Initiation à la Fouille de Données et à l'Apprentissage

Troisième séance
Apprentissage d'arbres de décision (1/2)

M2 I2A
2011-2012

Valentin Emiya

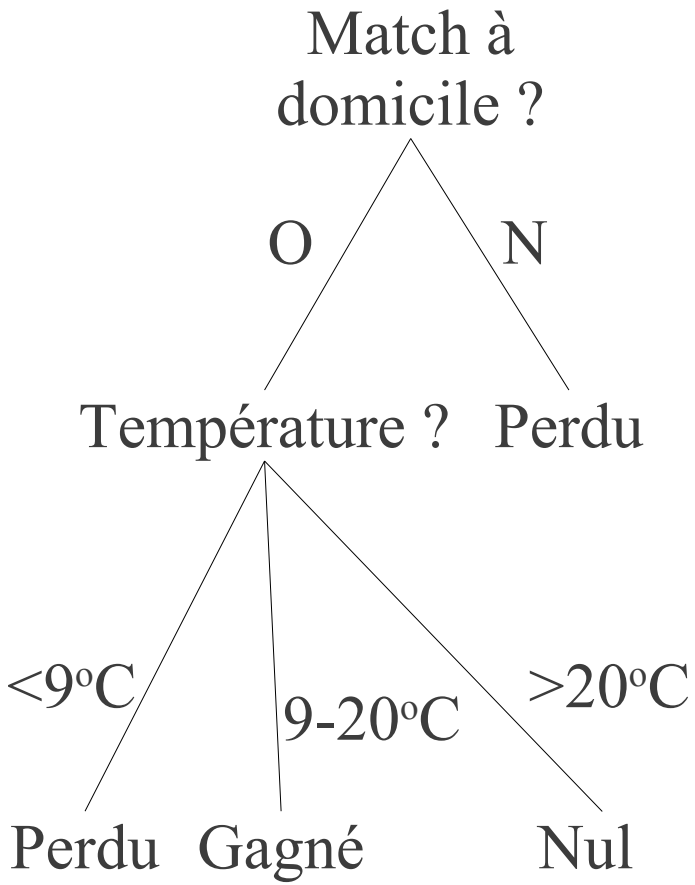
Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision

Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision

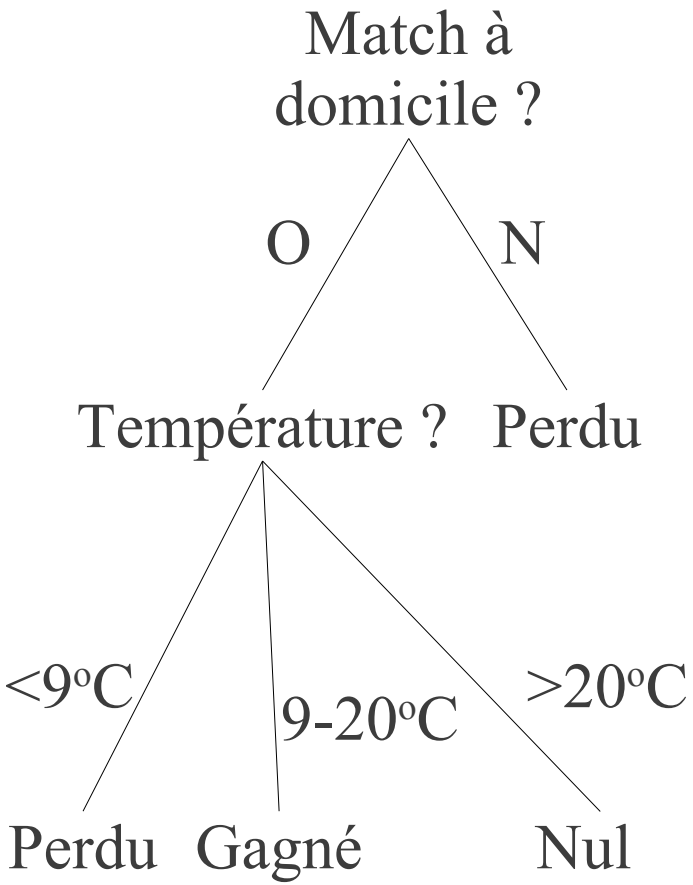
Description d'un arbre de décision



Un arbre de décision est un arbre orienté dont :

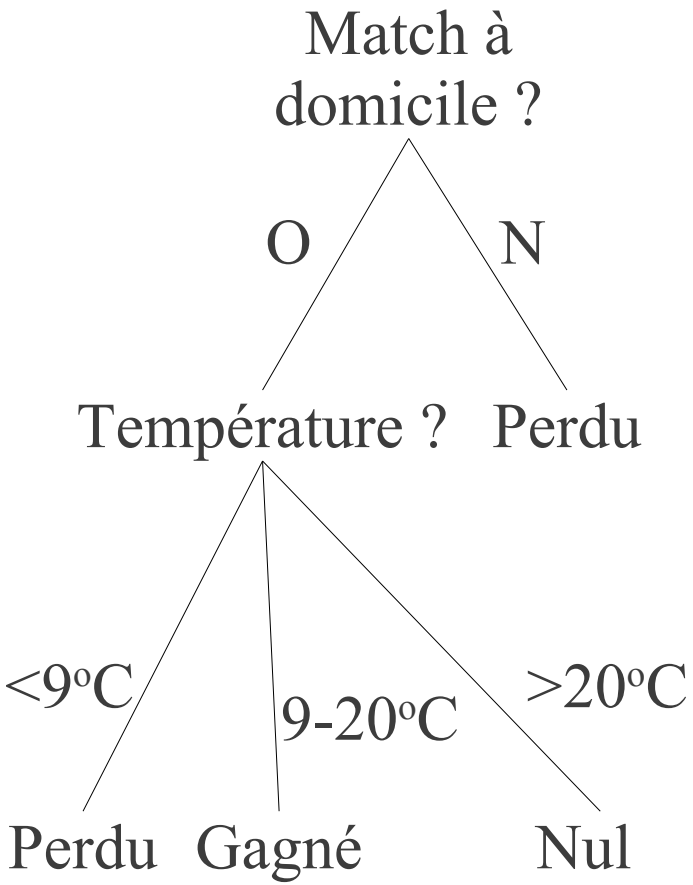
- Les **nœuds internes** sont étiquetés par un test applicable généralement sur un attribut de description.
- Les **arcs** contiennent les résultats du test.
- Chaque **feuille** est étiquetée par une classe (une classe peut apparaître plusieurs fois).
- Chaque feuille ou nœud est repérable par sa **position** : la liste (unique) des valeurs des arcs qui permettent d'y accéder.

Classification par arbre de décision



- Un arbre de décision est un classifieur organisé de manière arborescente.
- On parcourt l'arbre depuis la racine en testant successivement les attributs.
  Ex : x=(match à domicile, T<9), h(x)=Perdu
- Ce classifieur a une traduction immédiate en terme de règles de décision, mutuellement exclusives et ordonnées (si ... alors ... sinon ...).

Intérêt des arbres de décision



- Facilement interprétables
- Classification rapide :
  - Tests peu coûteux
  - Descente rapide

  Utile si grand nombre d'attributs
- Attributs continus utilisables dans les tests binaires/n-aires.
  Ex. : température
- Classification aisée de C classes

Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision

Apprentissage d'arbre : premiers pas

- **Problème :** Construire un arbre de décision à partir d'un échantillon de données
- Caractéristiques des **données** :
  - Apprentissage supervisé : nécessite un expert
  - Attributs à valeurs discrètes (vs continus)
- **Question :** quel attribut choisir en premier ? En second ? ...

