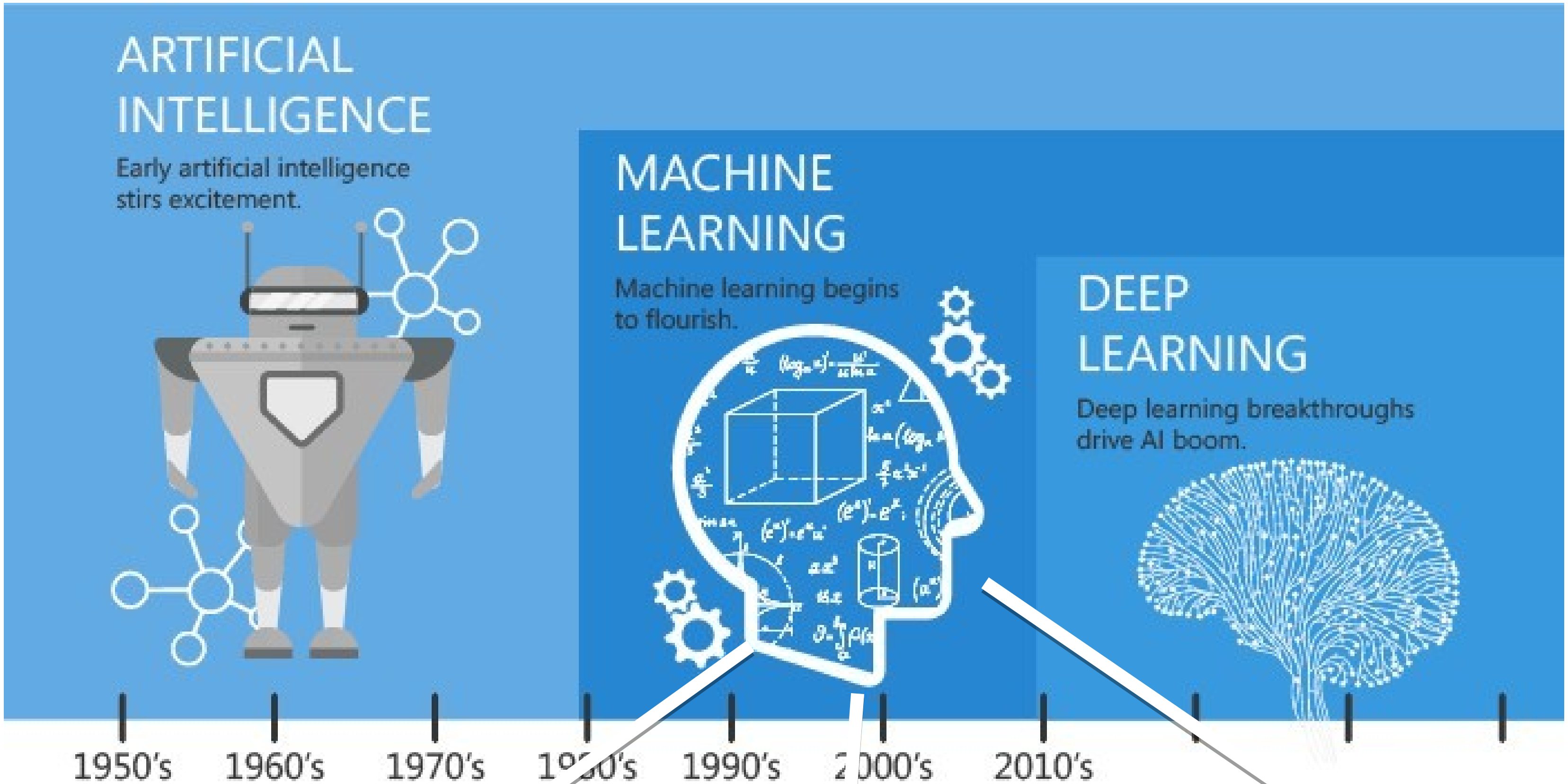
Intelligence Artificielle
Apprentissage

Apprentissage Automatique (Machine Learning)

Perspective historique



Statistiques Analyse de données,…

Optimisation

Automatique

Apprentissage

Définition 1 (Larousse)
- apprendre = acquérir de nouvelles connaissances: savoir, connaître,
- apprendre = contracter de nouvelles habitudes: savoir-faire

Définition 2 (Intelligence artificielle - Simon)
- **L'apprentissage induit des changements dans le système qui sont adaptatifs dans le sens qu'ils permettent au système de faire la même tâche une nouvelle fois plus efficacement**

AGENTS

pourquoi apprendre ?

(Complexité, système ouvert, comportement inconnu, environnement inconnu)

apprendre quoi ?

