Les raisons majoritaires : des explications abductives pour les forêts aléatoires

Gilles Audemard*, Steve Bellart*
Louenas Bounia*, Frédéric Koriche*
Jean-Marie Lagniez*, Pierre Marquis***

*Univ. Artois, CNRS, CRIL, F-62300 Lens
** Institut universitaire de France
"nom"@cril.fr,
http://www.cril.univ-artois.fr/

Résumé. Les forêts aléatoires constituent un modèle d'apprentissage automatique efficace, ce qui explique qu'elles soient encore massivement utilisées aujourd'hui. S'il est assez facile de comprendre le fonctionnement d'un arbre de décision, il est beaucoup plus complexe d'interpréter la décision prise par une forêt aléatoire, car elle est typiquement issue d'un vote majoritaire entre de nombreux arbres. Nous examinons ici diverses définitions d'explication abductive du classement prédit par une forêt aléatoire. Nous nous intéressons au problème de leur génération (trouver une explication) et au problème de leur minimisation (trouver une explication parmi les plus courtes). Nous montrons notamment que les explications abductives irredondantes (ou raisons suffisantes) peuvent être difficiles à calculer pour les forêts aléatoires. Nous proposons à leur place les raisons majoritaires, des explications abductives en théorie moins concises mais que l'on peut calculer en temps polynomial.

1 Introduction

Les progrès remarquables réalisés durant les deux dernières décennies en matière d'apprentissage automatique ont conduit à utiliser maintes fois ces technologies. Ce faisant, des modèles de prédiction offrant de hauts degrés de précision ont été intégrés ou en cours d'intégration dans des applications issues de domaines variés. Mais la haute précision de ces modèles se fait souvent au détriment de leur interprétabilité, ce qui peut se révéler critique dans certains domaines. Dans le monde médical par exemple, si un classeur identifie un patient comme malade suite à une radiographie, le médecin doit pouvoir demander au classeur ce qui sur la radiographie réalisée explique le classement obtenu.

Dans cet article, nous nous concentrons sur les classements réalisés à partir du modèle des forêts aléatoires, un modèle d'apprentissage automatique bien connu et couramment utilisé, basé sur le vote majoritaire de plusieurs arbres de décision appris à partir d'instances tirées aléatoirement avec remise dans l'ensemble d'apprentissage considéré (Breiman, 2001). Ces forêts sont en pratique faciles à apprendre et elles offrent une bonne robustesse au bruit. C'est

pourquoi elles sont utilisées dans des domaines divers, comme la vision par ordinateur (Criminisi et Shotton, 2013), l'écologie (Cutler et al., 2007) ou encore pour le diagnostic médical (Azar et al., 2014).

En dépit de leur succès et des liens proches que ces deux modèles entretiennent, les décisions prises à partir de forêts aléatoires sont de loin plus difficiles à expliquer que celles obtenues à partir d'arbres de décision. Si, pour une instance à classer donnée, un arbre de décision peut fournir directement une raison de la décision prise comme l'unique chemin de l'arbre compatible avec l'instance, il n'existe pas de telles raisons natives pour une forêt aléatoire en raison du vote majoritaire réalisé, qui est un facteur clé de la robustesse du modèle.

1.1 Travaux existants

Ces dernières années, l'étude de l'interprétabilité des forêts aléatoires a connu un intérêt croissant comme l'attestent les papiers de Bénard et al. (2021), Choi et al. (2020), ou Izza et Marques-Silva (2021). Parmi les types d'explications possibles pour le classement d'une instance donnée figurent les *explications abductives*. Une *explication abductive* d'une instance $\mathbf{x}$ est une sous-ensemble des couples (attribut, valeur) de $\mathbf{x}$ tel que toute instance partageant ce sous-ensemble est classée de la même manière que $\mathbf{x}$. Les *raisons suffisantes* de $\mathbf{x}$ sont les explications abductives de $\mathbf{x}$ qui ne contiennent aucun couple (attribut, valeur) inutile (en d'autres termes, elles sont minimales pour l'inclusion ensembliste). Quand les attributs considérés sont booléens et qu'il n'y a que deux classes possibles, tout classeur correspond à une fonction booléenne f et une raison suffisante pour une instance $\mathbf{x}$ classée comme positive (resp. négative) par f est un *impliquant premier* t de f (resp. de $\neg f$) inclus dans l'ensemble des couples (attribut, valeur) de $\mathbf{x}$. Pour une instance $\mathbf{x}$, quand f est représentée par un arbre de décision, une raison suffisante de $\mathbf{x}$ étant donnée f est calculable en temps polynomial. Malheureusement, quand f est représentée par une forêt aléatoire, identifier une raison suffisante de $\mathbf{x}$ est un problème calculatoirement difficile. En pratique, la génération d'une telle raison peut être réalisée (sans garantie sur le temps de calcul requis pour l'obtenir) via l'utilisation d'un solveur MUS (Izza et Marques-Silva, 2021).

En parallèle des approches d'explication basées sur des connaissances du modèle, des explications « modèle agnostique » ont fait leur apparition avec notamment (Ribeiro et al., 2016) et l'approche LIME. Cette approche a pour but de déterminer une fonction linéaire des attributs approchant le comportement du modèle cible autour de l'instance étudiée. S'il est possible de calculer efficacement une explication abductive de l'instance considérée étant donnée cette fonction, il n'y a aucune garantie que l'explication produite soit réellement une explication abductive de l'instance étant donné le modèle de départ car elle est issue d'une approximation du modèle et non du modèle initial lui-même.

