Arbres de décision

Principe des arbres de décisions (AD)

- On se donne un ensemble $\mathcal{D}$ de N exemples d'apprentissage avec d **attributs qualitatifs** ou **quantitatifs**.

- Chaque donnée $\mathbf{x}$ est étiquetée par une classe (variable cible) y appartenant à l'espace $\mathcal{Y}$. À partir de ces exemples, on construit un arbre dit de décision, tel que:

  - **Un nœud** correspond à un test sur la valeur d'un ou plusieurs attributs;
  - **Une branche** partant d'un nœud correspond à une ou plusieurs valeurs de ce test;
  - **Une feuille** est associée une valeur de l'attribut cible.

Principe des arbres de décisions (AD)

- **L'arbre de décision** peut être ensuite exploité de différentes manières :

    1. En y classant de nouvelles données;

    2. En faisant de l'estimation d'attributs;

    3. En extrayant un jeu de règles de classification;

    4. En interprétant la pertinence des attributs.

Construction d'un AD

Exemple:

- Soit à un ensemble de jours (un jour = un exemple). Chaque jour est caractérisé par un numéro et ses conditions météorologiques (***ciel, température***, ***humidité de l'air***, ***force du vent***).

- L'attribut cible étant « ***jouer au tennis ? »***, dont les valeurs possibles sont $\mathcal{Y} = \{\textbf{oui}, \textbf{non}\}$.

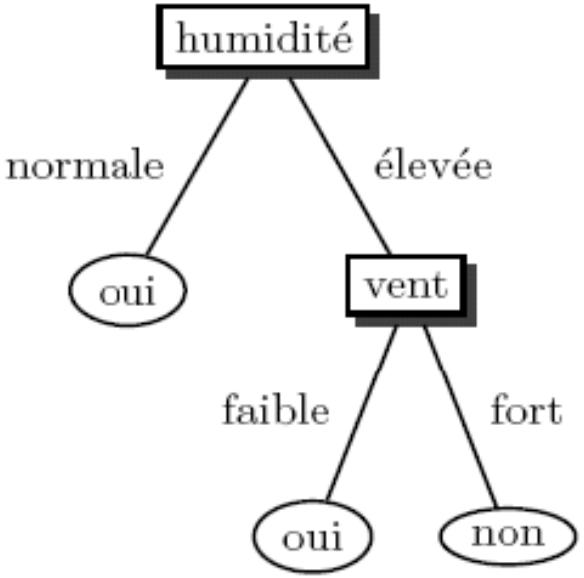
- Une fois l'arbre de décision construit, on pourra classer une nouvelle donnée pour savoir « ***si on joue ou non ce jour-là*** ».

Construction d'un AD

Jour	Ciel	Température	Humidité	Vent	Jouer au tennis ?
1	Ensoleillé	Chaude	Élevée	Faible	Non
2	Ensoleillé	Chaude	Élevée	Fort	Non
3	Couvert	Chaude	Élevée	Faible	Oui
4	Pluie	Tiède	Élevée	Faible	Oui
5	Pluie	Fraîche	Normale	Faible	Oui
6	Pluie	Fraîche	Normale	Fort	Non
7	Couvert	Fraîche	Normale	Fort	Oui
8	Ensoleillé	Tiède	Élevée	Faible	Non
9	Ensoleillé	Fraîche	Normale	Faible	Oui
10	Pluie	Tiède	Normale	Faible	Oui
11	Ensoleillé	Tiède	Normale	Fort	Oui
12	Couvert	Tiède	Élevée	Fort	Oui
13	Couvert	Chaud	Normale	Faible	Oui
14	Pluie	Tiède	Élevée	Fort	Non

Construction d'un AD

- Soit un exemple d'AD:



Jour	Ciel	Température	Humidité	Vent	Jouer au tennis ?
1	Ensoleillé	Chaude	Élevée	Faible	Non

- L'exemple de ligne 1 du tableau sera classé comme «oui» et les exemples 2 et 5 seront classés « non » et « oui ».

Construction d'un AD

- Il est à noter que la construction d'un **AD optimal** (**au sens où il minimise le nombre d'erreurs de classification**) est un problème NP-complet.

- Il ne faut pas donc avoir l'espoir de construire l'arbre de **décision optimal** pour un jeu d'exemples donné. On va se contenter d'en construire un qui soit **correct**.

- Plusieurs **algorithmes** ont été proposés pour construire des AD, dont **CART**, **ID3** et **C4.5**. **ID3** ne prend en compte que des attributs **nominaux**. **C4.5** prend également en charge des attributs quantitatifs.

Construction d'un AD

- Dans notre exemple, on suppose que tous les attributs sont **nominaux.** ID3 et C4.5 fonctionnent **récursivement** en:

  ➢ Déterminant un **attribut** à placer en **racine** de l'arbre. **La racine** possède autant de **branches** que cet **attribut prend de valeurs**.

  ➢ À **chaque branche** est associé un ensemble d'exemples dont l'attribut prend la valeur qui étiquette cette branche.

  ➢ On accroche alors au bout de cette branche l'arbre de décision construit sur ce **sous-ensemble** d'exemples, en considérant **tous les attributs** excepté celui qui vient d'être mis à la racine.

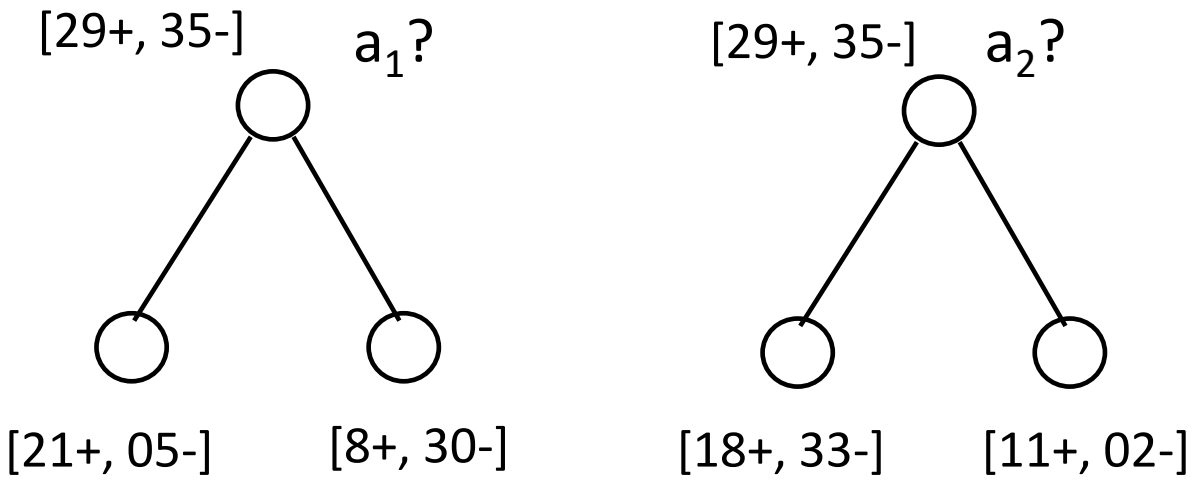
Arbres de décision: Algorithmes ID3, C4.5

Nœud = racine;

1- a ← meilleur attribut de décision pour prochain nœud;
2- Assigner a comme attribut de décision au nœud;
3- Pour chaque valeur de a, créer un nœud descendant**;**
4- Trier les exemples d'entrainement dans les feuilles de l'arbre;
5- **Si** (les exemples sont parfaitement classés)
STOP;
sinon
Itérer sur un nœud feuille.
fin

Choix de l'attribut de décision

Soit un exemple de nœud pour lequel on doit choisir un attribut de décision:



Quel attribut de décision choisir, a_1 ou a_2?