Exemple : scénario

- Objectif : évaluation du risque cardiaque à partir d'une table Individu contenant les attributs :
  - Age (entier positif)
  - Fumeur (O ou N)
  - Taille (entier positif)
  - Poids (entier positif)
  - Sexe (H ou F)
- On demande à un cardiologue d'étiqueter une partie de la base (disons 5%) en 2 classes : individu à risque ou non.

9

Exemple : discrétisation des attributs

- Ces attributs doivent être discrétisés :
  - Age (entier positif)
  - Taille (entier positif)
  - Poids (entier positif)
- Proposition :
  - Age en trois catégories : jeune (<20 ans), actif (entre 21 et 50), senior (>50)
  - On applique une formule liant âge et poids et on obtient un attribut Corpulence prenant trois valeurs : faible, moyenne, forte.

10

Exemple : échantillon

Voici les données étiquetées par le cardiologue :

Sexe	Fumeur	Age	Corpulence	À risque
F	O	15 (<20)	faible	N
F	O	19 (<20)	forte	O
F	O	30 (20-50)	faible	O
F	N	45 (20-50)	forte	O
F	N	65 (>50)	moyenne	O
F	N	98 (>50)	moyenne	O
H	O	34 (20-50)	faible	O
H	N	22 (20-50)	moyenne	N
H	N	45 (20-50)	forte	N
H	N	70 (>50)	forte	O

Exercice : construire un arbre de décision

Exemple : construction d'un arbre

Constat :

- plusieurs arbres sont possibles
- dans cet exemple, ils permettent tous de classer parfaitement les données d'apprentissage : le risque empirique est nul.
- ce n'est pas toujours le cas : exemple ?

Arbre de décision de risque empirique minimal

- Il est toujours possible de trouver un arbre de décision minimisant le risque empirique sur un jeu de données. Mais cet arbre est bien souvent un mauvais classifieur. Pourquoi ?
- Le plus petit arbre de décision compatible avec les données est l'hypothèse la meilleure en généralisation. Pourquoi ?
- La théorie de l'apprentissage statistique de Vapnick permet de répondre formellement à ces questions.
- Trouver le plus petit arbre de décision compatible avec un échantillon est un problème NP-complet :-(

Stratégie

Construire *un petit arbre de décision* compatible avec le maximum de données.

Conforme à 2 principes :

- Le rasoir d'Occam (XIV siècle) :

  “Les multiples ne doivent pas être utilisés sans nécessité”

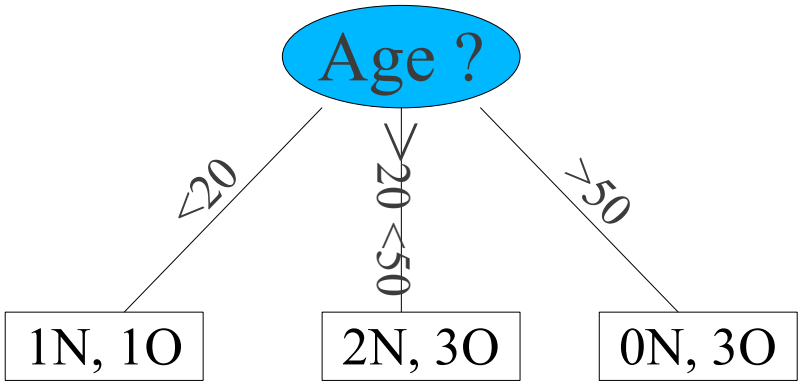
Autrement dit : entre deux représentations **équivalentes**, il faut choisir la moins complexe.

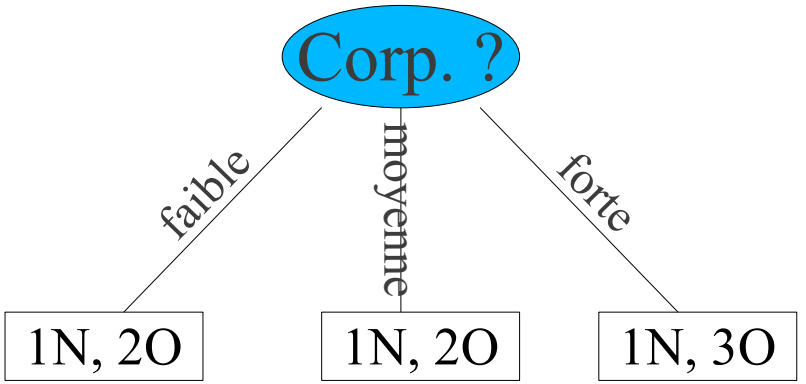
- Le principe MDL (*Minimum Description Length*) :

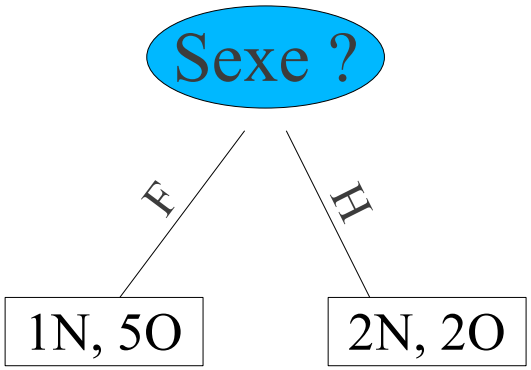
Soit S l'échantillon. Apprendre c'est trouver l'hypothèse H minimisant ||H|| + ||S|H||, c'est à dire un compromis entre la taille de l'hypothèse et celle du codage des données par cette hypothèse.

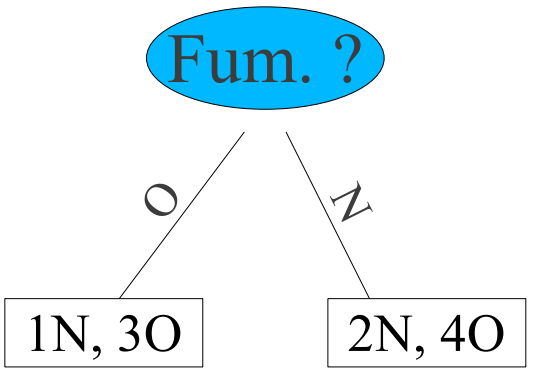
Fouille avec des arbres de décision exemple (apprentissage)