1.2 Notre contribution

Dans cet article, nous proposons deux nouvelles notions d'explication abductive pour les forêts aléatoires. Premièrement, nous étendons les *raisons directes* des arbres de décision aux forêts (une première forme d'explication abductive, naïve mais simple à calculer). Ensuite, nous présentons les *raisons majoritaires*, une version plus faible des raisons suffisantes qui sont, elles, propres aux forêt aléatoires. Une *raison majoritaire* d'une instance $\mathbf{x}$ étant donnée une forêt aléatoire donnée correspond à un impliquant d'une majorité d'arbres de la forêt. Pour

chacune de ces familles d'explication abductive, nous avons étudié leur problème de génération (trouver une explication) et leur problème de minimisation (trouver une explication de taille minimale). Nous montrons que calculer une raison directe ou une raison majoritaire d'une instance étant donnée une forêt aléatoire se fait en temps polynomial alors que le problème de minimisation des raisons majoritaires est NP-difficile, là où celui des raisons suffisantes est Σ_2^p-difficile.

Nous proposons également des algorithmes afin de calculer de telles explications, donnant la possibilité de réaliser des comparaisons empiriques. Les tests réalisés montrent que le ratio concision de l'explication / temps de calcul est en faveur des raisons majoritaires en comparaison aux raisons suffisantes. En effet, expérimentalement, la taille des raisons majoritaires est proche de celle des raisons suffisantes, mais leur temps de calcul est largement plus court. Nous avons ainsi tiré profit de la capacité à calculer efficacement des raisons majoritaires pour en générer plusieurs, de façon à ne conserver que les plus courtes. La taille des raisons majoritaires obtenues est souvent plus petite que celles des raisons suffisantes qui ont été produites. De plus, en utilisant un solveur *anytime* PARTIAL MaxSAT pour minimiser les raisons majoritaires, nous pouvons extraire des explications encore plus concises. Les preuves des propriétés présentées et des résultats empiriques plus détaillés sont disponibles depuis notre site web `www.cril.univ-artois.fr/expekctation/`.

2 Préliminaires

Pour tout entier naturel n, nous notons $[n] = \{1, ..., n\}$. Soit F_n l'ensemble des fonctions booléennes de $\{0,1\}^n$ à $\{0,1\}$, et soit $X_n = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$ l'ensemble des variables booléennes d'entrée. Nous nommons *instance* tout $\mathbf{x} \in \{0,1\}^n$. Pour chaque fonction $f \in F_n$, une instance $\mathbf{x}$ est un exemple *positif* si $f(\mathbf{x}) = 1$, sinon nous disons que $\mathbf{x}$ est un exemple *négatif*.

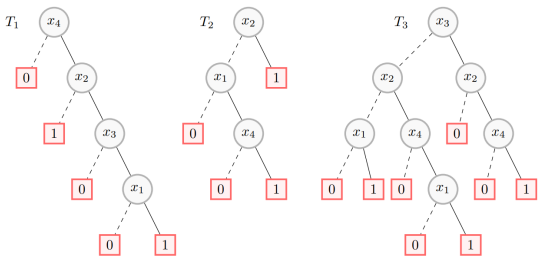

FIG. 1: Une forêt aléatoire $F = \{T_1, T_2, T_3\}$ reconnaissant les chats. Le fils gauche (resp. droit) de chaque nœud étiqueté $\mathbf{x}_i$ correspond à $\mathbf{x}_i = 0$ (resp. $\mathbf{x}_i = 1$).

f est vue comme une formule propositionnelle quand nous l'écrivons avec les connecteurs booléens $\wedge$ (conjonction), $\vee$ (disjonction) ou $\neg$ (négation) et en utilisant les constantes 1 (vrai) et 0 (faux). f est *satisfaisable* quand f n'est pas équivalente à 0. Un *littéral* l_i sur X_n décrit une variable booléenne $\mathbf{x}_i$ ou sa négation $\neg\mathbf{x}_i$, également notée $\bar{\mathbf{x}}$. Un *terme* t sur X_n est une conjonction de littéraux sur X_n et une *clause* c sur X_n est une disjonction de litté-

raux sur X_n. Souvent, t et c sont aussi vus comme des ensembles de littéraux. Une formule DNF (Forme Normale Disjonctive) est une disjonction de *termes* et une formule CNF (Forme Normale Conjonctive) est une conjonction de *clauses*. Dans la suite, nous identifions souvent les instances à des termes. Soit une instance $\mathbf{z} \in \{0,1\}^n$, le terme correspondant $t_\mathbf{z}$ est défini par :

$$t_\mathbf{z} = \bigwedge_{i=1}^{n} \mathbf{x}_i^{\mathbf{z}_i} \text{ où } \mathbf{x}_i^0 = \bar{\mathbf{x}}_i \text{ et } \mathbf{x}_i^1 = \mathbf{x}_i$$

Un terme t *couvre* une instance $\mathbf{z}$ si l'ensemble des littéraux composant t est inclus dans l'ensemble des littéraux composant le terme représentant z, noté $t_\mathbf{z}$. On note cela $t \subseteq t_\mathbf{z}$. Un *impliquant* d'une fonction booléenne f est un terme qui implique f, soit un terme t tel que $f(\mathbf{z}) = 1$ pour tout $\mathbf{z}$ couvert par t. Un *impliquant premier* de f est un impliquant t de f tel qu'aucun sous-ensemble strict de t n'est un impliquant de f.

