Structures de données et Algorithmes

ESILV S5 ⋈ 2017–2018

TD 8 & 9 – **Problème noté**

Attention bien lire cet encadré :

- Ce travail est à réaliser individuellement.
- Il est à rendre au plus tard à 22h00 samedi 13 janvier 2018.
- Vous déposerez **sur Moodle un fichier** `.zip` contenant :
  - le **code source** (les fichiers `.c` et `.h`)
  - le **mini-rapport** en `PDF` de 4 pages maximum.
- Un rendu sans le rapport sera noté 00/20.
- Un programme avec des *Warnings* à la compilation sera noté avec un malus pouvant aller jusqu'à -5 points selon la nature du/des *Warnings*.
- Un programme ne compilant pas ou ne s'exécutant pas sera noté 00/20.
- Une triche entraîne une note de 00/20 à l'ensemble du module.

Préambule

Un arbre de décision est un outil d'aide à la décision pour un ensemble de choix représenté sous la forme d'un arbre (*e.g.* Figure 1). Les différentes décisions possibles sont situées au niveau des feuilles de l'arbre, et sont atteintes en fonction des choix pris à chaque étape.

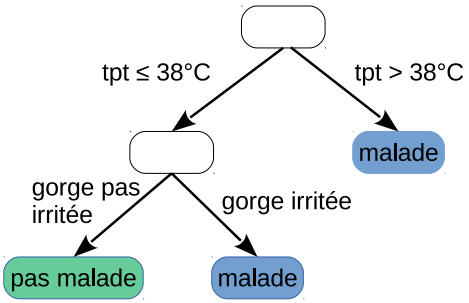


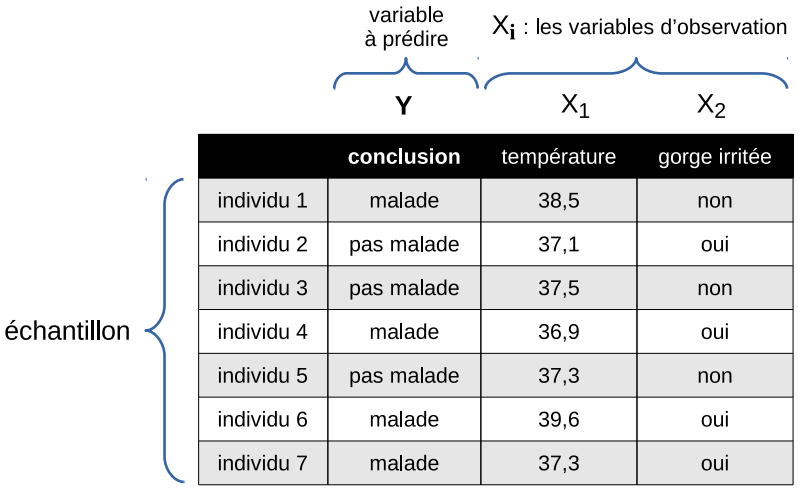
Figure 1 – Exemple très simple d'arbre de décision

La construction d'un arbre de décision (*e.g.* Figure 3) se fait à partir d'un échantillon de données, appelées « données d'apprentissage » (*e.g.* Figure 2), où

- chaque individu de l'échantillon (chaque **ligne**) dispose de valeurs pour un ensemble de variables $\{X_1, X_2, ...\}$, dites « variables d'observation » (les **colonnes**).
- Ces variables X_i sont utilisées pour diviser les individus en sous-échantillons ayant des valeurs proches ou communes.
- L'objectif étant d'être capable – pour de nouveaux individus – de prédire la valeur (appelée *classe*) de la variable Y, dite « variable à prédire ».

