



Arbres de décision

Algorithme CART



Plan

- Introduction
- Construction d'un arbre
- Algorithme CART
- Avantages
- Limites
- Fonctions S-Plus, R
- Pronostic des patients d'oncohématologie en réanimation



Introduction

- Jeu de données
 - N individus
 - P variables décrivant ces individus
- Variable cible (ou à prédire)
 - Variable classe/groupe (Qualitative)
 - Ex : malade / non malade
- Variables explicatives
 - Autres variables (Qualitatives et Quantitatives)
 - Ex : Température, pupille dilatée...



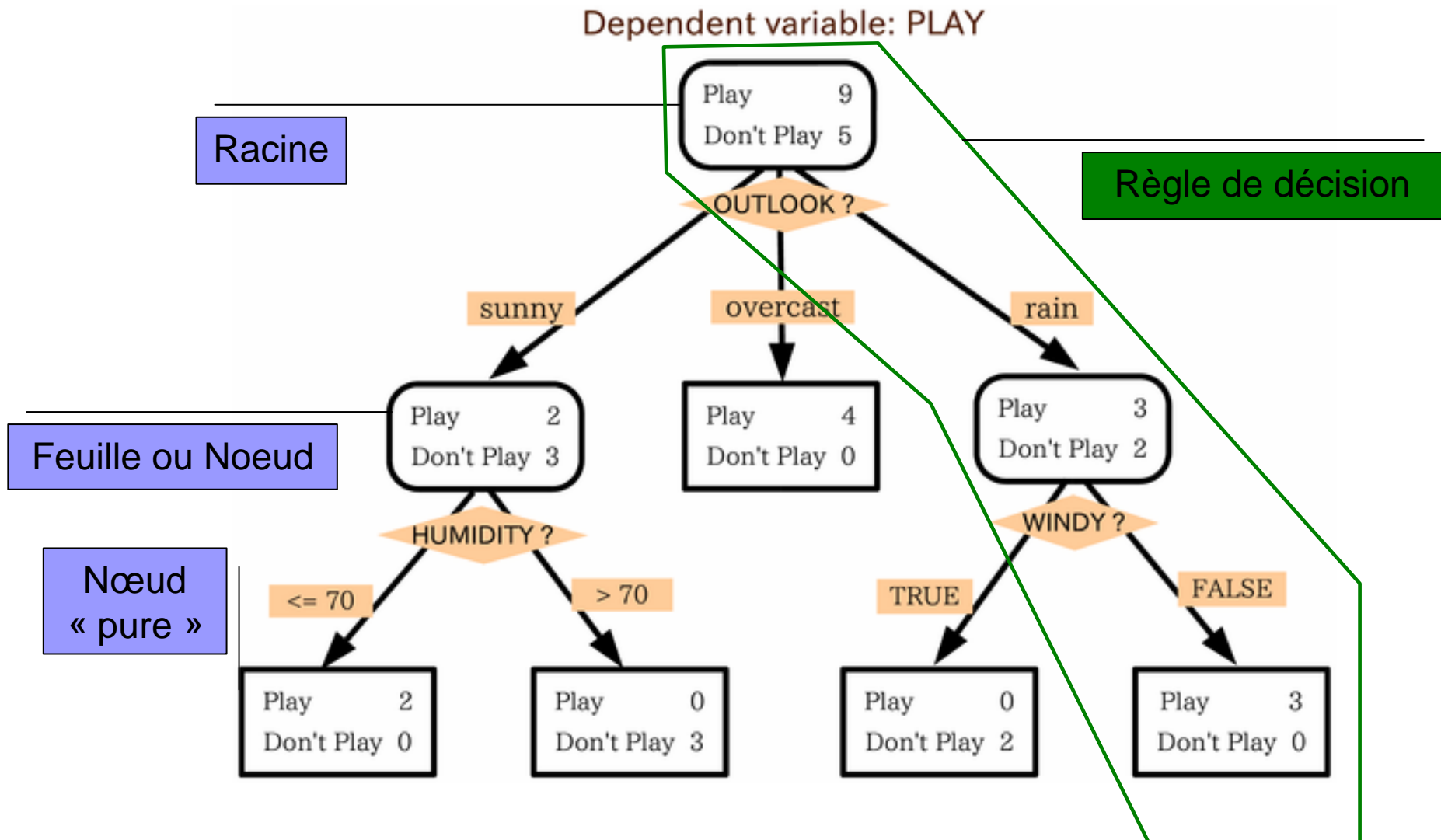
Introduction

- 2 familles de modèles de classification
 - Classification non supervisée
 - Établir des représentations des données dans des espaces à faible dimensions pour y lire des typologies d'individus (ex : ACP)
 - → Nombre de classes initialement inconnu
 - Classification supervisée
 - Obtenir un critère de séparation destiné à prédire l'appartenance à une classe
 - → Nombre de classes initialement connu
 - Méthodes : Analyse discriminante, régression logistique, arbre de décision, réseau de neurones

Introduction

- Arbre de décision
 - Utiliser les variables explicatives pour subdiviser les individus en groupes hétérogènes (classes)
 - Obtention de règles de décisions intelligibles :
 - Si Neige=Oui ET Visibilité=Mauvaise ALORS Ski=Non
 - Si Température<-10 ALORS Ski=Non
 - Représentation graphique hiérarchisé intuitive

Introduction