Choix de l'attribut de décision

- Les algorithmes de construction d'AD fonctionnent de haut en bas, en choisissant à chaque étape **une variable qui divise le mieux l'ensemble des exemples**.

- **Les métriques** utilisées pour choisir **un meilleur attribut** mesurent en général **l'homogénéité de la variable cible** dans les sous-ensembles résultants de la division de nœuds.

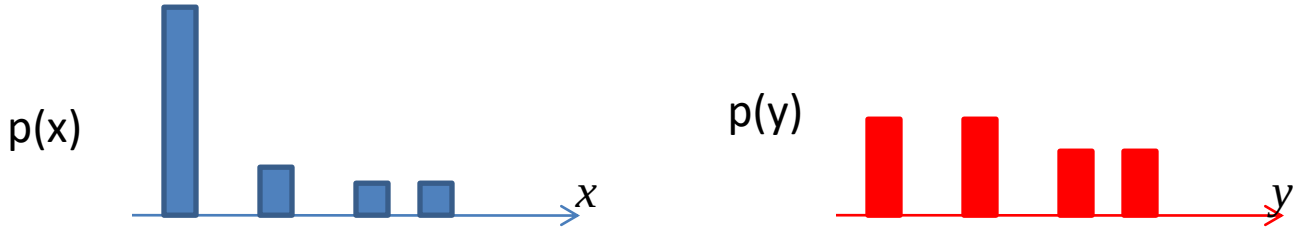
- Une autre mesure est **le gain d'information**, qui se base sur le concept **d'entropie** de la **théorie de l'information**.

Choix de l'attribut de décision

Entropie (cas d'une variable avec un attribut x**)**

- Soit $p(\mathrm{x})$ la distribution de la variable x mesurant **l'incertitude** sur la valeur de la variable.

- Certaines valeurs de variables sont moins sures que d'autres.



- **L'entropie** mesure **le nombre de bits** requis pour encoder x.

Choix de l'attribut de décision

- L'entropie $H(\mathrm{x})$ d'une variable aléatoire x est définie par:

$$H(\mathrm{x}) = -\sum_{\mathrm{x}=x_i} p(\mathrm{x} = x_i)\log_2(p(\mathrm{x} = x_i))$$

- $H(\mathrm{x})$ est le nombre de bits attendu pour encoder les valeurs de la variable aléatoire x.

- Un code efficace assigne $-\log_2(p(\mathrm{x} = x_i))$ bits pour encoder un message $\mathrm{x} = x_i$. **La valeur attendu de la longueur du code** est alors définie par la formule en haut.

Choix de l'attribut de décision (cas binaire)

- Pour K = 2 (deux classes), les éléments de la classe C_1 seront dénotés par $\oplus$ et ceux de la classe C_2 par $\ominus$. On aura:

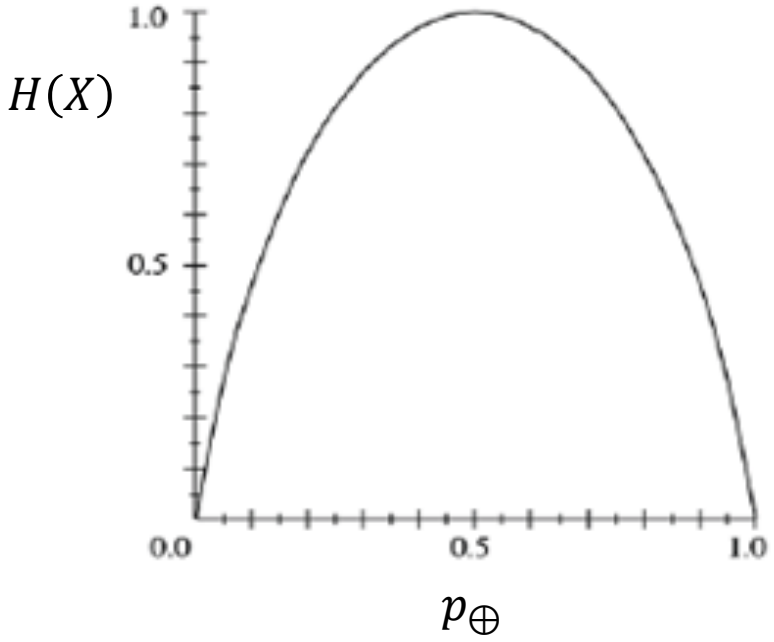
$$H(\mathrm{x}) = -p_{\oplus}\log_2(p_{\oplus}) - p_{\ominus}\log_2(p_{\ominus})$$

- $p_{\oplus}$ et $p_{\ominus}$ sont les proportions des deux classes dans le nœud, avec $p_{\oplus} + p_{\ominus} = 1$. On aura aussi:

- $0 \leq H(\mathrm{x}) \leq 1$.
- Si $p_{\oplus} = 0$ ou $p_{\ominus} = 0$, alors $H(\mathrm{x}) = 0$.
- Si $p_{\oplus} = p_{\ominus} = 0.5$, alors $H(\mathrm{x}) = 1$ (entropie maximale).

Choix de l'attribut de décision (cas binaire)

- Pour le cas de 2 classes, on aura le graphe suivant de $H(\mathrm{x})$:



- L'entropie mesure **le degré d'impureté** d'un nœud.

Choix de l'attribut de décision (cas binaire)

- Soit une population d'exemples $\mathcal{D}$. Le gain d'information de $\mathcal{D}$ par rapport à un attribut a_j donné est la variation d'entropie causée par la partition de $\mathcal{D}$ selon a_j.

$$Gain(\mathcal{D}, a_j) = H(\mathcal{D}) - \sum_{v \in valeurs(a_j)} \frac{|\mathcal{D}_{aj=v}|}{|\mathcal{D}|} H(\mathcal{D}_{aj=v})$$

- $\mathcal{D}_{aj=v} \subset \mathcal{D}$ est l'ensemble des exemples ayant $a_j = v$.

- $|\mathcal{D}|$ indique la cardinalité de l'ensemble $\mathcal{D}$.