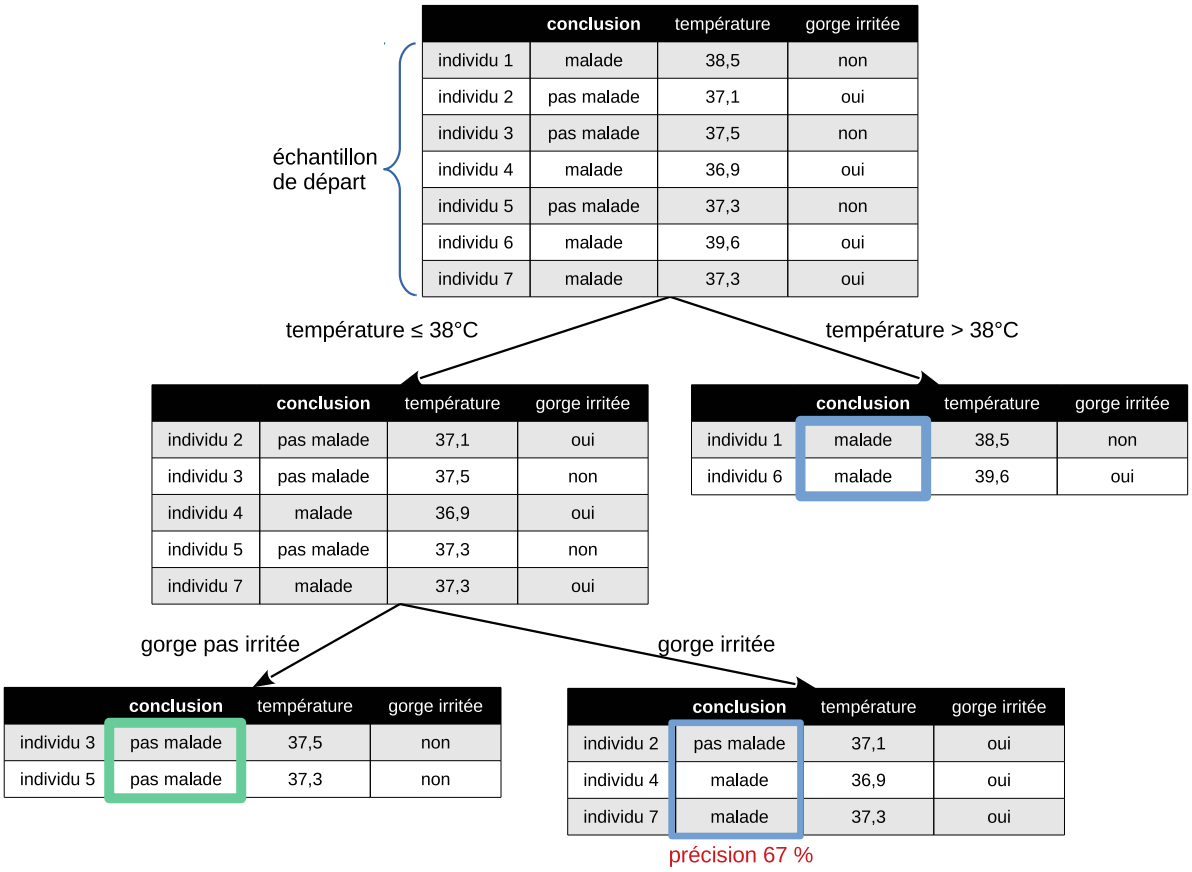
✎ Dans la mesure où un expert a pu valider (superviser) la valeur de la variable Y pour chaque individu des données d'apprentissage, on appelle cela un **« apprentissage supervisé »**.

L'utilisation d'un arbre de décision (une fois construit !) permet de prédire – avec une certaine précision – la valeur inconnue de Y pour un **nouvel** individu selon les valeurs observées pour les variables X_i de ce nouvel individu.



	conclusion	température	gorge irritée
individu 1	malade	38,5	non
individu 2	pas malade	37,1	oui
individu 3	pas malade	37,5	non
individu 4	malade	36,9	oui
individu 5	pas malade	37,3	non
individu 6	malade	39,6	oui
individu 7	malade	37,3	oui

