



# Data Mining

---

**Pr.Mohamed EL FAR**

# Plan

---

## ❖ Introduction au Datamining

- ✓ Définition du au Datamining
- ✓ Pourquoi le Datamining
- ✓ Description du processus de découverte de connaissances KDD (Knowledge Database Discovery) et étapes du processus KDD
- ✓ Applications

## ❖ Les données

- ✓ Types de variables
- ✓ Transformation d'une variable quantitative en variable qualitative
- ✓ Les données
- ✓ Nuage de points
- ✓ Description d'une variable quantitative
- ✓ Matrice de Corrélation

## ❖ Traitement des données

- ✓ Nettoyage des données
- ✓ Intégration des données
- ✓ Transformation des données
- ✓ Sélection des données
- ✓ Réduction des données

## ❖ Taches du Data Mining

- ✓ Classification
- ✓ Clustering (Segmentation)
- ✓ Règles d'association

# Chapitre 1

## Introduction au Data Mining

---

# Introduction Générale

---



# Pourquoi le Data Mining

---



# Data Mining




---

Un procédé d'exploration et d'analyse de grands volumes de données en vue d'une part de les rendre plus compréhensibles et d'autre part de découvrir des corrélations significatives, c'est-à-dire des règles de classement et de prédiction dont la finalité est l'aide à la décision.



# Exemples

---

- Combien de clients ont acheté tel produit pendant telle période ?  SQL
  - A-t-on vendu plus d'un tel produit cette année que l'année dernière?  OLAP
  - Quel est le profil des clients ?
  - Quels autres produits les intéresseront ?
  - Quand seront-ils intéressés ?
-  Data Mining



# Intérêt Data Mining

---

1) Exploitation des données de l'entreprise pour améliorer la rentabilité d'une activité.

2) Production de la connaissance :

- pour comprendre les phénomènes : SAVOIR
- et prendre des décisions : PREVOIR pour DECIDER

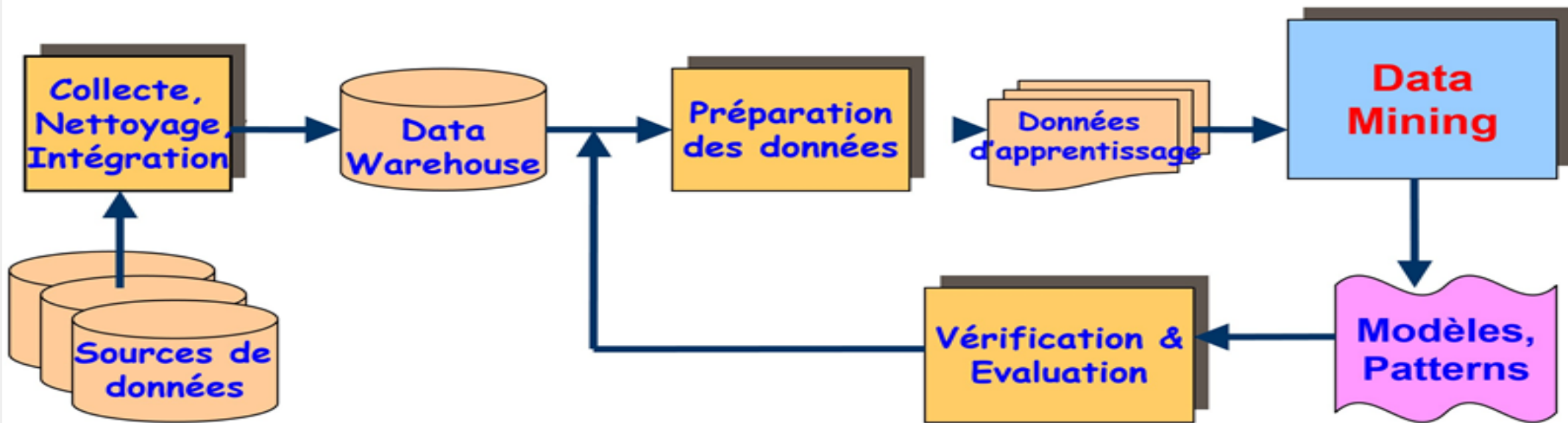
3) Permet d'augmenter le retour sur investissement des systèmes d'information.



# Processus KDD

---

- Data mining : coeur de KDD (Knowledge Data Discovery).



# Processus KDD

---

## ❖ Démarche méthodologique

### ✓ Comprendre l'application

- Connaissances a priori, objectifs, etc.

### ✓ Sélectionner un échantillon de données

- Choisir une méthode d'échantillonnage

### ✓ Nettoyage et transformation des données

- Supprimer le «bruit» : données superflues, marginales, données manquantes, etc.
- Effectuer une sélection d'attributs, réduire la dimension du problème, etc.

### ✓ Appliquer les techniques de fouille de données

- Choisir le bon algorithme

### ✓ Visualiser, évaluer et interpréter les modèles découverts

- Analyser la connaissance(intérêt)
- Vérifier sa validité (sur le reste de la base de données)
- Reiterer le processus si nécessaire

### ✓ Gérer la connaissance découverte

- La mettre à la disposition des décideurs
- L'échanger avec d'autres applications ( système expert, ..)
- Etc

# Rencontre de plusieurs disciplines

---



# Rencontre de plusieurs disciplines

---



# Rencontre de plusieurs disciplines

---

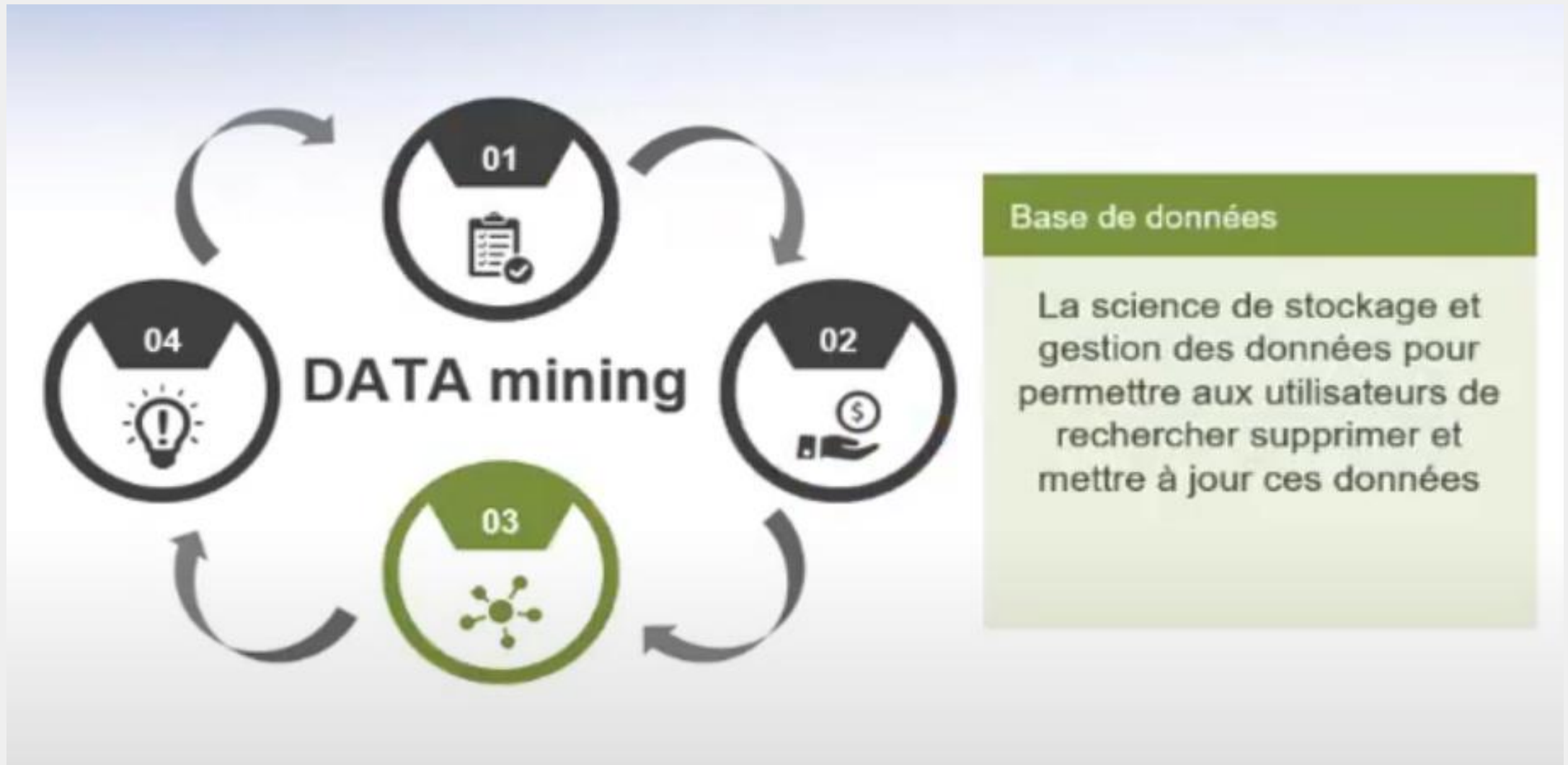


## Apprentissage automatique

Un ensemble d'algorithmes menant à l'acquisition des connaissances pour permettre à une machine d'évoluer.