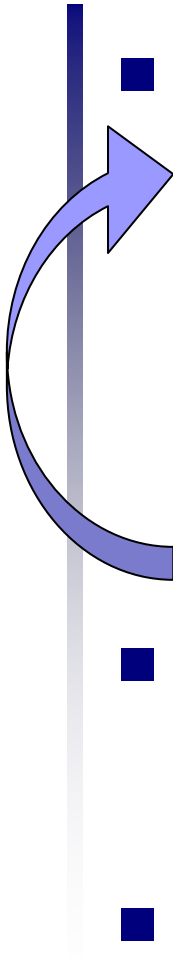


Construction de l'arbre

- Comment trouver les variables qui séparent le mieux les individus de chaque classe ?
 - Plusieurs critères de choix de variables correspondant à différents types d'arbres
 - CART (Classification And Regression Tree : Indice de Gini)
 - CHAID (Chi square Automatic Interaction Detection)
 - C5.0 (Entropie de Shannon)



Construction de l'arbre

- 
- Déroulement de la construction :
 - Recherche de la variable et du seuil qui sépare le mieux
 - Applique la séparation à la population
 - Obtention de nouveaux nœuds
 - Arrêt de l'approfondissement de l'arbre lorsque les conditions d'arrêts sont rencontrées
 - Éventuel « élagage » de l'arbre



Construction de l'arbre

- Conditions d'arrêts existantes :
 - Profondeur de l'arbre atteint une limite fixée (=nombre de variables utilisées)
 - Nombre de feuilles atteint un maximum fixé
 - L'effectif de chaque nœud est inférieur à un seuil fixé
 - La qualité de l'arbre est suffisante
 - La qualité de l'arbre n'augmente plus de façon sensible



Algorithme CART

- Classification And Regression Tree
 - 1984, L. Breiman, J.H Friedman, R.A. Olshen et C.J. Stone
 - Parmi les plus performants et plus répandus
 - On le trouve dans : SAS, R, S-Plus, SPAD...
 - Binaire : Deux nœuds fils pour chaque nœud parent
 - Accepte tout type de variables
 - Critère de séparation : Indice de Gini

