

APPRENTISSAGE D'ARBRES DE DÉCISION OPTIMAUX GRÂCE À LA PROGRAMMATION PAR CONTRAINTES

JFPC21

Hélène Verhaeghe¹, Siegfried Nijssen¹, Gilles Pesant², Claude-Guy Quimper³, and Pierre Schaus¹

Juin 2021

¹ICTEAM, UCLouvain, Place Sainte Barbe 2, 1348 Louvain-la-Neuve, Belgium, {*firstname.lastname*}@uclouvain.be

²Polytechnique Montréal, Montréal, Canada, *gilles.pesant@polymtl.ca*

³Université Laval, Québec, Canada, *claudc* — *guy.quimper@ift.ulaval.ca*

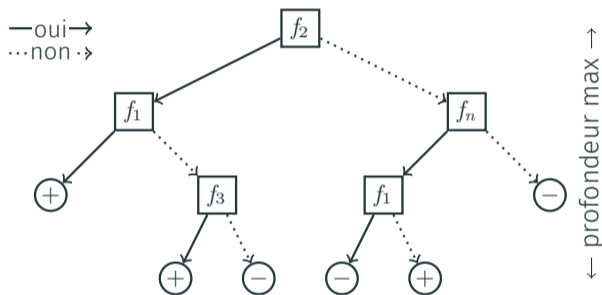


Base de données

f_1	f_2	f_3	\dots	f_n	c
1	0	1	\dots	1	+
0	1	0	\dots	1	-
1	1	0	\dots	0	+
0	0	0	\dots	0	+
1	0	0	\dots	0	+
0	1	1	\dots	1	-
1	1	1	\dots	0	-
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots
1	1	1	\dots	1	+

Base de données

f_1	f_2	f_3	...	f_n	c
1	0	1	...	1	+
0	1	0	...	1	-
1	1	0	...	0	+
0	0	0	...	0	+
1	0	0	...	0	+
0	1	1	...	1	-
1	1	1	...	0	-
⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	...	1	+



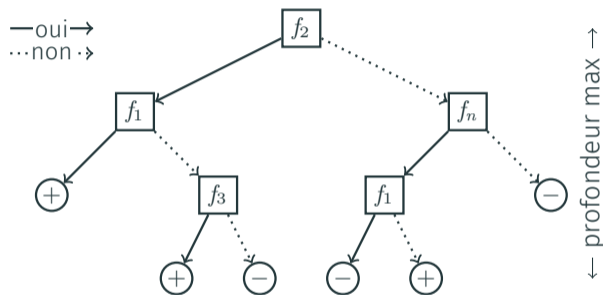
$$\min \sum (pred(i) - c(i))$$

Base de données

f_1	f_2	f_3	...	f_n	c
1	0	1	...	1	+
0	1	0	...	1	-
1	1	0	...	0	+
0	0	0	...	0	+
1	0	0	...	0	+
0	1	1	...	1	-
1	1	1	...	0	-
⋮	⋮	⋮	⋱	⋮	⋮
1	1	1	...	1	+

Nouvelle entrée

0	0	1	...	0	?
---	---	---	-----	---	---



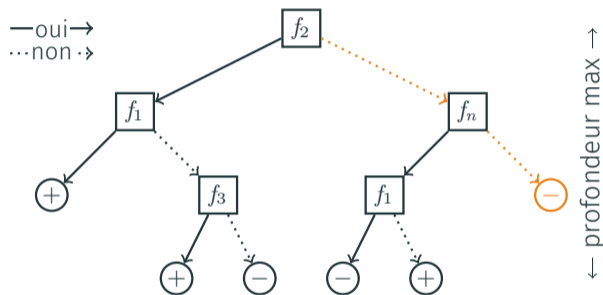
$$\min \sum (pred(i) - c(i))$$

Base de données

f_1	f_2	f_3	...	f_n	c
1	0	1	...	1	+
0	1	0	...	1	-
1	1	0	...	0	+
0	0	0	...	0	+
1	0	0	...	0	+
0	1	1	...	1	-
1	1	1	...	0	-
⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	...	1	+

