Interprétabilité, forêts aléatoires, et ensemble de règles.

Clément Bénard [1,*] & Sébastien Da Veiga [2] & Erwan Scornet [3] & Gérard Biau [4]

[1,2] *Safran Tech, Modelling & Simulation, Rue des Jeunes Bois, Châteaufort, 78114 Magny-Les-Hameaux, France*
[3] *CMAP, Ecole Polytechnique, route de Saclay 91128 Palaiseau, France*
[4,1] *Sorbonne Université - LPSM, 4 place Jussieu, 75005 Paris, France*
[*] *clement.benard@safrangroup.com*

Résumé. Les forêts aléatoires introduites par Breiman (2001) sont des algorithmes de régression et de classification parmi les plus performants. Cependant, le grand nombre d'opérations nécessaires pour effectuer une prédiction leur confère un aspect "boîte noire". A l'opposé, les arbres de décisions ont une structure très simple mais instable, et une prédictivité limitée. Ces caractéristiques limitent fortement l'utilisation des arbres et des forêts pour certaines applications, par exemple l'analyse des processus de production dans l'industrie manufacturière. En effet, les décisions impactant des chaînes de production ont des conséquences lourdes, et ne peuvent reposer aveuglement sur des modélisations aléatoires. Les modèles se doivent d'être interprétables, c'est à dire à minima, simples, stables et prédictifs. Un troisième type de modèles, les ensembles de règles, présentent un compromis intéressant avec une structure simple et une capacité de prédiction comparable aux forêts, mais se caractérisent aussi par une certaine instabilité. Nous proposons un nouvel algorithme de classification d'ensemble de règles, extraites d'une forêt aléatoire. Pour les problèmes avec des interactions d'ordre faible, la méthode hérite d'une capacité de prédiction approchant celle des forêts, de la simplicité des arbres de décision, et d'une structure stabilisée. L'algorithme proposé présente à la fois des garanties théoriques asymptotiques, et de bonnes performances sur des données réelles.

Mots-clés. Forêts aléatoires, interprétabilité, ensemble de régles, arbre de décision.

Abstract. Random forests are state-of-the-art regression and classification methods, introduced by Breiman (2001). However, they are often qualified as "black-boxes" because of the high number of operations involved in the prediction mechanism. On the other hand, decision trees exhibit a simple structure, but are also instable, and have a limited predictive accuracy. These properties are a strong limitation to the practical use of trees and forests for specific applications, for example, the analysis of production processes in the manufacturing industry. Indeed, any decision impacting a production process has long-term and heavy consequences, and therefore cannot simply rely on a blind stochastic modeling. Models have to be interpretable, i.e., at least simple, stable and predictive. A third type of models, rule ensembles, show an interesting tradeoff between a simple structure and a predictive accuracy similar to random forests. These models are also

quite unstable. We propose a new rule ensemble classification algorithm, extracted from a random forest. For problems with low order interactions, our method inherits a predictive accuracy close to random forests, the simplicity of decision trees and a stable structure. The algorithm comes with asymptotics guarantees, and shows good practical performance on real datasets.

Keywords. Random forests, interpretability, rule ensemble, decision tree.

1 Introduction

Dans l'industrie manufacturière, les processus de production mettent en jeu des phénomènes physiques et chimiques complexes, dont le contrôle et l'efficacité sont d'une importance critique. Les algorithmes d'apprentissage statistique permettent de modéliser efficacement ces comportements non linéaires, caractérisés par des interactions d'ordre faible. Cependant, les décisions impactant des chaînes de production ont des conséquences lourdes, et ne peuvent reposer aveuglement sur des modélisations aléatoires. Une compréhension profonde des phénomènes physiques en jeu est nécessaire. Les modèles doivent donc être interprétables, c'est à dire expliciter comment les entrées et la sortie sont liées, afin de guider l'analyse physique.
Il n'y a pas à ce jour de consensus sur la définition d'interprétabilité dans la communauté scientifique (e.g. Lipton, 2016; Doshi-Velez and Kim, 2017). Cependant, dans la continuité de Yu and Kumbier (2019), on peut définir des prérequis à l'interprétabilité à travers le tryptique : simplicité (e.g. Lipton, 2016; Doshi-Velez and Kim, 2017), stabilité (Yu, 2013) et prédictivité (Breiman et al., 2001).