Algorithme CART

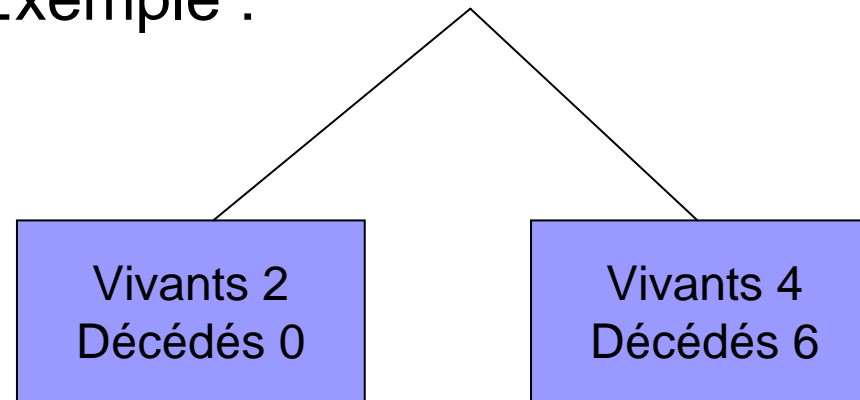
- Indice de Gini

$$I = 1 - \sum_i^n f_i^2$$

- N = nombre de classes à prédire
 - Fi = fréquence de la classe i dans le nœud
- Plus l'indice de Gini est bas, plus le nœud est pure

Algorithme CART

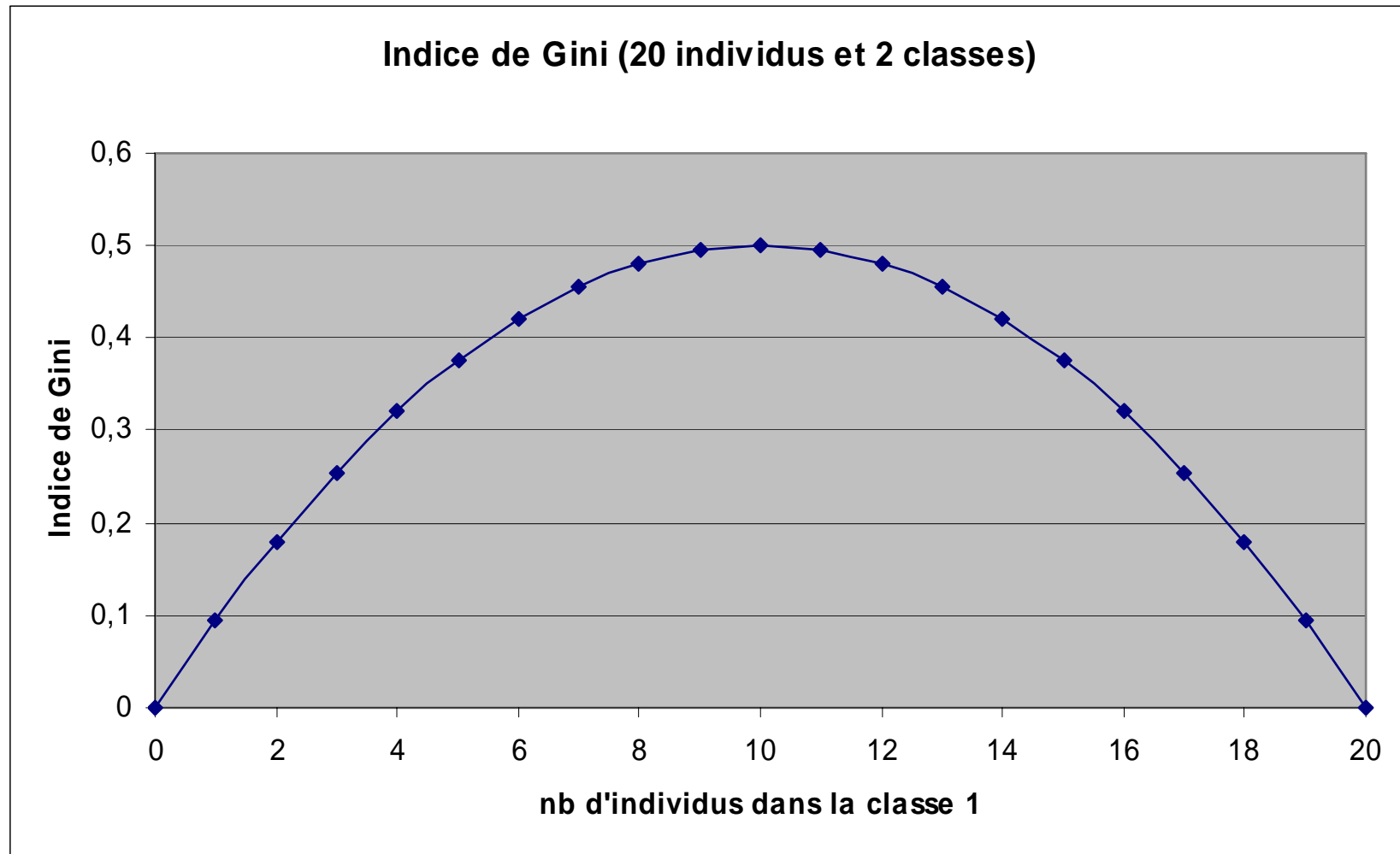
- Exemple :