Nouvelle entrée

0	0	1	...	0	-
---	---	---	-----	---	---



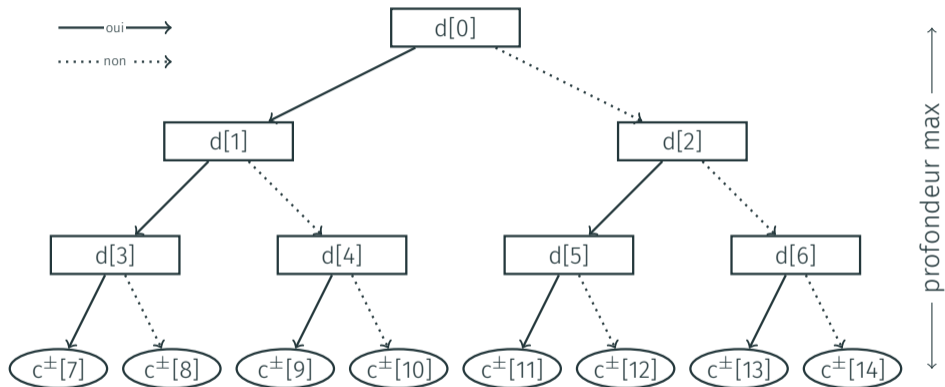
$$\min \sum (pred(i) - c(i))$$

Méthodes gloutonnes :

- ✓ construction facile
- ✗ difficulté d'imposer des contraintes additionnelles
- ✗ arbre potentiellement inutilement complexe

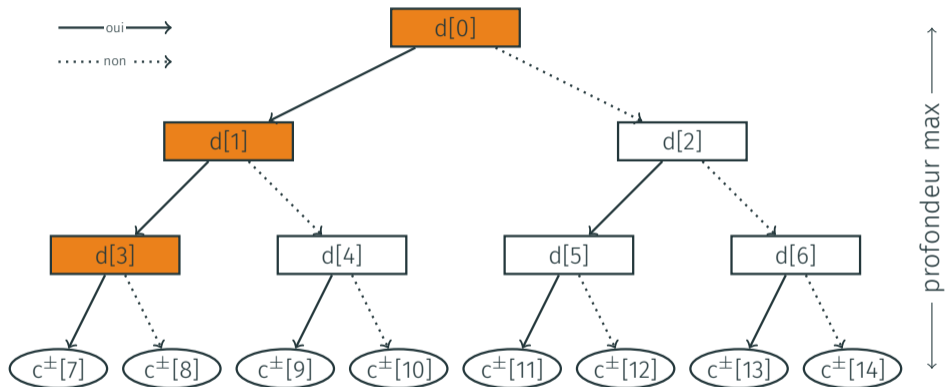
- Mining optimal decision trees from itemset lattices, Nijssen, S., Fromont, E., 2007
- Minimising decision tree size as combinatorial optimisation, Bessiere, C., Hebrard, E., O'Sullivan, B., 2009
- Optimal constraint-based decision tree induction from itemset lattices, Nijssen, S., Fromont, É., 2010
- **Optimal classification trees**, Bertsimas, D., Dunn, J., 2017
- Learning optimal decision trees with sat, Narodytska, N., Ignatiev, A., Pereira, F., Marques-Silva, J., RAS, I., 2018
- Learning optimal and fair decision trees for non-discriminative decision-making, Aghaei, S., Azizi, M.J., Vayanos, P., 2019
- Learning optimal classification trees using a binary linear program formulation, Verwer, S., Zhang, Y., 2019

MODÈLE PPC



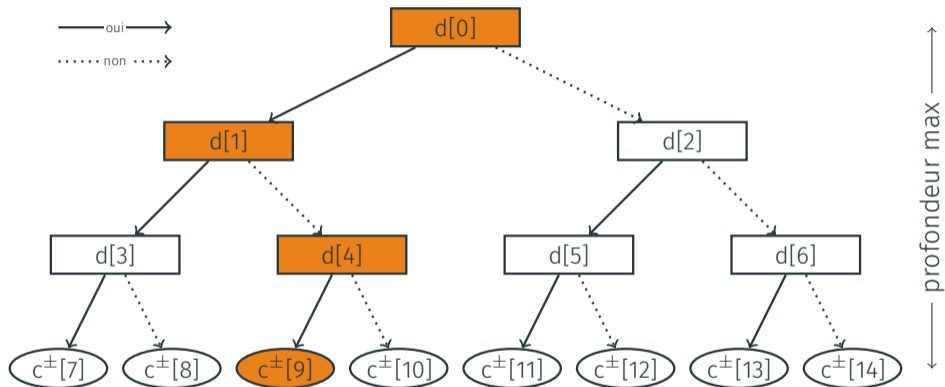
$$\text{dom}(d[i]) = \{0, 1, \dots, n\}$$