Jour	Ciel	Température	Humidité	Vent	Jouer au tennis ?
1	Ensoleillé	Chaude	Élevée	Faible	Non
2	Ensoleillé	Chaude	Élevée	Fort	Non
3	Couvert	Chaude	Élevée	Faible	Oui
4	Pluie	Tiède	Élevée	Faible	Oui
5	Pluie	Fraîche	Normale	Faible	Oui
6	Pluie	Fraîche	Normale	Fort	Non
7	Couvert	Fraîche	Normale	Fort	Oui
8	Ensoleillé	Tiède	Élevée	Faible	Non
9	Ensoleillé	Fraîche	Normale	Faible	Oui
10	Pluie	Tiède	Normale	Faible	Oui
11	Ensoleillé	Tiède	Normale	Fort	Oui
12	Couvert	Tiède	Élevée	Fort	Oui
13	Couvert	Chaud	Normale	Faible	Oui
14	Pluie	Tiède	Élevée	Fort	Non

Choix de l'attribut de décision (cas binaire)

- Si a_j prends par exemple 3 valeurs $a_j \in \{v_1, v_2, v_3\}$, la partition de $\mathcal{D}$ va former 3 parties: $\mathcal{D}_{aj=v1}, \mathcal{D}_{aj=v2}, \mathcal{D}_{aj=v3}$. Le gain d'information est donné par:

$$H(\mathcal{D}) - \frac{|\mathcal{D}_{aj=v1}|}{|\mathcal{D}|} H(\mathcal{D}_{aj=v1}) - \frac{|\mathcal{D}_{aj=v2}|}{|\mathcal{D}|} H(\mathcal{D}_{aj=v2}) - \frac{|\mathcal{D}_{aj=v3}|}{|\mathcal{D}|} H(\mathcal{D}_{aj=v3})$$

- On peut dire donc que le gain est la différence entre l'entropie moyenne des exemples $\mathcal{D}$ et l'entropie moyenne une fois que $\mathcal{D}$ a été partitionné selon les valeurs de a_j.

- On notera que plus cette différence (le gain) est grande, plus l'homogénéisation est grande.

Choix de l'attribut de décision (cas binaire)

Exercice:

- Dans notre exemple, il existe 9 $\oplus$ et 5 $\ominus$. Parmi ces exemples, 6 $\oplus$ et 2 $\ominus$ prennent la valeur «alpha» pour l'attribut a, tandis que les autres exemples prennent la valeur «beta» pour cet attribut.
- Calculer le gain d'information $Gain(\mathcal{D}, a)$ pour l'attribut a, si on choisi de le placer en racine.

$$Gain(\mathcal{D}, a_j) = H(\mathcal{D}) - \sum_{v \in valeurs(a_j)} \frac{|\mathcal{D}_{aj=v}|}{|\mathcal{D}|} H(\mathcal{D}_{aj=v})$$

$$H(\mathrm{x}) = -\sum_{\mathrm{x}=x_i} p(\mathrm{x} = x_i) \log_2(p(\mathrm{x} = x_i))$$

Solution:

	a	y
Parmi ces exemples, 6 ⊕ et 2 ⊖ prennent la valeur «alpha» pour l'attribut a	**6 ⊕ → alpha**	**9 ⊕**
	2 ⊖ → alpha	
les autres exemples prennent la valeur «beta» pour cet attribut.	**3 ⊕ → beta**	**5 ⊖**
	3 ⊖ → beta	

$$Gain(\mathcal{D}, a) = H(D) - \frac{8}{14} H(\mathcal{D}_{a=alpha}) - \frac{6}{14} H(\mathcal{D}_{a=beta})$$

$$H(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940$$

$$H(\mathcal{D}_{a=alpha}) = -\frac{6}{8}\log_2\left(\frac{6}{8}\right) - \frac{2}{8}\log_2\left(\frac{2}{8}\right) = 0.811$$

$$H(\mathcal{D}_{a=beta}) = -\frac{3}{6}\log_2\left(\frac{3}{6}\right) - \frac{3}{6}\log_2\left(\frac{3}{6}\right) = 1$$

$$Gain(\mathcal{D}, a) = 0.940 - \frac{8}{14}0.811 - \frac{6}{14}1 = 0.940 - 0.8920 = 0.048$$

Choix de l’attribut de décision (cas binaire)

- Le principe de l'algorithme ID3 pour déterminer l'attribut à placer à la racine de l’AD est de:

  ➢ Chercher l'attribut qui possède **le gain d'information maximum**, le placer en racine.

  ➢ Itérer pour chaque fils, c.à.d. pour chaque valeur de l'attribut.

  ➢Arrêter l'algorithme quand **le gain est négligeable**.

Étude d'un exemple d'AD

Pour notre exemple: « jouer au tennis?»,

- les exemples n'étant ni tous $\oplus$, ni tous $\ominus$, l'ensemble des attributs n'étant pas vide, on calcule les gains d'information pour chaque attribut:

Attribut	Gain
Ciel	$0,246$
Humidité	$0,151$
Vent	$0,048$
Température	$0,029$

- Donc, la racine de l'AD est l'attribut « Ciel ».

Étude d'un exemple d'AD

- L'attribut « **Ciel** » peut prendre 3 valeurs: **Ensoleillé**, **Pluie** et **Couvert.**
- **La branche « Ensoleillé »:** ID3, appelé récursivement avec 5 exemples: $x^{(1)}, x^{(2)}, x^{(8)}, x^{(9)}, x^{(11)}$.
- **Les gains d'information** des 3 attributs restants sont alors :

Attribut	Gain
Humidité	$0,970$
Vent	$0,570$
Température	$0,019$

- L'attribut « **Humidité** » sera donc choisi ; on continue la construction de l'AD récursivement.

Étude d'un exemple d'AD

- **La branche « Pluie »:** partant de la racine, ID3 est appelé récursivement avec 5 exemples: $x^{(4)}$, $x^{(5)}$, $x^{(6)}$, $x^{(10)}$, $x^{(14)}$.

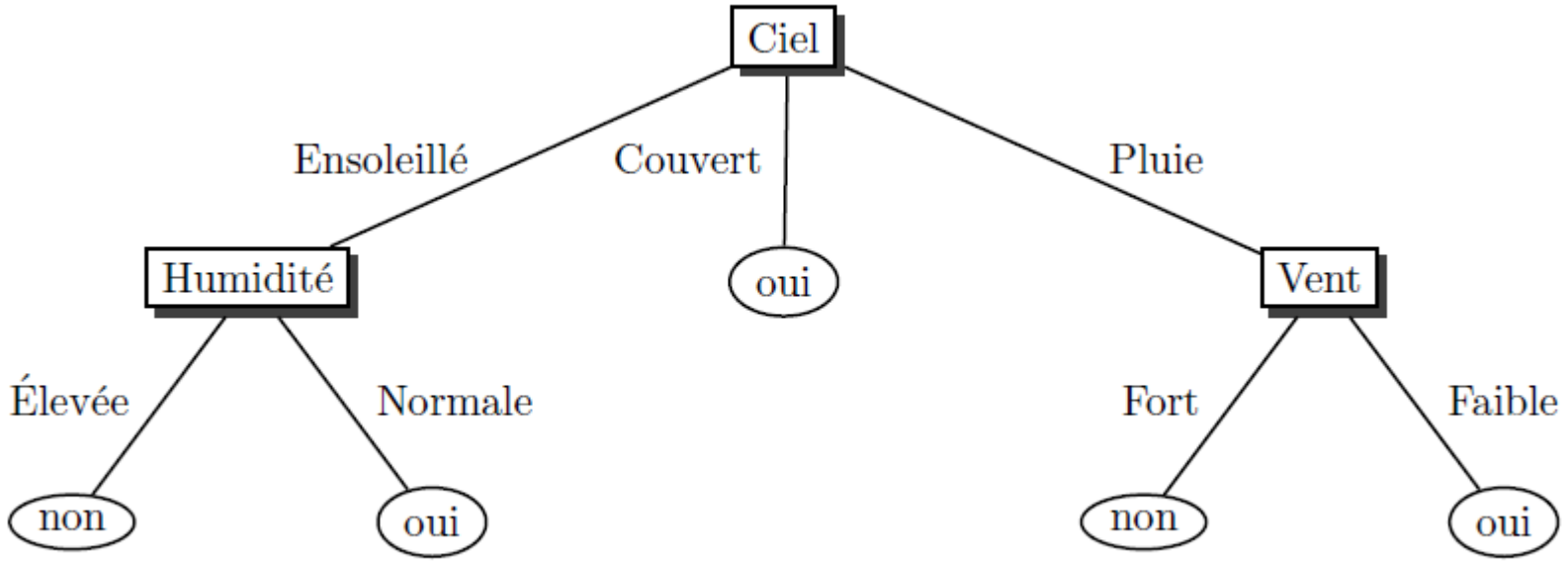
- On continue la construction de l'AD récursivement .

