Boosting de gradient équitable : une approche adverse

Vincent Grari[1,2], Boris Ruf[2], Sylvain Lamprier[1], et Marcin Detyniecki[2]

[1] *Sorbonne Université* , *LIP6/CNRS* , Paris, France
[2] *AXA* , *REV Research* , Paris, France

7 juin 2020

Résumé

L'entrainement d'algorithme de classification de manière équitable est devenue un sujet essentiel en recherche dans le domaine de l'apprentissage automatique. Alors que la plupart des stratégies d'atténuation des biais se concentrent sur les algorithmes de réseaux de neurones, nous avons constaté très peu de recherches sur les modèles de classifications équitables basés sur des arbres de décision. Une comparaison récente sur le pouvoir prédictif des algorithmes de classification appliqués sur des données tabulaires a mis en évidence que les arbres par boosting de gradient surpassent de manière empirique les algorithmes d'apprentissage profond [ZLZA17]. Motivé par les performances prédictives de ces modèles, nous avons mis au point une nouvelle approche de boosting de gradient équitable. L'objectif de l'algorithme est de prédire la sortie Y avec des arbres de boosting de gradient tout en minimisant la capacité d'un réseau de neurones adverses à prédire les différents attributs sensibles. L'approche incorpore, à chaque itération, le gradient du réseau neuronal directement dans le gradient de l'arbre de boosting. Nous évaluons empiriquement notre approche sur 4 ensembles de données populaires et comparons avec différents algorithmes de l'état de l'art. Les résultats montrent que notre algorithme obtient des performances prédictives plus intéressantes tout en obtenant le même niveau d'équité, mesuré sur différentes définitions communes d'équité.

Mots-clef : Apprentissage automatique équitable, Apprentissage adverse, Boosting de gradient

1 Introduction

Les modèles d'apprentissage automatique ont un nombre très vaste d'applications qui peuvent avoir de vastes répercussions sur les individus (approbation de crédit, score de récidive, etc.), on se préoccupe de plus en plus de leur potentiel discrimination à l'encontre d'un groupe particulier de personnes sur la base de caractéristiques sensibles telles que le sexe, l'origine, la religion et tant d'autres.

De nombreuses stratégies d'atténuation de biais en apprentissage automatique ont été proposées ces dernières années, mais la plupart d'entre elles se concentrent uniquement sur les réseaux neuronaux. Les méthodes d'ensemble combinant plusieurs classifieurs d'arbres de décision se sont avérées très efficaces pour diverses applications. Il en résulte qu'en pratique, pour les ensembles de données tabulaires, les actuaires et les scientifiques des données préfèrent l'utilisation des arbres de boosting de gradient par rapport aux réseaux neuronaux en raison des performances prédictives généralement plus élevés. Notre domaine d'intérêt est le développement de classifieurs équitables basés sur les arbres de décision. Dans cet article, nous proposons une nouvelle approche pour combiner la performance de l'apprentissage des algorithmes d'arbres de boosting de gradient en introduisant une contrainte d'équité adverse.

Les contributions de cet article sont les suivantes :

— Nous appliquons l'apprentissage adverse pour une classification équitable à la famille des arbres de décision ;

— Nous comparons empiriquement notre proposition et ses variantes avec plusieurs approches de l'état de l'art, pour deux mesures d'équité différentes. Les expériences montrent des meilleurs résultats à notre approche.

2 L'apprentissage automatique équitable

2.1 Définitions de l'équité

Tout au long de ce document, nous considérons une tache d'apprentissage de classification supervisée avec n observations $(x_i, s_i, y_i)_{i=1}^n$, où $x_i \in \mathbf{R}^p$ est le vecteur des caractéristiques avec p prédicteurs pour une observation donnée i, s_i est son attribut sensible binaire et y_i son label binaire à prédire.