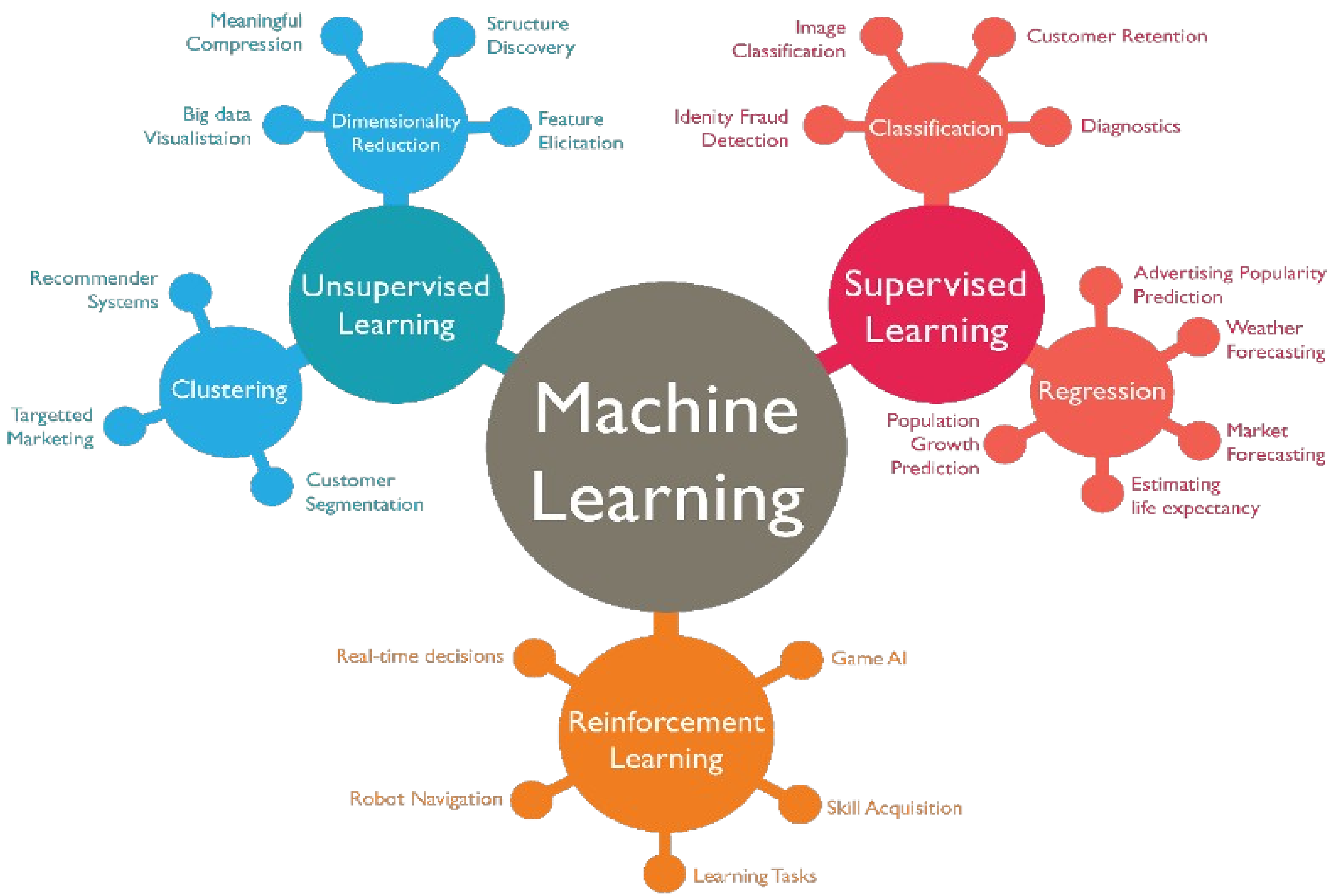
(Compétence, organisation, coordination, communication)

comment apprendre ?

(isolé ou interactif, intégrer l'expérience des autres ...)

Techniques d’apprentissages

Caractérisation des techniques de Machine Learning



Trois (Quatre) types d'apprentissage

Types of Learning

	With Teacher	Without Teacher
Active	Reinforcement Learning / Active Learning	Intrinsic Motivation / Exploration
Passive	Supervised Learning	Unsupervised Learning

Apprentissage supervisé

- Le retour désiré est connu.

Apprentissage non supervisé

- Aucun indice. L'agent apprend à partir des relations entre les perceptions. Il apprend à prédire les perceptions à partir de celles du passé.

Apprentissage semi-supervisé

- Une partie de données avec le retour désiré, l'autre partie est sans retour. On utilise le supervisé qui va guider le non-supervisé

Apprentissage par renforcement

- Le résultat désiré est inconnu. L'évaluation de l'action est faite par récompense ou punition.

Types d'apprentissages

1. Apprentissage *supervisé*

À partir de l'*échantillon d'apprentissage* $S = \{(x_i, u_i)\}_{1,m}$

on cherche une loi de dépendance sous-jacente

- Par exemple une fonction h aussi proche possible de f (fonction cible)

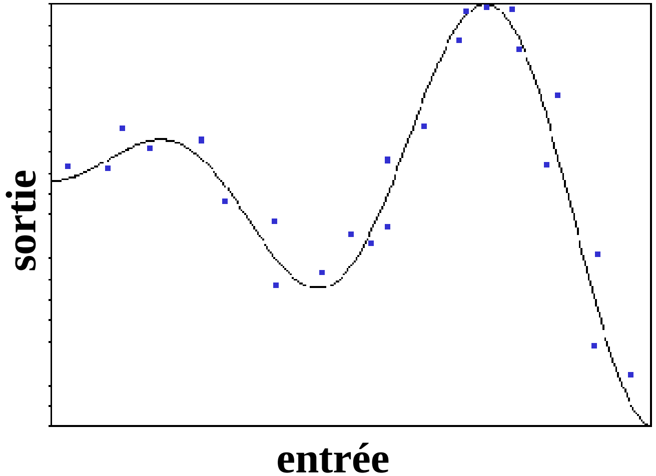
  tq : $u_i = f(x_i)$

- Ou bien une distribution de probabilités $P(x_i, u_i)$

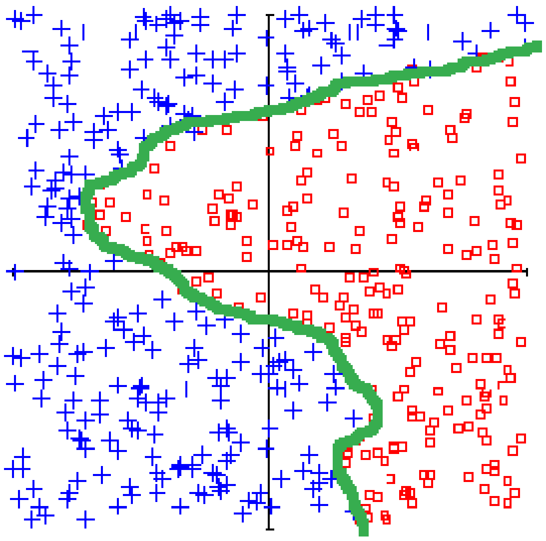
afin de prédire l'avenir

APPRENTISSAGE SUPERVISÉ : régression et classification

Régression (approximation)	Classification (y_i = « étiquettes »)



points = exemples ➔ courbe = régression



entrée = position point

sortie désirée = classe (□ =-1,+=+1)

➔

Fonction étiquette=f(x) (et frontière de séparation)

