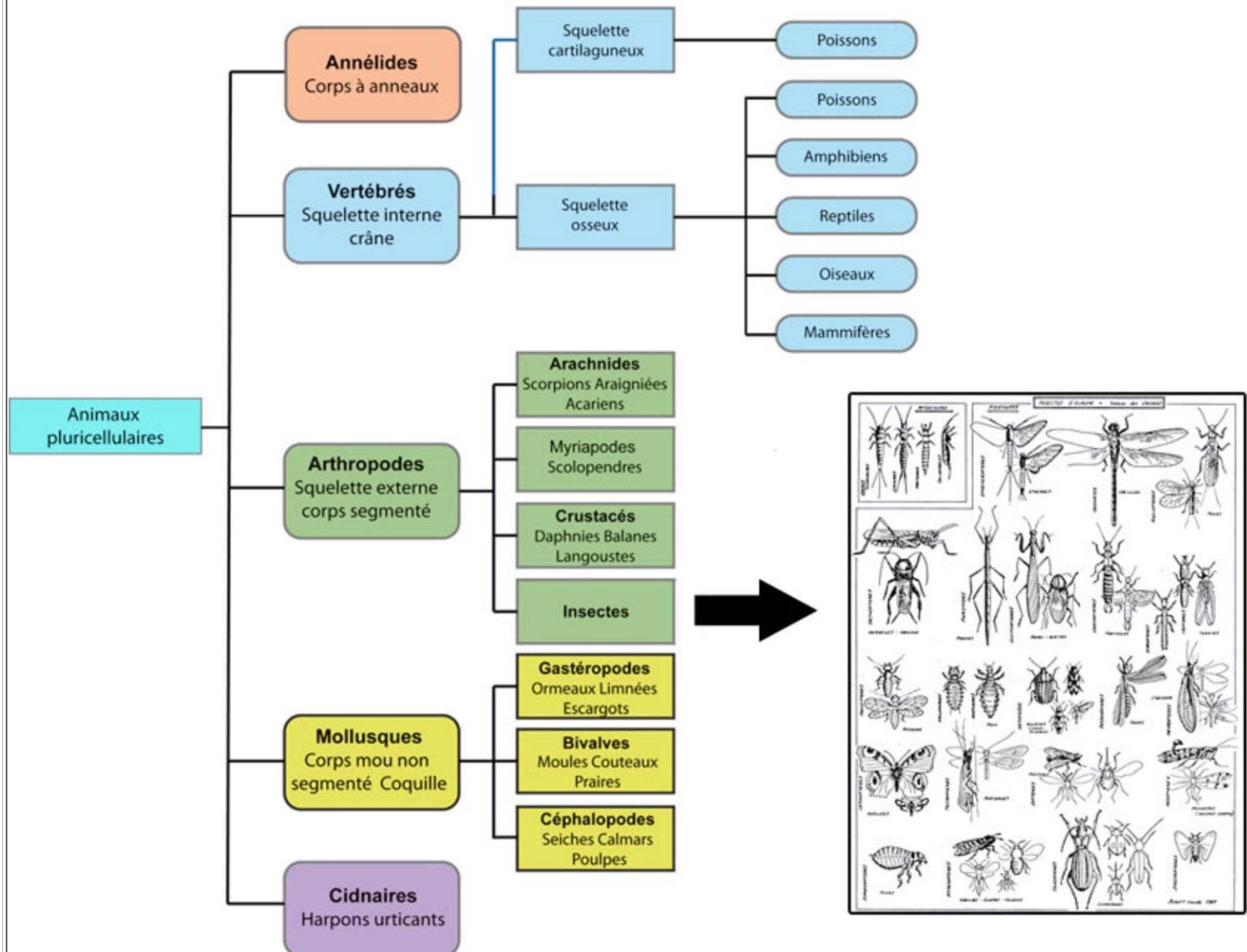
AGGLOMÉRATIF



CAH

**LA CLASSIFICATION ASCENDANTE
HIÉRARCHIQUE**





CAH - PRINCIPES

- Objectifs : production d'une structure (arborescence) permettant :
 - la mise en évidence de liens hiérarchiques entre individus ou groupes d'individus
 - la détection d'un nombre de classes « naturel » au sein de la population



CAH - PRINCIPES

- **Définition**

- Classification : action de constituer ou construire des classes
- Classe : ensemble d'individus (ou d'objets) possédant des traits de caractères communs (groupe, catégorie)