Afin de parvenir à l'équité, il est essentiel d'établir une compréhension claire de sa définition formelle. Dans ce qui suit, nous présentons les définitions les plus courantes utilisées dans la recherche récente. Tout d'abord, il y a le préretraitement des données qui limite les informations utilisées pour l'entraînement du classifieur. Ensuite, il y a l'équité individuelle, qui lie au niveau individuel et suggère que l'équité signifie que des individus similaires doivent être traités de la même manière. Enfin, il y a l'équité statistique ou de groupe. Ce type d'équité divise le monde en groupes définis par un ou plusieurs attributs sensibles de haut niveau. Elle exige qu'une statistique pertinente spécifique concernant le classifieur soit égale dans tous ces groupes. Dans ce qui suit, nous nous concentrons sur cette famille de mesures d'équité et nous expliquons les définitions les plus populaires de ce type utilisées dans les recherches récentes.

2.1.1 Parité démographique

Sur la base de cette définition, un classifieur est considéré comme équitable si la prédiction $\widehat{Y}$ des caractéristiques X est indépendante de l'attribut protégé S [DHP+11].

Definition 1. $P(\widehat{Y} = 1|S = 0) = P(\widehat{Y} = 1|S = 1)$

Il existe de multiples moyens d'évaluer cet objectif. L'évaluation selon la règle-p garantit que le ratio de taux positif pour le groupe non privilégié n'est pas inférieur à un seuil fixé : $\frac{p}{100}$. Le classifieur est considéré comme totalement équitable lorsque ce ratio satisfait une valeur de 100%. Inversement, une valeur de 0% indique un modèle totalement injuste.

$$\textit{P-rule :}\ \min\left(\frac{P(\widehat{Y}=1|S=1)}{P(\widehat{Y}=1|S=0)}, \frac{P(\widehat{Y}=1|S=0)}{P(\widehat{Y}=1|S=1)}\right)$$

2.1.2 Egalilité des chances

Un algorithme est considéré comme équitable si, pour les deux catégories démographiques, $S = 0$ et $S = 1$, avec un résultat $Y = 1$, le prédicteur $\widehat{Y}$ a des taux de *vrai* positifs égaux, et lorsque $Y = 0$, le prédicteur $\widehat{Y}$ a des taux de *faux* positifs égaux [HPS16]. Cette contrainte impose que le taux d'exactitude des prédictions soit la même pour l'ensemble des catégories démographiques puisque le taux de classification positif et négatif est égal dans tous les groupes.

Definition 2. $P(\widehat{Y} = 1|S = 0, Y = y) = P(\widehat{Y} = 1|S = 1, Y = y), \forall y \in \{0, 1\}$

Pour évaluer cet objectif, il convient de mesurer la disparité de mauvais traitements (DM) [ZVGG17]. Cette métrique mesure la différence absolue entre le taux de faux positifs (TFP) et le taux de faux négatifs (FNR) pour les deux groupes démographiques.

$$D_{FPR} : |P(\widehat{Y} = 1|Y = 0, S = 1) - P(\widehat{Y} = 1|Y = 0, S = 0)|$$
$$D_{FNR} : |P(\widehat{Y} = 0|Y = 1, S = 1) - P(\widehat{Y} = 0|Y = 1, S = 0)|$$

Plus les valeurs de D_{FPR} et D_{FNR} sont proches de 0, plus le degré de mauvais traitement du classifieur est faible.

2.2 État de l'art

Les recherches récentes sur l'apprentissage automatique équitable ont fait des progrès considérables dans la quantification et l'atténuation des biais non désirés. Il existe trois types de stratégies d'atténuation : La famille des algorithmes de « prétraitement » qui garantit que les données d'entrée sont équitables, les méthodes de « traitement en cours » où le biais indésirable est directement atténué pendant la phase de d'apprentissage, et enfin les algorithmes de « post-traitement » où la sortie du classifieur entraîné en amont est modifiée par la suite.

Nous proposons un algorithme de « traitement en cours ». Ici, le biais indésirable est directement atténué pendant la phase d'entraînement. Une approche simple pour atteindre cet objectif consiste à intégrer une pénalité d'équité directement dans la fonction de perte. Un tel algorithme peut intégrer une contrainte de covariance par borne de décision pour la régression logistique ou SVM linéaire [ZVRG15]. Dans une autre approche, un méta-algorithme prend la métrique d'équité comme partie en entrée et renvoie un nouveau classifieur optimisé suivant cette métrique d'équité [CHKV19]. En outre, l'émergence des réseaux antagonistes génératifs (GAN) a apporté la base nécessaire à une classification équitable par le biais d'un processus antagoniste [GPAM+14]. Dans ce domaine, un classifieur par réseau de neurones est entraîné pour prédire l'étiquette Y, tout en minimisant

simultanément la capacité d'un réseau neuronal antagoniste à prédire l'attribut sensible S [ZLM18, WVP18, LKC16].