Un *arbre de décision* (booléen) sur X_n est un arbre binaire T dont chaque nœud interne correspond à une des variables d'entrée et dont chaque feuille est étiquetée soit par 0, soit par 1. Chaque variable est supposée n'apparaître qu'une fois dans chaque chemin racine/feuille de l'arbre. La valeur de $T(\mathbf{x}) \in \{0,1\}$ de T sur une instance $\mathbf{x}$ est donnée par la valeur de la feuille atteinte en partant de la racine : à chaque nœud, si la variable $\mathbf{x}_i$ associée au nœud considéré vaut 1 alors nous continuons sur le fils droit, sinon sur le fils gauche. Une *forêt aléatoire* (booléenne) sur X_n est un ensemble $F = \{T_1, \ldots, T_m\}$, où chaque $T_i (i \in [m])$ est un arbre de décision sur X_n et tel que la valeur de $F(\mathbf{x}) \in \{0,1\}$ d'une instance $\mathbf{x}$ est donnée par :

$$F(\mathbf{x}) = \begin{cases} 1 & \text{si } \frac{1}{m}\sum_{i=1}^{m} T_i(\mathbf{x}) > \frac{1}{2} \\ 0 & \text{sinon} \end{cases}$$

La taille de F est donnée par $|F| = \Sigma_{i=1}^{m} |T_i|$, où $|T_i|$ est le nombre de nœuds apparaissant dans T_i. L'ensemble des arbres de décision définis sur X_n est noté DT_n et l'ensemble des forêts aléatoires définies sur X_n et possédant au moins m arbres est noté $RF_{n,m}$. De plus, on note RF_n l'union des $RF_{n,m}$ pour tout $m \in \mathbb{N}$.

Exemple 1 *La forêt aléatoire $F = \{T_1, T_2, T_3\}$ de la figure 1 est composée de trois arbres de décision. Cette forêt aléatoire sépare les chats des autres animaux en sachant si : $\mathbf{x}_1$: possède une moustache, $\mathbf{x}_2$: est un petit animal, $\mathbf{x}_3$: possède des coussinets et $\mathbf{x}_4$: possède 4 pattes.*

On sait qu'un arbre de décision T peut être transformé en sa négation en inversant les valeurs de ses feuilles. Calculer la négation d'une forêt aléatoire peut également être réalisé efficacement :

Proposition 1 *Il existe un algorithme en temps linéaire qui étant donnée une forêt aléatoire F retourne une forêt aléatoire représentant $\neg F$.*

Une propriété importante des arbres de décision est que quel que soit $T \in DT_n$, T peut être transformé en temps linéaire en une DNF équivalente où chaque terme correspond à l'un des chemins de l'arbre se concluant sur une feuille ayant la valeur 1, ou encore en une CNF équivalente où chaque clause est la négation d'un terme correspondant à un chemin de la racine à une feuille 0. La situation n'est pas aussi simple pour les forêts aléatoires :

Proposition 2 *Il existe un algorithme en temps linéaire qui transforme toute* CNF *et toute* DNF *en une forêt aléatoire équivalente, mais il n'existe pas d'algorithme polynomial en espace pour convertir une forêt aléatoire en une* CNF *ou en une* DNF *équivalente.*

3 Des explications abductives (ou raisons)

Notre article traite de la notion d'*explication abductive* pour les forêts aléatoires. Formellement, pour $f \in F_n$ et $\mathbf{x} \in \{0,1\}^n$, une *explication abductive* (aussi appelées *raisons*) de $\mathbf{x}$ étant donnée f est un impliquant t de f (ou de $\neg f$ dans le cas où $f(\mathbf{x}) = 0$) qui couvre $\mathbf{x}$. Il existe toujours une explication abductive t de $\mathbf{x}$ étant donnée f car $t = t_{\mathbf{x}}$ est une telle explication triviale. Ce faisant, nous allons dans le reste de cette section, nous concentrer sur des formes plus concises d'explication abductive. Quand f est représentée par une forêt aléatoire F, il suffit d'étudier le cas où $\mathbf{x}$ est un exemple positif, puisque nous pouvons calculer $\neg F$ en temps linéaire dans la taille de F (proposition 1).

3.1 Raisons directes

Pour un arbre de décision $T \in DT_n$ et une instance $\mathbf{x} \in \{0,1\}^n$, la *raison directe* de $\mathbf{x}$ étant donné T est le terme $t_{\mathbf{x}}^T$ correspondant à l'unique chemin racine/feuille de T qui couvre $\mathbf{x}$. Cette forme d'explication abductive peut être étendue aux forêts aléatoires comme suit :

Définition 1 *Soient $F = \{T_1, ..., T_n\}$ une forêt aléatoire dans $RF_{n,m}$ et $\mathbf{x} \in \{0,1\}$ une instance. La* raison directe *de $\mathbf{x}$ étant donnée F est définie par le terme $t_{\mathbf{x}}^F$ donné par :*

$$t_{\mathbf{x}}^F = \begin{cases} \bigwedge_{T_i \in F : T_i(\mathbf{x})=1} t_{\mathbf{x}}^{T_i} & \text{si } F(\mathbf{x}) = 1 \\ \bigwedge_{T_i \in F : T_i(\mathbf{x})=0} t_{\mathbf{x}}^{T_i} & \text{si } F(\mathbf{x}) = 0 \end{cases}$$

Exemple 2 *Considérons l'exemple 1 avec l'instance $\mathbf{x} = (1,1,1,1)$ qui est reconnue comme étant un chat, puisque $F(\mathbf{x}) = 1$. La raison directe de $\mathbf{x}$ étant donnée F est $t_{\mathbf{x}}^F = \mathbf{x}_1 \wedge \mathbf{x}_2 \wedge \mathbf{x}_3 \wedge \mathbf{x}_4$. Elle coïncide avec $t_{\mathbf{x}}$. Maintenant considérons l'instance $\mathbf{x}' = (0,1,0,0)$ qui n'est pas reconnue comme un chat. La raison directe est cette fois $t_{\mathbf{x}'}^F = \mathbf{x}_2 \wedge \bar{\mathbf{x}_3} \wedge \bar{\mathbf{x}_4}$ qui est une explication abductive plus concise que $t_{\mathbf{x}'}$.*