- **La branche « Couvert »:** partant de la racine, ID3 est appelé récursivement avec 4 exemples: $x^{(3)}$, $x^{(7)}$, $x^{(12)}$, $x^{(13)}$.

- Dans ce dernier cas, tous les exemples sont $\oplus$: on affecte donc tout de suite la classe « **oui** » à cette feuille.

Étude d'un exemple d'AD

- L'arbre final sera donné par le graphe:



Étude d'un exemple d'AD

On peut faire **les interprétations** suivante sur le graphe:

- L'attribut « **Température** », n'est pas utilisé dans l'AD, signifie que cet attribut **n'est pas pertinent** pour déterminer la classe.

- Si l'attribut « **Ciel** » vaut « **Ensoleillé**» , l'attribut « **Vent** » **n'est pas pertinent**.

- Si l'attribut « **Ciel** » vaut « **Pluie** », l'attribut « **Humidité**» **n'est pas pertinent**.

Classification par AD

En ayant un AD construit à partir d'exemples d'apprentissage $\mathcal{D}$, **la classification d'une nouvelle donnée** *x* se fait par l'algorithme suivant:

Algorithme: entrées (**AD**, *x*)

- **Nc** = *racine* (**AD**)
- *Tant-que* (**Nc** ≠ feuille) *faire*:
  - En fonction de l'attribut testé dans **Nc** et de sa valeur dans **x**, suivre l'une des branches de **Nc**.
  - Le nœud atteint devient **Nc**.
- *Fin tant-que*
- Retourner Étiquette (**Nc**).

Classification par AD

Exercice: Classer les exemples suivants:

x = (Ensoleillé, Fraîche, Elevée, Fort) ;

Exercice: Classer les exemples suivants:

x = (Ensoleillé, Fraîche, Elevée, Fort) ;

Classification par AD

Exercice: Classer les exemples suivants:

x = (Ensoleillé, Fraîche, Elevée, Fort) ;

Non

x = (Ensoleillé, Fraîche, Normale, Fort) ;

Oui

x = (Pluie, Chaude, Normale, Faible);

Oui

x = (Pluie, Fraîche, Elevée, Fort).

Non

Attributs numériques dans les ADs

- Successeur d'ID3, C4.5 prend en compte les **attributs numériques**, c.-à-d., des attributs dont **l'arité** est enlevée (voire infinie).

- Hormis cela et quelques détails décrits plus loin, la construction d'un AD par **C4.5** est **identique** dans son principe à la construction par **ID3**.

- Dans C4.5, un nœud de l'AD peut contenir **un test** du fait que la valeur **d'un attribut numérique** est inférieure ou égale à un certain **seuil**.

  ⇒ correspond donc à un nouveau **pseudo-attribut binaire**.

Attributs numériques dans les ADs

- Nous considérons notre exemple illustratif «**jouer au tennis ?**» dans lequel les attributs «Température» et «Humidité » ont été **numérisés**.

- Considérons les exemples dont l'attribut « **Ciel** » vaut la valeur «*Ensoleillé*», et soit l'ensemble $\mathcal{D}_{\text{Ciel=Ensoleillé}}$ d'exemples ayant un seul attribut numérique comme suit :

Jour	Température	« jouer au tennis »
1	27, 5	non
2	25	non
8	21	non
9	19, 5	oui
11	22, 5	oui

Attributs numériques dans les ADs

- On commence par **trier** les exemples sur **la valeur de leur attribut numérique**.

- À chaque attribut, on associe le numéro de son exemple associé ainsi que la valeur de l'attribut cible:

Température	19,5	21	22,5	25	27,5
Jour	9	8	11	2	1
« jouer au tennis ? »	oui	non	oui	non	non

- On détermine le seuil ***s*** pour **partitionner** cet ensemble d'exemples.

Attributs numériques dans les ADs

C4.5 utilise les règles suivantes :

- Ne pas séparer 2 exemples successifs ayant la même classe:

  $\Rightarrow$ On ne peut couper qu'entre les exemples $x^{(9)}$ et $x^{(8)}$, $x^{(8)}$ et $x^{(11)}$, $x^{(11)}$ et $x^{(2)}$;

- Si on coupe entre 2 valeurs v et w ($v < w$) de l'attribut, le seuil s est fixé à v (on aurait pu aussi utiliser $(v + w) / 2$) ;

- Choisir le seuil s de manière que **le gain d'information** soit maximal.

Attributs numériques dans les ADs

Remarque :

- Une fois le seuil s fixé et le nœud créé, chaque **sous-arbre** pourra à nouveau tester la valeur de cet attribut.

- Contrairement au cas des **attributs qualitatifs** qui produisent des nœuds ayant autant de branches que l'attribut prend de valeurs différentes, **l'ensemble des valeurs prises par un attribut numérique est coupé en 2**.

- Chaque partie peut donc encore être raffinée jusqu'à ne contenir que **des exemples ayant la même valeur cible**.

Attributs numériques dans les ADs

- Dans notre exemple, l'entropie de l'ensemble est donnée par:

$$H(\mathcal{D}) = -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right) \approx 0.971$$

- Pour $s = 21$, le gain d'information est:

$$Gain(\mathcal{D}, Température, s = 21)$$
$$= H(\mathcal{D}) - \left(\frac{2}{5}H\left(\mathcal{D}_{Température \leq 21}\right) + \frac{3}{5}H\left(\mathcal{D}_{Température > 21}\right)\right)$$

$$H\left(\mathcal{D}_{Température \leq 21}\right) = -\left(\frac{1}{2}\log_2\left(\frac{1}{2}\right) + \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) \approx 1$$

$$H\left(\mathcal{D}_{Température > 21}\right) = -\left(\frac{1}{3}\log_2\left(\frac{1}{3}\right) + \frac{2}{3}\log_2\left(\frac{2}{3}\right)\right) \approx 0.39$$

Attributs numériques dans les ADs

- Il s'en suit que:

$$Gain(\mathcal{D}, Température, s = 21) = 0.971 - \left(\frac{2}{5} \cdot 1 + \frac{3}{5} \cdot 0.39\right) \approx 0.337$$

Exercice:

De la même manière, en fonction du seuil, remplir le tableau:

Seuil	Gain($\mathcal{D}$, *Température*, s)
$s = 21$	0.337
$s = 19.5$	??
$s = 22.5$	??
S=25	??