3 Fair Adversarial Gradient Tree Boosting (FAGTB)

Notre objectif est d'apprendre un classifieur qui soit à la fois efficace pour prédire les vrais labels et équitable, dans le sens où il se focalise sur les métriques définies dans la section 2.1 pour la parité démographique ou l'égalité des chances. L'idée étant de tirer parti des performances intéressantes des arbres de boosting de gradient (GTB) pour la classification, tout en l'adaptant à un apprentissage automatique équitable par le biais d'un apprentissage antagoniste.

Le GTB est formé de manière séquentiel par itération de gradient avec une fonction de prédiction de la forme suivante :

$$F_M(x_i) = \sum_{m=0}^{M} \gamma_m h_m(x_i) \tag{1}$$

où x_i est le vecteur des caractéristiques, M est le nombre total d'itérations, et $h_m(x_i)$ correspond à un modèle d'apprentissage faible à l'étape m sous la forme d'une prédiction d'un arbre CART. Étant donné une fonction de perte $\mathcal{L}(y_i, F(x_i))$ à minimiser pour tous les (x_i, y_i) de l'ensemble d'entraînement, le GTB calcule à chaque étape m les « pseudo résidus » :

$$r_{im} = -\left[\frac{\partial \mathcal{L}(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} \quad \text{for } i = 1, \ldots, n \tag{2}$$

Ensuite, à chaque étape, le GTB ajuste un nouvel apprentissage faible $h_m(x)$ à ces pseudo résidus et l'ajoute au modèle actuel. Cette étape est répétée jusqu'à ce que l'algorithme converge.

Cette architecture nous permet de réaliser une classification équitable avec des algorithmes d'arbre de décision en utilisant le concept d'apprentissage antagoniste. Cela correspond à un jeu à deux joueurs avec deux composantes adverse, comme dans les réseaux antagonistes génératifs (GAN) [GPAM+14].

3.1 Min-Max formulation

Dans la continuité de [ZLM18, LKC16, WVP18] pour la classification équitable, nous considérons une fonction prédictive F, qui produit la probabilité qu'un vecteur d'entrée X soit étiqueté $Y = 1$, et un modèle antagoniste A qui tente de prédire l'attribut sensible S à partir de la sortie de F. En fonction du taux de d'exactitude des prédictions de l'algorithme contradictoire, nous pénalisons le gradient du GTB à chaque itération. L'objectif est d'obtenir un classifieur F dont les probabilités prédites de sortie ne permettent pas au modèle antagoniste de reconstruire la valeur de l'attribut sensible. Si cet objectif est atteint, le biais des données en faveur de certaines données démographiques disparaît de la prédiction de sortie.

Le prédicteur et le classifieur adverses sont optimisés simultanément dans un jeu min-max défini comme :

$$\arg\min_F \max_{\theta_A} \sum_{i=1}^{n} \mathcal{L}_{F_i}(F(x_i)) - \lambda \sum_{i=1}^{n} \mathcal{L}_{A_i}(F(x_i); \theta_A) \tag{3}$$

où $\mathcal{L}_{F_i}$ et $\mathcal{L}_{A_i}$ sont respectivement la perte du prédicteur et de l'adversaire pour l'échantillon d'entraînement i donné $F(x_i) \in \mathbb{R}$, qui fait référence à la sortie du prédicteur pour l'entrée x_i. L'hyperparamètre λ contrôle l'impact de la fonction de perte de l'adversaire.