- Choix de la racine de l'arbre : le pivot qui “disperse” le mieux les 2 classes



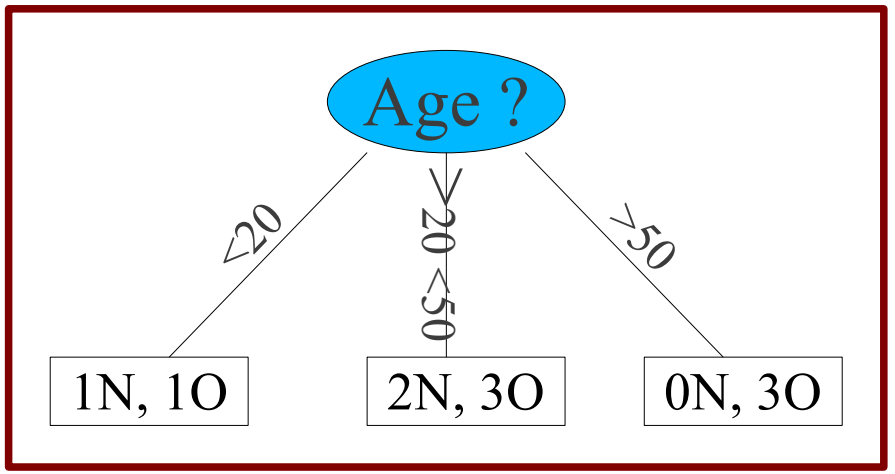


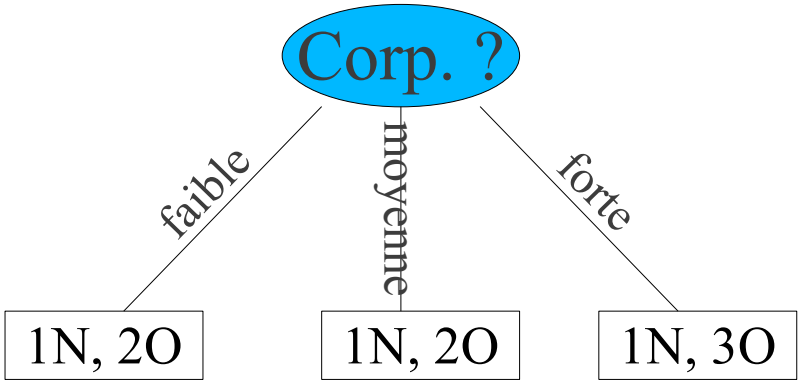


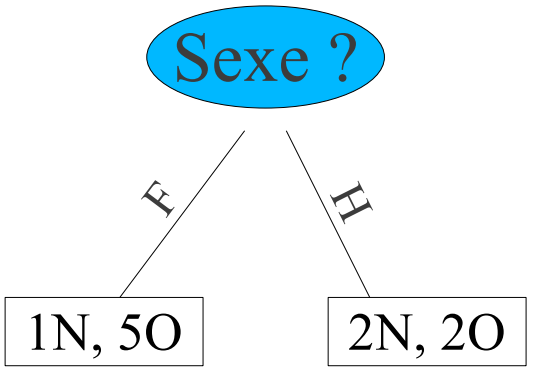


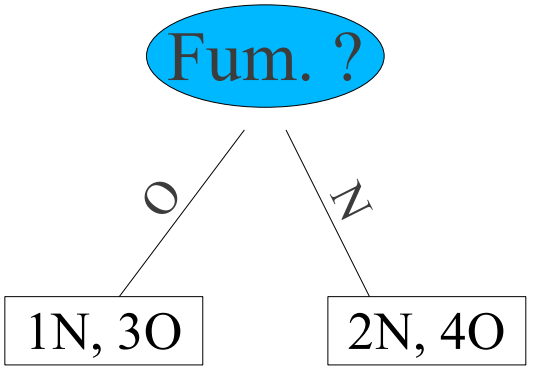
Fouille avec des arbres de décision exemple (apprentissage)

- Choix de la racine de l'arbre : le pivot qui “disperse” le mieux les 2 classes



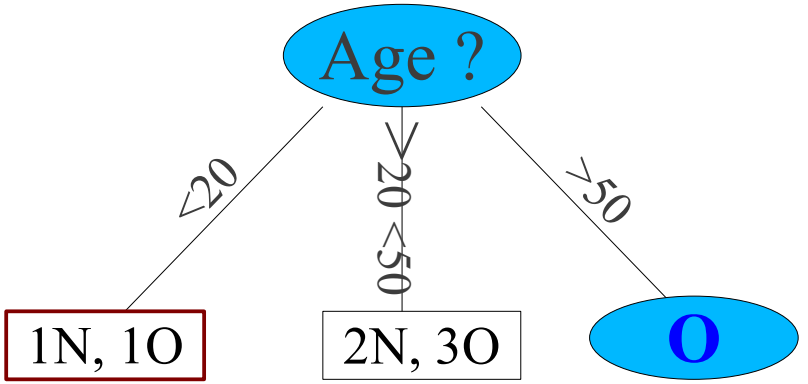




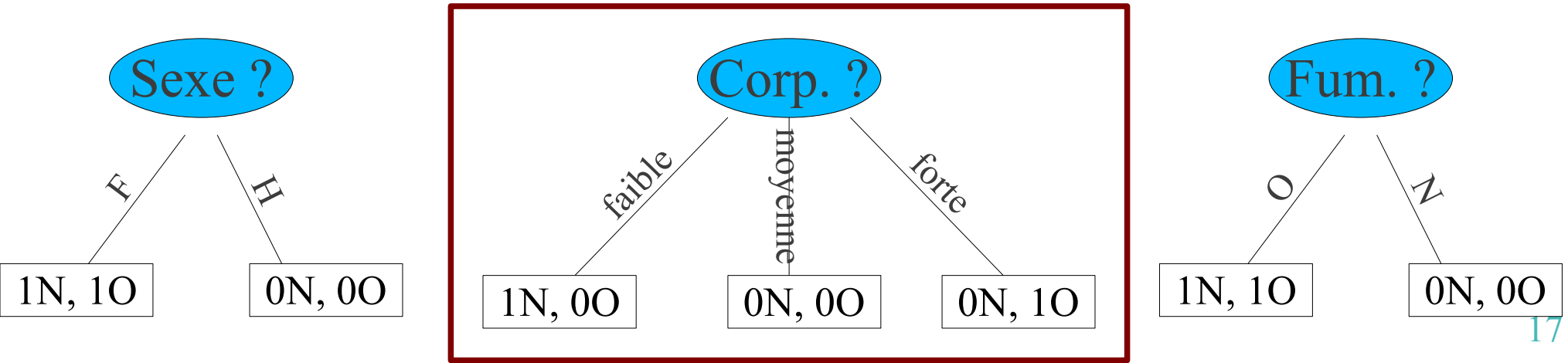


Fouille avec des arbres de décision exemple (apprentissage)

- On continue récursivement sur chacune des branches à partir de :