- **Exemples**

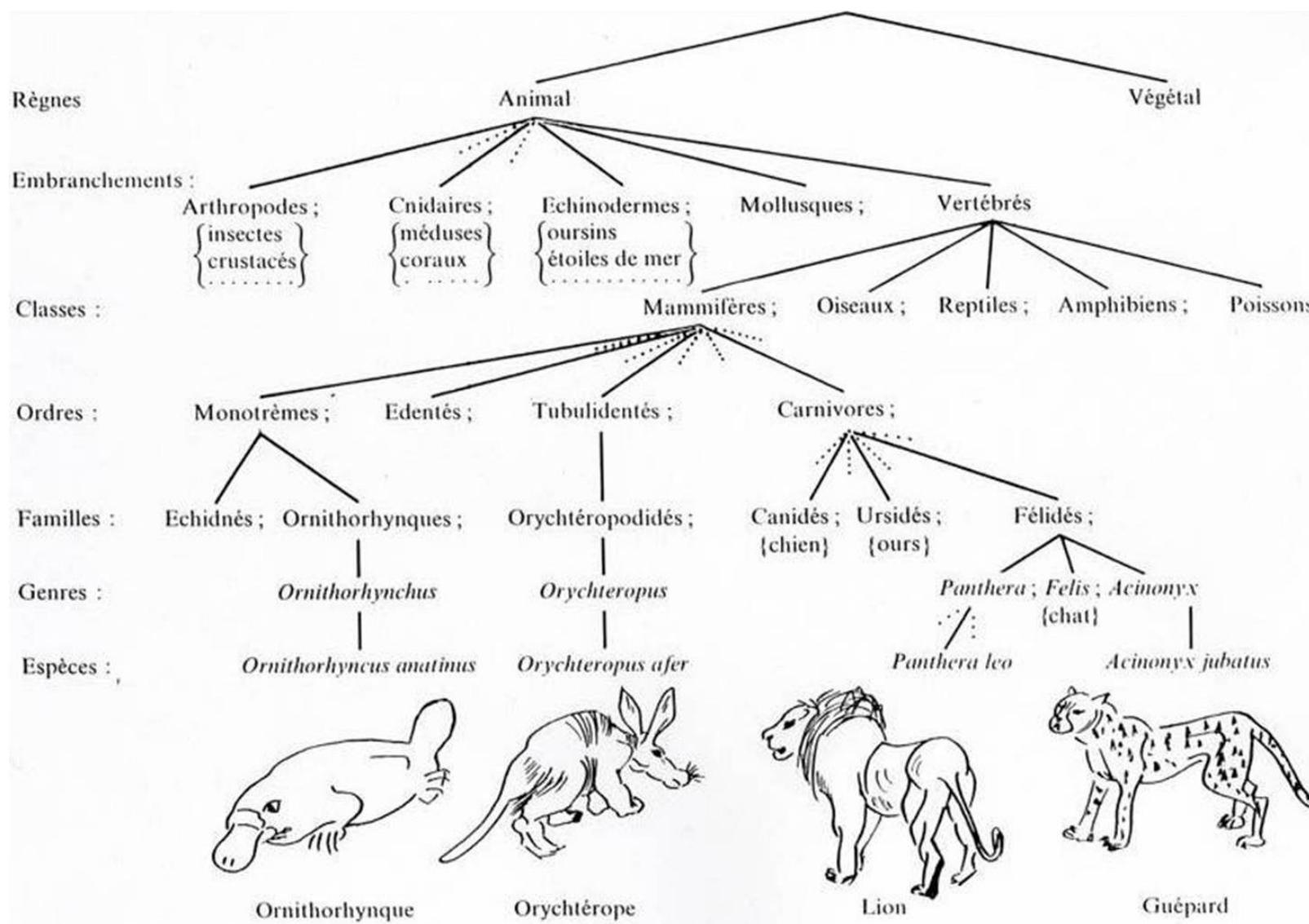
- de classification : règne animal, disque dur d'un ordinateur, division géographique de la France, etc.
- de classe : classe sociale, classe politique, etc.

- **Deux types de classification**

- hiérarchique : arbre, CAH
- méthode de partitionnement : partition



CAH - PRINCIPES



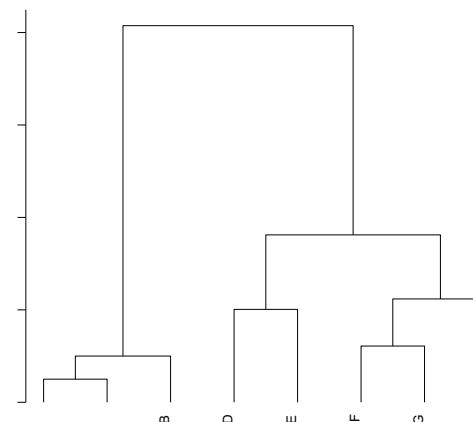
QUELLES DONNÉES POUR QUELS OBJECTIFS

La classification s'intéresse à des tableaux de données individus \times variables quantitatives

Objectifs : production d'une structure (arborescence) permettant :

- la mise en évidence de liens hiérarchiques entre individus ou groupes d'individus
- la détection d'un nb de classes « naturel » au sein de la population

k	K
x_{ik}	



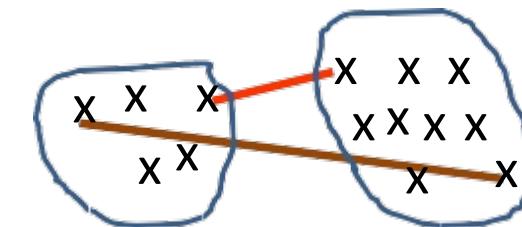
CRITÈRES

Ressemblance entre individus :

- distance euclidienne
- indice de similarité
- ...