Attributs numériques dans les ADs

- Dans son action, **C4.5** effectue ce traitement **pour chaque attribut quantitatif** et détermine donc pour chacun **un seuil** produisant **un gain d'information maximal**.

- **Le gain d'information** associé à chacun des attributs quantitatifs est celui pour lequel **le seuil entraine un maximum.**

- Finalement, l'attribut choisi (parmi les quantitatifs et les nominaux pour lesquels le principe est identique à ID3) est celui qui produit **un gain d'information maximal.**

Rapport de gain pour C4.5

- En présence **d'attribut numérique** ou **d'attribut d'arité élevée**, ceux-ci sont automatiquement **favorisés** pour être sélectionnés comme test dans les nœuds.

- Pour contrecarrer cet effet, C4.5 utilise **le rapport de gain** au lieu du **gain d'information** pour déterminer l'attribut à utiliser dans un nœud.

- **Le rapport de gain** est défini par :

$$Rapport\ gain(\mathcal{D}, a) = \frac{Gain(\mathcal{D}, a)}{Division\ Inf(\mathcal{D}, a)}$$

Rapport de gain pour C4.5

- Où on a:

$$Division\ Inf(\mathcal{D}, a) = \sum_{v \in valeurs(a)} \frac{|\mathcal{D}_{a=v}|}{|\mathcal{D}|} \log_2 \left(\frac{|\mathcal{D}_{a=v}|}{|\mathcal{D}|} \right)$$

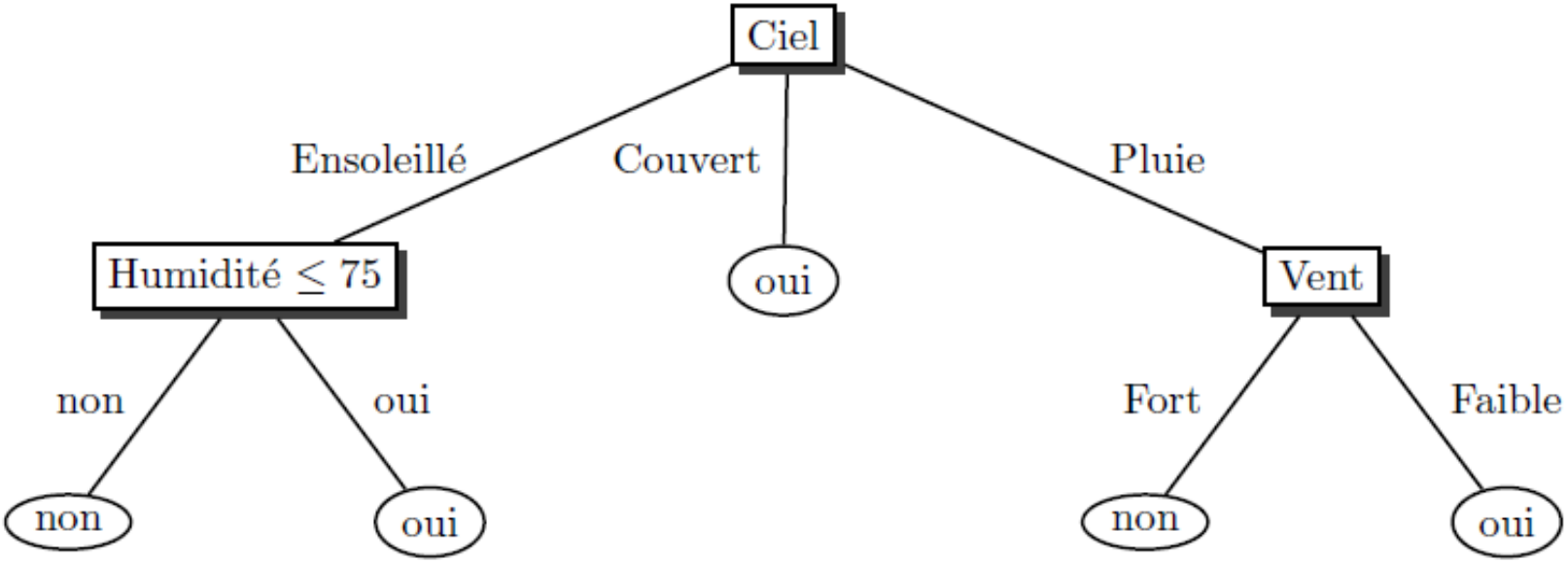
- Contrairement à ID3, qui choisit l'attribut ayant **le gain maximal**, C4.5 choisit l'attribut ayant **le rapport de gain maximal** pour le placer à **la racine de l'arbre de décision**.

- Pour notre exemple (**jouer au tennis ?**), avec des valeurs numériques pour les attributs « **Température** » (en °C) et «**Humidité** » (en %), on obtient l'AD suivant:

Rapport de gain pour C4.5

Jour	Ciel	Température	Humidité	Vent	Jouer au tennis ?
1	Ensoleillé	27, 5	85	Faible	Non
2	Ensoleillé	25	90	Fort	Non
3	Couvert	26, 5	86	Faible	Oui
4	Pluie	20	96	Faible	Oui
5	Pluie	19	80	Faible	Oui
6	Pluie	17, 5	70	Fort	Non
7	Couvert	17	65	Fort	Oui
8	Ensoleillé	21	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	90	Fort	Oui
13	Couvert	25, 5	75	Faible	Oui
14	Pluie	20, 5	91	Fort	Non

Rapport de gain pour C4.5



Valeurs d’attributs manquantes (VAM)

- ID3 ne dispose pas de possibilité pour gérer le problème des valeurs d’attributs manquantes (VAM); C4.5 dispose d'un mécanisme décrit comme suit:
- On distingue deux cas :

  ➢ Certains attributs dans **les exemples d'apprentissage** sont **non valués** ;

  ➢ Certains attributs de **la donnée à classer** sont **non valués**.

- On traite successivement ces deux cas dans la suite du cours.

VAM dans la phase d’apprentissage

Plusieurs solutions peuvent être envisagées, les plus générales:

- On laisse de côté les exemples ayant des valeurs manquantes ;

  ☞ennuyeux car le nombre d'exemples diminue ;

- Le fait que la valeur de l'attribut soit manquante est une information en soit :

  ☞ on ajoute alors une valeur à l'ensemble des valeurs de cet attribut qui indique que la valeur est inconnue ;

- La valeur **la plus courante** pour l'attribut parmi les exemples classés dans ce nœud est affectée à la valeur manquante.

VAM dans la phase d'apprentissage

- **Les différentes valeurs observées** dans les exemples couverts par le même nœud sont affectées avec **des poids différents** en fonction de **la proportion d'exemples** d'apprentissage couverts par ce nœud pour **les différentes valeurs** de cet attribut.

  ☞ C'est cette dernière possibilité qui est utilisée par **C4.5**.

  ☞ **Les fractions d'exemples** sont utilisées pour la construction de l'AD. Il faut alors **adapter le calcul du gain d'information**.

VAM dans la phase d'apprentissage

- Pour calculer **le gain d'information**, on ne tient compte que des exemples dont l'attribut est valué.

