

Induction d'arbres de décision

Antoine Cornuèjols
(antoine@lri.fr)

I.I.E.
&
L.R.I., Université d'Orsay

1- Les arbres de décision : le problème

- Chaque instance est décrite par un vecteur d'attributs/valeurs

	Toux	Fievre	Poids	Douleur
Marie	non	oui	normal	gorge
Fred	non	oui	normal	abdomen
Julie	oui	oui	maigre	aucune
Elvis	oui	non	obese	poitrine

- En entrée : un ensemble d'instances et leur classe (correctement associées par un "professeur" ou "expert")

	Toux	Fievre	Poids	Douleur	Diagnostic
Marie	non	oui	normal	gorge	rhume
Fred	non	oui	normal	abdomen	appendicite
.....					

- L'algorithme d'apprentissage doit construire un arbre de décision

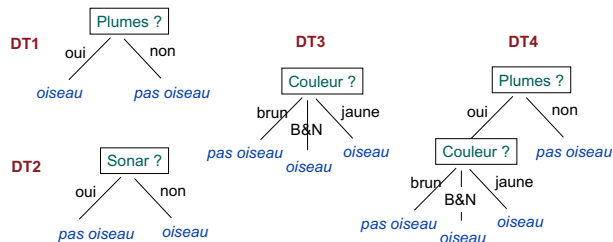
E.g. Un arbre de décision pour le diagnostic

Une des principales applications de l'apprentissage !

2- Les arbres de décision : le choix d'un arbre

	Couleur	Ailes	Plumes	Sonar	Concept
Faucon	jaune	oui	oui	non	oiseau
Pigeon	B&N	oui	oui	non	oiseau
chauve-souris	brun	oui	non	oui	pas oiseau

Quatre arbres de décision cohérents avec les données:

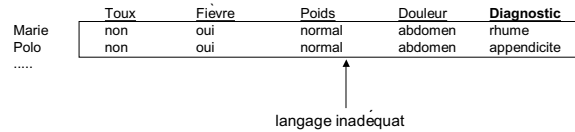


Induction d'arbres de décision

- La tâche**
 - Apprentissage d'une fonction de discrimination entre les formes de plusieurs classes
- Le protocole**
 - Apprentissage supervisé par approximation itérative gloutonne
- Le critère de succès**
 - Le taux d'erreur en classification
- Les entrées**
 - Données en attributs-valeurs (espace à N dimensions)
- Les fonctions cibles**
 - Arbres de décision

1- Les arbres de décision : pouvoir de représentation

- Le choix des attributs est très important !
- Si un attribut crucial n'est pas représenté on ne pourra pas trouver d'arbre de décision qui apprenne les exemples correctement.
- Si deux instances ont la même représentation mais appartiennent à deux classes différentes, le langage des instances (les attributs) est dit inadéquat.



2- Les arbres de décision : le choix d'un arbre

- Si le langage est adéquat, il est toujours possible de construire un arbre de décision qui classe correctement les exemples d'apprentissage.
- Il y a le plus souvent de nombreux arbres de décision possibles.

Quelle valeur attribuer à un arbre ?

- Impossibilité de procéder par énumération / évaluation (NP-complet)

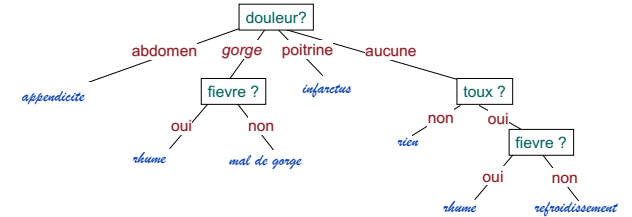
$$\prod_{i=1}^A i^{V^{A-i}}$$

4 attributs & 3 valeurs / attribut : 55296 arbres

Nécessité d'une démarche constructive itérative

1- Les arbres de décision : exemple

- Les arbres de décision sont des classificateurs pour des instances représentées dans un formalisme attribut/valeur
 - Les noeuds de l'arbre testent les attributs
 - Il y a une branche pour chaque valeur de l'attribut testé
 - Les feuilles spécifient les catégories (deux ou plus)