3.2 Raisons suffisantes

Une autre notion possible d'explication abductive est celle des raisons suffisantes définies pour n'importe quel classeur booléen (Darwiche et Hirth, 2020). Dans le cadre d'une forêt aléatoire, ces explications peuvent être définies comme suit :

Définition 2 *Soient $F \in RF_n$ une forêt aléatoire et $\mathbf{x} \in \{0,1\}^n$ une instance. Une* raison suffisante *de $\mathbf{x}$ étant donnée F est un impliquant premier t de F (resp. $\neg F$) si $F(\mathbf{x}) = 1$ (resp. $F(\mathbf{x}) = 0$) tel que t couvre $\mathbf{x}$.*

Exemple 3 *Dans notre exemple, $\mathbf{x}_2 \wedge \mathbf{x}_3 \wedge \mathbf{x}_4$ et $\mathbf{x}_1 \wedge \mathbf{x}_4$ sont des raisons suffisantes pour l'instance $\mathbf{x}$. $\bar{\mathbf{x}_4}$ et $\bar{\mathbf{x}_1} \wedge \mathbf{x}_2 \wedge \bar{\mathbf{x}_3}$ sont des raisons suffisantes pour l'instance $\mathbf{x}'$.*

On peut noter que tous les littéraux apparaissant dans une raison suffisante t de $\mathbf{x}$ étant donnée F sont *pertinents*, i.e., non redondants. En effet, par construction, $\forall l \in t$, le terme $t \backslash \{l\}$ n'implique pas F quand $F(\mathbf{x}) = 1$ (resp. n'implique pas $\neg F$ quand $F(\mathbf{x}) = 0$). La raison directe de $\mathbf{x}$ étant donnée F ne partage pas cette propriété en général.

Déterminer si un terme donné est une raison suffisante pour une instance $\mathbf{x}$ étant donnée une forêt aléatoire F a récemment été montré DP-complet (Izza et Marques-Silva, 2021). Vérifier qu'un terme est une explication abductive est déjà assez coûteux :

Proposition 3 *Soient F une forêt aléatoire dans RF_n, t un terme sur X_n et $\mathbf{x} \in \{0,1\}^n$ une instance. Déterminer si t est une explication abductive de $\mathbf{x}$ étant donnée F est un problème* coNP*-complet.*

Le résultat précédent montre une différence notable avec le problème résoluble en temps polynomial de tester si un terme t est une explication abductive d'une instance étant donné un arbre de décision T. Dans ce dernier cas, T peut être converti en une CNF équivalente en temps linéaire, et tester si un terme est un impliquant d'une CNF équivalente se résout en temps $O(n|T|)$. Dans le cas des forêts aléatoires, le test d'impliquant peut être réalisé via un oracle SAT :

Proposition 4 *Soient $F = \{T_1, ..., T_n\}$ une forêt aléatoire de $RF_{n,m}$ et t un terme satisfaisable sur X_n. Soit H tel que :*

$$H = \{(\bar{y_i} \vee c) : i \in [m], c \in \texttt{CNF}(\neg T_i)\} \cup \texttt{CNF}(\Sigma_{i=1}^{m} y_i > \tfrac{m}{2})$$

où $\{y_1, ..., y_n\}$ sont des variables booléennes utiles à l'encodage, $\texttt{CNF}(\neg T_i)$ est un encodage en CNF de $\neg T_i$ avec $i \in [m]$ et $\texttt{CNF}(\Sigma_{i=1}^{m} y_i > \frac{m}{2})$ un encodage de la contrainte de cardinalité $\Sigma_{i=1}^{m} y_i > \frac{m}{2}$ (représentant le vote majoritaire). t est un impliquant de F si et seulement si $H \wedge t$ n'est pas satisfaisable.

En se basant sur cet encodage, les raisons suffisantes d'une instance $\mathbf{x}$ étant donnée une forêt aléatoire F peuvent être vues comme des MUS (*minimal unsatisfiable sets*) (Izza et Marques-Silva, 2021). Cette caractérisation permet d'utiliser un algorithme de calcul de MUS comme ceux de Audemard et al., Liffiton et al. (2016), ou encore Marques-Silva et al. (2017) afin d'engendrer des raisons suffisantes.

Un moyen naturel d'améliorer la clarté des raisons suffisantes est de se focaliser sur des raisons suffisantes minimales, c'est-à-dire parmi les plus courtes / concises.

Exemple 4 *Dans notre exemple, $\mathbf{x}_1 \wedge \mathbf{x}_4$ est l'unique raison suffisante minimale de l'instance $\mathbf{x}$ étant donnée F et $\bar{\mathbf{x}_4}$ est l'unique raison suffisante minimale de $\mathbf{x}'$ étant donnée F.*

Rechercher une raison suffisante minimale revient à chercher un MUS de taille minimale. Malheureusement, bien que des algorithmes permettent de calculer de tels MUS existent (Ignatiev et al., 2015), le problème à résoudre est plus difficile que celui de calcul d'un MUS :

Proposition 5 *Soient $F \in RF_{n,m}$, $\mathbf{x} \in \{0,1\}^n$ et $k \in \mathbb{N}$. Déterminer s'il existe une raison suffisante minimale de $\mathbf{x}$ étant donnée F contenant au plus k littéraux est Σ_2^p-complet.*

3.3 Raisons majoritaires

Les résultats précédents montrent les raisons directes comme peu concises mais faciles à calculer, alors que les raisons suffisantes sont plus concises mais plus difficiles à obtenir. Une question naturelle surgit alors : peut-on trouver un bon compromis entre temps de calcul et concision de l'explication abductive ? Nous répondons positivement à cette question en introduisant la notion de *raison majoritaire* :