Apprentissage *non supervisé*

De l'*échantillon d'apprentissage* S = $\{(x_i)\}_{1,m}$

on cherche des régularités sous-jacente

- **Sous forme d'une fonction : *régression***
- **Sous forme de nuages de points (e.g. *mixture de gaussiennes*)**
- **Sous forme d'un modèle complexe (e.g. *réseau bayésien*)**

afin de résumer, détecter des régularités, comprendre …

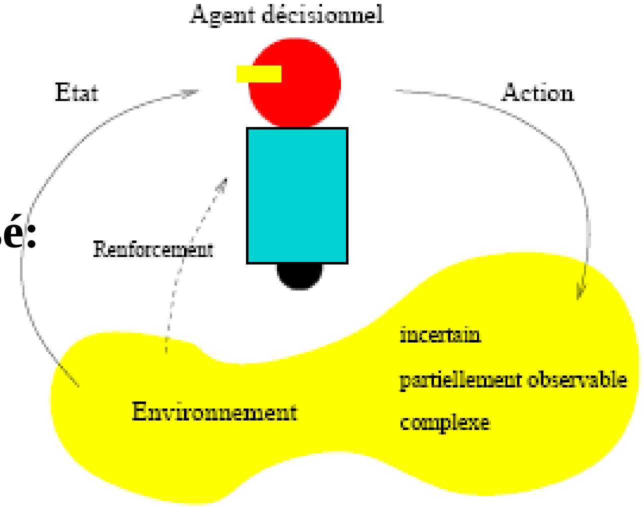
3. Apprentissage *par Renforcement* *(AR)*

Les données d'apprentissage

– Une séquence de perceptions, d'actions et de récompenses : $(s_t, a_t, r_t)_{t = 1, \infty}$

- Avec un renforcement r_t
- **r_t peut sanctionner des actions très antérieures à t**

Apprentissage Supervisé:



Le problème **:** inférer une application :

situation perçue → *action*

afin de maximiser un gain sur le long terme

(Comment sacrifier petit gain à court terme au profit du meilleur gain à long terme ?)

Apprentissage de réflexes ... -> … apprentissage de planification

Apprentissage Supervisé:

Plusieurs Techniques

Arbres De decision

KPP : Les K plus proches voisins

SVM:

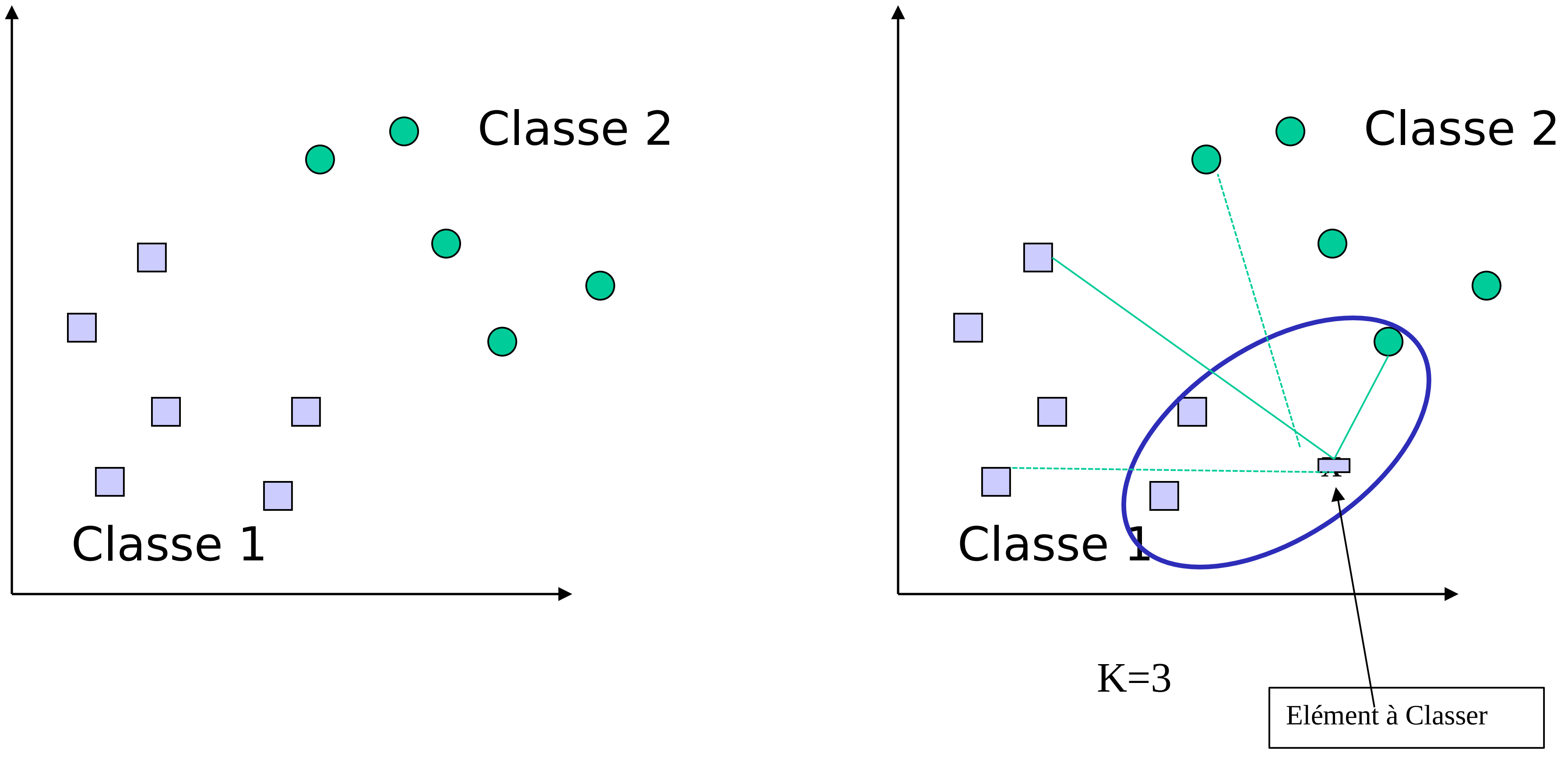
RNN

….