1- Les arbres de décision : pouvoir de représentation

- Toute fonction booléenne peut se représenter comme un arbre de décision
 - Rappel:** avec 6 attributs booléens, on peut définir environ 2 milliards de fonctions booléennes.
- Selon les fonctions à représenter les arbres sont plus ou moins grands
 - E.g. Pour les fonctions "parité" et "majorité", la taille de l'arbre peut grandir exponentiellement !
 - Pour d'autres fonctions, un seul nœud peut parfois suffire...
- Limité à la logique des propositions (on ne représente pas de relations)
- Un arbre peut se représenter par une disjonction de règles
 - (Si Plumes = non ou (Si Plumes = oui & Couleu r= brun) ou (Si Plumes = oui & Couleu r= B&N) ou (Si Plumes = oui & Couleu r= jaune) Alors Classe= pas-oiseau)
 - Alors Classe= pas-oiseau
 - Alors Classe= oiseau
 - Alors Classe= oiseau

2- Quel modèle pour la généralisation ?

- Parmi toutes les hypothèses cohérentes possibles, laquelle faut-il choisir en vue d'une bonne généralisation ?
 - La réponse intuitive ...
 - ... est-elle confirmée par la théorie ?
- Un peu de théorie de l'apprenabilité [Vapnik,82,89,95]
 - La consistance de la minimisation du risque empirique (ERM)
 - Le principe de minimisation du risque structurel (SRM)
- Bref, il faut faire bref ...
 - Comment ?
 - Méthodes d'induction d'arbres de décision

3- Induction d'arbres de décision : Exemple [Quinlan,86]

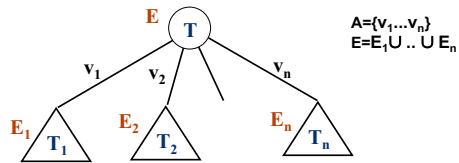
Attributs	Pif	Temp	Humid	Vent	Valeurs possibles
					soleil,couvert,pluie chaud,bon,frais normale,haute vrai,faux

N°	Pif	Temp	Humid	Vent	Golf
1	soleil	chaud	haute	faux	NePasJouer
2	soleil	chaud	haute	vrai	NePasJouer
3	couvert	chaud	haute	faux	Jouer
4	pluie	bon	haute	faux	Jouer
5	pluie	frais	normale	faux	Jouer
6	pluie	frais	normale	vrai	NePasJouer
7	couvert	frais	normale	vrai	Jouer
8	soleil	bon	haute	faux	NePasJouer
9	soleil	frais	normale	faux	Jouer
10	pluie	bon	normale	faux	Jouer
11	soleil	bon	normale	vrai	Jouer
12	couvert	bon	haute	vrai	Jouer
13	couvert	chaud	normale	faux	Jouer
14	pluie	bon	haute	vrai	NePasJouer

la classe

3- Induction d'arbres de décision : algorithme TDIDT

PROCEDURE AAD(T,E)
SI tous les exemples de E sont dans la même classe Ci
ALORS affecter l'étiquette Ci au noeud courant FIN
SINON sélectionner un attribut A avec les valeurs v₁...v_n
 Partitionner E selon v₁...v_n en E₁, ..., E_n
 Pour j=1 à n AAD(T_j, E_j).



3- Mesure d'impureté : le critère Gini

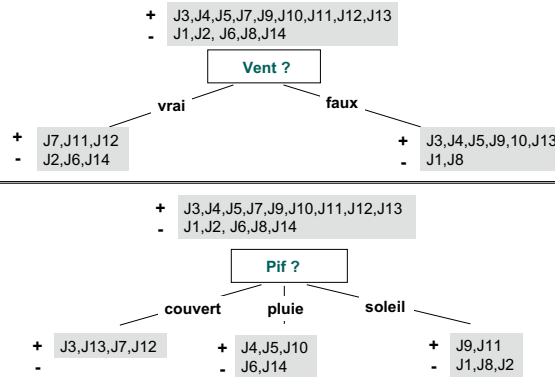
- Idealement :**
 - Mesure nulle si les populations sont homogènes
 - Mesure maximale si les populations sont maximalelement mélangées
- Index Gini** [Breiman et al.,84]

$$Gini(D) = 1 - \sum_{j=1}^k (p_j)^2$$

3- Induction d'arbres de décision

- Stratégie : Induction descendante : TDIDT**
 - Recherche en meilleur d'abord sans retour arrière (gradient) avec une fonction d'évaluation
 - Choix récursif d'un attribut de test jusqu'à critère d'arrêt
- Fonctionnement :**
 - On choisit le premier attribut à utiliser pour l'arbre : le plus informatif
 - Après ce choix, on se trouve face au problème initial sur des sous-ensembles d'exemples.
 - D'où un algorithme récursif.