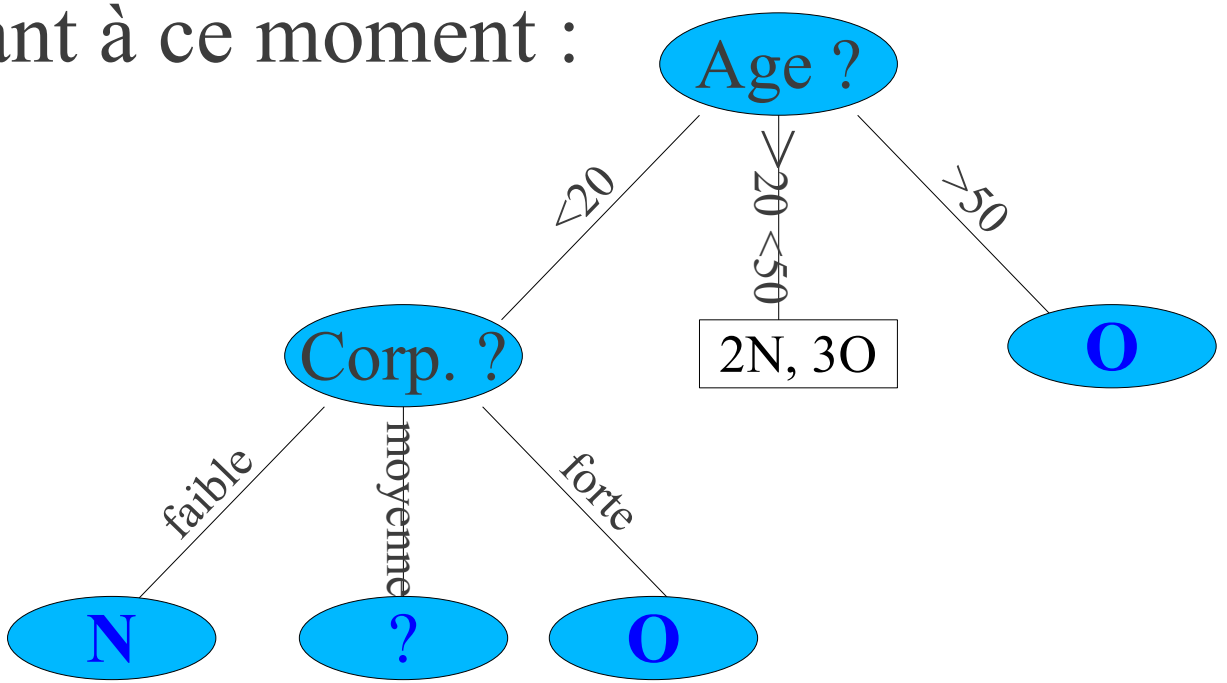


On a (première branche à gauche) :



Fouille avec des arbres de décision exemple (apprentissage)

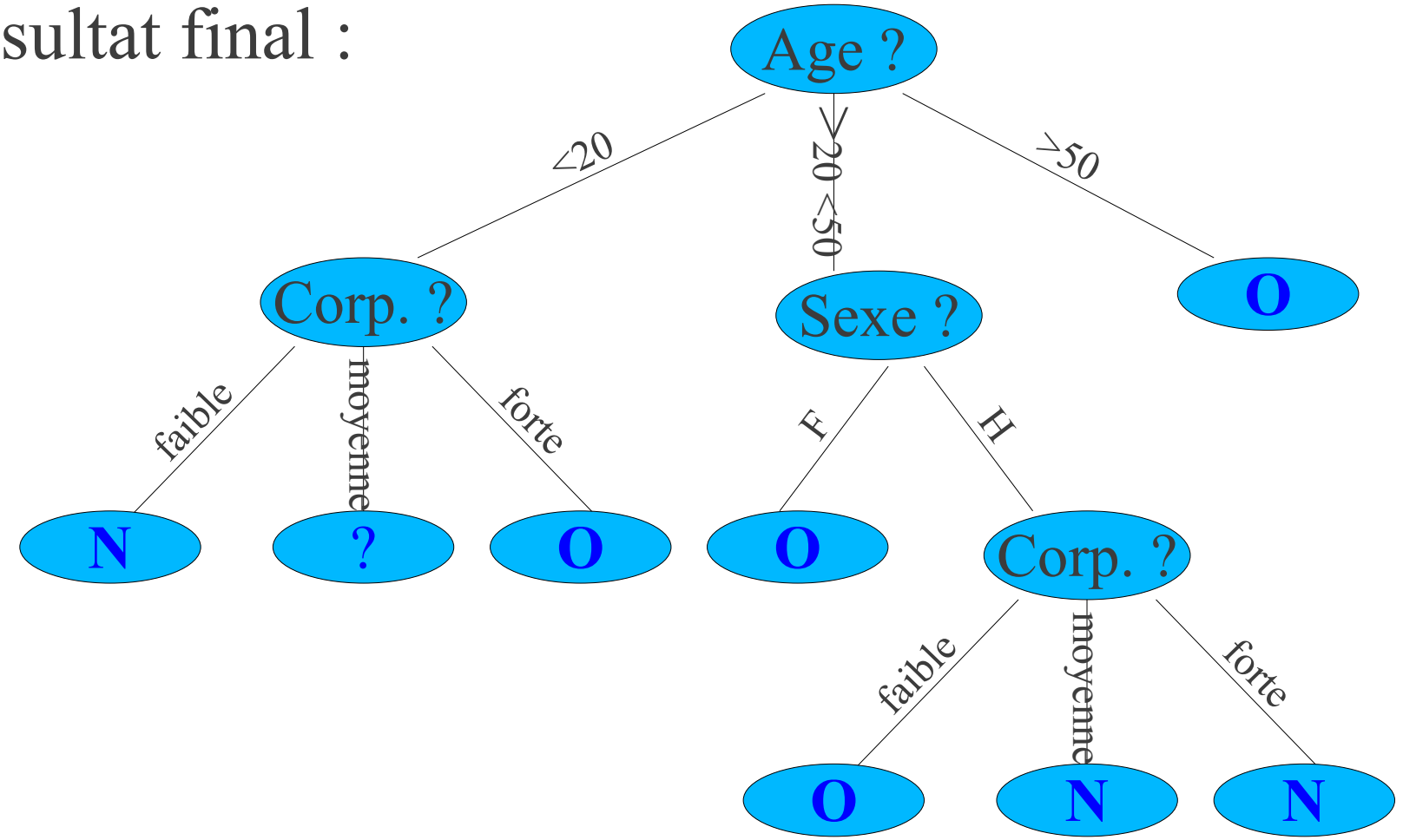
- L'arbre courant à ce moment :



- Après calcul, en testant sur l'attribut Sexe (puis corpulence) dans la branche restant à déterminer on disperse entièrement les classes.

Fouille avec des arbres de décision exemple (apprentissage)

- Résultat final :



- On peut alors classer toutes les données.

Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision

Algorithmes d'apprentissage d'arbres de décision

- Plusieurs algorithmes : CART [Breiman84], C4.5[Quinlan94].

- Algorithmes en deux étapes :
  - Construction d'un petit arbre de décision compatible
  - Elagage de l'arbre pour prévenir le surapprentissage

Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision
  - Étape 1 : construction d'un petit arbre
  - Étape 2 : prévenir le surapprentissage

Algorithmes d'apprentissage d'arbres de décision

Aperçu de la première étape :

- Méthodes de construction Top-Down, gloutonnes et récursives.
- Idée principale : diviser récursivement et le plus efficacement possible l'échantillon d'apprentissage par des tests définis à l'aide des attributs jusqu'à obtenir des sous-échantillons ne contenant (presque) que des exemples appartenant à une même classe.

Algorithmes d'apprentissage d'arbres de décision

- Algorithme générique :

Initialisation : arbre ← arbre vide ; nœud_courant ← racine
Répéter
    Décider si le nœud courant est terminal
    Si le nœud est terminal *alors* lui affecter une classe
    Sinon sélectionner un test et créer autant de nœuds fils qu'il y a de réponses au test
    Passer au nœud suivant (s'il existe)
Jusqu'à obtenir un arbre de décision consistant

- On a besoin de trois opérateurs permettant de :
    - Décider si un nœud est terminal
    - Si un nœud est terminal, lui affecter une classe
    - Si un nœud n'est pas terminal, lui associer un test

Les trois opérateurs (en général)

- Un nœud est terminal lorsque :
  - (presque) tous les exemples correspondant à ce nœud sont dans la même classe, ou
  - il n'y a plus d'attribut non utilisé dans la branche correspondante,
- On attribue à un nœud terminal la classe majoritaire (en cas de conflit, on peut choisir la classe majoritaire dans l'échantillon, ou en choisir une au hasard),
- On sélectionne le test qui fait le plus progresser la classification des données d'apprentissage. Comment mesurer cette progression ? CART utilise l'*indice de Gini* et C4.5 utilise le calcul d'*entropie*.

