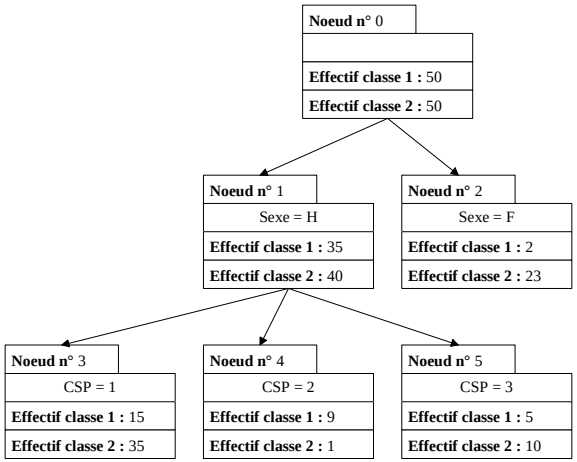
Département Informatique et Statistique, Faculté de SEG, Université Lumière Lyon 2
L3 IDS – Année 2011-2012

C – Sujet de dossier

J. Darmont (*http://eric.univ-lyon2.fr/~jdarmont/*), 03/04/12

On souhaite stocker en mémoire et manipuler un arbre de décision du type suivant. À partir d'une population de départ dont chaque individu appartient à une classe distincte (par exemple, riche, pauvre ou classe moyenne), on obtient des partitions en fonction des valeurs de variables descriptives de la population (Sexe, CSP…).



Un tel arbre permet de dériver des règles de décision telles que :

- Si Sexe = H et CSP = 1 alors p(classe 2) = 70 % (35/50) ;
- Si Sexe = H et CSP = 2 alors p(classe 1) = 90 % (9/10) ;
- Si Sexe = H et CSP = 3 alors p(classe 2) = 67 % (10/5) ;
- Si Sexe = F alors p(classe 2) = 92 % (23/25) ;

où p(classe *i*) est la probabilité qu'un individu d'un nœud défini par des critères données (par exemple, Sexe = H et CSP = 1) appartienne à la classe *i*.

L'objectif de ce dossier n'est pas de construire un arbre de décision de manière automatique, mais de permettre à l'utilisateur de le faire de manière interactive. Il faudra donc lui permettre de saisir les nœuds de l'arbre les uns après les autres et de les relier entre eux au fur et à mesure de la saisie. Le nombre de nœuds n'est ni connu, ni limité. En revanche, pour simplifier, on supposera que les individus ne peuvent appartenir qu'à deux classes, comme cela est représenté sur le schéma ci-dessus.

De plus, le programme C devra permettre de vérifier *a posteriori* que les effectifs saisis par l'utilisateur sont corrects, c'est-à-dire que la somme des effectifs totaux de tous les sous-nœuds d'un nœud donné est bien égale à la somme des effectifs de ce nœud (vérification à effectuer pour tous les nœuds de l'arbre). Par exemple, dans l'arbre ci-dessus, il faut que :

(Nœud1_EffectifClasse1 (35) + Nœud1_EffectifClasse2 (40))	75
+	
(Nœud2_EffectifClasse1 (2) + Nœud2_EffectifClasse2 (23))	25
=	
(Nœud0_EffectifClasse1 (50) + Nœud0_EffectifClasse2 (50))	100

Finalement, l'application devra permettre d'afficher toutes les règles de décision correspondant à l'arbre saisi.

Question subsidiaire : Comment passer de deux à N classes (N non connu *a priori* et non limité) ? Proposer une solution, mais ne pas l'implémenter.

Travail par binômes (pas de groupe de trois).
Code C commenté à rendre par courriel à
jerome.darmont@univ-lyon2.fr
au plus tard le lundi 30 avril 2012.