Définition 3 *Soient $F = \{T_1, \ldots, T_m\}$ une forêt aléatoire de $RF_{n,m}$ et $\mathbf{x} \in \{0,1\}^n$ une instance. Une* raison majoritaire *de $\mathbf{x}$ étant donnée F est un terme t couvrant $\mathbf{x}$ et qui est un impliquant d'au moins $\lfloor \frac{m}{2} \rfloor + 1$ arbres de décision T_i (resp. $\neg T_i$) si $F(\mathbf{x}) = 1$ (resp. si $F(\mathbf{x}) = 0$) et que $\forall l \in t, t \backslash \{l\}$ ne satisfait plus la condition précédente.*

Exemple 5 *Dans notre exemple, l'instance $\mathbf{x}$ possède la raison majoritaire $\mathbf{x}_1 \wedge \mathbf{x}_2 \wedge \mathbf{x}_4$ impliquant T_2 et T_3. De même, nous pouvons expliquer l'instance $\mathbf{x}'$ via une raison majoritaire $\mathbf{x}_2 \wedge \bar{\mathbf{x}_4}$ qui implique T_1 et T_3.*

En général, les notions de raison suffisante et de raison majoritaire ne coïncident pas. En effet, une raison suffisante t est un impliquant premier (couvrant $\mathbf{x}$) de F, alors qu'une raison majoritaire t' est un impliquant (couvrant $\mathbf{x}$) d'une majorité d'arbres de F. Nous pouvons voir les raisons majoritaires comme des formes « faibles » des raisons suffisantes puisqu'elles contiennent potentiellement des littéraux redondants.

Proposition 6 *Soient $F = \{T_1, \ldots, T_m\}$ tel que $F \in RF_{n,m}$ et $\mathbf{x} \in \{0,1\}^n$. Sauf si $m < 3$, il est possible que chaque raison majoritaire de $\mathbf{x}$ étant donnée F soit de taille arbitrairement plus grande que celles de toutes les raisons suffisantes de $\mathbf{x}$ étant donnée F.*

Ce qui rend les raisons majoritaires intéressantes, c'est le fait que ces explications abductives peuvent être générées en temps linéaire. La preuve repose sur l'existence d'un algorithme glouton pour les calculer. Considérons le cas $F(\mathbf{x}) = 1$ et commençons avec $t = t_{\mathbf{x}}$. Nous itérons sur tous les littéraux $l \in t$ en vérifiant si $t \setminus \{l\}$ est un impliquant d'au moins $\lfloor \frac{m}{2} \rfloor + 1$ arbres de décision T_i de F. Si tel est le cas, on efface l de t et on reprend le calcul avec le littéral suivant. Une fois que tout les littéraux de $t_{\mathbf{x}}$ ont été examinés, le terme t résultant est une raison majoritaire par construction puisque $\forall l \in t$ le terme $t \setminus \{l\}$ n'implique pas une majorité d'arbres alors que c'est le cas de t. Le cas $F(\mathbf{x}) = 0$ est similaire, il suffit de raisonner sur les $\neg T_i$ au lieu des T_i. L'algorithme glouton opère en temps $O(n|F|)$ puisqu'à chaque itération, vérifier si t est un impliquant de T_i peut être réalisé en temps $O(n|T_i|)$. Comme pour les raisons suffisantes minimales, un moyen d'améliorer la qualité des raisons majoritaires est de se focaliser sur les plus courtes. Soit F une forêt aléatoire et $x \in \{0,1\}^n$ une instance. Une *raison majoritaire minimale* de $\mathbf{x}$ étant donnée F est une raison majoritaire de taille minimale.

Exemple 6 *Dans notre exemple, la taille des raisons majoritaires minimales de $\mathbf{x}$ est de 3 et elle est de 2 pour $\mathbf{x}'$.*

Bien évidement, optimiser la concision d'une raison majoritaire n'est pas gratuit. Cependant, cette optimisation est bien moins coûteuse que celle à mettre en place pour calculer des raisons suffisantes minimales :

Proposition 7 *Soient $F \in RF_n, \mathbf{x} \in \{0,1\}^n$ et $k \in \mathbb{N}$. Déterminer s'il existe une raison majoritaire minimale t pour $\mathbf{x}$ étant donnée F contenant au plus k littéraux est un problème* NP*-complet.*

Nous pouvons traduire le problème du calcul d'une raison majoritaire minimale en un problème d'optimisation PARTIAL MaxSAT. Les entrées du problème PARTIAL MaxSAT consistent en deux ensembles finis de clauses C_{soft} et C_{hard}. Le but est de calculer (quand elle existe) une interprétation qui satisfait le plus de clauses de C_{soft} et toutes les clauses de C_{hard}.

Proposition 8 *Soient $F \in RF_n$ et $\mathbf{x} \in \{0,1\}^n$ tels que $F(\mathbf{x}) = 1$. Soient C_{soft} et C_{hard} tels que :*

$$C_{soft} = \{\bar{\mathbf{x}}_i : \mathbf{x}_i \in t_{\mathbf{x}}\} \cup \{\mathbf{x}_i : \bar{\mathbf{x}}_i \in t_{\mathbf{x}}\}$$

$$C_{hard} = \{(\bar{y}_i \vee c_{|\mathbf{x}}) : i \in [m], c \in \texttt{CNF}(T_i)\} \cup \texttt{CNF}\left(\Sigma_{i=1}^{m} y_i > \frac{m}{2}\right)$$