Ressemblance entre groupes d'individus :

- saut minimum ou lien simple (**plus petite distance**)
- lien complet (**plus grande distance**)
- critère de Ward



DISTANCES POUR LES CRITÈRES

Notion importante, cf distances

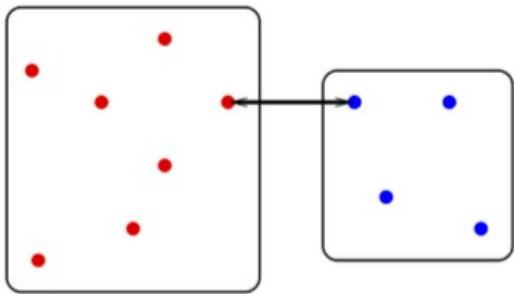
- ▶ ressemblance entre individus = distance
- ▶ ressemblance entre groupes d'individus = critère d'agrégation
 - ▶ lien simple
 - ▶ lien complet
 - ▶ lien moyen
 - ▶ critère de Ward



DISTANCES POUR LES CRITÈRES

Distances entre groupes

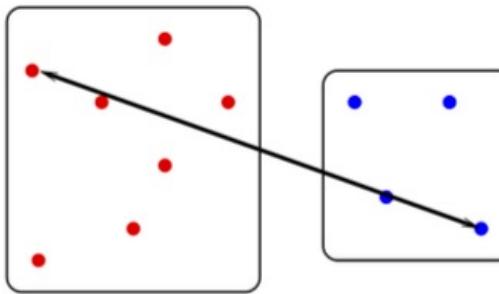
Single



Classe 1

Classe2

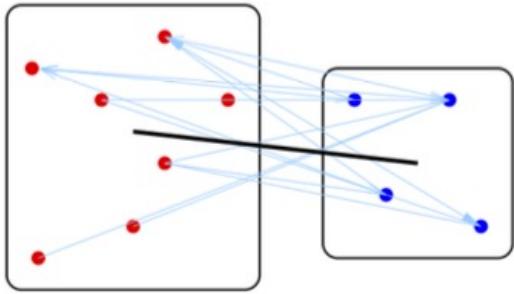
Complete



Classe 1

Classe2

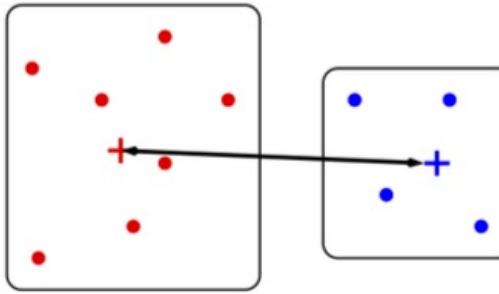
Average