Le classifieur ainsi obtenu produit l'étiquette $\widehat{Y}$ qui maximise la probabilité postérieur $P(\widehat{Y}|X)$. Ainsi, pour un échantillon donné x_i, on obtient :

$$\hat{y}_i = \arg\max_{y \in \{0;1\}} p_F(Y = y | X = x_i) \tag{4}$$

où $p_F(Y = 1|X = x_i) = \sigma(F(x_i))$, avec σ désignant la fonction sigmoïde. Avec, $\mathcal{L}_{F_i}$ qui est défini comme la log-vraisemblance négative du prédicteur pour l'échantillon d'entraînement i :

$$\begin{aligned} \mathcal{L}_{F_i}(F(x_i)) &= -\log p_F(Y = y_i | X = x_i) \\ &= -\mathbf{1}_{y_i=1} \log(\sigma(F(x_i))) \\ &\quad -\mathbf{1}_{y_i=0} \log(1 - \sigma(F(x_i))) \end{aligned} \tag{5}$$

où $\mathbf{1}_{cond}$ est égal à 1 si *cond* est vrai, 0 sinon.

L'adversaire A correspond à un réseau de neurones avec les paramètres θ_A, qui prend comme entrée le sigmoïde de la sortie du prédicteur pour tout échantillon i (c'est-à-dire $P_F(Y = 1|X = x_i)$), et produit la probabilité P_{F,θ_A} que la valeur de la variable sensible soit égale à 1 :

- Pour la tâche de parité démographique, $P_F(Y = 1|X = x_i)$ est la seule entrée donnée à l'adversaire pour la prédiction de l'attribut sensible s_i. Dans ce cas, le réseau A produit la sortie de la probabilité conditionnelle $P_{F,\theta_A}(S = 1|V = v_i) = A(v_i)$, avec $V = (\sigma(F(X)))$.
- Pour la tâche de l'égalité des chances, l'étiquette y_i est concaténée à $P_F(Y = 1|X = x_i)$

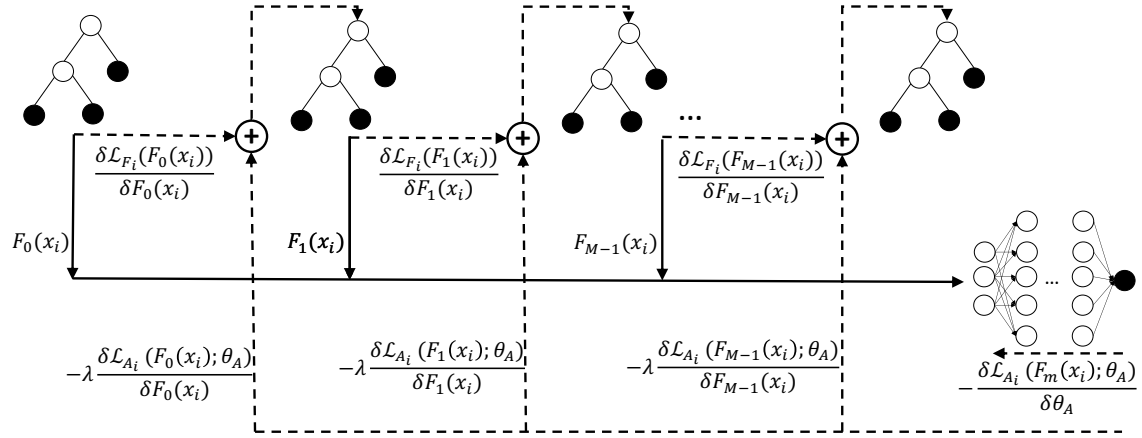

FIGURE 1 – L'architecture du Fair Adversarial Gradient Tree Boosting (FAGTB). Quatre étapes sont représentées, chacune correspond à un arbre h qui est ajouté au classifieur global F. Le réseau neuronal de droite est l'adversaire qui prédit les attributs sensibles à partir des sorties du classifieur. Les lignes pleines représentent les opérations en avant, tandis que les lignes pointillées représentent la rétropropagation du gradient. À chaque étape m, les gradients des fonctions de perte du prédicteur et de l'adversaire sont additionnés pour former la cible du nouvel arbre de décision h_{m+1}.

pour former le vecteur d'entrée de l'adversaire $v_i = (\sigma(F(x_i)), y_i)$, de sorte que la fonction A puisse sortir différentes probabilités conditionnelles $P_{F,\theta_A}(S = 1|V = v_i)$ selon l'étiquette y_i de i.