Indice de Gini (G) et Entropie (E)

Soit S l'échantillon et S_1, S_2, ... S_k sa partition suivant les classes de l'attribut du test.

On définit :

- l'indice de Gini

$$G(S) \triangleq \sum_i \frac{|S_i|}{|S|}\left(1 - \frac{|S_i|}{|S|}\right) = \sum_{i \neq j} \frac{|S_i|}{|S|}\frac{|S_j|}{|S|}$$

- l'entropie

$$E(S) \triangleq -\sum_i \frac{|S_i|}{|S|} \log\left(\frac{|S_i|}{|S|}\right)$$

Indice de Gini et Entropie : k=2

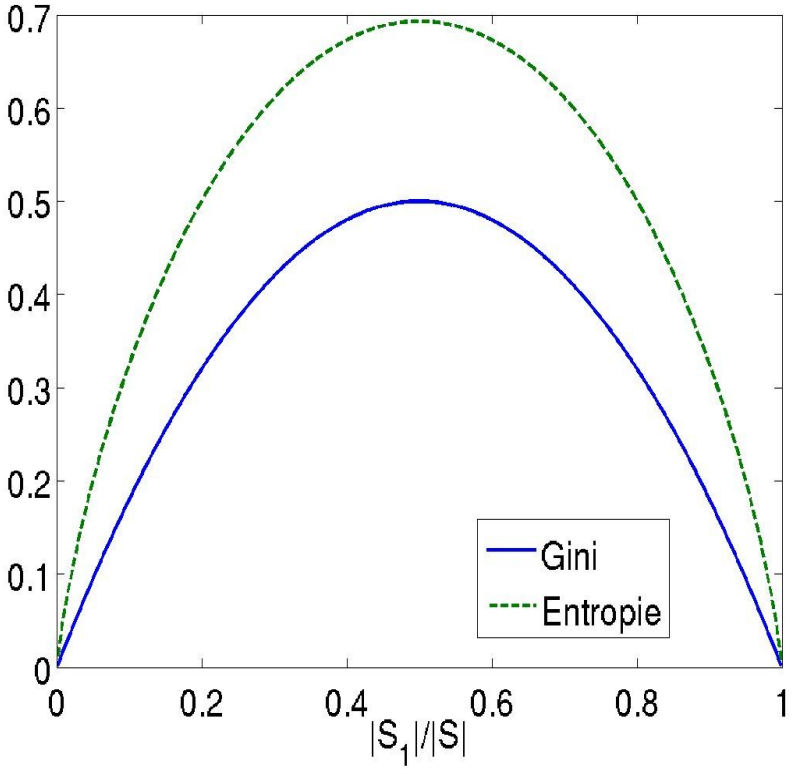
$$G(S) \triangleq \sum_i \frac{|S_i|}{|S|}\left(1 - \frac{|S_i|}{|S|}\right) = \sum_{i \neq j} \frac{|S_i|}{|S|}\frac{|S_j|}{|S|}$$

$$E(S) \triangleq -\sum_i \frac{|S_i|}{|S|} \log\left(\frac{|S_i|}{|S|}\right)$$

Cas binaire : $k = 2, x \triangleq \frac{|S_1|}{|S|}$

$$G(x) = 2x(1-x)$$

$$E(x) = -x\log(x) - (1-x)\log(1-x)$$



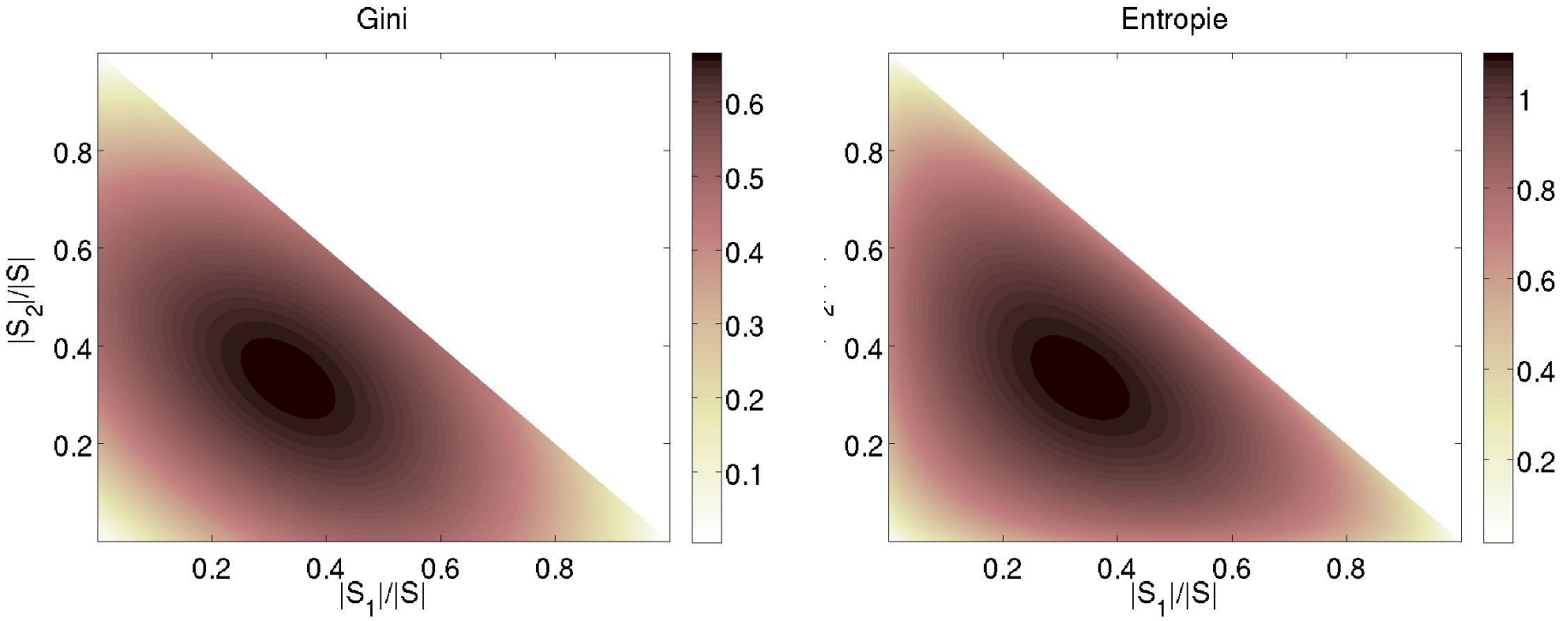
Ces fonctions :

- Ont des valeurs dans >0
- S'annulent pour x = 0 et x= 1
- Sont maximales pour x=1/k=1/2

Indice de Gini et Entropie : k=3

$$G(S) \triangleq \sum_{i=1}^{3} \frac{|S_i|}{|S|}\left(1 - \frac{|S_i|}{|S|}\right)$$

$$E(S) \triangleq -\sum_{i=1}^{3} \frac{|S_i|}{|S|} \log\left(\frac{|S_i|}{|S|}\right)$$



Gain et sélection du test

- Soit p la position courante de l'arbre en construction et T un test. On définit (avec f=E ou f=Gini) :

$$\mathrm{Gain}_f\,(p,T) \triangleq f\,(S_p) - \sum_j P_j \times f\,(S_{p_j})$$

où S_p est l'échantillon associé à p et P_j est la proportion des éléments de S_p qui satisfont la j-ème branche de T.