3- Induction d'arbres de décision : sélection de l'attribut



3- Le critère entropique (1/3)

- 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:

$$I = - \sum_{i=1..k} p_i \times \log_2(p_i)$$
 ou p_i est la probabilité de la classe C_i.

3- Induction d'arbres de décision : exemple

Si on choisissait l'attribut Temp?

N°	Pif	Temp	Humid	Vent	Golf
1	soleil	chaud	haute	faux	NePasJouer
2	soleil	chaud	haute	vrai	NePasJouer
3	couvert	chaud	haute	faux	Jouer
4	pluie	bon	haute	faux	Jouer
5	pluie	frais	normale	faux	Jouer
6	pluie	frais	normale	vrai	NePasJouer
7	couvert	frais	normale	vrai	Jouer
8	soleil	bon	haute	faux	NePasJouer
9	soleil	frais	normale	faux	Jouer
10	pluie	bon	normale	faux	Jouer
11	soleil	bon	normale	vrai	Jouer
12	couvert	bon	haute	vrai	Jouer
13	couvert	chaud	normale	faux	Jouer
14	pluie	bon	haute	vrai	NePasJouer

+ J3,J4,J5,J7,J9,J10,J11,J12,J13
 - J1,J2, J6,J8,J14

Temp?

chaud bon frais

+ J3,J13
 - J1,J2

+ J4,J10,J11,J13
 - J8,J14

+ J5,J7,J9
 - J6

3- La sélection d'un bon attribut de test

- Comment obtenir un arbre "simple" ?**
 - Arbre simple :
 - Minimise l'espérance du nombre de tests pour classer un nouvel objet
 - Comment traduire ce critère global en une procédure de choix locale ?
- Critères de choix de chaque noeud**
 - On ne sait pas associer un critère local au critère global objectif
 - Recours à des heuristiques
 - La notion de **mesure d'impureté**
 - Index Gini
 - Critère entropique (ID3, C4.5, C5.0)
 - ...

3- Le critère entropique (2/3)

Entropie d'information de S (en C classes) :

$$I(S) = - \sum_{i=1}^C p(c_i) \cdot \log p(c_i)$$

p(c_i) : probabilité de la classe c_i

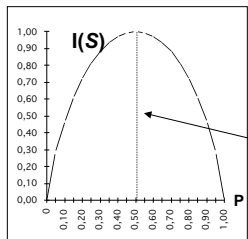
- Nulle quand il n'y a qu'une classe
- D'autant plus grande que les classes sont équiprobables
- Vaut log₂(k) quand les k classes sont équiprobables
- Unité: le bit d'information

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

- Pour $C=2$ on a : $I(S) = -p_v \times \log_2(p_v) - p_{\bar{v}} \times \log_2(p_{\bar{v}})$
 D'après l'hypothèse on a $p_v = p / (p+n)$ et $p_{\bar{v}} = n / (p+n)$

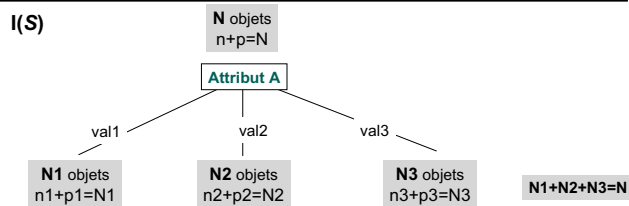
d'où $I(S) = - \frac{p}{p+n} \log_2 \left(\frac{p}{p+n} \right) - \frac{n}{p+n} \log_2 \left(\frac{n}{p+n} \right)$

et $I(S) = -P \log_2 P - (1-P) \log_2 (1-P)$



$P=p/(p+n)=n/(n+p)=0.5$
 équiprobable

3- Exemple (2/4)