plus proches voisins

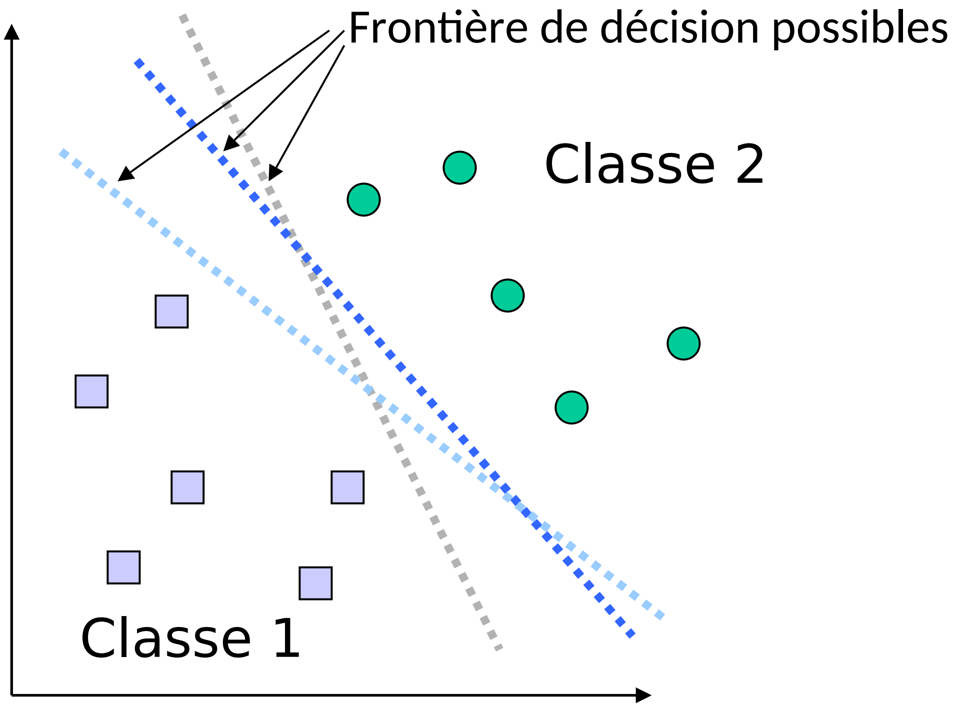
Exemple:

On cherche les K plus proches voisins de l’élément qu’on veut classifier: on classe l’élément selon la classe de la majorité des k voisins trouvés

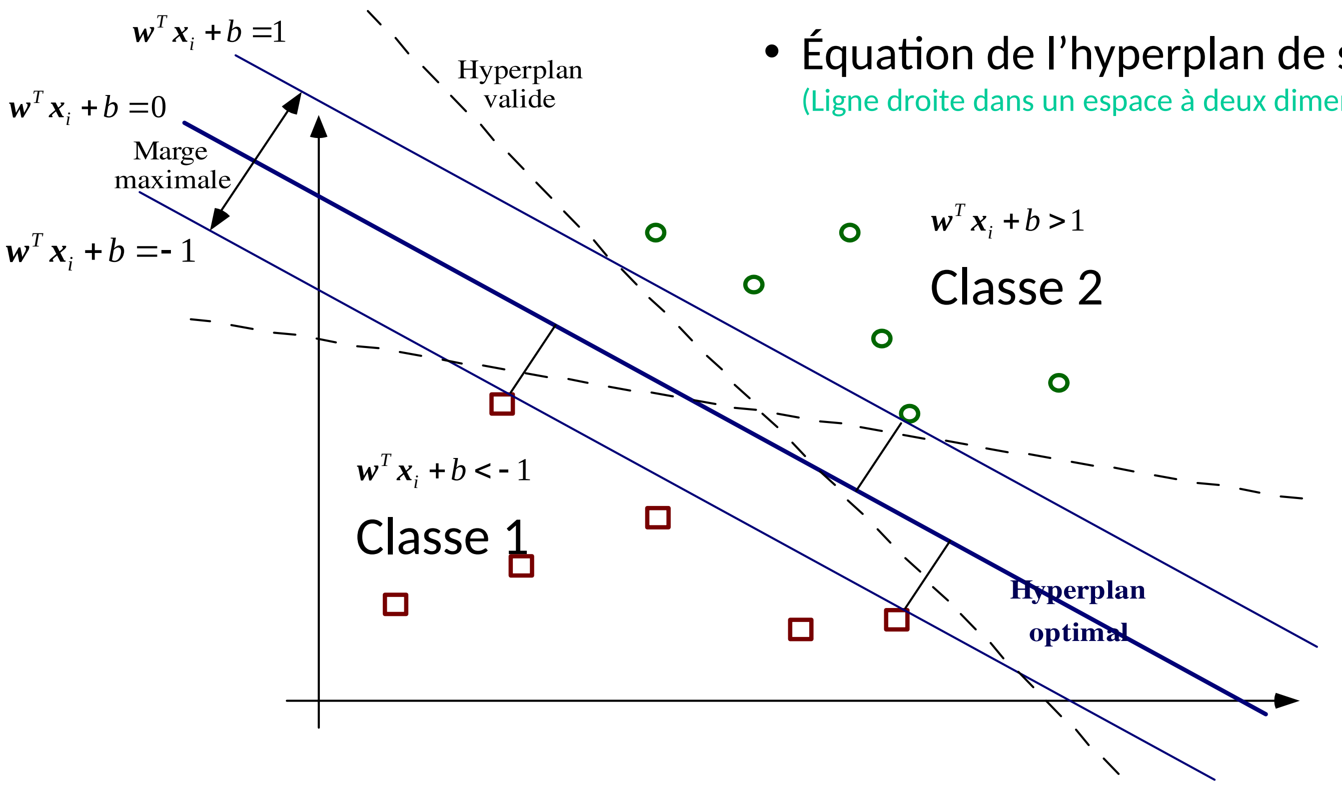


Sur les 3 voisins : 2 de la classe (1) et 1 classe (2), l’élément sera donc classifié en (1)

Problème à deux classes linéairement séparables



SVM (Support Vector Machine)
(Hyperplan de plus vaste marge)



- Équation de l’hyperplan de séparation : $y = w^T x + b$
  (Ligne droite dans un espace à deux dimensions)

- Si $\{x_i\} = \{x_1, \ldots, x_n\}$ est l’ensemble des données et $y_i \in \{-1,1\}$ est la classe de chacune, on devrait avoir :

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \qquad \forall i$$

tout en ayant une distance optimale entre x_i et le plan de séparation

Problème d'optimisation quadratique

- Maximiser le pouvoir de généralisation du classifieur revient donc à trouver $\boldsymbol{w}$ et b tels que :

$$\frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{est minimum}$$

et

$$y_i\left(\boldsymbol{w}^T \boldsymbol{x}_i + b\right) \geq 1, \qquad i = 1,\ldots,n$$

- Si d est la dimension des $\boldsymbol{x}_i$ (nombre d'entrées), cela revient à régler d+1 paramètres (les éléments de $\boldsymbol{w}$, plus b)
  - Possible par des méthodes d'optimisation classiques (optimisation quadratique) seulement si d pas trop grand (< qqs 10^3)
  - L'approche SVM utilise les **multiplicateurs de Lagrange** pour une solution plus simple

Apprentissage Supervisé:

arbres de décision

Une des formes les plus simples d’apprentissage, mais tout de même une de celles qui connaissent le plus de succès.