où $c_{|\mathbf{x}} = c \cap t_{\mathbf{x}}$ est la restriction de c aux littéraux de $t_{\mathbf{x}}$, $\{y_1, ..., y_m\}$ sont des variables utiles à l'encodage et $\texttt{CNF}(\Sigma_{i=1}^{m} y_i > \frac{m}{2})$ est un encodage en CNF de la contrainte de cardinalité (pour le vote majoritaire) $\Sigma_{i=1}^{m} y_i > \frac{m}{2}$. L'intersection de $t_{\mathbf{x}}$ avec $t_{\mathbf{z}*}$, où $\mathbf{z}*$ est une solution optimale du problème PARTIAL MaxSAT est une raison majoritaire minimale de $\mathbf{x}$ étant donnée F. Dans le cas où $F(\mathbf{x}) = 0$, on peut calculer une raison majoritaire minimale de $\mathbf{x}$ étant donnée F de façon similaire en remplaçant C_{hard} par $C_{hard} = \{(\bar{y}_i \vee c_{|\mathbf{x}}) : i \in [m], c \in \texttt{CNF}(\neg T_i)\} \cup \texttt{CNF}(\Sigma_{i=1}^{m} y_i > \frac{m}{2})$.

Grâce à cette caractérisation, nous pouvons exploiter les nombreux solveurs PARTIAL MaxSAT existants (voir en particulier, (Ansótegui et al., 2013; Morgado et al., 2014; Narodytska et Bacchus, 2014; Saikko et al., 2016)) pour dériver des raisons majoritaires minimales.

4 Expérimentations

4.1 Conditions expérimentales

Voici le protocole expérimental que nous avons suivi : nous avons considéré une sélection de 15 datasets connus de la littérature et disponibles sur Kaggle (`www.kaggle.com`), OpenML (`www.openml.org`), ou UCI (`archive.ics.uci.edu/ml/`). Les datasets étudiés sont *compas*, *placement*, *recidivism*, *adult*, *ad_data*, *mnist38*, *mnist49*, *gisette*, *dexter*, *dorothea*, *farm-ads*, *higgs_boson*, *christine*, *gina* et *bank*. *mnist38* et *mnist49* sont des sous-ensembles du dataset *mnist* réduit aux instances représentant des « 3 » ou des « 8 » et respectivement des « 4 » ou des « 9 ».

Nous n'avons utilisé que des datasets associés à un classement binaire. Les attributs catégoriels sont traités comme des attributs numériques en affectant à chaque catégorie un entier (ex : *l'attribut « genre » ayant pour valeur « H » ou « F » est codée avec les valeurs « 1 » ou « 2 »*). Comme pour tous les autres attributs numériques, aucun pré-traitement n'a été réalisé : les données ont été binarisées « à la volée » par la binarisation induite/apprise par la forêt aléatoire.

Pour chaque dataset d, nous avons réalisé une *validation croisée à 10 blocs*. Cela consiste à séparer d en 10 parties $p_i, i \in \{1, ..., 10\}$ et pour chaque p_i, nous entraînons la forêt aléatoire sur toutes les autres parties afin de la tester sur p_i. Pour tout d, nous calculons la précision (et les autres statistiques) en considérant les 10 forêts générées (une par p_i). Nous avons utilisé l'implémentation des forêts aléatoires de la librairie python *Scikit-Learn* (Pedregosa et al., 2011) dans sa version v0.23.2. Tous les hyper-paramètres des forêts aléatoires ont été mis à leur valeur par défaut, sauf le nombre d'arbres retenus pour chaque dataset afin d'optimiser la précision de la forêt associée. Pour chaque benchmark d, pour chacune des 10 forêts apprises avec leur p_i associée et pour chaque type d'explication abductive mentionné en section 3, nous avons testé pour 25 instances de p_i les algorithmes de génération et de minimisation de ces

explications. Nous avons donc, pour chacun de ces algorithmes et sur chaque dataset, réalisé des tests sur 250 instances différentes.

Afin de calculer des raisons suffisantes et des raisons majoritaires minimales, nous avons utilisé la librairie python PYSAT (Ignatiev et al., 2018) dans sa version 0.1.6.dev15. Elle fournit une interface python facilitant la manipulation des formules `CNF`. De plus, pour les raisons suffisantes, PYSAT propose aussi une interface au solveur MUS, appelé MUSER (Belov et Marques-Silva, 2012). Pour les raisons majoritaires, nous avons implémenté notre algorithme glouton en C++. Nous avons lancé 50 fois cet algorithme pour chaque instance $\mathbf{x}$ en changeant à chaque fois l'ordre de parcours des littéraux de $t_{\mathbf{x}}$ représentant $\mathbf{x}$ et nous avons conservé la raison majoritaire de plus petite taille ainsi générée.

Nous avons aussi calculé une « explication LIME » pour chaque instance testée. Comme esquissé en amont, LIME (Ribeiro et al., 2016) cherche à approcher localement (c'est-à-dire pour des instances « proches » d'une instance $\mathbf{x}$ donnée) le comportement d'un classeur. L'implémentation de LIME utilisée génère une approximation via une fonction linéaire, ainsi chaque $l_i \in t_{\mathbf{x}}$ est associé à un poids w_i. Pour calculer l'explication LIME d'une instance $\mathbf{x}$ étant donnée une forêt F, nous trions les l_i par ordre décroissant de leur w_i associé en valeur absolue. Nous sommons tous les poids négatifs si $F(\mathbf{x}) = 1$, positifs sinon. Ensuite, nous parcourons les l_i dans l'ordre du tri et nous conservons dans l'explication LIME tous les littéraux ayant un poids positif (resp. négatif) jusqu'à ce que leur somme soit supérieure à la somme des poids négatifs (resp. positifs) en valeur absolue. Les littéraux ainsi conservés forment l'explication LIME et cette explication est obtenue en temps polynomial (voir aussi (Marques-Silva et al., 2020)).