- Soit $\mathcal{X}$ l'ensemble d'exemples couverts par le nœud courant (dont on est en train de déterminer l'attribut à tester), et soit $\mathcal{D}_{sans?} \subset \mathcal{D}$ les exemples dont l'attribut est valué.

- On redéfinit: $H(\mathcal{D}) = H(\mathcal{D}_{sans?})$ et on a:

$$Gain(\mathcal{D},a) = \left(H(\mathcal{D}) - \sum_{v \in valeurs(a)} \frac{|\mathcal{D}_{sans?,a=v}|}{|\mathcal{D}_{sans?}|} H(\mathcal{D}_{sans?,v=a}) \right) \frac{|\mathcal{D}_{sans?}|}{|\mathcal{D}|}$$

VAM dans la phase d'apprentissage

Exemple : Supposons que $x^{(12)}$ ait ? a la place de « *Couvert* » comme valeur de son attribut « **Ciel** ».

$$H(\mathcal{D}) = -\left(\frac{8}{13}\log_2\left(\frac{8}{13}\right) - \frac{5}{13}\log_2\left(\frac{5}{13}\right)\right) \approx 0.961$$

$$Gain(\mathcal{D}, Ciel)$$
$$\approx \frac{13}{14}\left(0.961 - \frac{5}{13}\left(-\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right)\right.$$
$$- \frac{3}{13}\left(-\frac{3}{3}\log_2\left(\frac{3}{3}\right) - \frac{0}{3}\log_2\left(\frac{0}{3}\right)\right)$$
$$\left. - \frac{5}{13}\left(-\frac{3}{5}\log_2\left(\frac{3}{5}\right) - \frac{2}{5}\log_2\left(\frac{2}{5}\right)\right)\right) \approx 0.199$$

VAM dans la phase d'apprentissage

- Demeurant l'attribut fournissant **un gain maximal**, « **Ciel** » est placé à la racine de l'arbre.

- L'exemple 12 est affecté avec les poids $\frac{5}{13}$, $\frac{3}{13}$ et $\frac{5}{13}$ à chacune des branches, respectivement « *Ensoleillé* » , « *Couvert* » et « *Pluie* ».

- Les autres exemples sont affectés à leur branche respective avec un poids 1 pour chacun.

Ex. Dans la branche « *Ensoleillé* », la proportion des $\oplus$ sera de $\frac{5/13+2}{5/13+2+3} \approx 0.44$ et celle des $\ominus$ sera de $\frac{3}{5/13+2+3} \approx 0.56$.

VAM dans la phase de classification

- Quand on descend dans l'arbre, si un nœud teste un attribut dont la valeur est inconnue, **C4.5 estime la probabilité pour la donnée de suivre chacune des branches** en fonction de la répartition des exemples d'apprentissage couverts par ce nœud.

- Cela détermine **une fraction de donnée** qui poursuit sa descente le long de chacune des branches.

- Arrivé aux feuilles, **C4.5 détermine la classe la plus probable** à partir de ces probabilités estimées. Pour chaque classe, il fait la somme des poids;

  ☞ La classe prédite est celle dont le poids est maximal.

VAM dans la phase de classification

Exemple: classons la donnée:

(Ciel =?, Température = Tiède, Humidité =?, Vent = Faible)?

- La valeur **« Ciel »** étant inconnue, on calcule la proportion d'exemples correspondant à chaque valeur: 5 Ensoleillé ; 4 Couvert ; 5 Pluie .

- Donc, on poursuit la classification en transmettant les poids $\frac{5}{14}$ vers le nœud testant l'attribut « **Humidité** », $\frac{4}{14}$ le long de la branche « *Couvert* » vers l'étiquette « oui » et $\frac{5}{14}$ vers le nœud testant l'attribut « **Vent** ».

VAM dans la phase de classification

- La valeur de « **Humidité** » est inconnue également, .

- Parmi les exemples « Ensoleillé », il y en a 3 dont la cible vaut «*Non*», 2 dont la cible vaut «*Oui*», soit $\frac{3}{5}$ et $\frac{2}{5}$ respectivement.

- Puisque $\frac{5}{14}$ exemple a suivi cette branche depuis la racine, on obtient $\frac{5}{14} \times \frac{3}{5} = \frac{3}{14}$ exemple atteignant l'étiquette « non » et $\frac{5}{14} \times \frac{2}{5} = \frac{1}{7}$ exemple atteignant l'étiquette « oui ».

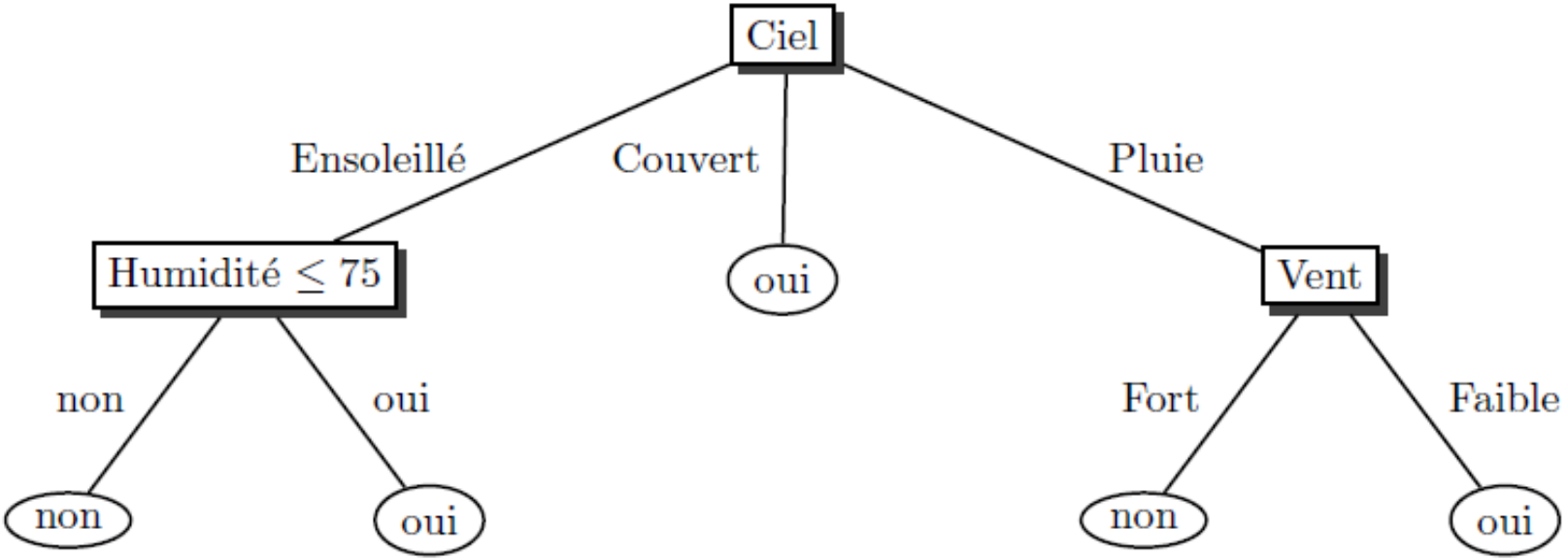
VAM dans la phase de classification

- L'attribut « **Vent** » a la valeur « *Faible* »; le $\frac{5}{14}$ d'exemple qui ont suivi cette branche depuis la racine est donc classé comme « oui ».