$E(N,A) = N1/N \times I(p1,n1) + N2/N \times I(p2,n2) + N3/N \times I(p3,n3)$

Le gain d'entropie de A vaut: **GAIN(A) = I(S) - E(N,A)**

3- Des systèmes TDIDT

Entrée : vecteur d'attributs valués associés à chaque exemple

Sortie : arbre de décision

- CLS (Hunt, 1966) [analyse de données]
- **ID3 (Quinlan 1979)**
- ACLS (Paterson & Niblett 1983)
- ASSISTANT (Bratko 1984)
- **C4.5 (Quinlan 1986)**
- CART (Breiman, Friedman, Ohlson, Stone, 1984)

3- Gain entropique associé à un attribut

$$Gain(S, A) = I(S) - \sum_{v \in \text{valeurs}(A)} \frac{|S_v|}{|S|} \cdot I(S_v)$$

$|S_v|$: taille de la sous-population dans la branche v de A

En quoi la connaissance de la valeur de l'attribut A m'apporte une information sur la classe d'un exemple

3- Exemple (3/4)

- Pour les exemples initiaux $I(S) = -9/14 \log_2(9/14) - 5/14 \log_2(5/14)$
- Entropie de l'arbre associé au test sur Pif ?
 - $E(\text{Pif}) = 4/14 I(p_1, n_1) + 5/14 I(p_2, n_2) + 5/14 I(p_3, n_3)$
 - ⇒ **Gain(Pif) = 0.940 - 0.694 = 0.246 bits**
 - **Gain(Temp) = 0.029 bits**
 - **Gain(Humid) = 0.151 bits**
 - **Gain(Vent) = 0.048 bits**

⇒ **Choix de l'attribut Pif pour le premier test**

4- Les problèmes potentiels

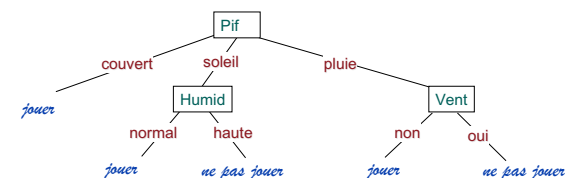
1. Attributs à valeur continue
2. Attributs à facteurs de branchement différents
3. Valeurs manquantes
4. Sur-apprentissage
5. Recherche gloutonne
6. Le choix des attributs
7. Variance des résultats :
 - arbres différents à partir de données peu différentes

3- Exemple (1/4)

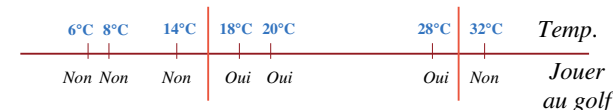
- Entropie de l'ensemble initial d'exemples $I(p,n) = -9/14 \log_2(9/14) - 5/14 \log_2(5/14)$
- Entropie des sous-arbres associés au test sur **Pif ?**
 - $p_1 = 4 \quad n_1 = 0 : I(p_1, n_1) = 0$
 - $p_2 = 2 \quad n_2 = 3 : I(p_2, n_2) = 0.971$
 - $p_3 = 3 \quad n_3 = 2 : I(p_3, n_3) = 0.971$
- Entropie des sous-arbres associés au test sur **Temp ?**
 - $p_1 = 2 \quad n_1 = 2 : I(p_1, n_1) = 1$
 - $p_2 = 4 \quad n_2 = 2 : I(p_2, n_2) = 0.918$
 - $p_3 = 3 \quad n_3 = 1 : I(p_3, n_3) = 0.811$

3- Exemple (4/4)

• Arbre final obtenu :



4.1. Discretisation des attributs à valeur continue



Ici, deux seuils candidats : 16°C et 30°C

L'attribut $Temp_{>16°C}$ est le plus informatif, on le retient

4.2. Facteurs de branchement différents

- **Problème :**
le critère de gain entropique favorise les attributs ayant un facteur de branchement plus élevé
- **Deux solutions :**
 - ❑ Rendre tous les attributs binaires
 – Mais perte d'intelligibilité des résultats
 - ❑ Introduire un facteur de normalisation dans le calcul

$$Gain_norm(S,A) = \frac{Gain(S,A)}{\sum_{i=1}^{nb\ valeurs\ de\ A} \frac{|S_i|}{|S|} \cdot \log \frac{|S_i|}{|S|}}$$