À partir d’exemples, le but est d’apprendre des structures d’arbres permettant de prendre des décisions.

Chaque noeud représente un test à faire.

Chaque branche représente une valeur possible résultant du test.

Une feuille correspond à la décision.

Exemple (Jouer au tennis)

On a enregistré les différents états des journées où on a joué ou pas dans le tableau suivant:

Journée	Ciel	Température	Humidité	Vent	JouerTennis
J1	Ensoleillé	Chaude	Élevée	Faible	Non
J2	Ensoleillé	Chaude	Élevée	Fort	Non
J3	Nuageux	Chaude	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaude	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

Les valeurs de l’attribut Ciel sont : ensoleillé, nuageux et pluvieux

Les valeurs de l’attribut Température sont: chaude, tempérée, et froide

Les valeurs de l’attribut Humidité sont: élevée et normale

Les valeurs de l’attribut Vent sont: faible et fort

QUESTION: Y a t- il une règle permettant de décider (jouer ou pas) pour aujourd’hui (pluvieux et vent faible)

Construire un arbre de décision

L’apprentissage d’un arbre de décision est fait à partir d’exemples de valeurs d’attributs et de la valeur résultante du prédicat à apprendre.

Exemple = ensemble de valeurs d’attributs (propriétés)

La valeur du prédicat est appelée la classification de l’exemple (ex: Vrai / Faux).

L’ensemble des exemples est appelé l’ensemble d’entraînement.

Le but est de trouver le plus petit arbre qui respecte l’ensemble d’entraînement.
(NP-complex)

L’arbre doit extraire des tendances ou des comportements (règles) à partir des exemples.

Algorithme

Procédure : construire-arbre(S) , S=ensemble d'exemples

Si tous les exemples de **S** appartiennent à la même classe **alors**

créer une feuille portant le nom de cette classe

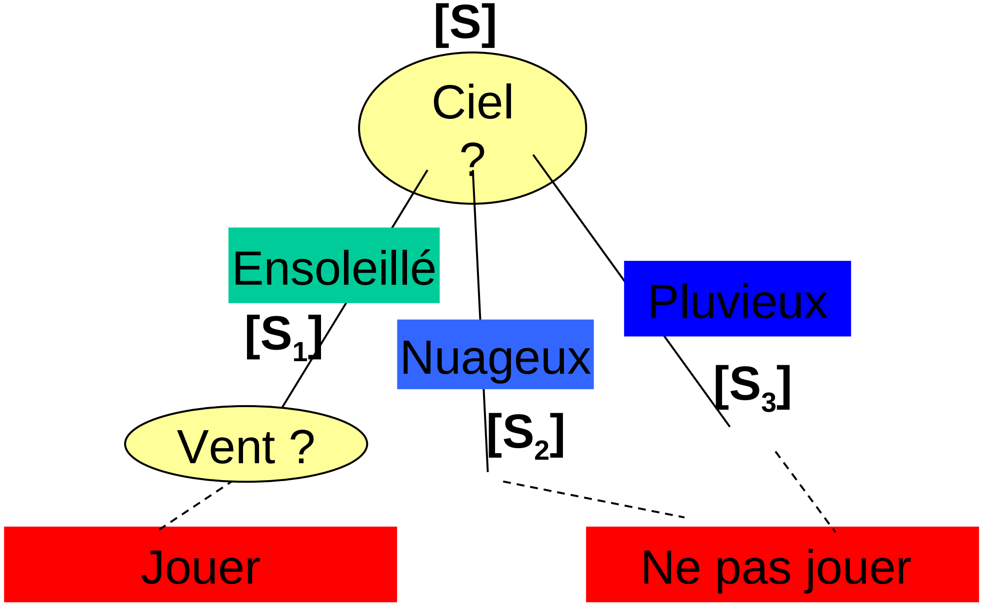
sinon

choisir un (Le meilleur) attribut A pour créer un nœud

Le test associé à ce nœud sépare **S** en V parties : S_1,,, S_v

construire-arbre (S_k) k=1,,,V (V : nombre de valeurs de A)

finsi



Exemple

Est-ce une bonne journée pour jouer au tennis ?
Attributs : ciel, Température, Humdité et vent (caracteristiques d'une journée)
2 Classes: Jouer(OUI) , ne pas jouer (NON)
Les exemples: les journées

Journée	*Ciel*	*Température*	*Humidité*	*Vent*	*JouerTennis*
J1	Ensoleillé	Chaude	Élevée	Faible	Non
J2	Ensoleillé	Chaude	Élevée	Fort	Non
J3	Nuageux	Chaude	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaude	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

Les valeurs de l'attribut Ciel sont : Ensoleillé, nuageux et pluvieux

Les valeurs de l'attribut Température sont: chaude, tempérée, et froide

Les valeurs de l'attribut Humidité sont: élevée et normal

Les valeurs de l'attribut Vent sont: faible et fort

Le Choix du meilleur Attribut (plusieurs Algorithmes)

Algorithme (ID3)

Il construit les arbres de décision de haut en bas.

Il place à la racine l'attribut le plus important, c'est-à-dire celui qui sépare au mieux les exemples positifs et négatifs.

Par la suite, il y a un nouveau noeud pour chacune des valeurs possibles de cet attribut. Pour chacun de ces noeuds, on recommence le test avec le sous-ensemble des exemples d'entraînement qui ont été classés dans ce noeud.