# Rencontre de plusieurs disciplines

---



# Rencontre de plusieurs disciplines

---



## Intelligence artificielle

l'ensemble de théories et de techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence



# Données et Information

---



Donnée = 3.89



**Donnée + contexte = information**

**Que signifier ces données ????**



# Lien entre donnée Information et connaissance

Patient	Température	La tension	Vertige	TSS	La Maladie?
N:1	37.5	15	Non	1.8	HYPER/dia
N:2	39.5	13	Oui	1.1	grippe
N:3	36	15	Non	2.36	HYPER/dia
N: 4	40	18	Oui	0.7	Fièvre
N: 5	38.2	12	Oui	0.48	Diabétique
N: 6	37.9	14	Oui	1.83	Diabétique
N: 7	36.5	13	Oui	1.3	Grippe
N: 8	35.6	14	Non	0.98	Normal
N: 9	38.4	15	Non	0.78	HYPER
N: 10	37.6	13	Non	2.69	Diabétique



# Taches de Data Mining

---

1. Classification
2. Estimation
3. Prédiction
4. Segmentation ( clustering)
5. Association
6. Description

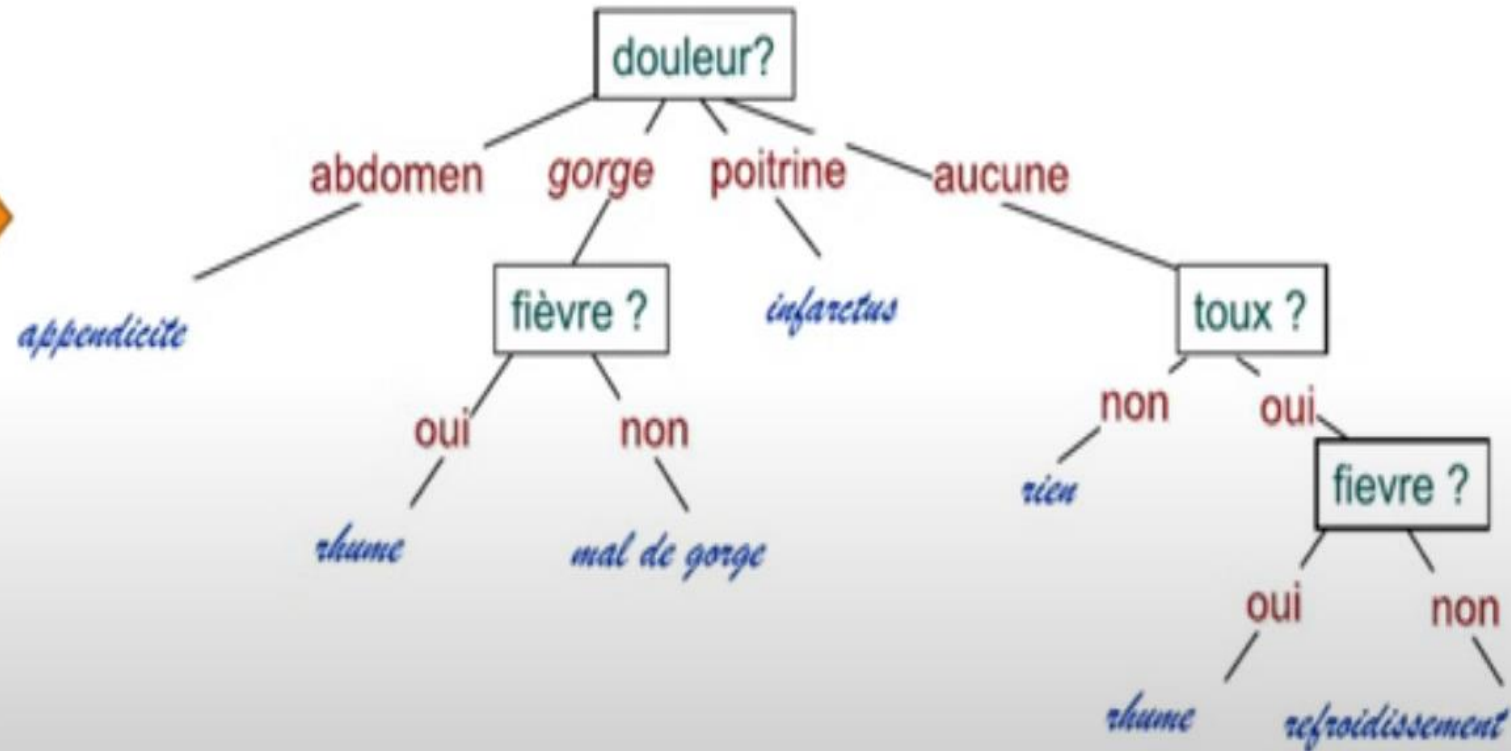


# Classification

Consiste à examiner des caractéristiques d'un élément nouvellement présenté afin de l'affecter à une classe d'un ensemble prédéfini.

Technique : Arbres de décision

Omar    Toux    Fièvre    Douleur  
non    oui    gorge



Id	categorical		categorical		continuous	class
	Refund	Marital Status	Taxable Income	Cheat		
1	Yes	Single	125K	No		
2	No	Married	100K	No		
3	No	Single	70K	No		
4	Yes	Married	120K	No		
5	No	Divorced	95K	Yes		
6	No	Married	60K	No		
7	Yes	Divorced	220K	No		
8	No	Single	85K	Yes		
9	No	Married	75K	No		
10	No	Single	90K	Yes		



Training Data

Model: Decision Tree



# Estimation

---

Contrairement à la classification, le résultat d'une estimation est une variable continue obtenue par une ou plusieurs fonctions combinant les données en entrée.

Le résultat d'une estimation permet de procéder aux classifications grâce à un barème.



**Exemple:** Estimer le revenu d'un ménage selon divers critères (type de véhicule , profession ou catégorie socioprofessionnelle, type d'habitation, etc.).

Il sera ensuite possible de définir des tranches de revenus pour classer les individus.

Le tableau ci-dessous résume les données de l'exemple.  
Dans le Panneau de configuration, ouvrez Système pour