$$I = 1 - \left[\left(\frac{2}{2} \right)^2 + \left(\frac{0}{2} \right)^2 \right] = 0$$

$$I = 1 - \left[\left(\frac{4}{10} \right)^2 + \left(\frac{6}{10} \right)^2 \right] = 1 - 0,52 = 0,48$$

Algorithme CART





Algorithme CART

- Ainsi,
 - En séparant 1 nœud en 2 nœuds fils on cherche la plus grande hausse de la pureté
 - La variable la plus discriminante doit maximiser :

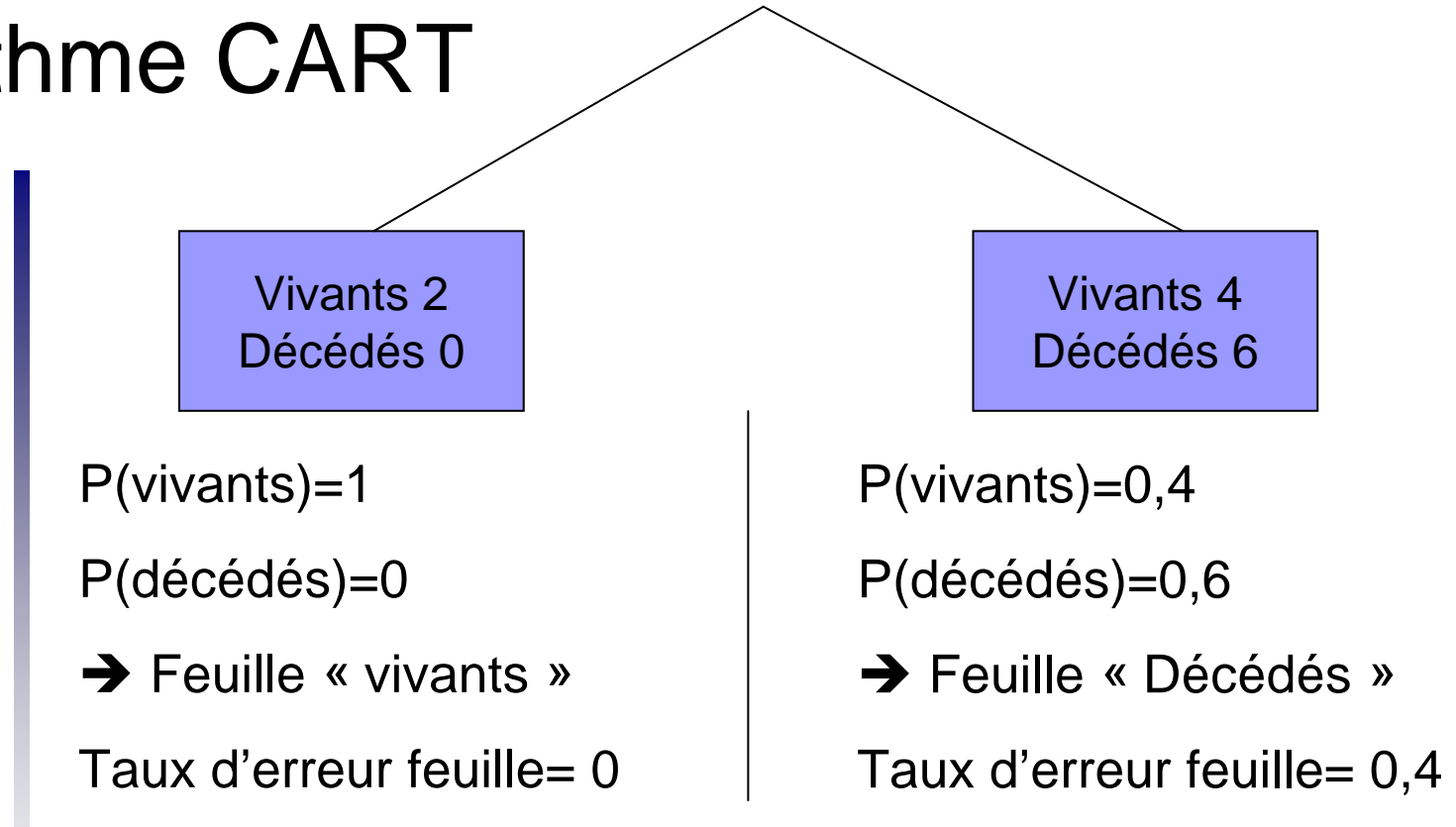
$$IG(\text{avant sep.}) - [IG(\text{fils1}) + IG(\text{fils2})]$$