Les arbres de décision (Breiman et al., 1984), une classe d'algorithme d'apprentissage supervisé, sont capables de modéliser des phénomènes non linéaires tout en ayant une structure simple, et sont donc de bons candidats lorsque l'interprétabilité est requise. Cependant les arbres sont particulièrement instables, ce qui limite fortement leur utilisation opérationnelle. Les forêts aléatoires, developpées par Breiman (2001), éliminent l'instabilité en agrégeant un grand nombre d'arbres randomisés. Le modèle obtenu est stable et prédictif, mais devient alors une boîte noire.

Un autre type d'algorithme peut modéliser des tendances non linéaires tout en conservant une structure simple : les ensembles de règles. Une règle est un estimateur constant par morceaux qui se lit simplement "si *conditions sur les entrées*, alors *réponse*, sinon *réponse par défaut*." Une multitude d'algorithmes a été dévelopée, parmi lesquelles on peut citer SLIPPER Cohen and Singer (1999), RuleFit (Friedman et al., 2008), Node harvest (Meinshausen, 2010), et BRL (Bayesian Rule Lists, Letham et al., 2015). Malgré leur simplicité et leur excellente capacité prédictive, ces approches sont instables comme les arbres de décision.

Nous proposons un nouvel algorithme de classification, fondé sur l'extraction d'un ensemble de règles d'une forêt aléatoire. La méthode hérite de la capacité de prédiction des forêts, de la structure simple d'un arbre, tout en ayant une structure stable, pour des problèmes impliquant des effets d'intéractions d'ordre faible.

2 Description de l'algorithme

On se place dans le cadre habituel de la classification supervisée, où l'on dispose d'un échantillon i.i.d $\mathscr{D}_n = \{(\mathbf{X}_i, Y_i), i = 1, \ldots, n\}$. Chaque $(\mathbf{X}_i, Y_i)$ est distribué comme la paire générique $(\mathbf{X}, Y)$, où $\mathbf{X} = (X^{(1)}, \ldots, X^{(p)})$ est un vecteur aléatoire à valeurs dans $\mathbb{R}^p$, et $Y \in \{0, 1\}$ une sortie binaire. La distribution de $(\mathbf{X}, Y)$ est supposée inconnue, et est notée $\mathbb{P}_{\mathbf{X},Y}$. Pour $\mathbf{x} \in \mathbb{R}^p$, notre objectif est d'estimer $\eta(\mathbf{x}) = \mathbb{P}(Y = 1|\mathbf{X} = \mathbf{x})$ à partir de quelques règles simples et interprétables.

Forêt aléatoire La procédure se fonde sur la forêt aléatoire (Breiman, 2001), que l'on modifie légèrement : la profondeur des arbres est limitée à $d = 2$, et la recherche de la meilleure coupure à chaque noeud est restreinte aux q-quantiles empiriques des distributions marginales de chaque composante de $\mathbf{X}$, $\hat{q}_{n,r}^{(j)}$ avec $r \in \{1, ..., q\}$ et $j \in \{1, ..., p\}$.
La deuxième étape de l'algorithme est l'extraction des règles de cette forêt modifiée. Pour définir rigoureusement cette étape, on introduit une notation supplémentaire. Le chemin $\mathscr{P}$ de longueur $d = 1$ ou $d = 2$, conduisant de la racine d'un arbre à un noeud donné, est défini par la représentation symbolique,

$$\mathscr{P} = \left\{(j_k, r_k, s_k)\right\}_{k=1,\ldots,d},$$

où pour $k \in \{1, \ldots, d\}$, le triplet (j_k, r_k, s_k) décrit comment passer du niveau $(k-1)$ au niveau k, avec une coupure suivant la coordonnée $j_k \in \{1, \ldots, p\}$, l'indice $r_k \in \{1, \ldots, q-1\}$ du quantile correspondant, et $s_k = L$ pour le noeud de gauche, $s_k = R$ pour celui de droite. L'ensemble déterministe de tous les chemins possibles est noté Π. Un Θ-arbre aléatoire de profondeur 2 génère au plus 6 chemins, un pour chaque noeud et feuille terminale. Θ est une variable aléatoire, utilisée pour rééchantillonner les données et randomiser le choix des coupures dans la construction de l'arbre. Dans la suite, on note $T(\Theta, \mathscr{D}_n)$ la liste des chemins extraits, un sous-ensemble aléatoire de Π. Un exemple est donné sur la figure 1 pour $p = 2$.