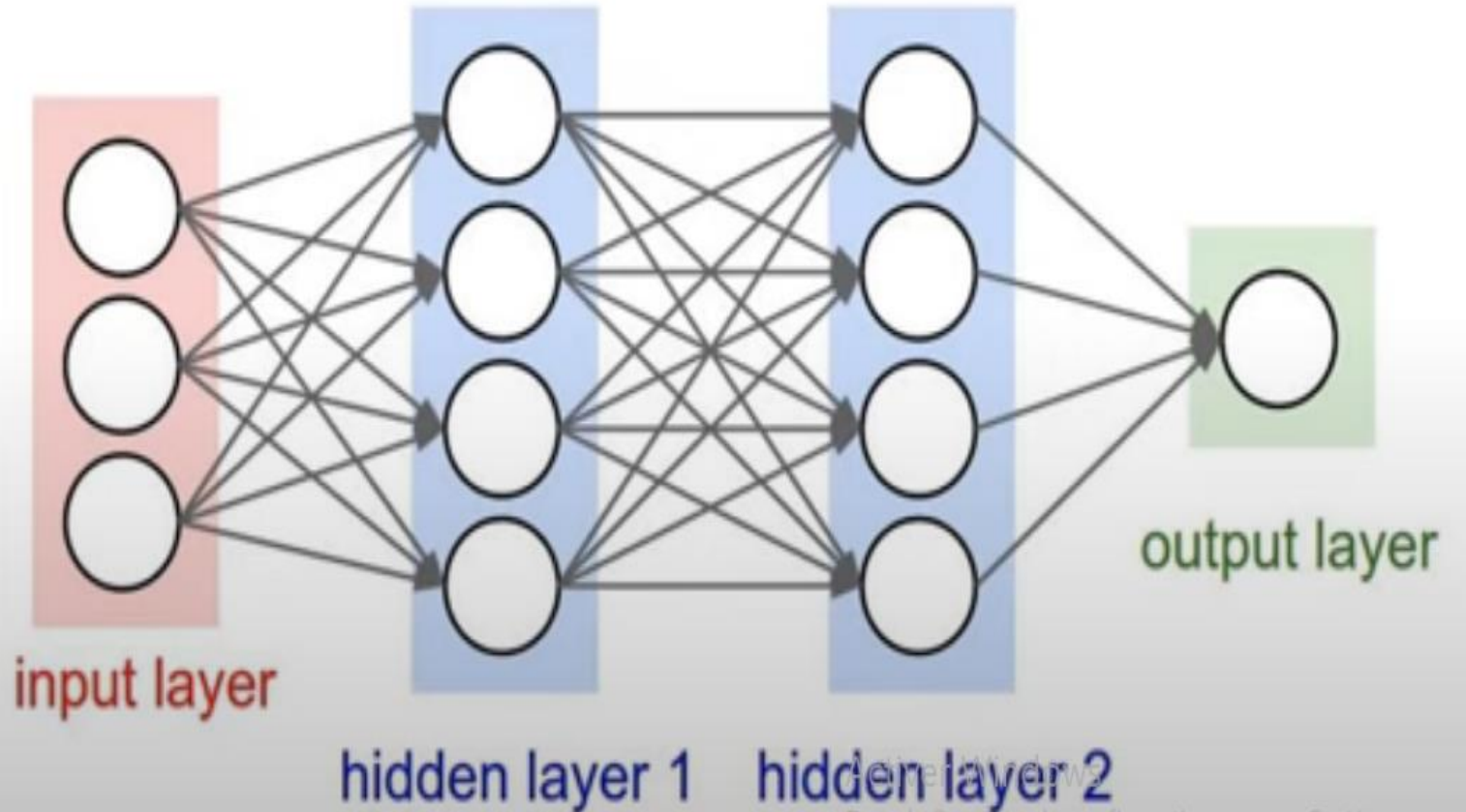
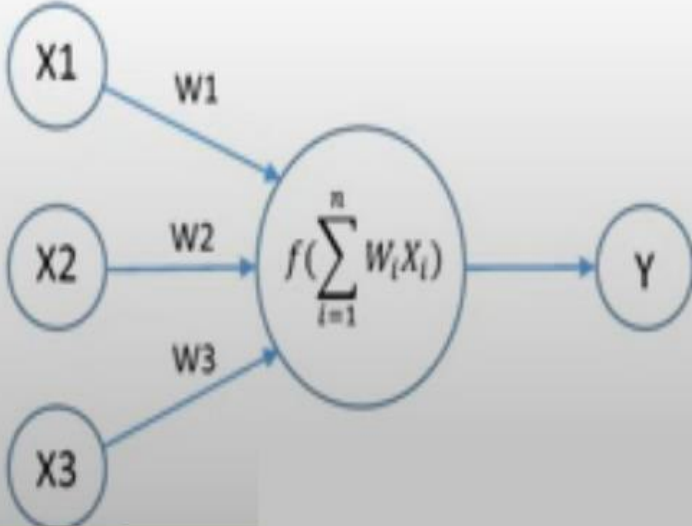
# Prédiction

Ressemble à la classification mais dans une échelle temporelle différente.

Elle s'appuie sur le passé et le présent mais son résultat se situe dans un futur généralement précisé.

## Technique :

### Les réseaux de neurones

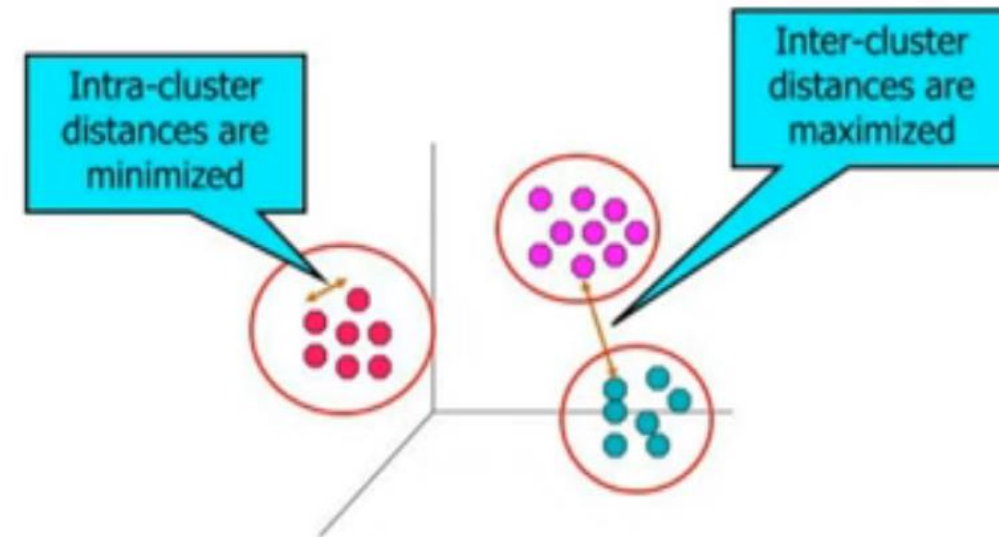
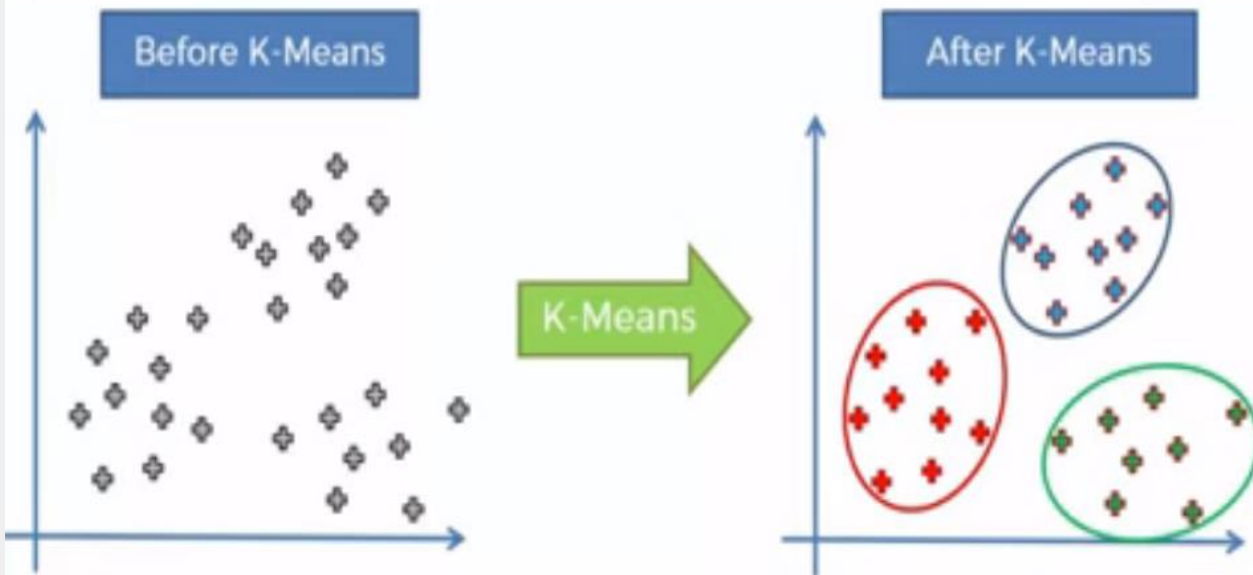


Dans le Panneau de configuration, ouvrez Système pour activer Windows.

# Clustering

Segmentation d'une population hétérogène en sous-populations homogènes.  
Contrairement à la classification, les sous-populations ne sont pas préétablies (classification non supervisée).

Technique : Clustering : Kmeans





# Association

Cherche à découvrir les règles de quantification ou de relation entre deux ou plusieurs attributs.