Algorithme CART

- Répartition des individus dans les nœuds
 - Quand l'arbre est construit : critères de division connus
 - On affecte chaque individu selon les règles obtenues → remplissage des feuilles
 - Pour chaque feuille : plusieurs classes C
 - P_c = Proportion d'individus de la feuille appartenant à la classe c
 - On affecte à la feuille la classe pour laquelle P_c est la plus grande

Algorithme CART



Taux d'erreur global de l'arbre = somme pondérée des taux d'erreur des feuilles

Pondération = proba qu'un individu soit dans la feuille (= taille de la feuille)



Algorithme CART

- Problèmes des arbres trop étoffés
 - Complexité de l'arbre, trop de règles
 - Trop spécifique aux données d'apprentissage
 - Règles non reproductibles (« surapprentissage »)
 - Trop peu d'individus dans les feuilles (aucune signification réelle)
 - minimum conseillé : 20-30 individus

- Solution → Élagage



Algorithme CART

- Processus d'élagage de CART
 - Création de l'arbre maximum
 - Toutes les feuilles des extrémités sont pures
 - Élagages successifs de l'arbre
 - Retient l'arbre élagué pour lequel le taux d'erreur estimé mesuré sur un échantillon test est le plus bas possible



Avantages

- Résultats explicites
 - Arbre
 - Règles de décisions simples
 - Modèle facilement programmable pour affecter de nouveaux individus
- Peu de perturbation des individus extrêmes
 - Isolés dans des petites feuilles
- Peu sensible au bruit des variables non discriminantes
 - Non introduites dans le modèle



Avantages

- CART permet l'utilisation de variables de tous types
 - Continues, discrètes, catégoriques
- Traitement d'un grand nombre de variables explicatives
- Peu d'hypothèses préalables



Inconvénients

- **Arbre non optimaux**
 - Utilisation de règles heuristiques
 - Utilisation des variables non simultanée mais séquentielle
 - « Effet papillon » → On change une variable dans l'arbre, tout l'arbre change
- **Nécessité d'un grand nombre d'individus**
 - Pour avoir 20-30 individus minimum par nœud pour que les règles aient une valeur



Inconvénients

- Temps de calculs importants
 - Recherche des critères de division
 - Élagage



Fonctions S-PLUS, R

- **Fonction `Rpart`, `Cart`, `Tree`**
 - Fonctions équivalentes : la plus complète est `Rpart`
 - `Rpart()` = ajuste un modèle
 - `Rpart.control()` = Paramètres pour l'ajustement
 - `Prune.Rpart()` = Élague l'arbre
 - `Plot.rpart()` = trace l'arbre
 - `Predict.rpart()` = Calcul des prédictions
 - ...