FIGURE 2 – Exemple de données d'apprentissage



	conclusion	température	gorge irritée
individu 1	malade	38,5	non
individu 2	pas malade	37,1	oui
individu 3	pas malade	37,5	non
individu 4	malade	36,9	oui
individu 5	pas malade	37,3	non
individu 6	malade	39,6	oui
individu 7	malade	37,3	oui

	conclusion	température	gorge irritée
individu 2	pas malade	37,1	oui
individu 3	pas malade	37,5	non
individu 4	malade	36,9	oui
individu 5	pas malade	37,3	non
individu 7	malade	37,3	oui

	conclusion	température	gorge irritée
individu 1	malade	38,5	non
individu 6	malade	39,6	oui

	conclusion	température	gorge irritée
individu 3	pas malade	37,5	non
individu 5	pas malade	37,3	non

	conclusion	température	gorge irritée
individu 2	pas malade	37,1	oui
individu 4	malade	36,9	oui
individu 7	malade	37,3	oui

FIGURE 3 – Exemple de construction d'un arbre de décision

1 Énoncé du problème

On se propose dans ce double TD de coder en `C`

1. la construction automatique d'un arbre de décision à partir de données d'apprentissage supervisé.
2. à l'aide de l'arbre de décision construit, de prédire si oui ou non un/des nouveaux individus possède une certaine valeurs de Y ? (d'où le nom variable à prédire de Y)

Critère de division Le critère pour diviser un échantillon à chaque étape de construction de l'arbre de décision est déterminant. Beaucoup de chercheurs se sont penchés dessus, notamment selon la nature des données quantitatives et/ou qualitatives (des nombres et/ou des chaînes de caractères) des différentes variables.

Pour les besoins du problème (et par simplification),
- les valeurs des variables Y et X_i seront des **réels**.
- la division d'un échantillon se fera de façon binaire en fonction de la « **médiane corrigée** » d'une des variables d'observation X_i.

La médiane corrigée d'une variable d'observation X_i, avec $\{v_1, v_2, ..., v_n\}$ les valeurs rangées dans l'ordre croissant de X_i,
- n'est **pas** définie si $n < 2$ ou si toutes les valeurs sont égales ;
- est définie de la manière suivante pour $n \geqslant 2$, avec m la médiane de la série statistique $\{v_1, v_2, ..., v_n\}$ (*cf.* ci-dessous) :
  - si m n'est pas la valeur maximale de la série, alors la médiane corrigée est égale à m (donc pas de correction).
  - sinon, la médiane corrigée est égale à la plus grande valeur de la série qui ne soit pas le maximum (en d'autres termes la "seconde" valeur maximale de la série).

On rappelle que la médiane d'une série statistique $\{v_1, v_2, ..., v_n\}$ rangée dans l'ordre croissant, avec $n > 0$, est définie telle que :
- si n est pair, avec $n = 2p$, alors la médiane de la série statistique est $\frac{x_p + x_{p+1}}{2}$;
- si n est impair, alors la médiane de la série statistique est la valeur x_p où $p = \frac{n+1}{2}$.

Exemples :
- `{1,2,3,3}` → médiane corrigée = médiane de la série statistique = $\frac{2+3}{2} = 2,5$
- `{1,2,2,3,3}` → médiane corrigée = médiane de la série statistique = 2
- `{1,2,3,3,3}` → médiane corrigée = 2 (et non 3, puisque $3 = \max(1, 2, 3)$)
- `{3,3,3,3,3}` → médiane corrigée non définie (toutes les valeurs identiques).

Données d'apprentissage Les données d'apprentissage utilisées concernent la classification de trois espèces d'iris (*cf.* fichier `iris.txt`). À partir de là on construira un arbre de décision pour déterminer si oui ou non un iris est de l'espèce *« versicolor »* (on pourra faire de même pour l'espèce *« setosa »* et pour l'espèce *« virginica »*).