## Technique :

- Règles d'association : Apriori

*Exemple:* L'analyse du panier de la ménagère

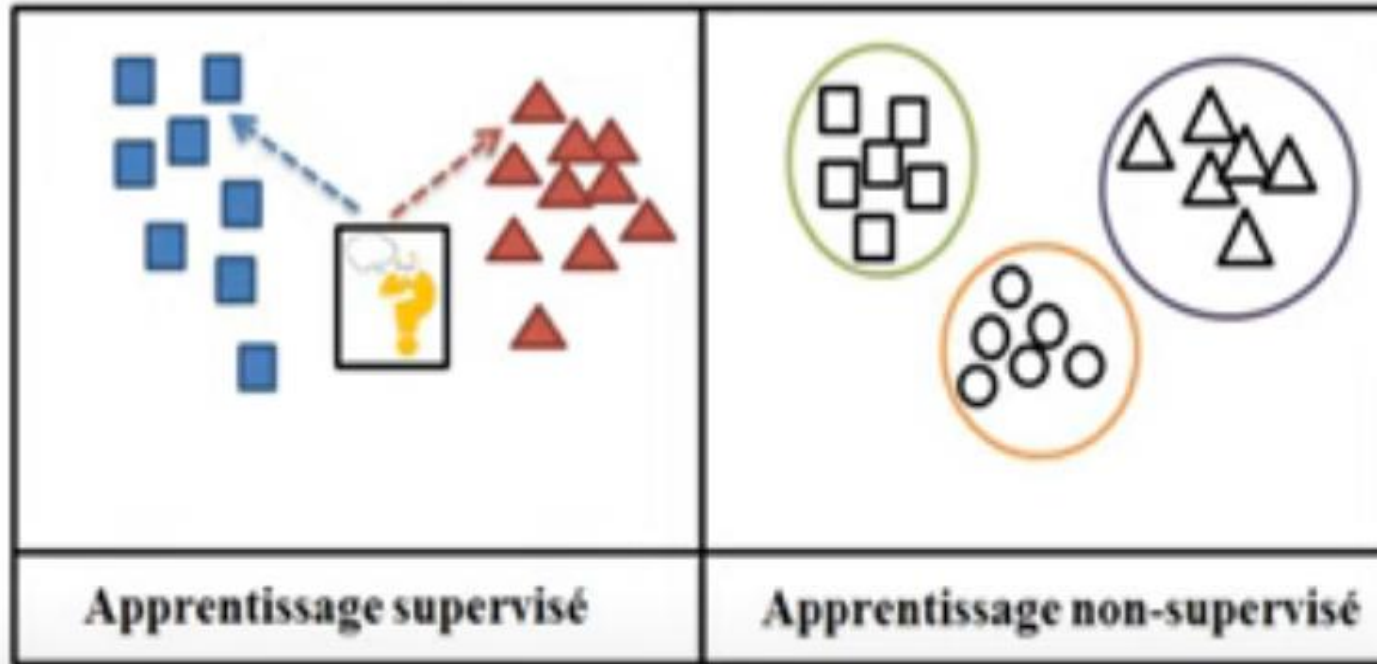
Clients	Dates	Itemsets
Client1	02/03/2008	Pain, TV
	03/04/2008	beurre
Client2	10/02/2008	Lecteur DVD
	11/02/2008	Dattes, pain
Client3	12/02/2008	Pain
	13/02/2008	Beurre



Activer Windows  
Dans le Panneau de configuration, ouvrez Sy  
activer Windows.

# Typologie des méthodes

## Selon le type d'apprentissage

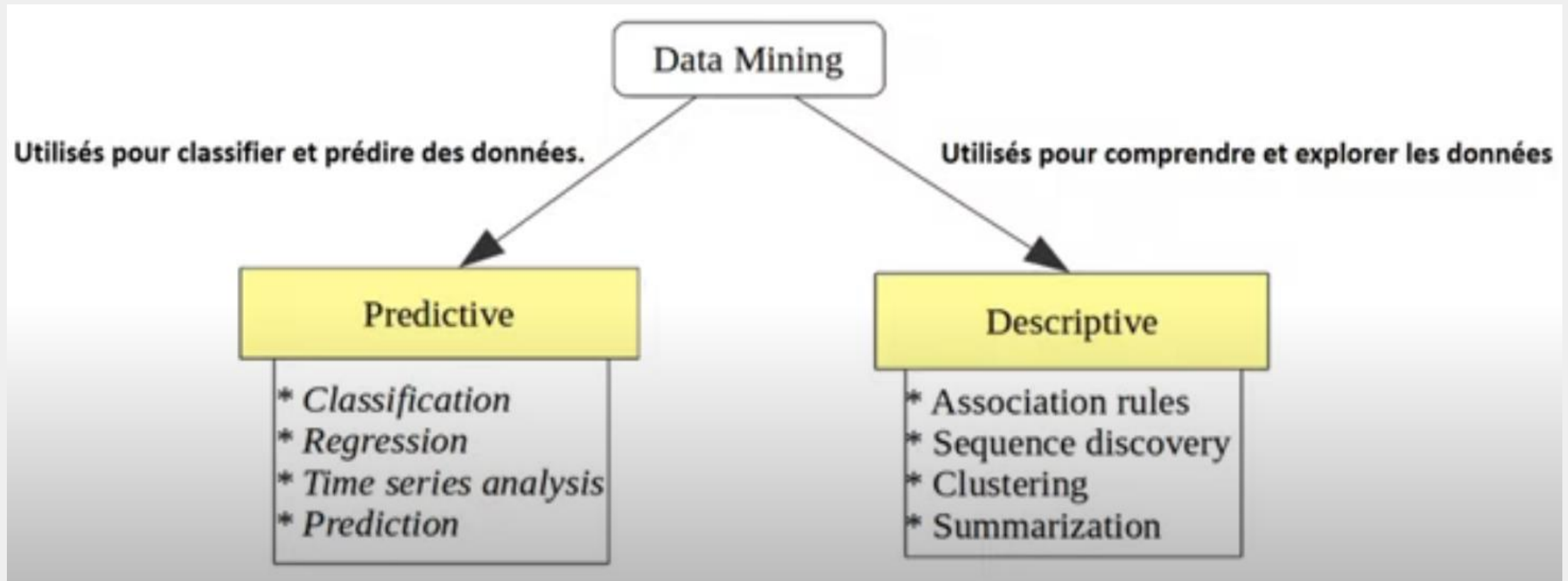


- Nombre de classes connu
- Utilisées principalement en classification et prédiction

- Nombre de classes inconnu
- Utilisées principalement en clustering et association

# Typologie des méthodes

## Selon le type de modele



# Exemples

---

**Classification :** Déterminer grade en fonction de l'âge, l'ancienneté, le salaire et les affectations.

**Estimation :** (sur Variables continues )

Estimer le salaire en fonction de l'âge, ancienneté et affectations

**Prédiction :** Prédire quelle sera la prochaine affectation d'un militaire.

**Association:** Déterminer des règles de type :

Le militaire qui est sergent entre 25 et 30 ans sera lieutenant colonel entre 45 et 50 ans (fiable à n %).

**Segmentation (ou clustering) :**

Segmenter les militaires en fonction de leurs parcours (carrière) et affectations.

**Description :** Indicateurs statistiques traditionnels :

Age moyen, pourcentage de femmes, salaire moyen

# Chapitre 2

## Données

---

# Analyse exploratoire (descriptive)

---

Une variable est une propriété ou caractéristique d'un individu

- ❖ Exemple : Couleur des yeux d'une personne, température, état civil, ...
- ❖ Une collection de variables décrivant à un individu

On dit individu ou enregistrement, point, cas, objet, entité, exemple d'observation

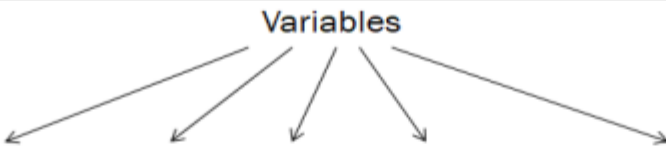


Diagram illustrating the variables used in the dataset:

Variables

- age
- Revenus
- Etudiant
- Taux\_crédit
- Achat\_PC

age	Revenus	Etudiant	Taux_crédit	Achat_PC
<=30	élevé	non	faible	non
<=30	élevé	non	excellent	non
31...40	élevé	non	faible	oui
>40	moyen	non	faible	oui
>40	faible	oui	faible	oui
>40	faible	oui	excellent	non
31...40	faible	oui	excellent	oui
<=30	moyen	non	faible	non
<=30	faible	oui	faible	oui
>40	moyen	oui	faible	oui
<=30	moyen	oui	excellent	oui
31...40	moyen	non	excellent	oui
31...40	élevé	oui	faible	oui
>40	moyen	non	excellent	non

# Analyse exploratoire (descriptive)

---

## Types de variables

Qualitative : les variables représentent des catégories différentes au lieu des numéros. Les opérations mathématiques comme la somme et la soustraction n'ont pas de sens.