- Maximiser le gain revient à minimiser $\sum_j P_j \times f\,(S_{p_j})$
  - Gain maximal : l'attribut permet de classer correctement toutes les données
  - Gain nul : données sont aussi mal classées après le test qu'avant
  - Sélectionner l'attribut dont le gain est maximum correspond à une stratégie gloutonne : rechercher le test faisant le plus progresser la classification localement.

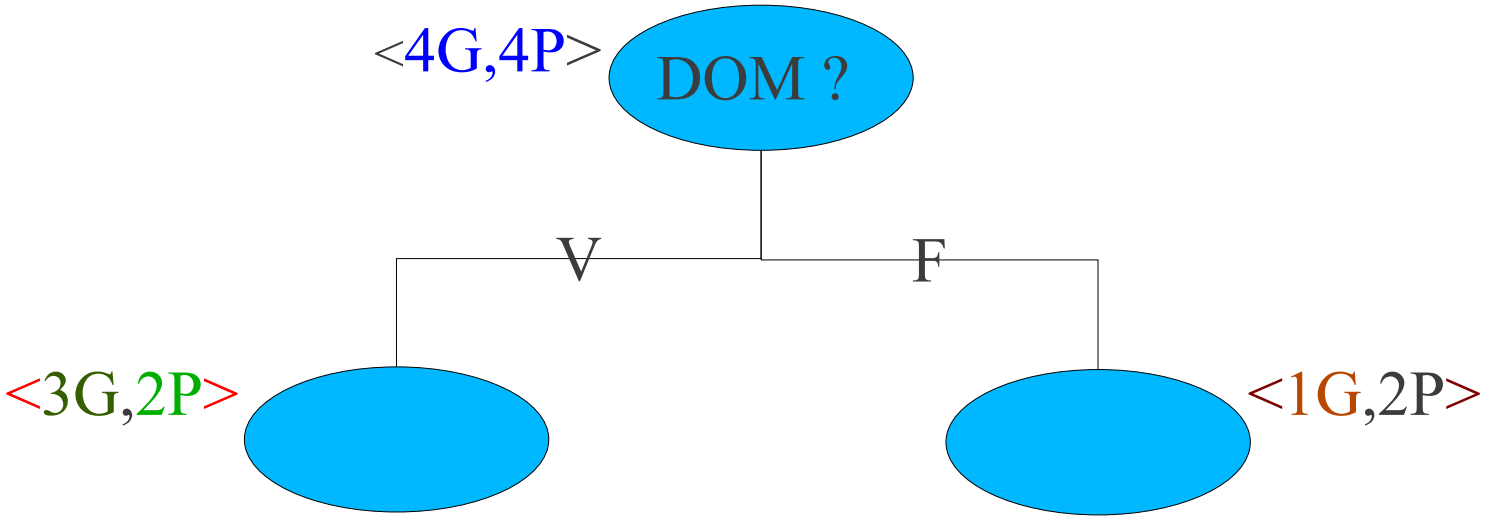
Exemple d'utilisation de l'algo de CART

Match à domicile ?	Balance positive ?	Mauvaises conditions climatiques ?	Match précédent gagné ?	Résultat
V	V	F	F	G
F	F	V	V	G
V	V	V	F	G
V	V	F	V	G
F	V	V	V	P
F	F	V	F	P
V	F	F	V	P
V	F	V	F	P

cf. poly p. 15

Sur l'exemple des matchs

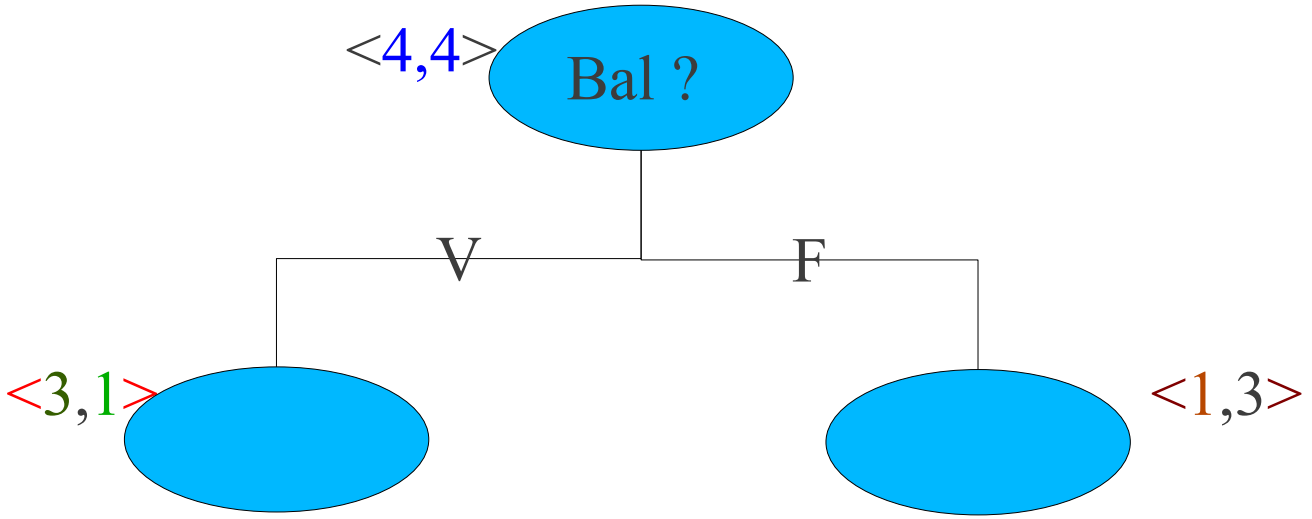
- Avec le critère de Gini et en désignant les attributs descriptifs *Dom, Bal, MCC* et *MPG* nous avons :



$$Gain(\varepsilon,Dom) = G(S)-(5/8G(S_1) + 3/8\ G(S_2))$$
$$= G(S) - 2*5/8*3/5*2/5 - 2*3/8*1/3*2/3$$
$$= G(S) - 7/15$$