- En résumé:
  - Il y a $\frac{3}{14}$ exemple qui atteint une étiquette « non » ;
  - Il y a $\frac{1}{7} + \frac{4}{14} + \frac{5}{14} = \frac{11}{14}$ exemple qui atteint une étiquette « oui ».

- On conclut que la classe la plus probable de la donnée est « oui ».

(Ciel =?, Température = Tiède, Humidité =?, Vent = Faible)?



Validation d’un classificateur

Validation d'un arbre de décision

- Une fois un AD construit, il est essentiel de le valider en estimant **la probabilité que la classe prédite** pour une donnée quelconque soit correcte. Cette quantité est donc une variable aléatoire dont il faut estimer la valeur.

- **L'erreur de classification** *E* d'un classificateur est la probabilité que ce dernier ne prédise pas correctement la classe d'une donnée de $\mathcal{D}$. **Le taux de succès** est égal a 1 - *E*.

- **L'erreur apprentissage** E_{app} est mesurée avec les exemples $\mathcal{D}_{app}$: c'est la proportion d'exemples dont la classe est mal prédite par le classificateur.

Validation d'un arbre de décision

Remarques:

- E_{app} n'est **pas un bon estimateur de l'erreur qui serait commise face à de nouvelles données**.

- L'apprentissage doit pouvoir être généralisé à de nouvelles données: **l'enjeu est bien là !**

- Un algorithme ne peut être qualifié **de bon algorithme d'apprentissage** que s'il est capable de **généraliser** ce qu'il a appris **à de nouvelles données**.

Validation d'un arbre de décision

Pour la validation, on distingue :

- Le **jeu d'exemples d'apprentissage** $\mathcal{D}_{\text{app}}$ ou d'entrainement avec lequel l'AD est construit.

- Le **jeu d'exemples de test** $\mathcal{D}_{\text{test}}$ pour lesquels, on connait les valeurs des classes. On les classe avec l'AD construit avec $\mathcal{D}_{\text{app}}$, puis on regarde s'ils sont classés correctement.

- Si on ne dispose pas d'un jeu de test, on découpe $\mathcal{D}_{\text{app}}$ en deux parties; l'une constituera **le jeu d'apprentissage effectif** pour construire l'AD; l'autre servira à évaluer (technique de ***holdout***).

Mesure de qualité d'un classificateur

- On suppose que l'on a construit un classificateur (un AD ou un autre) à partir d'un jeu d'exemples d'apprentissage $\mathcal{D}_{\text{app}}$.

- Dans le cas de **classification binaire** (sachant que les définitions sont aisément extensibles aux autres cas), on définit :

  - ***vp*** : le nombre **de vrais positifs**, c.-à-d. les exemples de classe positive dont la classe est prédite comme positive;

  - ***vn*** : le nombre **de vrais négatifs**, c.-à-d. les exemples de classe négative dont la classe est prédites comme négative ;

Mesure de qualité d'un classificateur

- ***fp*** : le nombre **de faux positifs**, c.-à-d. les exemples de classe négative dont la classe est prédite comme positive ;

- ***fn*** : le nombre **de faux négatifs**, c.-à-d. les exemples de classe positive dont la classe négative.

	+	− ← classe prédite
+	VP	FN
−	FP	VN
↑ classe		

Tableau de contingence (pour K > 2, on l'appelle **matrice de confusion**)

Mesure de qualité d'un classifcateur

- S'il n'y a des nombres non nuls que sur la diagonale principale, c'est qu'aucun exemple n'est mal classé (**bonne classification**).

- On peut définir aussi deux statistiques, la **précision** et le **rappel** :

  - ***Précision*** *pour les positifs* = $\frac{vp}{vp + fp}$

  - ***précision*** *pour les négatifs* = $\frac{vn}{vn + fn}$

  - ***Rappel*** *pour les positifs* = $\frac{vp}{vp + fn}$

  - ***Rappel*** *pour les négatifs* = $\frac{vn}{vn + fp}$

Mesure de qualité d'un classificateur

- La **précision** mesure la proportion d'exemples vraiment positifs (resp. négatifs) parmi ceux qui sont classés comme positifs (resp. négatifs).
- Le **rappel** mesure la proportion d'exemples vraiment positifs (resp. négatifs) parmi tous les exemples de classes positive (resp. négative).

- Il est toujours plus pratique de manipuler un seul nombre qui synthétise les autres. Ainsi la **mesure -*F*** est définie par :

$$F = \frac{2 \times rappel \times précision}{rappel + précision} = \frac{2 \times vp}{2 \times vp + fp + fn}$$

Exercice

- Soit un classificateur binaire tel que le nombre de l'ensemble de test N = 100, le nombre d'éléments de la classe $\oplus = 60$, le nombre d'éléments de la classe $\ominus = 40$.

- Résultats de classification pour la classe $\oplus$:
  - $VP = 40 \oplus$
  - FN = 20 $\ominus$
- Résultats de classification pour la classe $\ominus$:
  - VN = 30 $\ominus$
  - FP = 10 $\oplus$

Précision = $\frac{vp}{vp+fp} = 0.8$

Rappel = $\frac{vp}{vp+fn} = 0.67$

- Calculer les mesures de qualité suivantes: Précision, Rappel, et F.

$$F = \frac{2 \times rappel \times précision}{rappel + précision} = \frac{2 \times vp}{2 \times vp + fp + fn} = 0.72$$

Validation croisée

- Une méthode plus sophistiquée est la **validation croisée**. Pour cela, **on découpe l'ensemble** des exemples en *n* sous-ensembles mutuellement disjoints.

- Il faut prendre garde à ce que **chaque classe apparaisse avec la même fréquence** dans les *n* sous-ensembles (**stratification des échantillons**).

- Si *n* = 3, on aura 3 ensembles A, B et C. On construit l'AD $AD_{A \cup B}$ avec $A \cup B$ et on mesure le taux d'erreur sur C: E_C, c.-à-d. le nombre d'exemples de C dont la classe est mal prédite par $AD_{A \cup B}$.

Validation croisée

- Ensuite, on construit l'AD $AD_{C \cup B}$ avec $C \cup B$ et on mesure le taux d'erreur sur A: E_A.

- Enfin, on construit l'AD $AD_{A \cup C}$ avec $A \cup C$ et on mesure le taux d'erreur sur B: E_B.