❖ Exemples : couleur des yeux, niveau académique, adresse IP

Quantitative : les variables sont les numéros

❖ Exemple : poids, la température, le nombre d'enfants

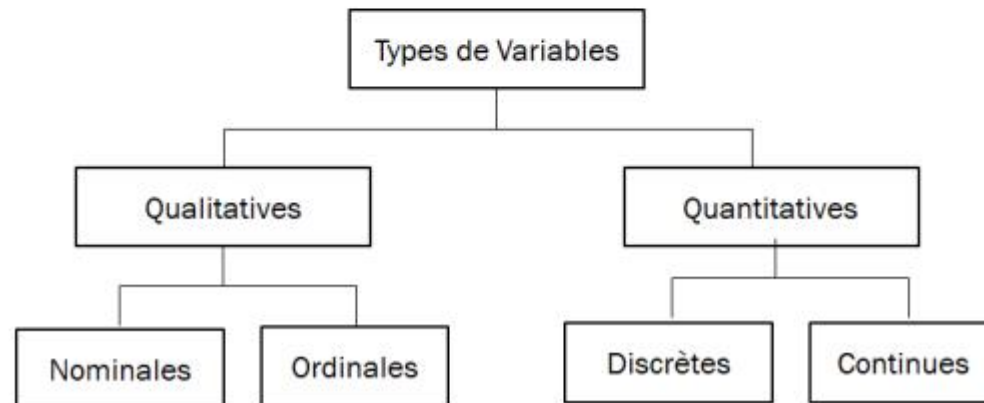


# Analyse exploratoire (descriptive)

## Types de variables

### Variables qualitatives

ind	SEXO	EDAD	INGRESO
1	F	5	Medio
2	F	3	Alto
3	M	4	Bajo
4	F	1	Bajo
5	M	2	Medio
6	M	5	Alto
7	F	2	Medio
8	M	3	Bajo
9	M	1	Alto
10	F	4	Medio



# Analyse exploratoire (descriptive)

---

## Types de variables

### Transformation d'une variable quantitative en variable qualitative

Pour les variables discrètes : considérer que les valeurs prises par la variable sont les modalités de la variable qualitative (ordonnée)

✓ Pour les variables continues :

- ❖ on divise l'intervalle  $[a ; b[$  où varie la variable en un certain nombre d'intervalles  $[a ; x_1[$ ,  $[x_1 ; x_2[$ ,  $[x_i ; x_{i+1}[$  ... ,  $[x_{p-1} ; b[$  et
- ❖ on dénombre pour chaque intervalle le nombre d'individus dont la mesure appartient à l'intervalle. En règle générale, on choisit des classes de même amplitude.
- ❖ Pour que la distribution en fréquence soit intéressante, il faut que chaque classe comprenne un nombre « suffisant » d'individus ( $n_i$ )
- ❖ Si la longueur des intervalles est trop grande, on perd trop d'information

# Analyse exploratoire (descriptive)

---

## Types de variables

### Transformation d'une variable quantitative en variable qualitative

Il existe des formules empiriques pour établir le nombre de classes pour un échantillon de taille  $n$

- ❖ Règle de Sturge
- ✓ Nombre de classes  $= 1 + 3.3 \log n$
- ❖ Règle de Yule
- ✓ Nombre de classes  $= 2.5 \sqrt{n}$
- ❖ L'intervalle entre chaque classe est calculé par
- ✓  $(b-a)/\text{nombre de classes}$
- ❖ On calcule ensuite à partir de  $a$  les classes successives par addition.

NB: il n'est pas obligatoire d'avoir des classes de même amplitude. Mais pas de chevauchement d'intervalle

# Les données

- ❖ Le point de départ est d'une table de données:

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \vdots & & \vdots & & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \vdots & & \vdots & & \vdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nm} \end{pmatrix} \leftrightarrow \text{individuo } i$$

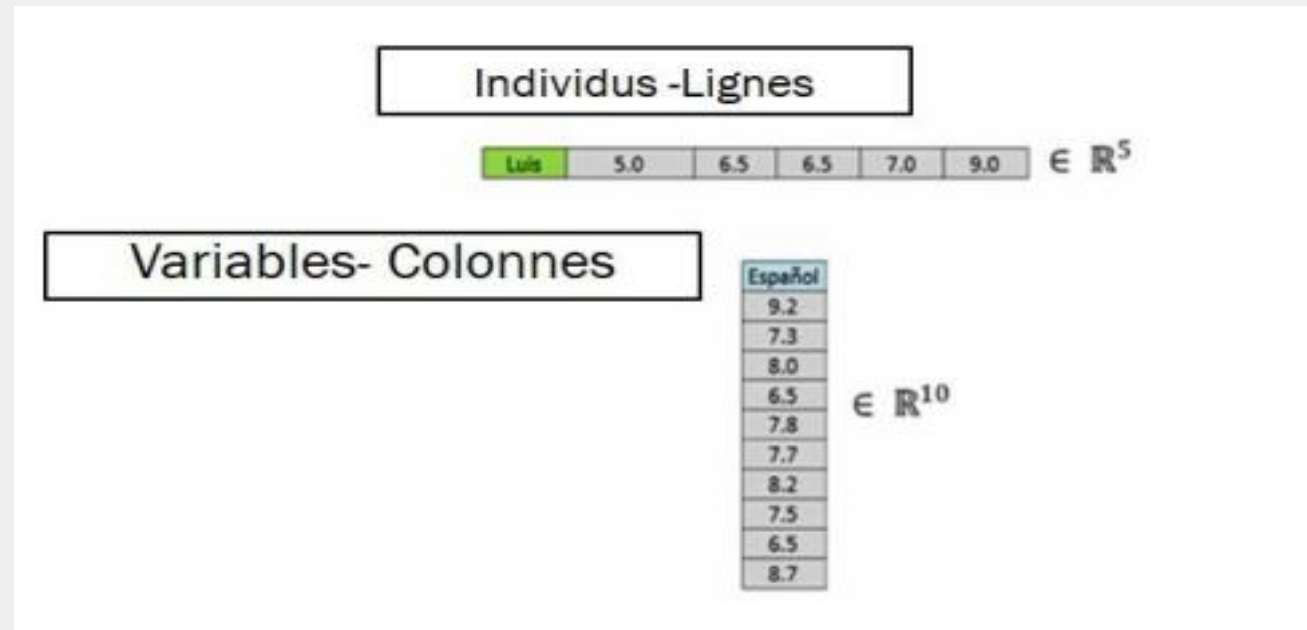
Variable  $j$

Exemple

	Matemáticas	Ciencias	Español	Historia	EdFísica
Lucía	7.0	6.5	9.2	8.6	8.0
Pedro	7.5	9.4	7.3	7.0	7.0
Inés	7.6	9.2	8.0	8.0	7.5
Luis	5.0	6.5	6.5	7.0	9.0
Andrés	6.0	6.0	7.8	8.9	7.3
Ana	7.8	9.6	7.7	8.0	6.5
Carlos	6.3	6.4	8.2	9.0	7.2
José	7.9	9.7	7.5	8.0	6.0
Sonia	6.0	6.0	6.5	5.5	8.7
María	6.8	7.2	8.7	9.0	7.0

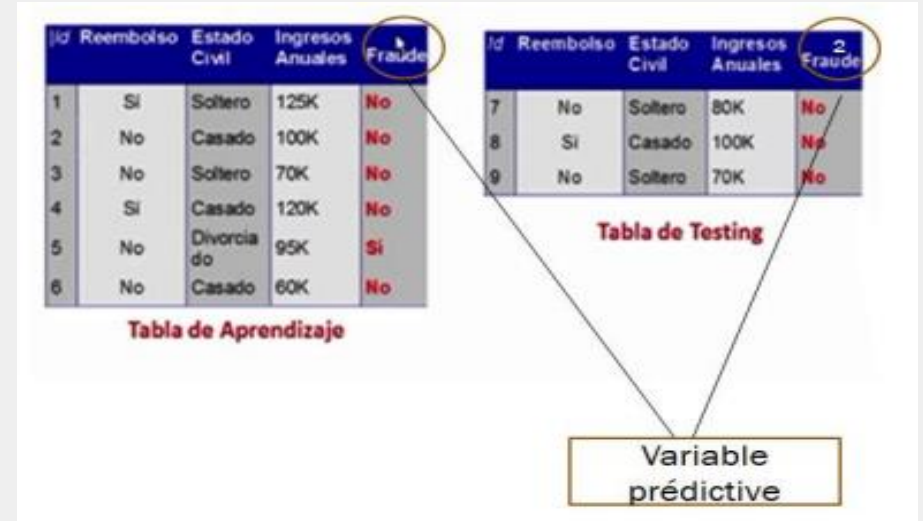
# Nuage de points

## Nuage de points



# Nuage de points

Données pour les méthodes prédictives



Exemple

	Matemáticas	Ciencias	Español	Historia	EdFísica	Tipo
Lucía	7.0	6.5	9.2	8.6	8.0	Regular
Pedro	7.5	9.4	7.3	7.0	7.0	Bueno
Inés	7.6	9.2	8.0	8.0	7.5	Bueno
Luis	5.0	6.5	6.5	7.0	9.0	Malo
Andrés	6.0	6.0	7.8	8.9	7.3	Regular
Ana	7.8	9.6	7.7	8.0	6.5	Bueno
Carlos	6.3	6.4	8.2	9.0	7.2	Regular
José	7.9	9.7	7.5	8.0	6.0	Bueno
Sonia	6.0	6.0	6.5	5.5	8.7	Regular
María	6.8	7.2	8.7	9.0	7.0	Malo