Classe 1

Classe2

Ward



Classe 1

Classe2



FONCTIONNEMENT DE L'ALGORITHME

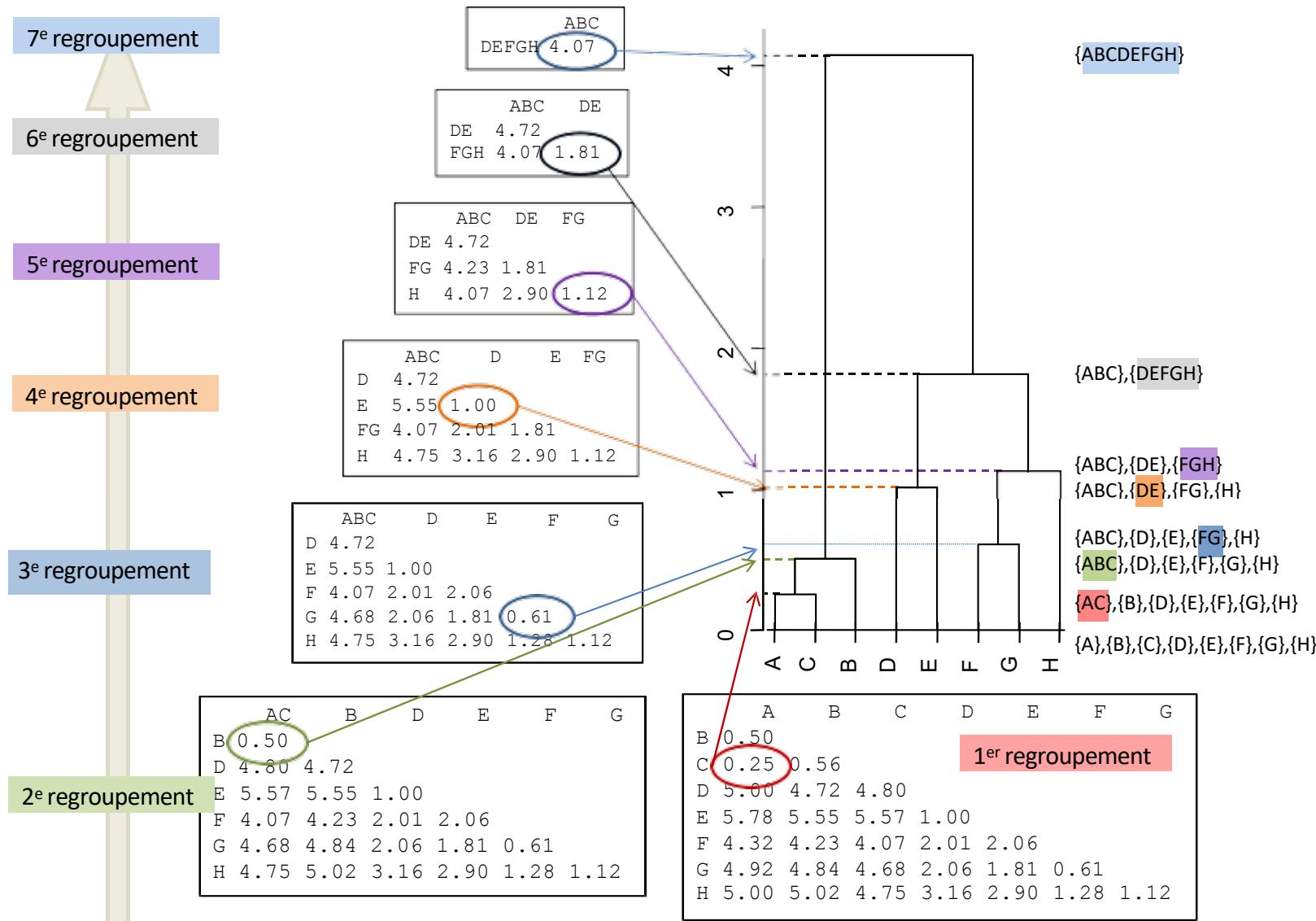
L'algorithme

étape 1 :

- ▶ départ : n individus = n clusters distincts
- ▶ calcul des distances entre tous les individus
 - ▶ choix de la métrique à utiliser en fonction du type de données
- ▶ regroupement des 2 individus les plus proches => $(n-1)$ clusters

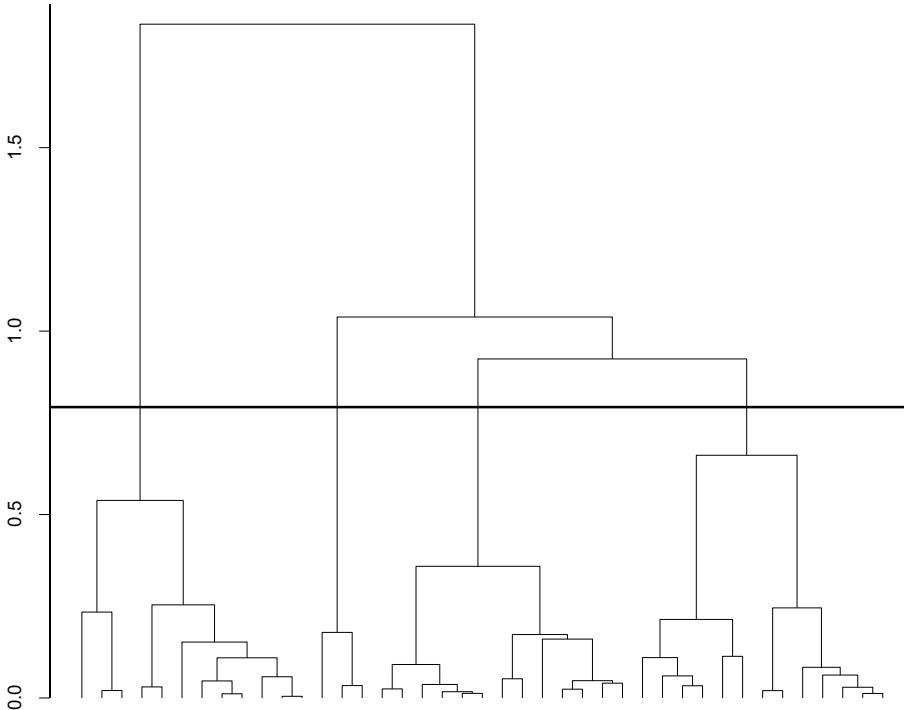


FONCTIONNEMENT DE L'ALGORITHME



ARBRES ET PARTITIONS

En définissant un niveau de coupure, on construit une partition



Remarque : vu le mode de construction, la partition n'est pas optimale mais est intéressante



QUALITÉ DE PARTITION

Quand une partition est-elle bonne ?