Regle élémentaire Pour chaque chemin $\mathscr{P} \in \Pi$, on définit la règle élémentaire associée $\hat{g}_{n,\mathscr{P}}$ par

$$\forall \mathbf{x} \in \mathbb{R}^p, \quad \hat{g}_{n,\mathscr{P}}(\mathbf{x}) = \begin{cases} \frac{1}{N_n(\hat{H}_n(\mathscr{P}))} \sum_{i=1}^n Y_i \mathbb{1}_{\mathbf{X}_i \in \hat{H}_n(\mathscr{P})} & \text{si } \mathbf{x} \in \hat{H}_n(\mathscr{P}) \\ \frac{1}{n - N_n(\hat{H}_n(\mathscr{P}))} \sum_{i=1}^n Y_i \mathbb{1}_{\mathbf{X}_i \notin \hat{H}_n(\mathscr{P})} & \text{sinon,} \end{cases}$$

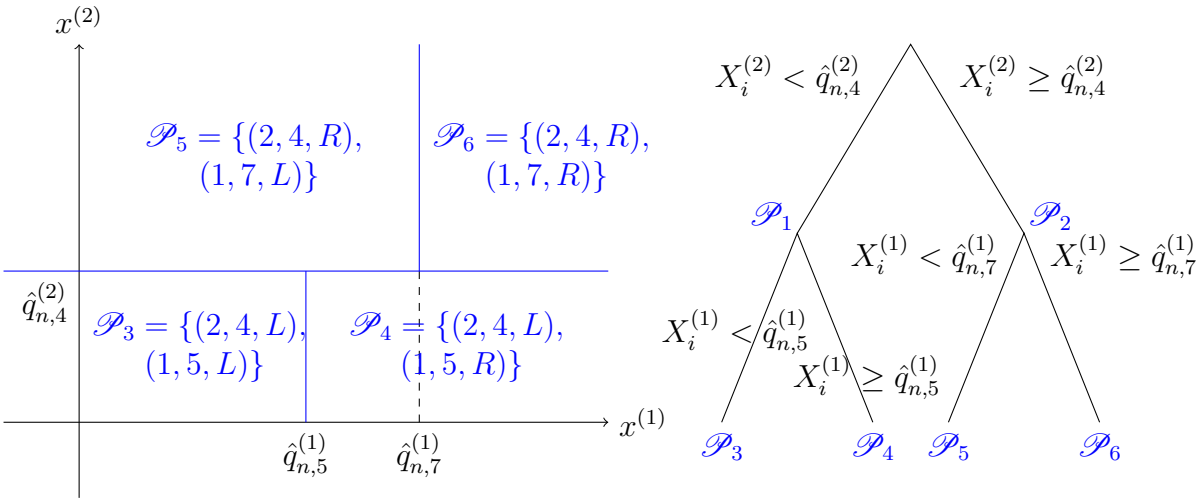

Figure 1: Exemple d'un noeud $\mathbb{R}^2$ partitionné par un arbre aléatoire de profondeur 2.

où $\hat{H}_n(\mathscr{P})$ est l'hyperrectangle (i.e. le noeud) associé au chemin $\mathscr{P}$ et à l'échantillon $\mathscr{D}_n$, et $N_n(\hat{H}_n(\mathscr{P}))$ est le nombre de points contenus dans $\hat{H}_n(\mathscr{P})$. (On utilise la convention $0/0 = 0$.) La règle élémentaire $\hat{g}_{n,\mathscr{P}}(\mathbf{x})$ est donc un estimateur de la probabilité que $\mathbf{x}$ soit de classe 1, selon que $\mathbf{x}$ appartienne à $\hat{H}_n(\mathscr{P})$ ou non.