Variable prédictive

# Description d'une variable quantitative

---

Une variable quantitative est décrite par les valeurs qui prennent l'ensemble de  $n$  individus pour lesquels a été définis

Exemple

individuo	tamaño
1	1.70
2	1.65
3	1.70
4	1.80

Pour résumer l'information d'une variable quantitative les indices les plus communes sont :

❖ La moyenne. Définit par :

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n x_i.$$

❖ La Variance : définit par

$$\text{var}(X) = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{X})^2$$

❖ L'écart type :

$$\sigma_x = \sqrt{\text{var}(X)}.$$

❖ Le Coefficient de détermination :

$$R^2 = \frac{\text{var}(aX + b)}{\text{var}(Y)}$$

❖ Le Coefficient de corrélation :

$$R = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}$$

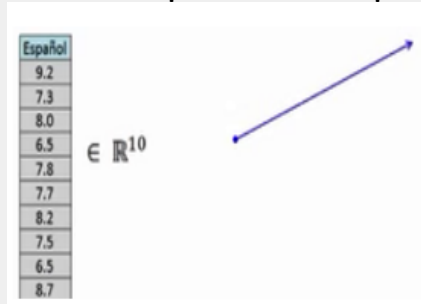


# Matrice de Corrélation

- Grande corrélation positive implique que si une variable augmente l'autre aussi augmente.
- Grande corrélation négative implique que si une variable augmente l'autre diminue et vice versa.
- Corrélation proche de 0 implique l'absence de relation entre les variables

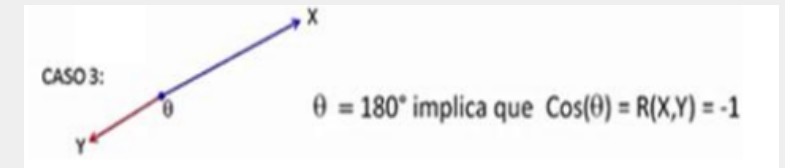
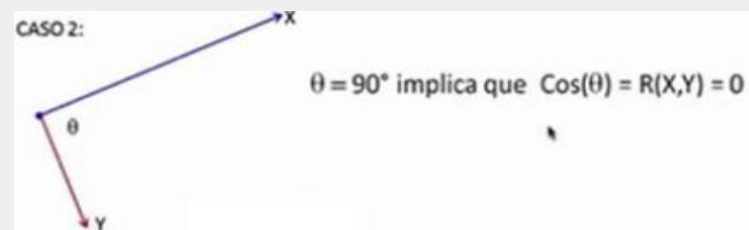
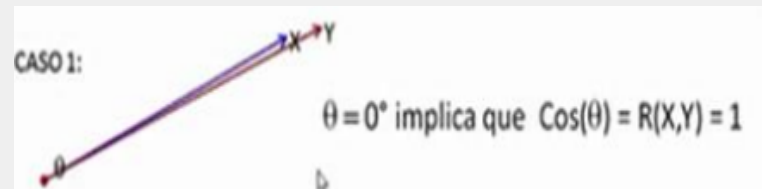
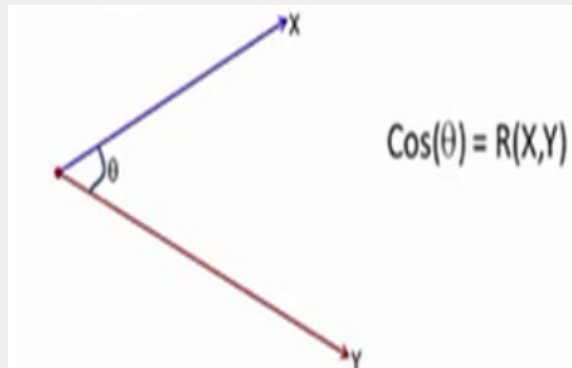
Interprétation géométrique du coefficient de corrélation

- ✓ Une variable  $x$  qui prend  $n$  valeurs peut être représentée comme un vecteur de  $\mathbb{R}^n$
- ✓ Variables –colonnes



❖ Théorème :

Dans l'espace vectoriel des variables  $\mathbb{R}^n$  le cosinus de l'angle entre 2 variables réduites et centrées est égale au coefficient de corrélation entre ses deux variables :

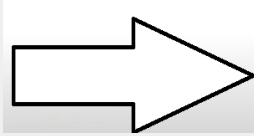


# Matrice de Corrélation

Matrice de Corrélation

$$\Sigma = \text{Corr}(X) = \frac{1}{n} X^t X$$

Exemple

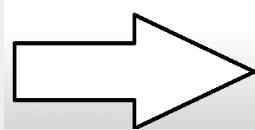
$$X = \begin{pmatrix} 12,04 & 23,7 & 5,9 \\ 17,18 & 15,5 & -1,8 \\ 11,83 & 13,1 & 2,8 \\ 6,23 & 13,5 & -2,4 \\ 16,99 & 21,1 & 7,2 \\ 3,87 & 20,30 & -0,90 \end{pmatrix}$$


Etape 01 : Matrice des données centrées réduites

$$X_{cr} = \begin{pmatrix} 0,14 & 1,44 & 1,09 \\ 1,17 & -0,59 & -0,96 \\ 0,1 & -1,18 & 0,27 \\ -1,03 & -1,08 & -1,12 \\ 1,13 & 0,8 & 1,44 \\ -1,5 & 0,6 & -0,72 \end{pmatrix}$$

Etape 2 :

Matrice Transposée

$$X^t = \begin{pmatrix} 0,14 & 1,17 & 0,1 & -1,03 & 1,13 & -1,5 \\ 1,44 & -0,59 & -1,18 & -1,08 & 0,8 & 0,6 \\ 1,09 & -0,96 & 0,27 & -1,12 & 1,44 & -0,72 \end{pmatrix}$$


Etape 03 :  
 $\Sigma = \text{Corr}(X) = \frac{1}{n} X^t X$

$$\Sigma = \text{Corr}(X) = \frac{1}{6} \begin{pmatrix} 0,14 & 1,17 & 0,1 & -1,03 & 1,13 & -1,5 \\ 1,44 & -0,59 & -1,18 & -1,08 & 0,8 & 0,6 \\ 1,09 & -0,96 & 0,27 & -1,12 & 1,44 & -0,72 \end{pmatrix} \begin{pmatrix} 0,14 & 1,44 & 1,09 \\ 1,17 & -0,59 & -0,96 \\ 0,1 & -1,18 & 0,27 \\ -1,03 & -1,08 & -1,12 \\ 1,13 & 0,8 & 1,44 \\ -1,5 & 0,6 & -0,72 \end{pmatrix}$$

$$\Sigma = \text{Corr}(X) = \begin{pmatrix} 1 & 0,9 & 0,42 \\ 0,9 & 1 & 0,62 \\ 0,42 & 0,62 & 1 \end{pmatrix}$$

# Chapitre 3

## Traitement des données

---

# Nettoyage des données

---

## Objectif :

Supprimer les données bruitées ou non pertinentes.

## Questions :

- Que faire si certaines données sont manquantes ?
  - Certains clients n'ont pas donné leur adresse.
- Toutes les données sont-elles fiables (problèmes d'inconsistance) ?
  - Un même article appartient à différentes catégories (dans des magasins différents).
  - Le prix d'un même article est très supérieur à la normale dans un magasin donné.
- Que faire si certaines données sont numériques dans le cas où la technique d'extraction ne peut manipuler que des données symboliques ?

# Nettoyage des données

---

## Données manquantes

### Solutions :

- Ne pas tenir compte des tuples contenant des données manquantes (valeurs nulles).
- Remplir manuellement les champs non remplis.
- Utiliser les valeurs connues :
  - Remplacer un salaire manquant par le salaire médian des clients.
  - Prédire les valeurs manquantes, en le déduisant d'autres paramètres (salaire à partir de l'âge et de la profession).

# Nettoyage des données

---

## Données bruitées

Plusieurs solutions :

lissage, segmentation, régression linéaire.

