# Regroupements pour la construction d'arbres de classification

Zacharie Ales[1,2], Valentine Huré[1], Amélie Lambert[1]

[1] CNAM, Paris, France
valentine.hure@lecnam.net,amelie.lambert@cnam.fr
[2] UMA, ENSTA Paris, Institut Polytechnique de Paris, Palaiseau, France.
zacharie.ales@ensta-paris.fr

**Mots-clés** : *classification supervisée, arbre de classification, clustering, PLNE.*

L'intérêt croissant de l'interprétabilité pour les algorithmes de Machine Learning rend d'autant plus pertinent l'utilisation des arbres de décision comme modèles de classification supervisée. Les algorithmes heuristiques (tels que CART [2]) sont encore très utilisés mais de plus en plus de méthodes exactes [1, 3, 4] se développent afin d'obtenir des arbres fournissant de meilleures performances. Le plus grand challenge pour ces approches est le passage à l'échelle. C'est pourquoi nous proposons plusieurs algorithmes de regroupements des données visant à réduire la taille des formulations tout en garantissant l'optimalité de la solution obtenue.

## 1 Regroupement de données : définition et propriétés

Soit $(P)$ un modèle d'optimisation visant à construire des arbres de décision. $(P)$ a classiquement deux familles de variables : celles définissant la structure de l'arbre et celles suivant le chemin de chaque donnée. Ainsi, afin de réduire significativement le nombre de ces variables, nous proposons de fusionner les données en groupes ayant de grandes chances de suivre le même chemin dans l'arbre. Nous présentons en Figure 1, un exemple de regroupement de données. Les points en couleurs sont les *représentants* de chaque groupe. Ainsi, nous définissons un modèle réduit, noté $(P_R)$, dans lequel ces représentant constituent les nouvelles données considérées.

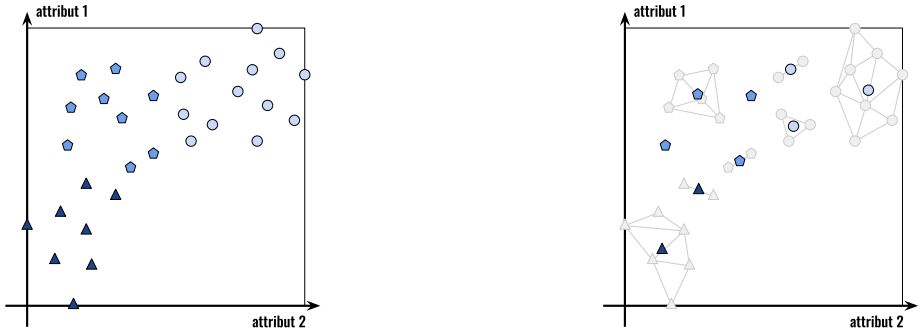


FIG. 1 – Jeu initial : 30 données, 3 classes (gauche), et jeu fusionné : 9 données (droite)

Nous introduisons maintenant quatre propriétés structurelles des regroupements afin de pouvoir comparer les regroupements calculés par nos algorithmes :

- **Homogénéité** : un regroupement est dit homogène si chaque groupe n'est constitué que de données de même classe.
- **Exclusion** : il y a exclusion au sein d'un regroupement si aucune des enveloppes convexes des groupes ne s'intersectent.
- **Cohérence des données** : toute donnée d'un groupe est de même classe que son plus proche voisin.
- **Cohérence des groupes** : toute donnée d'un groupe est dans un groupe de même classe que le groupe de son plus proche voisin.

## 2 Algorithmes de regroupement et méthodes exactes

Nous proposons plusieurs algorithmes de regroupements et caractérisons s'ils sont homogènes, avec exclusion ou cohérents. Parmi eux, certains sont des algorithmes de clustering hiérarchiques qui fusionnent itérativement des clusters. Les autres sont de nouvelles heuristiques visant à respecter nos trois propriétés.

Pour un regroupement donné, l'arbre obtenu par la résolution du modèle $(P_R)$ n'est généralement pas optimal pour les données d'origine. C'est pourquoi nous considérons un algorithme itératif. À chaque itération $k$, nous résolvons $(P_R)$ pour obtenir un arbre $\mathcal{A}_k$. Nous identifions ensuite les groupes dont les données ne suivent pas toutes le même chemin dans $\mathcal{A}_k$ (on dit qu'ils sont *intersectés* par $\mathcal{A}_k$). Si un groupe est intersecté, nous le partitionnons en 2 sous-groupes. Nous démontrons que cet algorithme converge vers une solution optimale du jeu de données initial (Propriété 1).

**Propriété 1** *Etant donné un regroupement $R$, une solution optimale de $(P_R)$ qui n'intersecte aucun groupe de $R$ est optimale pour $(P)$.*

**Données :** Un jeu de données et un regroupement associé $R$
**Résultat :** Un arbre $\mathcal{A}$
$R_0 \leftarrow R$ ;
$\mathcal{A}_0 \leftarrow$ solution de $(P_R)$ appliqué à $R_0$ ;
**Tant que** $\mathcal{A}_k$ *intersecte* $R_k$ **faire**
  Diviser les groupements intersectés par $\mathcal{A}_k$ pour obtenir $R_{k+1}$ ;
  $\mathcal{A}_{k+1} \leftarrow$ solution de $(P_R)$ appliqué à $R_{k+1}$ ;
**fin**

**Algorithme 1 :** Algorithme utilisant $(P_R)$ et fournissant un arbre optimal

## 3 Premiers résultats numériques et perspectives

Le premier intérêt de notre approche est que la formulation $(P_R)$ contient significativement moins de variables que $(P)$ car le nombre de variables est proportionnel au nombre de données et on ne considère qu'une donnée par groupe dans $(P_R)$. Ainsi, nos premiers résultats expérimentaux sont encourageants et montrent, sur les jeux de données considérés, que ce nouvel algorithme permet d'accélérer la résolution de $(P)$ par rapport à sa soumission à un solveur standard. De plus, un avantage significatif du regroupement de données est de pouvoir résoudre de façon exacte des instances de la littérature de tailles plus importantes que celles résolues jusqu'alors. Notons également qu'il est possible de borner le nombre d'itérations de l'algorithme pour obtenir une solution heuristique de $(P)$. Enfin, notre méthode de regroupement de données est très générale et peut s'appliquer à d'autres modèles d'apprentissage supervisé.

## Références


[1] D. Bertsimas and J. Dunn. Optimal classification trees. *Machine Learning* 106(7) :1039–1082, 2017.

[2] L. Breiman, J.H. Friedman, R. Olshen, and C.J. Stone. Classification and regression trees. *Routledge*, 2017.

[3] Z. Alès, A. Lambert, and V. Huré. New optimization models for optimal classification trees *preprint HAL* https://hal.archives-ouvertes.fr/hal-03865931, 2022.

[4] S. Verwer and Y. Zhang Learning Optimal Classification Trees Using a Binary Linear Program Formulation. *AAAI*, 33(01) :1625–1632, 2019.