5.1. Sur-apprentissage : Effet du bruit sur l'induction

- **Types de bruits**
 - ❑ Erreurs de description
 - ❑ Erreurs de classification
 - ❑ "clashes"
 - ❑ valeurs manquantes
- **Effet**
 - ❑ Arbre trop développé : « touffus », trop profond

5.2. Sur-apprentissage : Contrôle de la taille par pré-élagage

- **Idée : modifier le critère de terminaison**
 - ❑ Profondeur seuil (e.g. [Holte,93]: seuil =1 ou 2)
 - ❑ Test du chi2
 - ❑ Erreur Laplacienne
 - ❑ Faible gain d'information
 - ❑ Faible nombre d'exemples
 - ❑ Population d'exemples non statistiquement significative
 - ❑ Comparaison entre l'"erreur statique" et l'"erreur dynamique"
- **Problème : souvent trop myope**

4.3. Traitement des valeurs manquantes

- Soit un exemple $\langle x, c(x) \rangle$ dont on ne connaît pas la valeur pour l'attribut A
- Comment calculer $gain(S,A)$?
 1. Prendre comme valeur la valeur la plus fréquente dans S totale
 2. Prendre comme valeur la valeur la plus fréquente à ce noeud
 3. Partager l'exemple en **exemples fictifs** suivant les différentes valeurs possibles de A pondérés par leur fréquence respective
 - ❑ E.g. si 6 exemples à ce noeud prennent la valeur $A=a_1$ et 4 la valeur $A=a_2$
 $A(x) = a_1$ avec $prob=0.6$ et $A(x) = a_2$ avec $prob=0.4$
 - ❑ En prédiction, classer l'exemple par l'étiquette de la feuille la plus probable

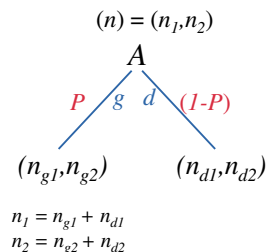
5.1. Sur-apprentissage : Le problème de la généralisation

- Le sur-apprentissage (over-fitting)
- **Risque empirique faible. Risque réel élevé.**
- Le principe SRM (Minimisation du Risque Structurel)
 - ❑ Justification [Vapnik,71,79,82,95]
 - Notion de "capacité" de l'espace des hypothèses
 - Dimension de Vapnik-Chervonensis

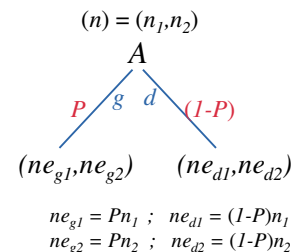
Il faut contrôler l'espace d'hypothèses

5.2. Exemple : le test du chi2

Soit un attribut binaire A



Hypothèse nulle



$$\chi^2 = \sum_{i=1}^2 \frac{(n_{gi} - Pn_i)^2}{Pn_i}$$

5- Le problème de la généralisation

A-t-on appris un bon arbre de décision ?

- Ensemble d'apprentissage. Ensemble test.
- Courbe d'apprentissage
- Méthodes d'évaluation de la généralisation
 - ❑ Sur un ensemble test
 - ❑ Validation croisée
 - ❑ "Leave-one-out"

5.1. Contrôle de l'espace H : motivations & stratégies

- **Motivations :**
 - ❑ Améliorer la performance en généralisation (SRM)
 - ❑ Fournir un modèle compréhensible des données (pour les experts)
- **Stratégies :**
 1. Contrôler directement la taille de l'arbre induit : *élagage*
 2. Modifier l'espace d'états (arbres) dans lequel se fait la recherche
 3. Modifier l'algorithme de recherche
 4. Restreindre la base de données
 5. Traduire les arbres obtenus dans une autre représentation

5.3. Sur-apprentissage : Contrôle de la taille par post-élagage

- **Idée : Elaguer après la construction de l'arbre entier, en remplaçant les sous-arbres optimisant un critère d'élagage par un noeud.**
- **Nombreuses méthodes. Encore beaucoup de recherches.**
 - ❑ Minimal Cost-Complexity Pruning (MCCP) (Breiman et al.,84)
 - ❑ Reduced Error Pruning (REP) (Quinlan,87,93)
 - ❑ Minimum Error Pruning (MEP) (Niblett & Bratko,86)
 - ❑ Critical Value Pruning (CVP) (Mingers,87)
 - ❑ Pessimistic Error Pruning (PEP) (Quinlan,87)
 - ❑ Error-Based Pruning (EBP) (Quinlan,93) (utilisé dans C4.5)
 - ❑ ...