Techniques de lissage (data smoothing) :

- ❖ Trier les différentes valeurs de l'attribut considéré.  
{4, 8, 15, 21, 21, 24, 25, 28, 34}
- ❖ Partitionner l'ensemble résultat.  
{{4, 8, 15}, {21, 21, 24}, {25, 28, 34}}
- ❖ Remplacer les valeurs initiales par de nouvelles valeurs en fonction du partitionnement réalisé :
  - par la valeur moyenne des regroupements réalisés  
{9, 22, 29}
  - par les min et max des regroupements réalisés.  
{{4, 4, 15}, {21, 21, 24}, {25, 25, 34}}

Implique une perte de précision ou d'information.

# Nettoyage des données

---

## Données bruitées

Plusieurs solutions :

lissage, segmentation, régression linéaire.

Techniques de segmentation (clustering) :

- ❖ Les valeurs similaires sont placées dans une même classe.
- ❖ On ne tient pas compte des valeurs isolées (dans une classe comportant trop peu d'éléments).

Techniques de régression linéaire :

- ❖ Hypothèse : un attribut Y dépend linéairement d'un attribut X.
  - Années d'expérience X et salaire Y
- ❖ Trouver les coefficients a et b tels que  $Y = aX + b$ .
- ❖ Remplacer les valeurs de Y par celles prédites.



# Nettoyage des données

---

## Données bruitées : régression linéaire

Données de départ :

Un ensemble de couples  $(X_i, Y_i)$ .

Détermination des coefficients :

- Soient  $\bar{X}$  et  $\bar{Y}$  les valeurs moyennes des attributs X et Y .

- $a = \frac{COV(X,Y)}{V(X)}$

- $b = \bar{Y} - a\bar{X}$  .

# Nettoyage des données

---

## Données inconsistantes

### Données inconsistantes dans une base de données :

- Contraintes d'intégrités ou dépendances fonctionnelles non respectées.
- Exemples :
  - La contrainte  $I\ ID \rightarrow I\ CATEGORY$  n'est pas respectée au moment de l'intégration des données.
  - Unicité de clés non respectée.

# Intégration des données

---

## Objectif :

Regrouper les données provenant de différentes sources.

→ Problématique typique lors de la construction d'entrepôts de données .

## Exemple :

Un attribut nommé C\_ID dans la BD de Rabat peut très bien se nommer CUST\_ID dans la BD de Fes.

→ Utilisation de Talend transformation pour la mise en correspondance.

# Transformation des données

---

- **Lissage de données :**

utilisation de techniques de régression.

- Normalisation des données : normaliser certains attributs numériques afin qu'ils varient entre 0 et 1.
  - Pour ne pas privilégier les attributs ayant les plus grands domaines de variation (salaire/âge).
- Agrégation des données : opérations OLAP (On-Line Analytical Processing) permettant une analyse multidimensionnelle sur les BD volumineuses afin de mettre en évidence une analyse particulière des données.
  - Calculer les niveaux de ventes réalisées de tel produit par mois plutôt que par jour.
- Généralisation des données : remplacer les données finies par des données de plus haut niveau.
  - Remplacer les adresses précises des clients par leur code postal.
  - Remplacer l'âge des clients par << jeune >>, << adulte >>, << senior >>.

# Transformation des données

---

## Discrétisation des connaissances

### Répartition des valeurs des attributs :

A chaque étape, on cherche à découper l'intervalle de variation des données en  $K$  intervalles comportant le même nombre de valeurs.

On divise  $C\_AGE = [0, 100]$  en  $A1 = [0, 20]$  et  $A2 = [20, 100]$  si 50 % des clients ont moins de 20 ans.

### Entropie et classification a priori des données :

On cherche à caractériser les individus achetant les différents types de lait (entier, demi-écrémé, écrème).

### Facilité à appréhender le découpage obtenu :

On veut obtenir des intervalles du type  $[-12.5, 0]$  plutôt que  $[-12.536, 0.0005]$ .

# Transformation des données

---

## Discrétisation basée sur l'entropie

Entropie d'un ensemble de données S :

Définition :

- S est découpé en k classes  $C_1, \dots, C_k$ .
- $\text{Ent}(S) = - \sum_i p_i \cdot \log(p_i)$  avec  $p_i = \frac{|C_i|}{|S|}$ .

Propriétés :

- $\text{Ent}(S)$  est maximale (égale à 0) si les données sont réparties dans une seule et même classe.
- $\text{Ent}(S)$  est minimale si les données sont uniformément réparties dans toutes les classes.



# Transformation des données

---

## Discrétisation basée sur l'entropie

Méthode :

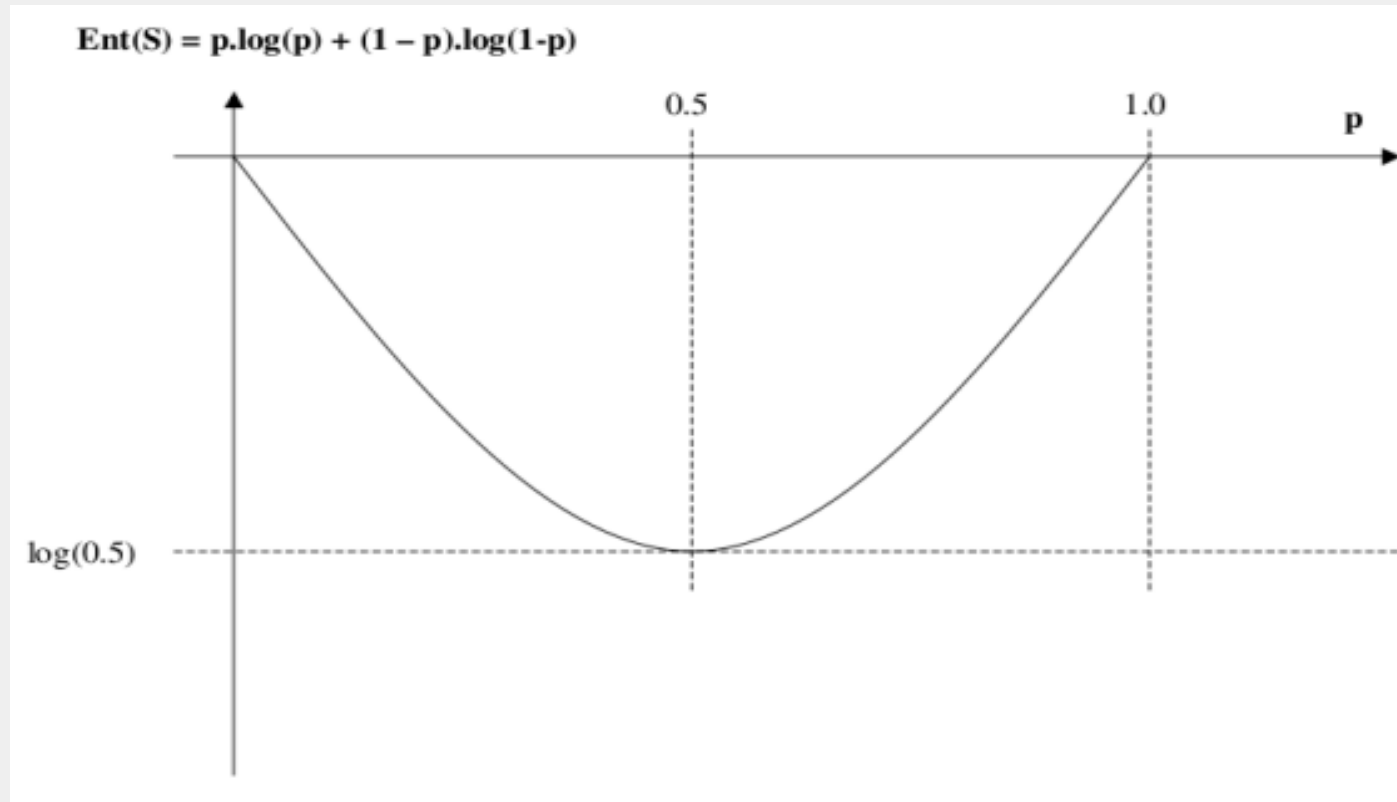
- Découper  $S = [a, b]$  en  $S1 = [a, c]$  et  $S2 = [c, b]$ .
- Maximiser le gain d'information

$$I(S, c) = \frac{|S1|}{|S|} \text{Ent}(S1) + \frac{|S2|}{|S|} \text{Ent}(S2) - \text{Ent}(S).$$

- Arrêt du découpage si le gain devient insuffisant, quel que soit  $c$ .