Sur l'exemple des matchs

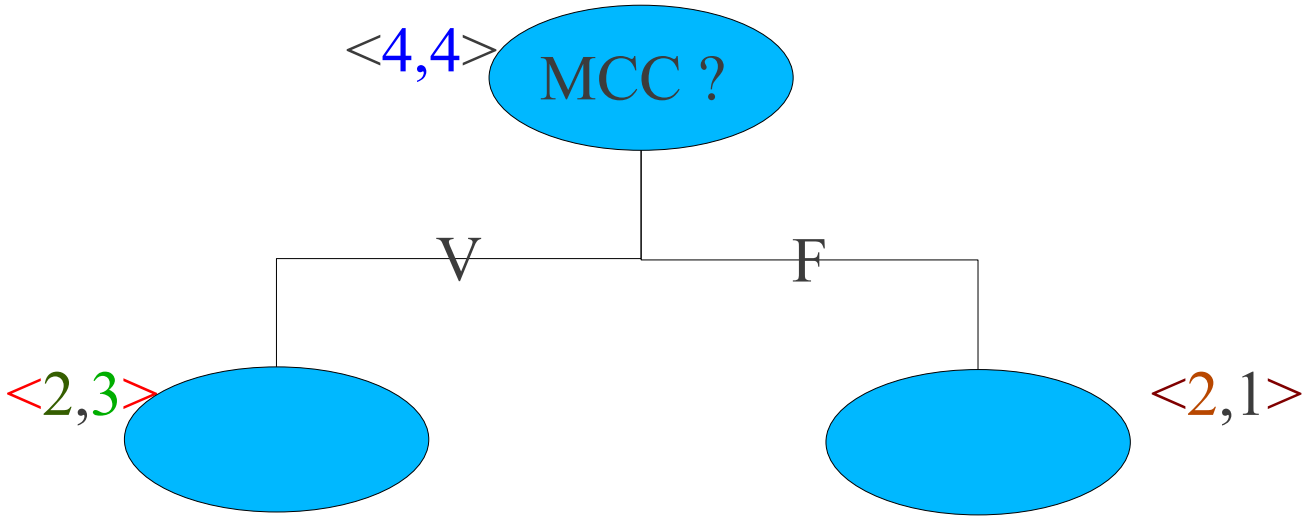
- Avec le critère de Gini et en désignant les attributs descriptifs *Dom, Bal, MCC* et *MPG* nous avons :



$$Gain(\varepsilon,Dom) = G(S)-(4/8G(S_1) + 4/8\ G(S_2))$$
$$= G(S) - 2*4/8*3/4*1/4 - 2*4/8*1/4*3/4$$
$$= G(S) - 3/8$$

Sur l'exemple des matchs

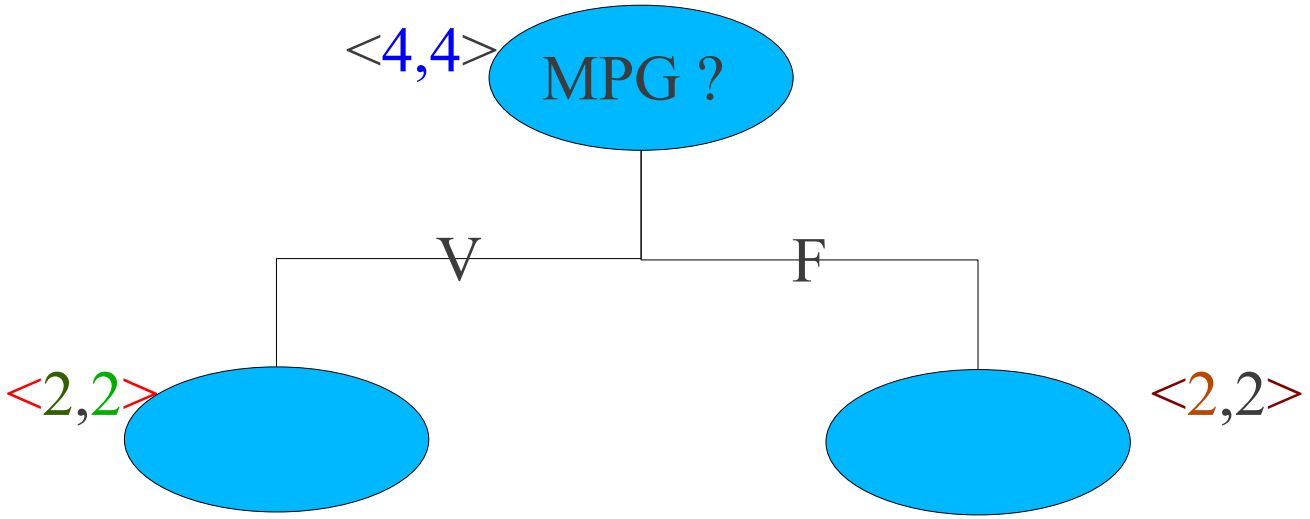
- Avec le critère de Gini et en désignant les attributs descriptifs *Dom, Bal, MCC* et *MPG* nous avons :



$$Gain(\varepsilon,Dom) = G(S)-(5/8G(S_1) + 3/8\ G(S_2))$$
$$= G(S) - 2*5/8*2/5*3/5 - 2*3/8*2/3*1/3$$
$$= G(S) - 7/15$$

Sur l'exemple des matchs

- Avec le critère de Gini et en désignant les attributs descriptifs *Dom, Bal, MCC* et *MPG* nous avons :



$$\text{Gain}(\varepsilon,\text{Dom}) = G(S) - (4/8\,G(S_1) + 4/8\,G(S_2))$$
$$= G(S) - 2*4/8*2/4*2/4 - 2*4/8*2/4*2/4$$
$$= G(S) - 1/2$$

Sur l'exemple des matchs

- Avec le critère de Gini et en désignant les attributs descriptifs *Dom, Bal, MCC* et *MPG* nous avons :

Gain(ε,Dom) = G(S) – 7/15 = G(S) – 0.466...

Gain(ε,Bal) = G(S) – 3/8 = G(S) – 0.375

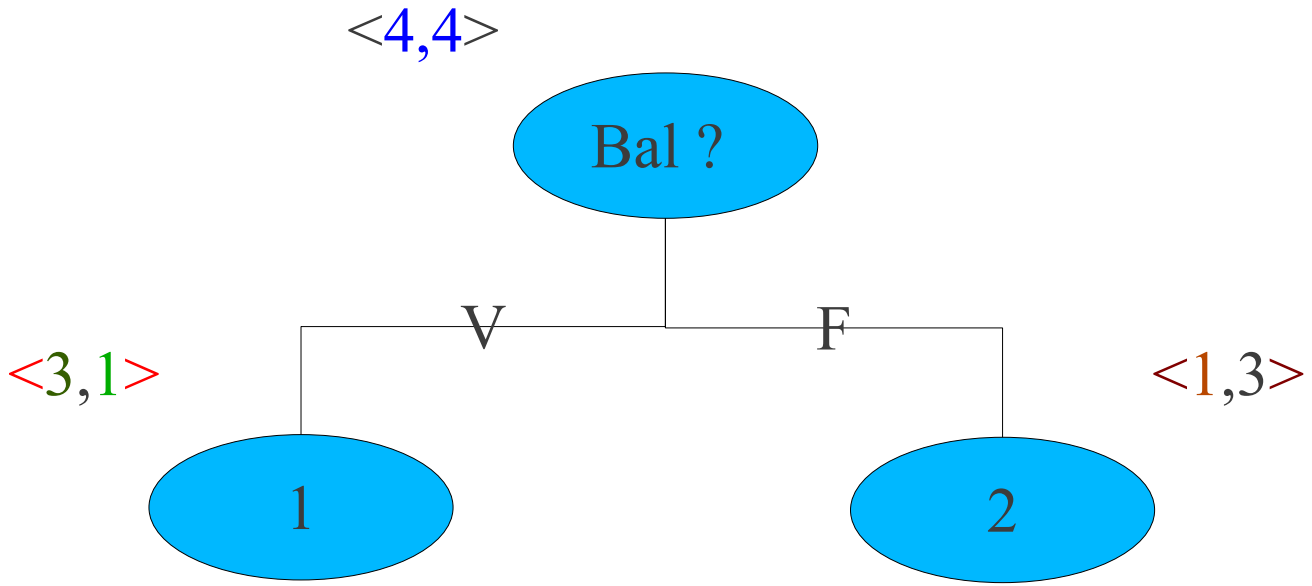
Gain(ε,MCC) = G(S) - 7/15 = G(S) – 0.466...