- Si les individus d'une même classe sont proches
- Si les individus de 2 classes différentes sont éloignés

Et mathématiquement ça se traduit par ?

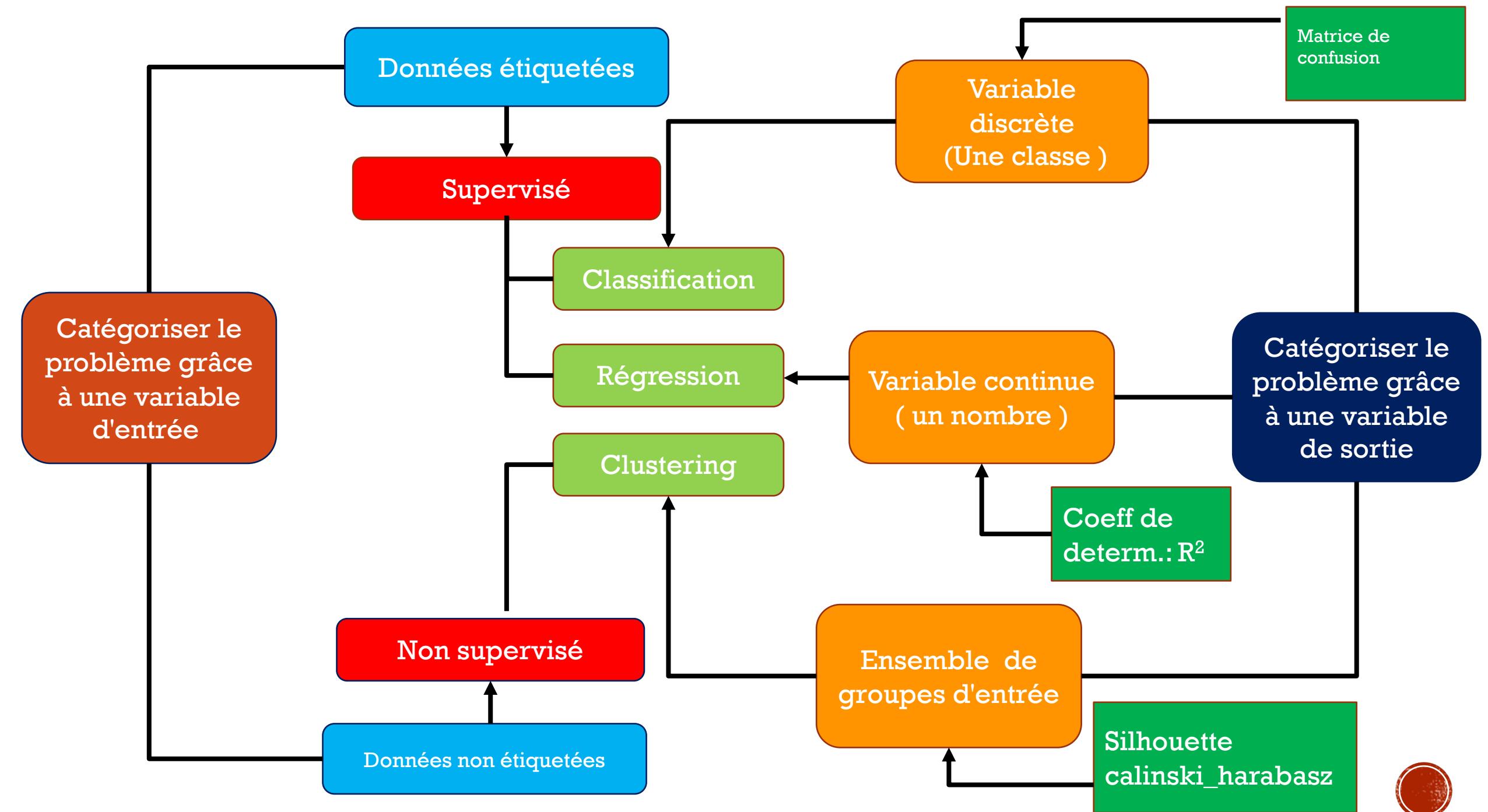
- Variabilité intra-classe petite
- Variabilité inter-classes grande

==> Deux critères, lequel choisir ?



EVALUATION DES MODÈLES





MATRICE DE CONFUSION

- La matrice de confusion est une corrélation entre les prédictions d'un modèle et les étiquettes de classe réelles des points de données.

Confusion Matrix

TPR (taux vraiment positif) = (vrai positif / réel positif)

TNR (taux vraiment négatif) = (vrai négatif / réel négatif)

FPR (taux de faux positifs) = (faux positifs / réels négatifs)

FNR (taux de faux négatifs) = (faux négatifs / réels positifs)

		Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)	
	False Negatives (FNs)	True Negatives (TNs)	

Un modèle intelligent: TPR ↑, TNR ↑, FPR ↓, FNR ↓



MATRICE DE CONFUSION

- La matrice de confusion est une corrélation entre les prédictions d'un modèle et les étiquettes de classe réelles des points de données.