5.3- Cost-Complexity pruning

- [Breiman et al.,84]
- Cost-complexity pour un arbre :

6. Modification de la stratégie de recherche

- **Idee** : ne plus utiliser une recherche en profondeur
- **Méthodes utilisant une autre mesure**:
 - Utilisation du principe de description minimale (MDLp)
 - Mesure de la complexité de l'arbre
 - Mesure de la complexité des exemples non codés par l'arbre
 - Garder l'arbre minimisant la somme de ces mesures
 - Mesure de la théorie de l'apprenabilité faible
 - Mesure de Kolmogorov-Smirnoff
 - Séparation de classes
 - Mesures hybrides de sélection de tests

7. Induction d'arbres obliques

- **Autre cause d'arbres touffus** : une représentation inadaptée
- **Solutions** :
 - Demander à l'*expert* (e.g. finale aux échecs [Quinlan,83])
 - Faire préalablement une *ACP*
 - Autre méthode de *sélection d'attributs*
 - Appliquer de l'*induction constructive*
 - *Induction d'arbres obliques*

5.3. Post-élagage par traitement en règles

1. Traduire chaque branche par une règle
2. Pour chaque règle, retirer les conditions qui permettent un accroissement de la performance en généralisation (ens. de test)
3. Ordonner les règles par performance décroissante et les tester dans cet ordre lors de la prédiction

Avantages supplémentaires :

- lisibilité des résultats
- souvent plus grande stabilité du résultat

7. Modification de l'espace de recherche

- **Modification des tests dans les noeuds**
 - Pour remédier aux effets d'une représentation inadaptée
 - Méthodes d'induction constructive (e.g. multivariate tests)

E.g. Arbres de décision obliques

- **Méthodes** :
 - Opérateurs numériques
 - Perceptron trees
 - Arbres et Programmation Génétique
 - Opérateurs logiques

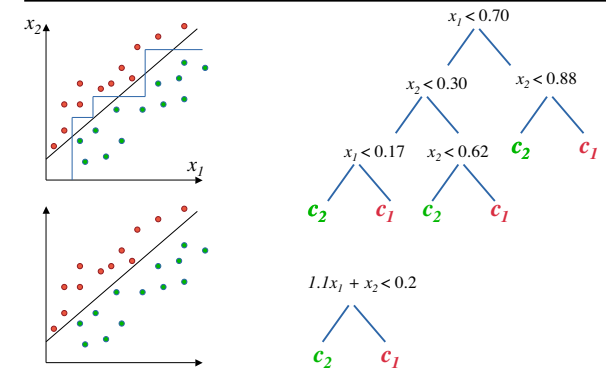
8. Traduction dans d'autres représentations

- **Idee** : Traduire un arbre complexe dans une représentation où le résultat est plus simple
- **Traduction en graphes de décision**
- **Traduction en ensemble de règles**

6. Recherche en avant

- Plutôt que de faire une recherche gloutonne, on peut faire une recherche à n coups en avant
 - Si je sélectionne d'abord tel attribut puis ensuite tel autre plutôt que ...
- Mais augmentation exponentielle de la complexité

7. Arbres obliques



9. Conclusions

- **Approprié pour** :
 - Classification de formes décrites en attributs-valeurs
 - Attributs à valeurs discrètes
 - Résistant au bruit
- **Stratégie** :
 - Recherche par construction incrémentale d'hypothèse
 - Critère local (gradient) fondé sur critère statistique
- **Engendre**
 - Arbre de décision interprétable (e.g. règles de production)
- Nécessite contrôle de la taille de l'arbre

Restriction de la base d'apprentissage

- **Ideé** : simplifier les arbres en simplifiant la base d'apprentissage
- **Méthodes de sélection d'exemples**
- **Méthodes d'élimination d'attributs**