Algorithme On peut générer un grand nombre de règles via une forêt aléatoire. On souhaite sélectionner celles qui représentent des tendances fortes entre les entrées et la sortie. On définit alors $p_{\mathscr{D}_n}(\mathscr{P}) = \mathbb{P}(\mathscr{P} \in T(\Theta, \mathscr{D}_n)|\mathscr{D}_n)$ la probabilité qu'un arbre aléatoire randomisé par Θ contienne le chemin $\mathscr{P}$. L'estimateur de Monte-Carlo associé $\hat{p}_{M,n}(\mathscr{P})$, calculé via la forêt aléatoire modifiée, s'exprime par

$$\hat{p}_{M,n}(\mathscr{P}) = \frac{1}{M}\sum_{\ell=1}^{M} \mathbb{1}_{\mathscr{P}\in T(\Theta_\ell,\mathscr{D}_n)}.$$

On extrait uniquement les chemins qui apparaissent dans la forêt avec une fréquence supérieure à $p_0 \in]0,1[$ (un hyper-paramètre de l'algorithme), soit l'ensemble $\hat{\mathscr{P}}_{M,n,p_0} = \{\mathscr{P} \in \Pi : \hat{p}_{M,n}(\mathscr{P}) > p_0\}$. (L'ensemble $\hat{\mathscr{P}}_{M,n,p_0}$ est post-traité pour supprimer la redondance entre les règles générées.)

Pour estimer $\eta(\mathbf{x}) = \mathbb{P}(Y = 1|\mathbf{X} = \mathbf{x})$, l'ensemble des règles sélectionnées est simplement moyenné

$$\hat{\eta}_{M,n,p_0}(\mathbf{x}) = \frac{1}{|\hat{\mathscr{P}}_{M,n,p_0}|}\sum_{\mathscr{P}\in\hat{\mathscr{P}}_{M,n,p_0}} \hat{g}_{n,\mathscr{P}}(\mathbf{x}).$$

La procédure de classification finale assigne la classe 1 à l'entrée $\mathbf{x}$ si $\hat{\eta}_{M,n,p_0}(\mathbf{x})$ est supérieur à un seuil donné, et la classe 0 sinon.

3 Propriétés théoriques

La construction de l'ensemble de règles repose essentiellement sur les estimateurs $\hat{p}_{M,n}(\mathscr{P})$, le Théorème 1 établit leur consistence. On définit le pendant théorique $T^\star(\Theta)$ de $T(\Theta, \mathscr{D}_n)$, la liste des chemins extraits de l'arbre théorique randomisé par Θ, où les coupures sont choisies en maximisant le critère CART théorique. Ainsi $T^\star(\Theta)$ ne dépend plus de $\mathscr{D}_n$ mais seulement de $\mathbb{P}_{\mathbf{X},Y}$. De même, on introduit $p^\star(\mathscr{P}) = \mathbb{P}(\mathscr{P} \in T^\star(\Theta))$ et $\mathscr{P}^\star_{p_0}$, les pendants théoriques de respectivement $p_{\mathscr{D}_n}(\mathscr{P})$ et $\hat{\mathscr{P}}_{M,n,p_0}$. On définit les hypothèses suivantes

(A1) Le nombre d'observations a_n, échantillonnées sans remise pour construire chaque arbre, satisfait $\lim_{n\to\infty} a_n = \infty$ and $\lim_{n\to\infty} \frac{a_n}{n} = 0$.

(A2) Le nombre d'arbres M_n satisfait $\lim_{n\to\infty} M_n = \infty$.

(A3) $\mathbf{X}$ admet une densité strictement positive f par rapport à la mesure de Lebesgue. Pour tout $j \in \{1, \ldots, p\}$, la densité marginale $f^{(j)}$ de $X^{(j)}$ est continue, bornée et strictement positive.

Theorem 1. *Si les hypothèses (A1)-(A3) sont vérifiées, pour tout* $\mathscr{P} \in \Pi$,

$$\lim_{n\to\infty} \hat{p}_{M_n,n}(\mathscr{P}) = p^\star(\mathscr{P}) \quad \textit{en probabilité.}$$

Les estimateurs $\hat{p}_{M_n,n}(\mathscr{P})$ étant la seule source d'aléa dans la sélection des chemins, on en déduit le corollaire suivant.