<i>Réel</i>		
<i>Estimé</i>	+	-
+	VP	FP
-	FN	VN

14% des poissons sont pris pour des papillons

	Av	Pl	Ut	Mi	Ch	Po	Ve	Pa	Por	Fi	Vo	Fl
Avi	67%	2%	-	-	2%	2%	10%	10%	4%	2%	-	-
Pla	-	21%	-	2%	7%	19%	10%	12%	5%	-	19%	5%
Uta	17%	-	33%	-	7%	-	-	3%	10%	10%	13%	7%
Min	-	-	-	100%	-	-	-	-	-	-	-	-
Chi	26%	5%	7%	-	14%	9%	12%	9%	5%	2%	12%	-
Poi	5%	13%	3%	8%	-	13%	18%	21%	-	3%	10%	8%
Ver	2%	2%	-	-	10%	7%	43%	-	21%	5%	7%	2%
Pap	6%	6%	-	-	2%	14%	14%	35%	6%	-	12%	4%
Por	2%	2%	-	-	-	2%	-	12%	70%	10%	-	2%
Fig	-	-	-	-	-	-	6%	-	24%	70%	-	-
Voi	21%	6%	-	-	4%	4%	8%	4%	4%	29%	19%	-
Fle	2%	9%	-	-	-	9%	21%	14%	-	-	16%	28%



EVALUATION DU MODÈLE



PROBLÈME DE CLASSIFICATION

- Taux de classification correcte - Accuracy
- Taux de vrais positifs - Precision
- Capacité à identifier les labels positifs - Recall (rappel)
- Mesure synthétique (moyenne harmonique) de la précision et du rappel - F1 Score



PROBLÈME DE CLASSIFICATION

PRÉCISION 1/4

- Précision = Prédiction correctes / Prédiction totale

En utilisant une matrice de confusion,

$$\text{Précision} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$



PROBLÈME DE CLASSIFICATION PRÉCISION ET RAPPEL (2/4)

- Précision: C'est le rapport des vrais positifs (TP) et du total des prédictions positives. Fondamentalement, cela indique combien de fois votre prédiction positive a été réellement positive
- Rappel : nous indique sur tous les points positifs combien ont été prédits positifs (TPR - True Positive Rate)



PROBLÈME DE CLASSIFICATION

MESURE F (3/4)

- Mesure F: Harmonique moyen de précision et de rappel.



RÉGRESSION § RAPPEL

- Prédire les ventes d'un produit particulier le mois prochain
- Impact de l'alcoolémie sur la coordination
- Prévoyez les ventes mensuelles de cartes-cadeaux et améliorez les projections de revenus annuels



REGRESSION COEFFICIENT DE DÉTERMINATION - R²

- Il est désigné par R². Lors de la prédiction des valeurs cibles de l'ensemble de test, nous rencontrons quelques erreurs (e_i), qui sont la différence entre la valeur prédictée et la valeur réelle.
- Disons que nous avons un ensemble de test avec n entrées. Comme nous le savons, tous les points de données auront une valeur cible, disons [y₁, y₂, y₃y_n]. Prenons les valeurs prédictées des données de test comme [f₁, f₂, f₃, f_n].
- Calculer la Somme résiduelle des carrés, qui est la somme de toutes les erreurs (e_i) au carré, en utilisant cette formule où f_i est la valeur cible prédictée par un modèle pour le i^{ème} point de données.



RÉGRESSION

- Mean squared error
- Root Mean squared error
- Mean Absolute Error
- Mean Absolute Percentage Error



Distances

APPRENTISSAGE NON SUPERVISE



DISTANCES / SIMILARITÉS

Rappels mathématiques

Distance : trois propriétés à satisfaire

- Symétrie : $\forall A, B \in E^2 \text{ on a } d(A, B) = d(B, A)$
- Séparation : $\forall A, B \in E^2 \text{ on a } d(A, B) = 0 \Leftrightarrow A = B$
- Inégalité triangulaire : $\forall A, B, C \in E^3 \text{ on a } d(A, C) \leq d(A, B) + d(B, C)$

Similarité : deux propriétés à satisfaire

- Symétrie : $\forall A, B \in E^2 \text{ on a } d(A, B) = d(B, A)$
- Séparation : $\forall A, B \in E^2 \text{ on a } d(A, B) = 0 \Leftrightarrow A = B$
- ~~Inégalité triangulaire : $\forall A, B, C \in E^3 \text{ on a } d(A, C) \leq d(A, B) + d(B, C)$~~

De nombreuses fonctions selon le type de données (binaires, numériques, catégorielles) et l'application