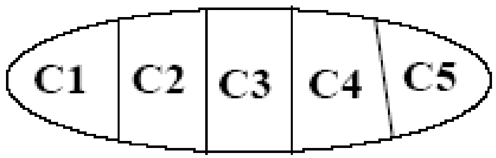
- **L'entropie de Boltzmann ...**
- **... et de Shannon**
  - Shannon en 1949 a proposé une mesure d'entropie valable pour les distributions discrètes de probabilité.
  - Elle exprime la quantité d'information, c'est à dire le nombre de bits nécessaire pour spécifier la distribution
  - L'entropie d'information est:

$$I = -\sum_{i=1..k} p_i \times \log_2(p_i)$$

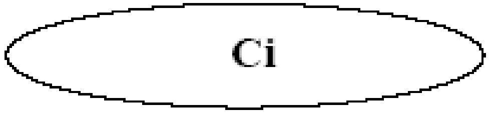
où p_i est la probabilité de la classe C_i.

$H_s(C|A) = - \sum_i P(X_i) \,.\, \sum_k P(C_k|X_i).\log(P(C_k|X_i$

Entropie d'information de N objets: $I = -\sum_{i=1..k} pr(Ci) \times \log_2 pr(Ci)$



k classes qui probables: $\mathbf{I = lg_2(k)}$



1 seule classe: **EI=0**

- Nulle quand il n'y a qu'une classe
- D'autant plus grande que les classes sont équiprobables
- Vaut $\log_2(k)$ quand les k classes sont <u>équiprobables</u>
- Unité: le bit d'information

l'Entropie mesure alors le degré de l'hétérogénité dans une population

Choix de l'attribut

On choisit l'attribut ayant le meilleur gain d'information:

$$Gain(S, A) \equiv Entropie(S) - \sum_{v \in V(A)} \frac{|S_v|}{|S|} Entropie(S_v)$$

$$Entropie(S) \equiv \sum_{i=1}^{c} (-p_i) log_2(p_i)$$

S **: les exemples d'entraînement.**
A **: l'attribut à tester.**
V**(*****A*****) : les valeurs possibles de l'attribut *****A*****.**
S_v : le sous-ensemble de *****S*** qui contient les exemples qui ont la valeur *****v***** pour l'attribut *****A*****.**
c **: le nombre de valeurs possibles pour la fonction visée (classes).**
pi **: la proportion des exemples dans *****S***** qui ont *****i***** comme valeur pour la fonction visée.**

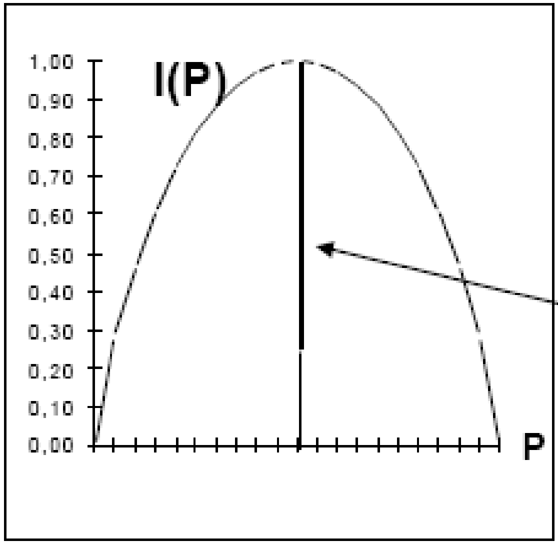
Le critère entropique (3/3) : le cas de deux classes

- **Pour k=2 on a :** $I(p,n) = -p_+ \times \log_2(p_+) - p_- \times \log_2(p_-)$

  D'après l'hypothèse H1 on a $p_+ = p/(p+n)$ et $p_- = n/(p+n)$

d'où $$I(p,n) = -\frac{p}{(p+n)} \log\left(\frac{p}{(p+n)}\right) - \frac{n}{(p+n)} \log\left(\frac{n}{(p+n)}\right)$$

et $$I(P) = -P \log P - (1-P) \log(1-P)$$



P=p/(p+n)=n/(n+p)=0.5
équiprobable

$$Gain(S,A) \equiv Entropie(S) - \sum_{v \in V(A)} \frac{|S_v|}{|S|} Entropie(S_v)$$

$$Entropie(S) \equiv \sum_{i=1}^{c} (-p_i) log_2(p_i)$$

Exemple (arbre ID3)

Premièrement, il faut choisir la racine de l'arbre.
Pour cela, nous allons choisir l'attribut qui a le plus grand gain d'information.
– Pour calculer le gain d'information, nous devons d'abord calculer l'entropie des exemples d'entraînement.
– Il y a 9 exemples positifs et 5 exemples négatifs, sur un total de 14, donc nous obtenons une entropie de:

$$\begin{aligned} Entropie(S) &= \sum_{i=1}^{c} (-p_i) log_2(p_i) \\ &= (-9/14)\log_2(9/14) + (-5/14)\log_2(5/14) \\ &= 0.94 \end{aligned}$$

Maintenant, nous allons calculer le gain d'information pour le premier attribut, l'attribut ***Ciel***.
– Cet attribut a 3 valeurs possibles, donc les exemples d'entraînement seront regroupés en 3 sous-ensembles.
– Nous commençons donc par calculer l'entropie des 3 sous-ensembles:

$S_{ensoleillé}=\{j_1^-, j_2^-, j_8^-, j_9^+, j_{11}^+\}$
$S_{nuageux}=\{j_3^+, j_7^+, j_{12}^+, j_{13}^+\}$
$S_{pluvieux}=\{j_4^+, j_5^+, j_6^-, j_{10}^+, j_{14}^-\}$

$$\begin{aligned} Entropie(S_{ensoleillé}) &= (-2/5)\log_2 2/5 + (-3/5)\log_2 3/5 = 0.971 \\ Entropie(S_{nuageux}) &= (-4/4)\log_2 4/4 + (-0/4)\log_2 0/4 = 0 \\ Entropie(S_{pluvieux}) &= (-3/5)\log_2 3/5 + (-2/5)\log_2 2/5 = 0.971 \end{aligned}$$

Journée	Ciel	Température	Humidité	Vent	JouerTennis
J1	Ensoleillé	Chaude	Élevée	Faible	Non
J2	Ensoleillé	Chaude	Élevée	Fort	Non
J3	Nuageux	Chaude	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaude	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