Nos calculs ont été réalisés sur un processeur Intel(R) XEON E5-2637 CPU @ 3.5GHz avec 128Go de RAM. Un temps limite (TO) a été fixé à 600 secondes sauf pour le calcul des explications LIME (pour qui le temps calcul n'a pas été contraint).

4.2 Résultats

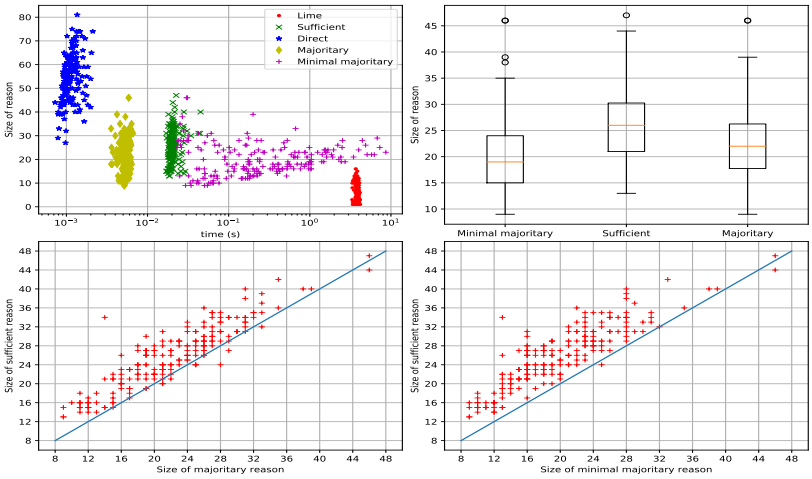

FIG. 2: Résultats expérimentaux sur le dataset *placement*

Le premier résultat dont nos expérimentations témoigne est la difficulté du calcul des raisons suffisantes minimales, le temps limite de 600s étant atteint de façon quasi-systématique. Pour des raisons d'espace, nous allons uniquement commenter ici les résultats obtenus pour les datasets *placement* et *gisette* [1]. Le dataset *placement* classe 215 étudiants sur le fait d'avoir un travail (classé positif) ou non (négatif) en fonction de 13 attributs décrivant leurs situations respectives (salaire, genre, études, ...). La forêt aléatoire générée possède 25 arbres de décision et a une précision de 97.6%. Le dataset *gisette* est beaucoup plus grand ; il classifie des images de « 4 » (négatif) et de « 9 » (positif) manuscrits en fonction de valeurs concernant les pixels de ces images, son nombre d'attributs s'élève à 5000 et cela pour 7000 images. La forêt générée possède 85 arbres de décisions pour une précision de 96.0%.

La figure 2 illustre les résultats obtenus pour *placement* par le biais de 4 graphiques. Chaque point représente une instance. Le premier graphique donne le nombre de littéraux (ordonnée) contenus dans l'explication, le temps de calcul de l'explication (abscisse) et sa nature (code couleur). Nous observons que la raison directe est rapide à calculer mais est nettement moins concise que les autres raisons. Inversement, si l'explication LIME reste concise, le temps de calcul de cette dernière est bien plus élevé (rappelons aussi que ces explications sont non abductives en général). Enfin, les boîtes à moustache et les diagrammes de dispersion montrent que les raisons majoritaires (et leurs versions minimales) introduites dans cet article semblent avoir une meilleure concision que les autres explications pour un temps de calcul inférieur.

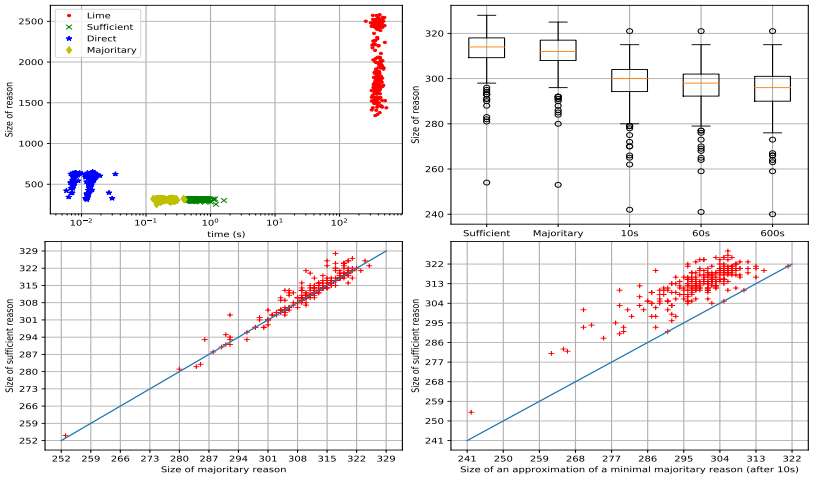

FIG. 3: Résultats expérimentaux sur le dataset *gisette*

La figure 3 pour le dataset *gisette* appelle d'autres remarques. D'abord, LIME est ici nettement moins performant, car LIME raisonne directement sur les 5000 attributs de base alors que les autres approches utilisent la binarisation induite par la forêt et que celle-ci contient moins de littéraux (attributs binaires). Ensuite, pour *gisette*, les raisons majoritaires minimales se sont révélées trop difficiles à calculer. C'est pourquoi nous avons eu recours au solveur PARTIAL MaxSAT appelé LMHS, Saikko et al. (2016) qui a l'avantage d'être « anytime ».