Le fichier `iris.txt` est formaté de la manière suivante :
- la première ligne contient deux entiers séparés d'un espace indiquant respectivement le nombre de lignes et le nombre de colonnes à lire.
- chaque ligne suivante correspond à une observation d'un iris ; elle est composée de 5 réels séparés d'un espace :

— le premier réel est la valeur (la classe) de la variable Y à prédire, avec valeur 1 ⇔ iris setosa, 2 ⇔ iris versicolor et 3 ⇔ iris virginica.
— les réels suivants constituent les valeurs pour les différentes variables d'observation X_i, avec X_1 ⇔ longueur des sépales, X_2 ⇔ largeur des sépales, X_3 ⇔ longueur des pétales et X_4 ⇔ largeur des pétales.

Algorithme de construction d'un arbre de décision L'idée est de diviser récursivement et le plus efficacement possible les individus de l'échantillon d'apprentissage, jusqu'à ce que l'on obtienne des sous-échantillons ne contenant (presque) que des individus appartenant tous ou non à la classe à prédire.

L'algorithme fonctionne de la manière suivante en trois étapes en partant du nœud racine :

0. étape d'initialisation, où la racine de l'arbre de décision a pour échantillon l'intégralité de l'échantillon d'apprentissage.
1. **tester** si l'échantillon peut être divisé. Un échantillon peut être divisé si :
   - la hauteur maximale (∗) de l'arbre de décision n'a pas été atteinte, et
   - le nombre d'individus n'est pas inférieur au nombre minimal possible (∗), et
   - la précision de l'échantillon n'est ni inférieure strict au seuil minimal, ni supérieure strict au seuil maximal de précision ciblée (∗).

   ↯ la précision est le pourcentage (réel $\in [0, 1]$) d'individus ayant pour valeur de Y celle à prédire (∗).
2. si oui, déterminer quelle variable d'observation X_i offre la **meilleure division**, c'est à dire où
   - en considérant l'ensemble de tous les individus dont la valeur de X_i est inférieure ou égale à la médiane corrigée de X_i (on appelle cet ensemble le sous-échantillon « de gauche »),
   - en considérant l'ensemble de tous les individus dont la valeur de X_i est strictement supérieure à la médiane corrigée de X_i (on appelle cet ensemble le sous-échantillon « de droite »),
   - la précision est maximale [1] pour l'un de ces deux sous-échantillons.

   ✎ si non, condition d'arrêt (pas de division de cet échantillon).
3. **associer** à la gauche du nœud courant un nouveau nœud ayant pour échantillon le sous-échantillon « de gauche » défini à l'étape 2, et associer à la droite du nœud courant un nouveau nœud ayant pour échantillon le sous-échantillon « de droite » défini à l'étape 2.
4. pour chacun des nœuds fils créés en étape 3, revenir à l'étape 1.

(∗) la valeur à prédire de Y, les seuils minimal et maximal de précision à atteindre, la hauteur maximale de l'arbre, et le nombre minimal d'individus par (sous-)échantillons sont fournis en paramètres du programme.

2 Travail à réaliser

2.1 Programme

Créer un programme `C`

1. par rapport à la précision qui pourrait être obtenue dans un des sous-échantillons théoriques de gauche ou de droite en utilisant une autre variable X_i.

— qui commence par demander à l'utilisateur la valeur de Y à prédire, les seuils minimal et maximal de précision à atteindre, le nombre minimal d'individus par échantillons et la taille maximale de l'arbre, puis qui **crée l'arbre de décision**.
  ✎ Exemple de paramètres du programme :
  $Y = 2$ (iris *versicolor*), seuil min = 10%, seuil max = 90%, nombre minimal d'individus par échantillon = 10% de la taille de l'échantillon d'apprentissage de départ, et hauteur maximal = 2 x nombre de variables d'apprentissage.