# Transformation des données

## Variation de l'entropie



# Sélection des données

---

## Objectif :

Garder uniquement les données pertinentes pour l'étude à réaliser.

## Exemple :

- Doit-on s'intéresser à toutes les catégories de produits de vente ?
- Doit-on s'intéresser aux ventes réalisées il y a plus d'un an ?

# Réduction des données

---

## Réduction en ligne par échantillonnage :

- Pour des raisons de performance.
- Du fait de la complexité importante des algorithmes d'extraction.
- Plusieurs méthodes : échantillonnage aléatoire (avec ou sans remise), échantillonnage par clustering/segmentation.

## Réduction en colonne par suppression des attributs redondants :

- Cas triviaux (âge et date de naissance).
- Via une analyse des corrélation entre attributs :

$$\text{corr}_{A,B} = \frac{P(A \wedge B)}{P(A) \cdot P(B)} = \frac{P(B/A)}{P(B)}$$

- Indépendance :  $\text{corr}_{A,B} = 1$  si  $P(B/A) = P(B)$ .
- Corrélation positive :  $\text{corr}_{A,B} > 1$  si  $P(B/A) > P(B)$ .

# Réduction des données

---

## Matrice de contingence

Exemple de matrice de contingence :

	Avec pain	Sans pain	Total
Avec beurre	4.000	3.500	7.500
Sans beurre	2.000	500	2.500
Total	6.000	4.000	10.000

Analyse de corrélation :

- $P(\text{Beurre}) = \frac{7.500}{10.000} = 0.75$  et  $P(\text{Pain}) = 0.6$ .
- $P(\text{Beurre} \wedge \text{Pain}) = \frac{4.000}{10.000} = 0.4$ .
- $\text{corr}_{\text{Pain,Beurre}} = \frac{0.4}{0.75 \times 0.6} = 0.89 < 1$   
→ Indique une corrélation négative.

# Réduction des données

---

## Qualité de la corrélation

Coefficient de corrélation :

$$r_{A,B} = \frac{\sum (A_i - \bar{A})(B_i - \bar{B})}{\sigma_A \cdot \sigma_B}$$

$$\text{avec } \sigma_X = \sqrt{\sum (X_i - \bar{X})^2}.$$

### Signification :

Plus  $r_{A,B}$  s'éloigne de zéro, meilleure est la corrélation :

- $r_{A,B} = +1$  : corrélation positive parfaite.
- $r_{A,B} = -1$  : corrélation négative parfaite.
- $r_{A,B} = 0$  : absence totale de corrélation.



# Equilibrage des données

---

## Introduction

Toutefois, en Data Science, on désignera par cette appellation les jeux de données où l'une des classes (ou plusieurs) est extrêmement minoritaire par rapport aux autres. Par exemple, un dataset où 98% des données appartiennent à la classe "A" contre seulement 2% à la classe "B" est un dataset fortement déséquilibré.

La notion de "déséquilibre de classe" est très importante en machine learning, et en particulier pour les modèles de type "supervisés" qui impliquent deux classes (ou plus)

### **Pourquoi est-ce important ?**

En règle générale, la plupart des modèles fonctionnent correctement si les proportions des classes dans un dataset sont relativement similaires : les légers déséquilibres de classes sont bien gérés. Cependant, passé un certain point, les modèles de machine learning auront du mal à identifier correctement la (ou les) classe(s) minoritaire(s).

Exemple:

détection de fraude ou de défauts, diagnostic médical, etc.

Si l'on reprend l'exemple typique de la classification d'emails frauduleux (de type spam, arnaque, hameçonnage, malwares, etc.), alors seulement une très faible proportion d'entre eux s'avèrent être frauduleux. Ce type d'emails est donc rare dans les datasets, et les modèles ont du mal à les classer : ils apprennent un biais vis-à-vis de la classe majoritaire (email "non frauduleux"), et ont alors tendance à toujours prédire cette dernière.

# Equilibrage des données

---

## ré-équilibrer

**ré-équilibrer le dataset:** En fonction de la quantité de données disponible, on choisira alors l'une ou l'autre des méthodes suivantes :

- **L'undersampling** lorsque l'on dispose d'un très grand nombre d'observations (à minima  $> 10K$ ). Il s'agit ici simplement de retirer aléatoirement des instances de la classe majoritaire afin de ré-équilibrer les proportions. On perd toutefois de l'information, et il y a donc un risque d'underfitting.
- **L'oversampling** lorsque l'on dispose d'un nombre limité d'observations ( $< 10K$ ), ou bien si le temps de calcul n'est pas un problème. Il s'agit ici de dupliquer aléatoirement certaines instances des classes minoritaires, rendant ainsi leur signal plus puissant. Il y a toutefois ici un risque d'overfitting.
- La méthode des **class weights** permet de prendre en compte le caractère biaisé de la distribution du dataset et de créer un *modèle pénalisé*. Il s'agit ici de simplement attribuer des poids différents aux différentes classes de notre dataset, en donnant un poids plus important aux classes minoritaires, afin d'influencer le modèle lors de son entraînement. Nous pénalisons ainsi plus fortement une erreur de classification d'une classe minoritaire par rapport à une erreur de classification d'une classe majoritaire.

# Equilibrage des données

---

## Exemple

```
# define oversampling strategy  
oversample = RandomOverSampler(sampling_strategy='minority')
```

Cela signifie que si la classe majoritaire avait 1 000 exemples et la classe minoritaire en avait 100, cette stratégie suréchantillonnerait la classe minoritaire afin qu'elle ait 1 000 exemples.

```
# define oversampling strategy  
oversample = RandomOverSampler(sampling_strategy=0.5)
```

Cela garantirait que la classe minoritaire était suréchantillonnée pour avoir la moitié du nombre d'exemples de la classe majoritaire, pour les problèmes de classification binaire. Cela signifie que si la classe majoritaire avait 1 000 exemples et la classe minoritaire en avait 100, l'ensemble de données transformé aurait 500 exemples de la classe minoritaire.

```
# fit and apply the transform  
X_over, y_over = oversample.fit_resample(X, y)
```

# Equilibrage des données

---

## Exemple

ajuster et appliquer en une seule étape en appelant la fonction `fit_sample()` :

# fit and apply the transform

```
X_over, y_over = oversample.fit_resample(X, y)
```

classification binaire synthétique avec un déséquilibre de classe de 1:100.

```
X, y = make_classification(n_samples=10000, weights=[0.99], flip_y=0)
```

exécution d'un sur échantillonnage aléatoire pour équilibrer la distribution de classe est répertorié ci-dessous.

```
from collections import Counter
```

```
from sklearn.datasets import make_classification
```

```
from imblearn.over_sampling import RandomOverSampler
```

**# définir dataset**

```
X, y = make_classification(n_samples=10000, weights=[0.99], flip_y=0)
```

*# résumer la distribution des classes*

```
print(Counter(y))
```

**# définir oversampling strategy**

```
oversample = RandomOverSampler(sampling_strategy='minority')
```

*# ajuster et appliquer la transformation*

```
X_over, y_over = oversample.fit_resample(X, y)
```

**# résumer la distribution des classes**

```
print(Counter(y_over))
```

L'exécution de l'exemple crée d'abord l'ensemble de données, puis résume la distribution de classe. On peut voir qu'il y a près de 10K exemples dans la classe majoritaire et 100 exemples dans la classe minoritaire

**Counter({0: 9900, 1: 100})**

Ensuite, la transformée de sur échantillonnage aléatoire est définie pour équilibrer la classe minoritaire, puis ajustée et appliquée à l'ensemble de données. La distribution de classe pour l'ensemble de données transformé est rapportée, montrant que maintenant la classe minoritaire a le même nombre d'exemples que la classe majoritaire.

**Counter({0: 9900, 1: 9900})**

# Equilibrage des données

---

## Exemple

```
# example of evaluating a decision tree with random oversampling
from numpy import mean
from sklearn.datasets import make_classification
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.tree import DecisionTreeClassifier
from imblearn.pipeline import Pipeline
from imblearn.over_sampling import RandomOverSampler

# define dataset
X, y = make_classification(n_samples=10000, weights=[0.99], flip_y=0)
# define pipeline
steps = [('over', RandomOverSampler()), ('model', DecisionTreeClassifier())]
pipeline = Pipeline(steps=steps)
# evaluate pipeline
cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
scores = cross_val_score(pipeline, X, y, scoring='f1_micro', cv=cv, n_jobs=-1)
score = mean(scores)
print('F1 Score: %.3f' % score)
```

F1 Score: 0.990

L'exécution de l'exemple évalue le modèle d'arbre de décision sur l'ensemble de données déséquilibré avec sur échantillonnage.