Fonctions S-PLUS, R

- **SYNTAXE RPART ()**
- **Formula** : formule de modélisation
 - `fit <- rpart(Kyphosis ~ Age + Number + Start)`
- **Data (Option)** : tableau avec libellé des modalités des variables
- **subset (Option)** : sous ensemble d'individu
- **Na.action** : Comment on gère les valeurs manquantes
 - Par défaut : Supprime les individus quand la variable Classe est manquante



Fonctions S-PLUS, R

- **Method** : Méthode de séparation utilisée
 - Dépend du type de la variable Classe
- **Parms** : Paramétrage de **method**

Fonctions S-PLUS, R

■ Exemple

<http://www.grappa.univ-lille3.fr/~ppreux/ensg/miashs/tp-R/ad.html>

Variable Cible : JOUER (OUI/NON)

```
> tennis<-read.table("http://www.grappa.univ-lille3.fr/~ppreux/ensg/miashs/tp-R/tennum.txt")
> tennis
```

	Ciel	Température	Humidité	Vent	Jouer
1	Ensoleillé	27.5	85	Faible	Non
2	Ensoleillé	25.0	90	Fort	Non
3	Couvert	26.5	86	Faible	Oui
4	Pluie	20.0	96	Faible	Oui
5	Pluie	19.0	80	Faible	Oui
6	Pluie	17.5	70	Fort	Non
7	Couvert	17.0	65	Fort	Oui
8	Ensoleillé	21.0	95	Faible	Non
9	Ensoleillé	19.5	70	Faible	Oui
10	Pluie	22.5	80	Faible	Oui
11	Ensoleillé	22.5	70	Fort	Oui
12	Couvert	21.0	90	Fort	Oui
13	Couvert	25.5	75	Faible	Oui
14	Pluie	20.5	91	Fort	Non

Fonctions S-PLUS, R

```
> cnt<-rpart.control(minsplit=1)
```

```
> arbre<-rpart(Jouer ~ Ciel + Température + Humidité + Vent, tennis, control=cnt)
```

On peut alors afficher le résultat de la construction sous forme de texte :

```
> arbre
n= 14

node), split, n, loss, yval, (yprob)
  * denotes terminal node

1) root 14 5 Oui (0.3571429 0.6428571)
 2) Ciel=Ensoleillé,Pluie 10 5 Non (0.5000000 0.5000000)
 4) Humidité>=82.5 5 1 Non (0.8000000 0.2000000)
 8) Température>=20.25 4 0 Non (1.0000000 0.0000000) *
 9) Température< 20.25 1 0 Oui (0.0000000 1.0000000) *
 5) Humidité< 82.5 5 1 Oui (0.2000000 0.8000000)
 10) Température< 18.25 1 0 Non (1.0000000 0.0000000) *
 11) Température>=18.25 4 0 Oui (0.0000000 1.0000000) *
 3) Ciel=Couvert 4 0 Oui (0.0000000 1.0000000) *
```

On peut également obtenir une représentation graphique :

```
> plot(arbre)
> text(arbre)
```



Pronostic des patients d'oncohématologie en réanimation

- Base OUTCOMEREA
 - Variables sur les 4 premiers jours en réanimation
- ➔ Prédire la survenu du Décès en réanimation
- Variable cible : Décès (OUI/NON)
 - Variables explicatives suspectée comme intéressantes



Pronostic des patients d'oncohématologie en réanimation

- **VARIABLES INITIALES (par patient)**
- Patients de Chirurgie programmée (protecteur)
- Score de Mac Cabe
- Aplasie
- Prise de corticoïdes
- Patient transférés d'un service
- DNR Order dans les 4 premiers jours
- Insuffisance rénale aigüe (Protecteur)
- Défaillance multi viscérale et chocs
- ...

Pronostic des patients d'oncohématologie en réanimation

- **VARIABLES JOURNALIERES (par jour et par patient)**

- Ventilation mécanique
- Intubation
- Inotropes
- Score du SOFA

} Défaillance d'organes