1. Pour les résultats obtenus sur les autres datasets, on pourra consulter `www.cril.univ-artois.fr/expekctation/`

Il est ainsi possible d'interrompre le calcul réalisé par LMHS avec la garantie d'obtenir alors un sur-ensemble de la solution recherchée. En pratique, le sur-ensemble obtenu avec LMHS au bout de 10s de calcul seulement conduit à des raisons majoritaires dont la taille est plus petite que celles des raisons suffisantes obtenues. À notre connaissance, pour les raisons suffisantes, aucun solveur pour le calcul d'un MUS de taille minimale et offrant un comportement « anytime » n'existe, nous empêchant d'utiliser le même type d'approche pour des raisons suffisantes approchant les raisons suffisantes minimales.

5 Conclusion

Nous avons introduit et comparé plusieurs notions d'explications abductives permettant d'interpréter les classements réalisés par des forêts aléatoires. Quand il s'agit de calculer efficacement des explications abductives relativement concises, les raisons majoritaires que nous avons introduites semblent offrir un bon compromis, qu'il s'agisse du problème de leur génération ou celui de leur minimisation. Même si une raison majoritaire minimale peut être difficile à obtenir pour des datasets de taille importante, nous avons mis en évidence une méthode efficace en pratique, qui permet d'approcher une raison majoritaire minimale en quelques secondes. Expérimentalement, cette approche fournit une explication plus concise que celles obtenues en utilisant les autres approches considérées dans cet article.

Remerciements

Merci aux relecteurs pour leurs remarques et leurs suggestions d'amélioration de l'article. Elles nous ont été très utiles. Le travail correspondant a été réalisé dans le cadre de la chaire ANR d'enseignement et de recherche EXPEKCTATION (ANR-19-CHIA-0005-01).

Références

Ansótegui, C., M. L. Bonet, et J. Levy (2013). Sat-based maxsat algorithms. *Artificial Intelligence 196*, 77–105.

Audemard, G., J.-M. Lagniez, et L. Simon. Improving glucose for incremental SAT solving with assumptions : Application to MUS extraction. In *SAT'13*, pp. 309–317.

Azar, A. T., H. I. Elshazly, A. E. Hassanien, et A. M. Elkorany (2014). A random forest classifier for lymph diseases. *Computer Methods and Programs in Biomedicine 113*(2), 465–473.

Belov, A. et J. Marques-Silva (2012). Muser2 : An efficient MUS extractor. *J. Satisf. Boolean Model. Comput. 8*(3/4), 123–128.

Bénard, C., G. Biau, S. D. Veiga, et E. Scornet (2021). Interpretable random forests via rule extraction. In *AISTATS'21*, pp. 937–945.

Breiman, L. (2001). Random forests. *Machine Learning 45*(1), 5–32.

Choi, A., A. Shih, A. Goyanka, et A. Darwiche (2020). On symbolically encoding the behavior of random forests. In *FoMLAS'20, Workshop at CAV'20*.

Criminisi, A. et J. Shotton (2013). *Decision Forests for Computer Vision and Medical Image Analysis*. Advances in Computer Vision and Pattern Recognition. Springer.

Cutler, R., C. E. J. Thomas, K. H. Beard, A. Cutler, K. T. Hess, J. Gibson, et J. J. Lawler (2007). Random forests for classification in ecology. *Ecology 88*(11), 2783–2792.

Darwiche, A. et A. Hirth (2020). On the reasons behind decisions. In *ECAI'20*, pp. 712–720.

Ignatiev, A., A. Morgado, et J. Marques-Silva (2018). PySAT : A python toolkit for prototyping with SAT oracles. In *SAT'18*, pp. 428–437.

Ignatiev, A., A. Previti, M. Liffiton, et J. Marques-Silva (2015). Smallest MUS extraction with minimal hitting set dualization. In *CP'15*, pp. 173–182.

Izza, Y. et J. Marques-Silva (2021). On explaining random forests with SAT. In *IJCAI'21*.

Liffiton, M., A. Previti, A. Malik, et J. Marques-Silva (2016). Fast, flexible MUS enumeration. *Constraints An Int. J. 21*(2), 223–250.

Marques-Silva, J., T. Gerspacher, M. C. Cooper, A. Ignatiev, et N. Narodytska (2020). Explaining naive bayes and other linear classifiers with polynomial time and delay. In *NeurIPS'20*.

Marques-Silva, J., M. Janota, et C. Mencía (2017). Minimal sets on propositional formulae. Problems and reductions. *Artificial Intelligence 252*, 22–50.

Morgado, A., A. Ignatiev, et J. Marques-Silva (2014). MSCG : robust core-guided MaxSAT solving. *J. Satisf. Boolean Model. Comput. 9*(1), 129–134.

Narodytska, N. et F. Bacchus (2014). Maximum satisfiability using core-guided MaxSAT resolution. In *AAAI'14*, pp. 2717–2723.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, et E. Duchesnay (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research 12*, 2825–2830.

Ribeiro, M. T., S. Singh, et C. Guestrin (2016). "why should I trust you ?" : Explaining the predictions of any classifier. In *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144. ACM.

Saikko, P., J. Berg, et M. Järvisalo (2016). LMHS : A SAT-IP hybrid MaxSAT solver. In *SAT'16*, pp. 539–546.

Summary

Random forests are an effective machine learning model, this is why they are still widely used today. However, whilst it is quite easy to understand how a decision tree works, it is much more complex to interpret the decision made by a random forest, because it is typically the result of a majority vote among many trees. Here we examine various definitions of abductive explanations for random forests based on binary attributes. We are particularly interested in the generation problem (finding an explanation) as well as the minimization problem (finding a shortest explanation). We show in particular that irredundant abductive explanations (or sufficient reasons) can be difficult to obtain for random forests. We propose instead the notion of "majority reasons", that are in principle less concise abductive explanations, but which can be computed in polynomial time.