Corollary 1. *Si les hypothèses (A1)-(A3) sont vérifiées, alors si* $p_0 \in [0, \max_{\mathscr{P}\in\Pi} p^\star(\mathscr{P})) \setminus \{p^\star(\mathscr{P}) : \mathscr{P} \in \Pi\}$, *on a*

$$\lim_{n\to\infty} \mathbb{P}\big(\hat{\mathscr{P}}_{M_n,n,p_0} \neq \mathscr{P}^\star_{p_0}\big) = 0$$

4 Expériences

Les expériences sont conduites sur des jeux de données réelles de l'UCI repository (Asunsion and Newman 2007). On utilise $q = 10$ quantiles. Les performances de l'agorithme sont évaluées par validation croisée (10 folds), répétée 10 fois pour calculer les écarts types. La capacité prédictive est mesurée par 1-AUC, la stabilité par le nombre relatif de règles en commun entre 2 modèles de la validation croisée.

Comme l'illustre la figure 2, quelques dizaines de règles suffisent pour construire un modèle avec une capacité prédictive proche de celle des forêts aléatoires. Dans cet exemple, deux modèles différents construits lors d'une validation croisée ont environ 80% de règles en commun en moyenne.

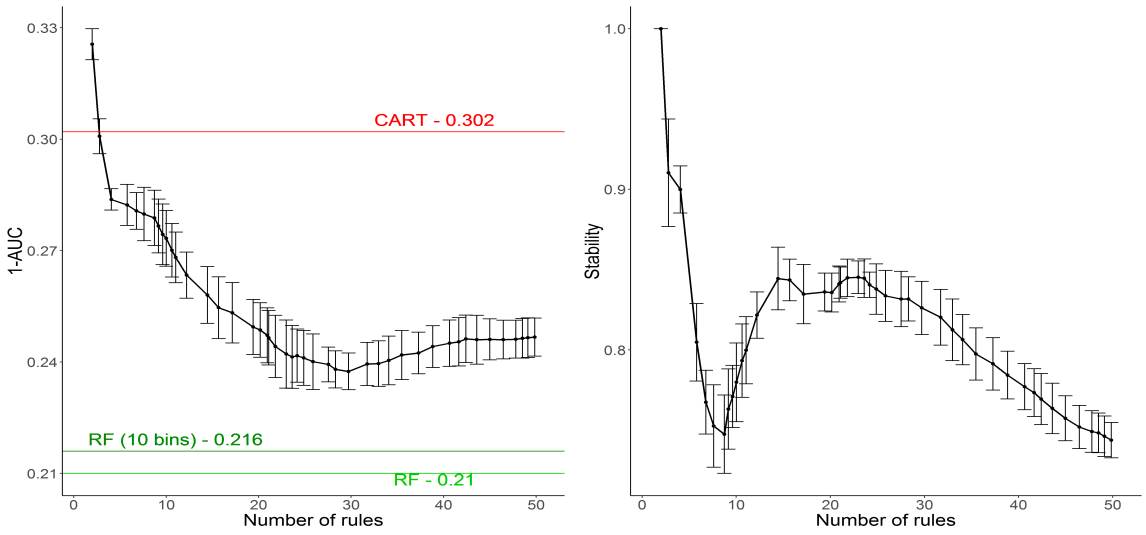

Figure 2: Capacité prédictive et stabilité du modèle en fonction du nombre de règles (p_0) pour le jeu de données "Credit German".

Bibliographie

L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. Classification and regression trees. *Wadsworth International Group*, 1984.

L. Breiman et al. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16:199–231, 2001.

W.W. Cohen and Y. Singer. A simple, fast, and effective rule learner. *AAAI/IAAI*, 99: 335–342, 1999.

F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

J.H. Friedman, B.E. Popescu, et al. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2:916–954, 2008.

B. Letham, C. Rudin, T.H. McCormick, D. Madigan, et al. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9:1350–1371, 2015.

Z.C. Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

N. Meinshausen. Node harvest. *The Annals of Applied Statistics*, pages 2049–2072, 2010.

B. Yu. Stability. *Bernoulli*, 19:1484–1500, 2013.

Bin Yu and Karl Kumbier. Three principles of data science: predictability, computability, and stability (pcs). *arXiv preprint arXiv:1901.08152*, 2019.