Le calcul du gain d'information pour l'attribut ***Ciel*** va donc donner:

$$\begin{aligned} Gain(S, Ciel) &= Entropie(S) - \sum_{v \in V(Ciel)} \frac{|S_v|}{|S|} Entropie(S_v) \\ &= 0.94 - ((5/14) \times Entropie(S_{ensoleillé}) + \\ &\quad (4/14) \times Entropie(S_{nuageux}) + \\ &\quad (5/14) \times Entropie(S_{pluvieux})) \\ &= 0.94 - ((5/14) \times 0.971 + (4/14) \times 0 + (5/14) \times 0.971) \\ &= 0.94 - 0.694 \\ &= 0.246 \end{aligned}$$

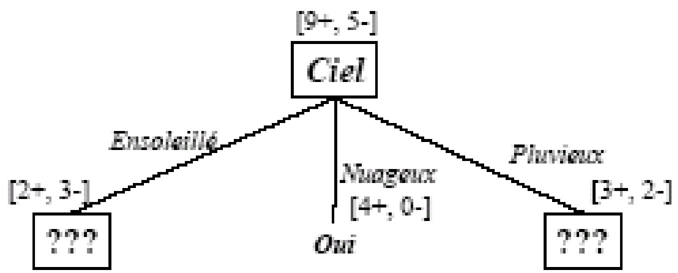
Exemple

On calcul le gain de la même manière pour les trois autres attributs:
L'attribut qui a le plus grand gain d'information est l'attribut *Ciel*, donc se sera la racine de l'arbre de décision.

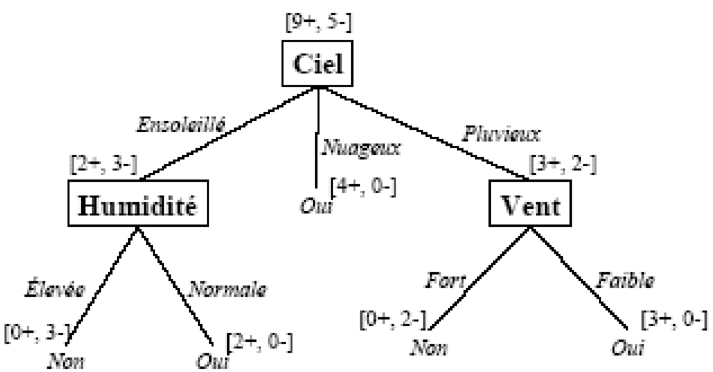
$$
\begin{aligned}
Gain(S, Ciel) &= 0.246 \\
Gain(S, Humidité) &= 0.151 \\
Gain(S, Vent) &= 0.048 \\
Gain(S, Température) &= 0.029
\end{aligned}
$$

En séparant les exemples selon les valeurs de l'attributs *Ciel*, on obtient l'arbre partiel:

$S_{ensoleillé}=\{j_1^-, j_2^-, j_8^-, j_9^+, j_{11}^+\}$
$S_{nuageuxé}=\{j_3^+, j_7^+, j_{12}^+, j_{13}^+\}$
$S_{pluvieux}=\{j_4^+, j_5^+, j_6^-, j_{10}^+, j_{14}^-\}$



On peut voir que lorsque le ciel est nuageux, il reste uniquement des exemples positifs, donc ce noeud devient une feuille avec une valeur de *Oui* pour la fonction visée.
Pour les deux autres noeuds, il y a encore des exemples positifs et négatifs, alors il faut recommencer le même calcul du gain d'information, mais avec les sous-ensembles restant.



On a alors les règles de décision:

SI le ciel est ensoleillé et l'humidité est élevée ALORS on ne joue pas

SI le ciel est ensoleillé et l'humidité est normale ALORS on joue

SI le ciel est nuageux ALORS on joue

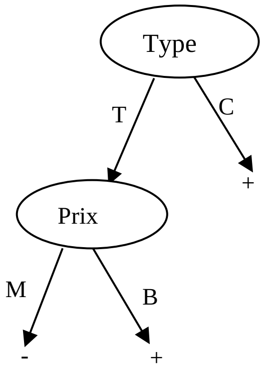
SI le ciel est pluvieux et le vent est fort ALORS on ne joue pas

SI le ciel est pluvieux et le vent est faible ALORS on joue

Exercice 1

On veut déterminer si on aime ou pas un restaurant donné
Les attributs étudiés sont : le prix et le type de ce qui est à manger
Les valeurs pour Prix : Bas(B), Moyen(M) et Haut(H)
Les valeurs pour Type : Tagine (T) et Couscous(C).
Les classes : Aimer (+) et Ne pas Aimer (-)
On utilise les exemples d'apprentissage suivants :

exemple	Type	Prix	Classe
1	T	B	+
2	C	B	+
3	T	M	-
4	C	M	+
5	C	H	-



T1

- Est-ce que l'arbre T1 classifie bien les exemples ?

- En commençant par l'attribut Prix, Donner l'arbre obtenu T2.

- Calculer le gain (par ID3) pour l'attribut Prix

- Calculer le gain (par ID3) pour l'attribut Type

- Construire alors l'arbre T3

- Traduire l'arbre T3 en règles

log2=1, log1/2=-1, log1/3 = -1.585, log2/3 = -0.585, log2/5=-1.322 ,log3/5=-0.737

Trois algorithmes principaux

Il existe 3 mesures différentes de la qualité du partitionnement

3 algorithmes d 'arbre différents

- Indice de pureté - coefficient de corrélation: On choisit la variable X qui maximise son coefficient de corrélation avec la classe à prédire: algorithme CART
- Écart à l 'indépendance - le lien du χ^2: Mesure très utilisée en statistique, on choisit la variable X qui maximise son lien avec la classe à prédire: algorithme ChAID
- Gain informationnel - entropie de Shannon: mesure très utilisée en Intelligence Artificielle: algorithme ID3
- Hétérogénéité, indice de Gini: Gini (S) = $\sum_{i \neq i} pi * pj$, i et j sont des classes

Sur-apprentissage

Un arbre peut avoir une erreur apparente nulle mais une erreur réelle importante, c'est-à-dire être bien adapté à l'échantillon mais avoir un pouvoir de prédiction faible.