— qui ensuite affiche un **menu** proposant les fonctionnalités suivantes :

1. afficher la hauteur de l'arbre
2. afficher la largeur de l'arbre, c'est à dire le nombre de feuilles.
3. afficher l'arbre sous forme arborescente (comme vu en TD 6)
   - où pour chaque nœud sera affiché la précision par rapport à la valeur de Y à prédire, le nombre d'individus de l'échantillon, et comment on y est arrivé (uniquement depuis le nœud parent)
   - Exemple de "comment" : $X_1 \leq 5.8$
4. afficher les feuilles
   - c'est à dire afficher la précision des feuilles, leur nombre d'individus, et le chemin depuis la racine pour y accéder.
   - Exemple de chemin : $X_1 \leq 5.8$ puis $X_2 > 3.1$ puis ...

   ✎ Affichage du chemin éventuellement « à l'envers », mais l'indiquer !
5. prédire
   - l'utilisateur doit saisir les différentes valeurs des variables X_i d'un nouvel individu.
   - le programme doit afficher la précision associé à ce nouvel individu pour que sa valeur de Y soit celle pour laquelle l'arbre de décision a été créé (éventuellement en précisant les différentes décisions prises, *i.e.* le chemin).

   ✎ le programme se servira donc simplement de l'arbre de décision créé.

Contraintes

Une attention particulière devra être portée sur la qualité du code :
— découpage en (sous-)fonctions ⇒ donc pas tout dans le `main` !
— noms des variables et fonctions explicites
— organisation adaptée du code (fichiers `.c` / `.h`)
— commentaires à bon escient

Seules les bibliothèques suivantes peuvent être utilisées : `<stdlib.h>`, `<stdbool.h>`, `<string.h>` et `<stdio.h>`.

Aides

AIDE 1 : sur Moodle est disponible en `C` les fichiers `donnees.h` et `donnees.c` permettant de manipuler une matrice d'entiers, de la charger depuis un fichier texte formaté comme le fichier `iris.txt`, et de la détruire.

AIDE 2 : en Annexe est présentée la construction automatique de l'arbre de décision basé sur l'échantillon de départ en Figure 2 (avec que des valeurs réelles [2]) tel que la valeur de Y à prédire soit 1 (*i.e.* malade), le nombre minimal d'individus par échantillon (avant division) soit 2.

2. pour Y : $0 \Leftrightarrow$ pas malade, $1 \Leftrightarrow$ malade ; pour X_2 : $0 \Leftrightarrow$ pas gorge irritée, $1 \Leftrightarrow$ gorge irritée.

AIDE 3 : les informations à stocker dans chaque nœud sont :
- l'accès aux données (le même pour tous les nœuds).
- la valeur de Y à préciser (la même pour tous les nœuds).
- le critère de division qui a été fait pour arriver à ce nœud (depuis le nœud parent) ; c'est à dire (i) la variable X_i (index de colonne dans les données), (ii) la médiane corrigée, ainsi que (iii) le test d'inégalité ($\leq$ ou $>$) qui ont été utilisés.
- la liste des individus de l'échantillon (*i.e.* la liste des index de ligne dans les données).
- la précision.
- les accès au nœud parent, au fils gauche et au fils droit.

2.2 Mini-rapport

Un mini-rapport de 4 pages maximum est à rendre avec le code.

Doivent y figurer :
- les nom et prénom de l'élève et le groupe de TD ;
- une explication en français du/des type/s choisi/s pour manipuler l'arbre de décision ;
- une explication en français de comment est construit automatiquement l'arbre de décision ;
- une explication en français de comment est codée chaque fonctionnalité proposée dans le menu (en indiquant le numéro de la fonctionnalité dans le menu). Un petit schéma peut éventuellement venir compléter l'explication.
- et enfin quelle précision donnez-vous pour prédire si ces nouveaux individus sont des *iris versicolor* ($Y = 2$) ?

	précision pour Y=2	X_1	X_2	X_3	X_4
1	?	7.7	3.0	6.1	2.3
2	?	6.2	2.8	4.8	1.8
3	?	5.5	2.5	4.0	1.3
4	?	6.7	3.3	5.7	2.5
5	?	6.0	2.2	5.0	1.5
6	?	6.0	2.7	5.1	1.6
7	?	5.7	2.6	3.5	1.0
8	?	5.8	2.6	4.0	1.2
9	?	5.1	3.4	1.5	0.2
10	?	5.4	3.9	1.3	0.4