- Le taux d'erreur E est alors estimé par la moyenne de ces trois erreurs, comme suit:

$$E = \frac{E_A + E_B + E_C}{3}$$

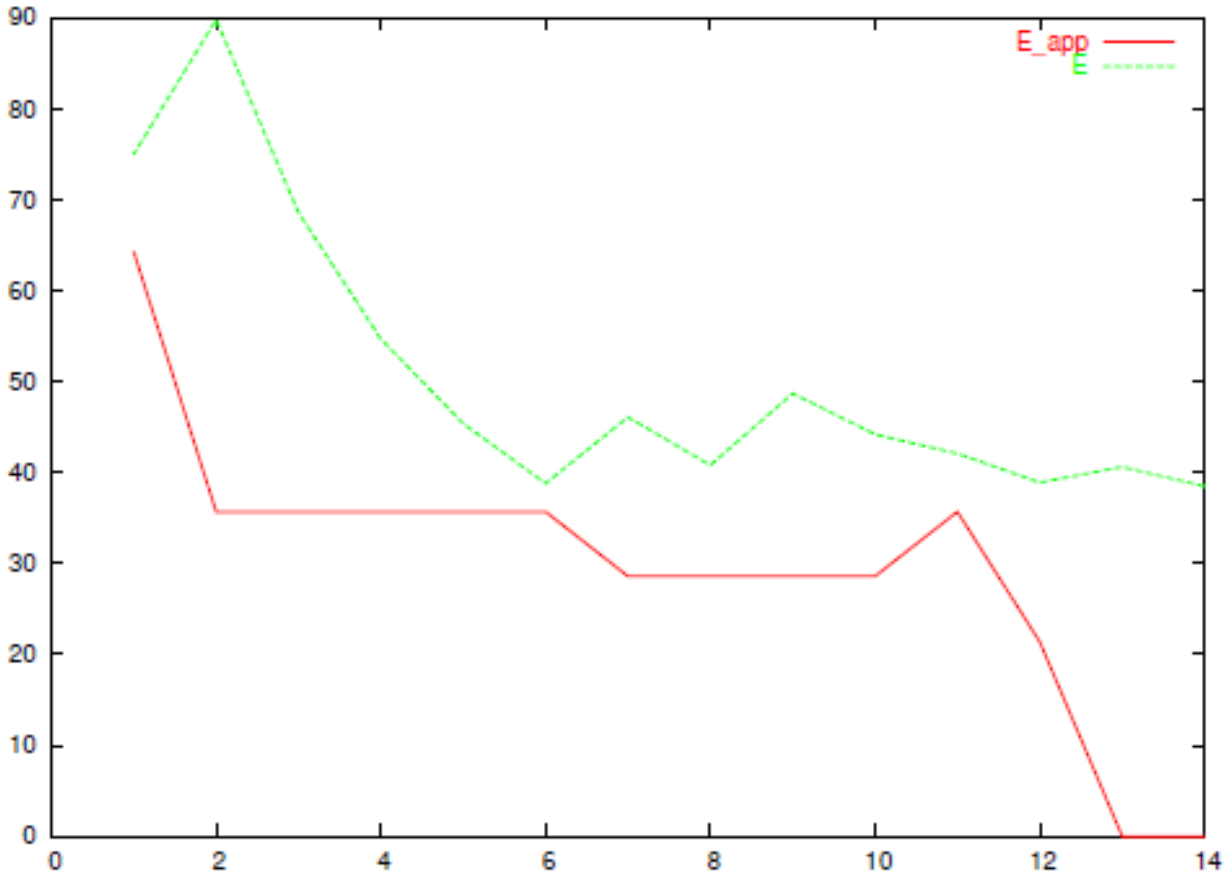
Sur-apprentissage

- Il est intéressant de dessiner **l'erreur d'apprentissage (E_{app})** et l'estimation de **l'erreur réelle *E*** en fonction de la proportion d'exemples utilisés pour l'apprentissage.

- Ainsi, sur l'exemple « **jouer au tennis** ? », on a pris successivement sur 1, 2, 3, ... 14 exemples pour construire l'AD.

- On note que E_{app} diminue constamment quand on utilise plus d'exemples pour construire l'AD, alors que *E* diminue d'abord puis **stagne**.

Sur-apprentissage



Sur-apprentissage

- L'erreur E_{app} atteint 0 et reste à 0 parce qu'on peut construire un AD qui classe correctement tous les exemples de $\mathcal{D}_{app}$.

- Si l'on ne peut pas construire d'AD parfait (à cause d'une incohérence dans les exemples: **ex.** deux exemples ayant la même description sont de classe différente), $\boldsymbol{E}_{app}$ reste non nulle.

- Ce qui compte vraiment, c'est ***E***. Quand *n* augmente, ***E*** commence par diminuer puis elle augmente : c'est précisément la où elle est minimale que l'on a construit le **meilleur modèle,** celui qui fait **une erreur minimale**.

Sur-apprentissage

- Au-delà de du **modèle optimal**, quand l'apprentissage se poursuit, **le modèle se complique**.

- La probabilité d'erreur augmente, et le modèle produit du **sur-apprentissage,** c.à.d. le modèle construit colle de plus en plus près aux exemples et sa capacité à prédire correctement la classe d'autres données diminue ;

- Intuitivement, on peut dire dans ce cas que le modèle **manque de recul par rapport aux exemples**.

Élagage

- **L'élagage** consiste à simplifier un AD en coupant des branches. Il possède deux objectifs :

  ☞ **Simplifier** l'arbre de décision ;

  ☞ **Diminuer le sur-apprentissage** (augmenter la capacité de généralisation) et, par la-même, **diminuer le taux d'erreur.**

- Deux possibilités s'offrent pour l'élagage:

  ☞ Élagage lors de la construction.

  ☞ Élagage après la construction (C4.5).

Élagage

- Dans C4.5, deux possibilités d'élagage:

  ☞ **Remplacement d'un sous arbre** : consiste a remplacer un sous arbre par une feuille ;

  ☞ **Promotion d'un sous arbre** : consiste à rassembler deux nœuds dans un seul nœud.

- Dans le premier cas (**remplacement d'un sous arbre**), par exemple, les nœuds sont considérés depuis les feuilles en remontant vers la racine et en effectuant un test au niveau de chaque nœud.

Élagage

- La décision d'appliquer une transformation est prise en fonction du **taux d'erreur du nœud et de ses fils,** comme suit.

- C4.5 estime l'erreur de chaque fils ainsi que l'erreur du nœud. Ces estimations sont combinées **en les pondérant chacune par la proportion d'exemples couvert par chaque fils**.

- C4.5 estime ensuite le taux d'erreur du nœud. Si celui-ci est plus petit que la combinaison des taux d'erreur des fils, alors C4.5 remplace le nœud par une feuille.

Élagage

Exemple : On suppose qu'un nœud possède 3 fils:

Le premier couvre 6 exemples, dont 2 sont mal classés ;

Le deuxième couvre 2 exemples, 1 étant mal classé ;

le troisième couvre 6 exemples dont 2 sont mal classés.

- Les taux d'erreur estimé de ces 3 fils sont 0.47, 0.72 et 0.47. La combinaison des 3 donne : $\frac{6}{14}0.47 + \frac{2}{14}0.72 + \frac{6}{14}0.47 = 0.504.$

- Le taux d'erreur du nœud est calculé avec les 14 exemples, dont 5 mal classés, ce qui donne 0.46. Ce taux d'erreur étant plus petit, C4.5 remplace le nœud par une feuille étiquetée avec la **valeur la plus présente** des 14 exemples.

Références

1. M. S. Allili. Techniques d'apprentissage automatique (Cours de 2e cycle). Université du Québec en Outaouais (UQO), Québec, Canada. Hivers 2015.
2. S. Rogers et M Girolami. A first Course in machine learning, CRC press, 2012.
3. C. Bishop. Pattern Recognition and Machine learning. Springer 2006.
4. R. Duda, P. Storck et D. Hart. Pattern Classification. Prentice Hall, 2002.