Gain(ε,MPG) = G(S) - 1/2

- Le gain maximal est obtenu pour le test sur l'attribut Balance positive

Sur l'exemple des matchs

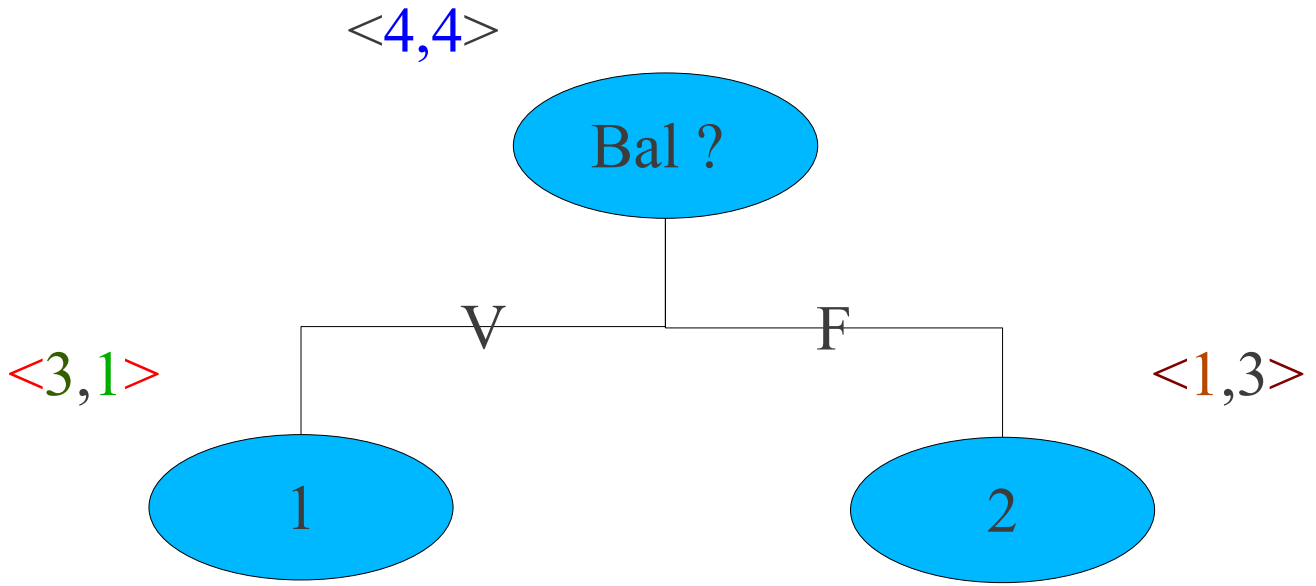
- L'arbre courant est alors :



- Il faut alors recommencer récursivement (et indépendamment) le calcul du gain en position 1 et en position 2 pour choisir les tests à ces niveaux.

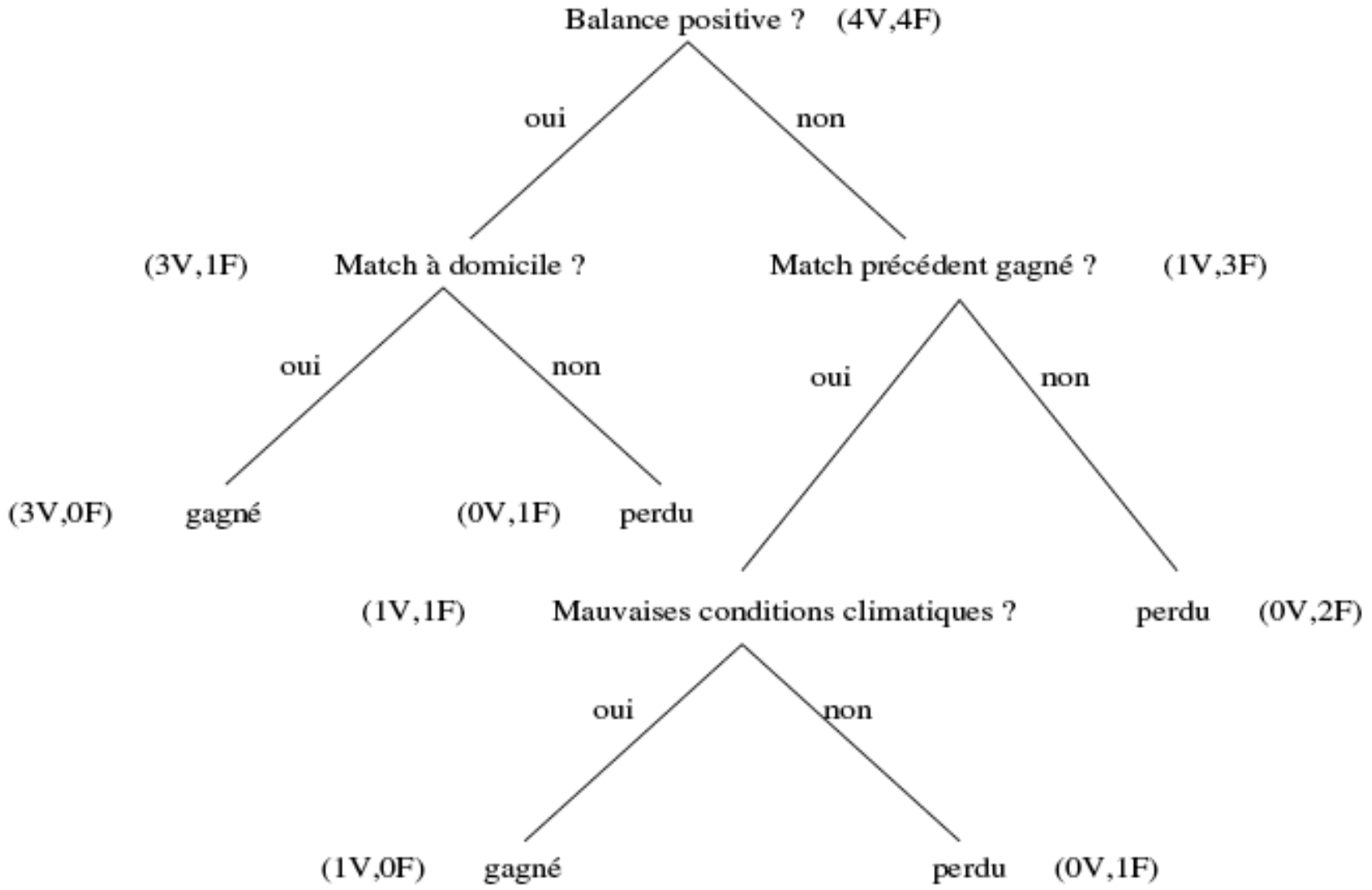
Sur l'exemple des matchs

- L'arbre courant est :



- Exercice : continuer l'algorithme.

Sur l'exemple des matchs (résultat)



TD

Exercices 2 et 3 p. 23-25 du poly

Plan

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision
  - Étape 1 : construction d'un petit arbre
  - Étape 2 : prévenir le surapprentissage

Plan : prochain cours

- Les arbres de décision
  - description, classification, intérêts
- L'apprentissage d'arbres de décision
  - principes et enjeux
- Algorithmes d'apprentissage d'arbres de décision
  - Étape 1 : construction d'un petit arbre
  - Étape 2 : prévenir le surapprentissage