Annexe

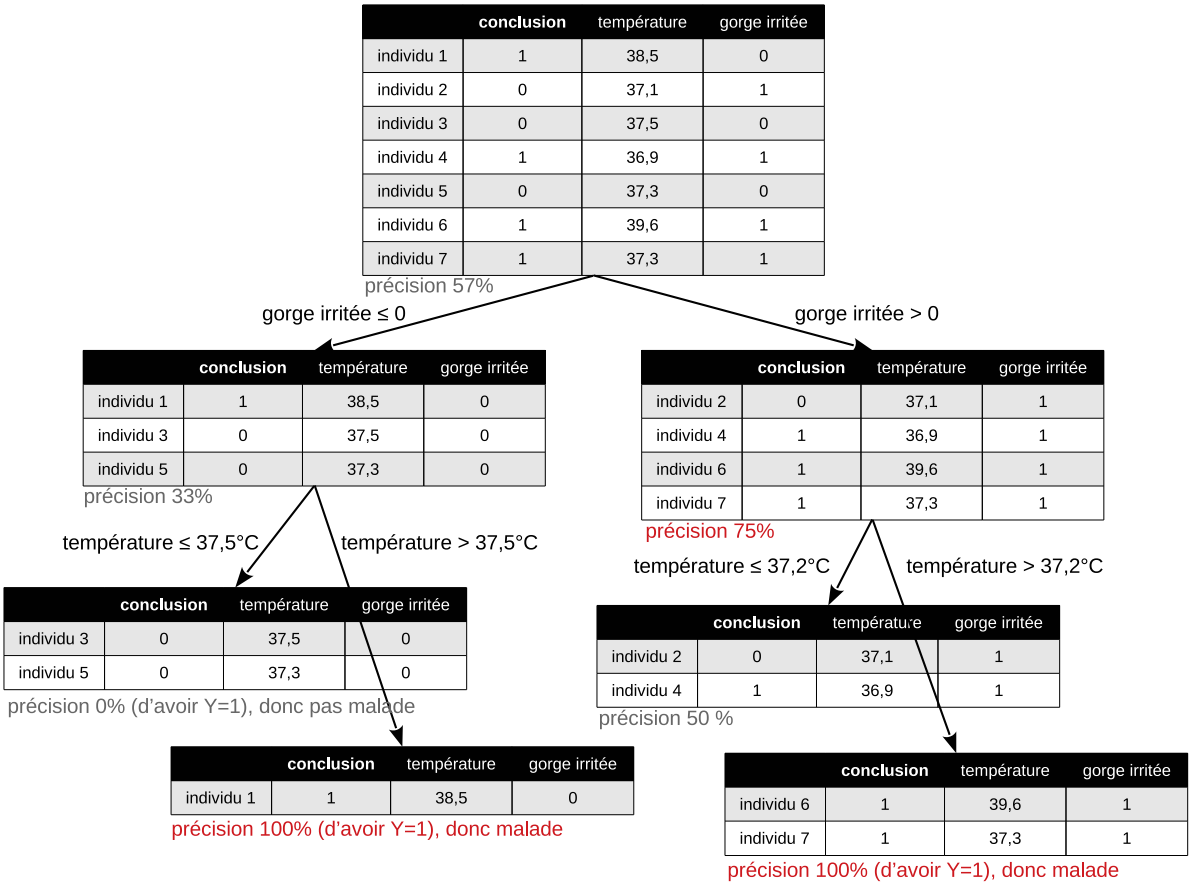


Figure 4 – Construction **automatique** de l'arbre de décision pour prédire $Y = 1$ (malade), en divisant selon la médiane corrigée sur le "meilleur" X_i (meilleure précision dans une des deux divisions), avec en sous-arbre de gauche les individus dont la valeur de X_i est inférieure ou égale à la médiane et en sous-arbre de droite lorsque la valeur est strictement supérieure.