DISTANCES

ATTRIBUTS NUMÉRIQUES

Notation

On note $\text{dist}(x_i, x_j)$ la distance entre x_i et x_j

Distance de Minkowski

$$\text{dist}(\mathbf{x}_i, \mathbf{x}_j) = ((x_{i1} - x_{j1})^h + (x_{i2} - x_{j2})^h + \dots + (x_{ir} - x_{jr})^h)^{\frac{1}{h}}$$

Deux cas spéciaux très utilisés sont :

1. Distance euclidienne
2. Distance de Manhattan (city block)



DISTANCES

Attributs numériques

Si $h = 2$ (distance euclidienne)

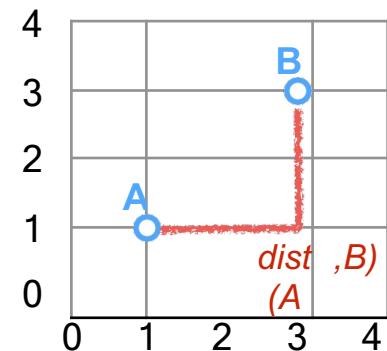
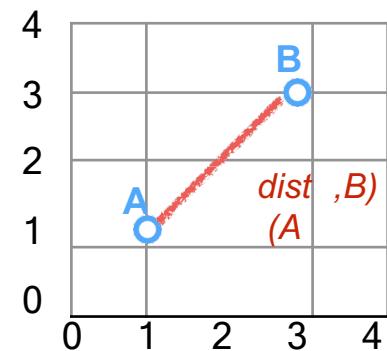
$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \dots + (x_{ir} - x_{jr})^2}$$

Si $h = 1$ (distance de Manhattan)

$$dist(\mathbf{x}_i, \mathbf{x}_j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ir} - x_{jr}|$$

Distance euclidienne pondérée

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{w_1(x_{i1} - x_{j1})^2 + w_2(x_{i2} - x_{j2})^2 + \dots + w_r(x_{ir} - x_{jr})^2}$$



DISTANCES

ATTRIBUTS NUMÉRIQUES

Distance euclidienne au carré

Donner plus de poids aux points qui sont vraiment loin (les défavoriser)

$$dist(\mathbf{x}_i, \mathbf{x}_j) = (x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \dots + (x_{ir} - x_{jr})^2$$

Distance de Chebychev

Définir un point comme différent si au moins un attribut diffère

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \max(|x_{i1} - x_{j1}|, |x_{i2} - x_{j2}|, \dots, |x_{ir} - x_{jr}|)$$



DISTANCES

ATTRIBUTS BINAIRES

Rappels

Un attribut qui n'a que deux valeurs

Pas de notion d'ordre

Exemple : le genre (masculin, féminin)

Comment ?

Utilisation d'une matrice de confusion pour définir les distances

Notation

x_i et x_j représentent deux points



DISTANCES

ATTRIBUTS BINAIRES

- **a** : le nombre d'attributs avec 1 pour les deux points
- **b** : le nombre d'attributs avec $x_{if}=1$ et $x_{jf}=0$
- **c** : le nombre d'attributs avec $x_{if}=0$ et $x_{jf}=1$
- **d** : le nombre d'attributs avec 0 pour les deux points

Point j

	1	0
1	a	b
0	c	d

Point i



DISTANCES

ATTRIBUTS BINAIRES SYMÉTRIQUES

Symétrique

- Les deux valeurs ont la même importance
- Le même poids
- *Exemple : le genre*

Distance

Le coefficient de correspondance simple :

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \frac{b + c}{a + b + c + d}$$



DISTANCES

ATTRIBUTS BINAIRES SYMÉTRIQUES

Exemple

<i>Point i</i>	1	0	1	0	0	1	1	0
<i>Point j</i>	0	1	1	0	1	1	0	0

Distance ?



DISTANCES

ATTRIBUTS BINAIRES ASYMÉTRIQUES

Asymétrique

- Les deux valeurs n'ont la même importance
- Ou la même fréquence
- Par convention, 1 représente l'état le plus important (le plus rare)

Distance

Le coefficient de Jaccard

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \frac{b + c}{a + b + c}$$

Remarque

Variations possibles (ajout de poids)



DISTANCES

ATTRIBUTS NOMINAUX

Nominaux

- Un attribut avec plusieurs modalités
- Pas d'ordre

Distance

- Basée sur le coefficient de correspondance simple
- Soient 2 points \mathbf{x}_i et \mathbf{x}_j , r le nombre d'attributs et q le nombre d'attributs avec des valeurs identiques

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \frac{r - q}{r}$$



DISTANCES

TEXTE

Texte

- Texte = séquence de phrases
- Phrase = séquence de mots
- Simplification : sac de mots
- 1 document = 1 vecteur de mots
- Similarité plus courante que distance

Similarité cosinus

$$\text{Similarity} = \frac{A \times B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n (A_i \times B_i)}{\sqrt{\sum_{i=1}^n (A_i^2)} \times \sqrt{\sum_{i=1}^n (B_i^2)}}$$



DISTANCES

TEXTE

Exemple

<i>Texte i</i>	2	1	1	1	0
<i>Texte j</i>	0	1	2	0	1

Similarité ?