- Mémorisation de tout l'ensemble d'apprentissage plutôt que d'induire un concept général..
- Plus on apprend, plus on "colle" aux données
- A la fin, tous les exemples sont appris par coeur, y compris le bruit

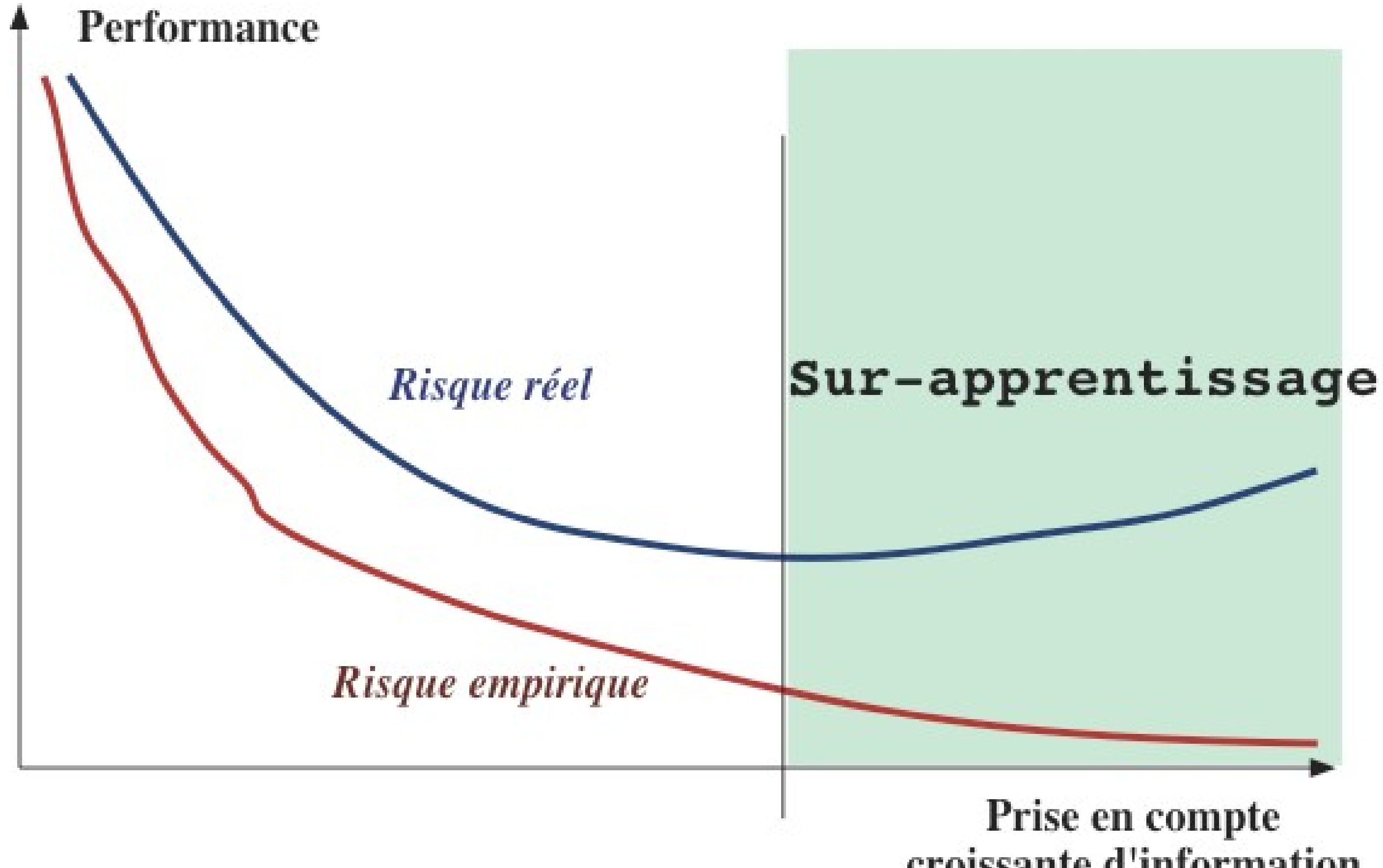
→ **Problème de surapprentissage**

Eviter le sur-apprentissage

● On peut utiliser un ensemble de test pour arrêter la construction de l'arbre quand l'estimation de l'erreur ne diminue plus.

● On peut construire l'arbre en entier, **puis l'élaguer** (réduire l'arbre): **processus d'élagage**

Le sur-apprentissage



Récapitulatif

- Méthode de référence en apprentissage supervisé
- Méthode très répandue, rapide et disponible *(http://www.cse.unsw.edu.au/~quinlan)*
- Méthode relativement sensible au bruit

Exercice 2

Les chercheurs de la chaîne de café Columbus ont collecté les informations suivantes concernant le fait si les clients aiment leur café avec différents arômes ajoutes. Les trois attributs sont des attributs binaires qui indiquent si l'arôme a été ajoute ou pas.

Menthe	Noisette	Vanille	Aimé ?
oui	oui	non	non
oui	non	non	oui
non	non	non	oui
non	oui	non	non

- Donner l’attribut à la racine de l’arbre de décision avec ID3. Donnez les détails du calcul. Est-ce qu'après avoir choisi la racine on doit choisir un autre noeud? Pourquoi ?

Exercice 3

Construire un arbre de décision relatif aux formules logiques suivantes:

· $a \wedge \neg b$

· $a \vee (b \wedge c)$

· a *xor* b

Exercice 4

Considérons un exemple simple ci-dessous. Nous remarquons que certains éléments sont redondants.

Exemple	a	b	Classe
1	1	1	+
2	1	1	+
3	0	0	+
4	1	0	
5	0	1	-
6	0	1	-

Question 1. En utilisant la mesure d'entropie, construisez l'arbre de décision associé à cet exemple.

Question 2. Est-ce que l'arbre de décision change lorsque nous enlevons les exemples redondants ?

Question 3. A présent nous rajoutons un septième exemple négatif.

7	1	1	-

Quelle est la structure de l'arbre de décision ?

Qu'en concluez vous ?

($\log_2(1/3)=-1,585$; $\log_2(2/3)=-0,585$; $\log_2(1/5)=-2,322$; $\log_2(2/5)=-1,322$; $\log_2(3/5)=-0,737$; $\log_2(4/5)=-0,322$; $\log_2(2/7)=-1,807$; $\log_2(5/7)=-0,485$)