$$\text{dom}(c[i]) = \{0, \dots, N\}$$



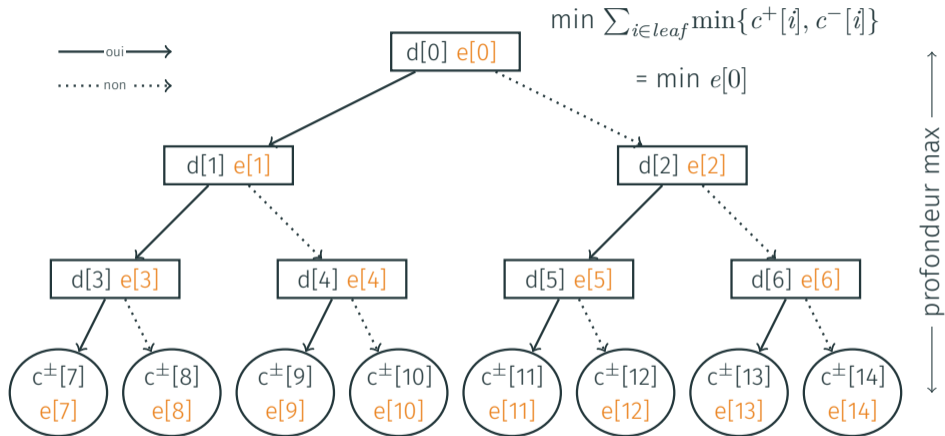
$$\text{dom}(d[i]) = \{0, 1, \dots, n\}$$

$$\text{dom}(c[i]) = \{0, \dots, N\}$$



$Coversize(\{d[0], d[4]\}, \{d[1]\}, c^+[9])$

$Coversize(\{d[0], d[4]\}, \{d[1]\}, c^-[9])$

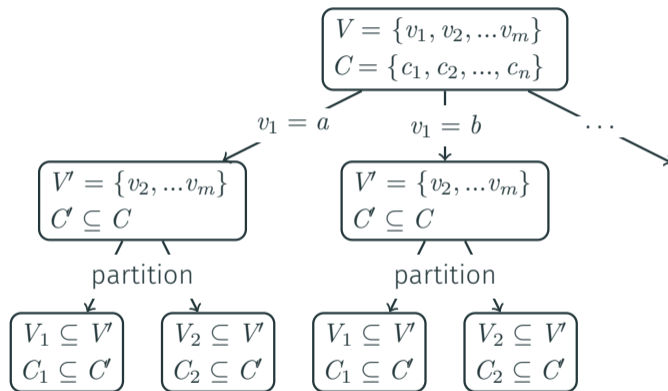


$$\text{dom}(d[i]) = \{0, 1, \dots, n\}$$

$$\text{dom}(c[i]) = \{0, \dots, N\}$$

$$\text{dom}(e[i]) = \{0, \dots, N\}$$

RECHERCHE

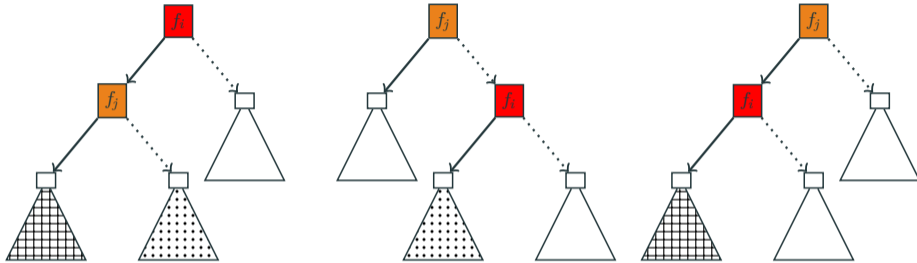


noeuds OU

SOL = SOL₁ ou SOL₂ ou ...

noeuds ET

SOL = SOL₁ et SOL₂ et ...



	oui	non	hash
			f_i, f_j^-
			$f_i - f_j$

RESULTATS

	$N_{\min} = 1$			$N_{\min} = 5$			
	DL8	BinOCT	CP	DL8	CP	CP-c	CP-m
Optimalité prouvée	49(64%)	13(17%)	57(75%)	54(71%)	56(74%)	56(74%)	58(76%)
Meilleure solution trouvée	49(64%)	21(28%)	76(100%)	54(71%)	74(97%)	74(97%)	70(92%)
Le plus rapide	23(30%)	11(14%)	49(64%)	28(37%)	40(53%)	33(43%)	22(29%)
Dépasse le temps alloué	27(36%)	63(83%)	19(25%)	22(29%)	21(28%)	21(28%)	19(25%)

23 instances, profondeur de 2 à 5, 10 min limite de temps

DL8 : Approche en programmation dynamique basée sur le minage d'ensemble d'objets fréquents

BinOCT : Approche basée sur un modèle MIP qui tourne avec CPLEX

Pour résumer

- méthode efficace
- basée sur la ppc
- exploitation de la structure du probleme
- meilleur solution actuelle

Pour aller plus loin

- arbres de décision multi-classe
- caractéristiques continues grâce à la binarisation
- autre fonctions de coût basées sur une somme
- ...

Merci pour votre attention!

Des questions?

Pour plus d'information, l'article journal est disponible