COMPARAION CAH & K-MEANS



CAH AVEC PYTHON

```
4
5 import seaborn as sns
6
7 #librairies pour la CAH
8
9 from matplotlib import pyplot as plt
10
11 from scipy.cluster.hierarchy import dendrogram, linkage, fcluster
12 from sklearn.cluster import KMeans
13 from sklearn import metrics as mc
14
15 import pandas as pd
16
17 # Charger le jeu de données iris
```



CAH AVEC PYTHON

```
22
23
24 # ici le fonctionnement de dendrogram – https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html
25 #générer la matrice des liens
26
27 # Plusieurs méthodes de linkage sont à tester : "single", "average", "weighted", "centroid" ou "ward".
28 # ward compare homogénéité avant & après fusion
29 # De même, la distance peut être "euclidean", "l1", "l2", "manhattan", "cosine", ou "precomputed"
30
31 Z = linkage(iris.iloc[:,0:4],method='ward', metric='euclidean')
32
33
34 #affichage du dendrogramme
```



CAH AVEC PYTHON

```
Z = linkage(iris.iloc[:,0:4],method='ward', metric='euclidean')

#affichage du dendrogramme

plt.title("CAH")

dendrogram(Z,labels=iris.index,orientation='top',color_threshold=4)

plt.show()
```

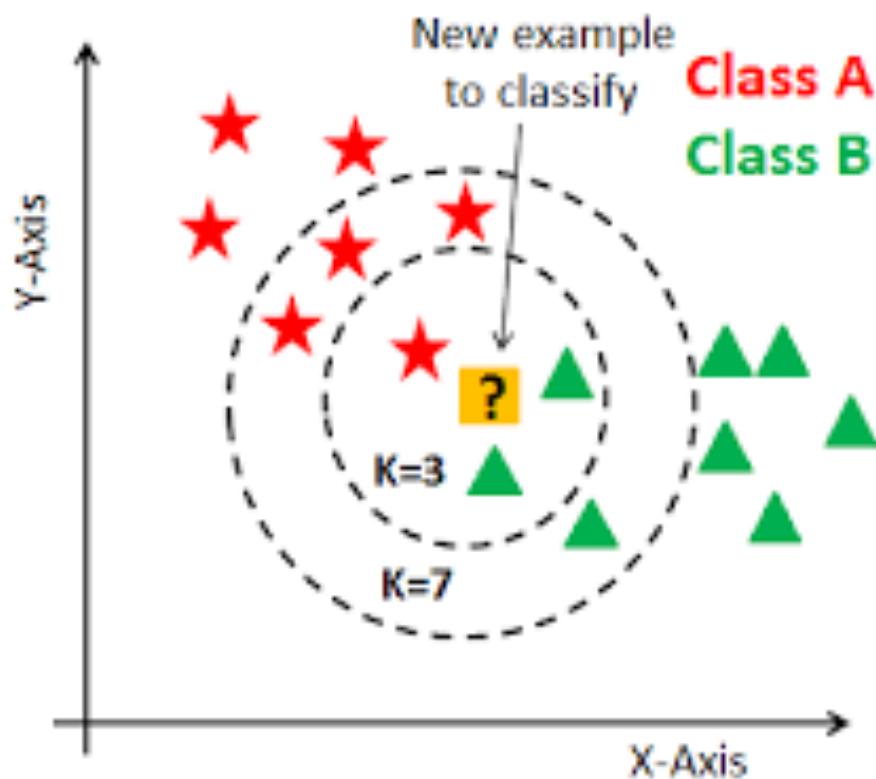


CLASSIFICATION SUPERVISEE § KNN

- Algorithme KNN (Plus proche voisin)
- Objectif : classifier les nouvelles données
 - Etape 1 : Input : K voisins (impair)
 - Etape 2 : Calculez la distance (Euclidienne, ...)
 - Pour déterminer les K plus proche
 - Parmi ces K voisins, comptez le nombre de points appartenant à chaque catégorie.
- Etape 3 : Affecter l'individu à la classe majoritaire



CLASSIFICATION SUPERVISEE



CLASSIFICATION SUPERVISEE

- Algorithme KNN - domaine d'application
 - Domaine de la recommandation
 - Technologies comme l'OCR
 - Prêt bancaire
 - Jeux collaboratifs



CLASSIFICATION SUPERVISEE

- KNN - Avantages
 - simple
 - Pas trop de paramétrage
 - Peut être utilisé aussi pour la régression
- KNN – Inconvénients
 - Mauvaises performances face à un jeu de données volumineux