La perte adverse est définie pour tout échantillon d'entraînement i comme :

$$\begin{aligned}\mathcal{L}_{A_i}(F(x_i);\theta_A) = &- \mathbf{1}_{s_i=1}\log(\sigma(A(v_i))) \\ &- \mathbf{1}_{s_i=0}\log(1-\sigma(A(v_i)))\end{aligned} \tag{6}$$

avec v_i défini en fonction de la tâche telle que détaillée ci-dessus.

Notez que, pour le cas de la parité démographique, s'il existe (F^*, θ_A^*) tel que $\theta_A^* = \arg\max_{\theta_A} P_{F^*,\theta_A}(S|V)$ sur l'ensemble d'entraînement, $P_{F^*,\theta_A^*}(S|V) = \widehat{P}(S)$ et $P_{F^*}(Y|X) = \widehat{P}(Y|X)$, avec $\widehat{P}(S)$ et $\widehat{P}(Y|X)$ les distributions correspondantes sur l'ensemble d'entraînement, (F^*, θ_A^*) est un optimum global de notre équation du problème min-max (3). Dans ce cas, nous avons à la fois un classifieur optimal en entraînement, et un modèle tout à fait équitable puisque le meilleur adversaire possible n'est pas capable de prédire les S avec plus de précision que la distribution antérieure estimée. Des observations similaires peuvent facilement être faites pour la tâche d'égalité des chances (en remplaçant $\widehat{P}(S)$ par $\widehat{P}(S|Y)$ et en utilisant la définition correspondante de V dans l'assertion précédente). Bien qu'un tel ajustement optimal n'existe pas toujours dans les données, il montre que le modèle est capable d'identifier une solution lorsqu'il en atteint une. Si une solution optimale n'existe pas dans les données, l'optimum de notre problème min-max est un compromis entre le taux d'exactitude des prédictions et l'équité, contrôlé par l'hyperparamètre λ.

3.2 L'apprentissage

Le processus d'apprentissage est décrit sous forme de pseudo-code dans Algorithme 1. L'algorithme initialise d'abord le classifieur F_0 avec des valeurs constantes pour toutes les entrées, comme cela se fait pour le GBT classique. En outre, il initialise les paramètres θ_A du réseau neuronal antagoniste A (une initialisation Xavier est utilisée dans nos expériences). Ensuite, à chaque itération m, en plus du calcul des pseudo résidus r_{im} pour tout échantillon d'entraînement i w.r.t. la perte de prédiction ciblée $\mathcal{L}_{F_i}$, le calcule des pseudo résidus t_{im} pour la perte antagoniste $\mathcal{L}_{A_i}$ est également calculé. Les deux résidus sont alors combinés dans $u_{im} = r_{im} - \lambda * t_{im}$, où λ contrôle l'impact du réseau antagoniste. L'algorithme ajuste ensuite un nouveau régresseur faible h_m (un arbre de décision CART dans notre étude) aux résidus en utilisant l'ajustement d'entraînement $\{(x_i, u_{im})\}_{i=1}^n$. Ce régresseur entraîné sur les pseudo-résidus permet de corriger à la fois les biais de prédiction et les biais contradictoires de l'ancien classifieur F_{m-1}. Il y est ajouté après une

étape de recherche linéaire, qui détermine le meilleur poids γ_m à attribuer à h_m dans le nouveau classifieur F_m. Enfin, l'adversaire adapte ses poids en fonction des nouvelles sorties (c'est-à-dire en utilisant l'ensemble d'entraînement $\{(F_m(x_i), s_i)\}_{i=1}^n$). Cela se fait par rétropropagation de gradient. Une représentation schématique de notre approche se trouve dans la figure 1.

4 Expériences

Pour nos expériences, nous utilisons quatre ensembles de données populaires souvent utilisés dans la classification équitable : L'ensemble de données sur les revenus (Adult UCI [DG17]), l'ensemble de données sur le risque de récidive (COMPAS [ALMK16]), l'ensemble de données sur le risque de défaut de crédit (Default [YL09]) et enfin l'ensemble de données sur le marketing bancaire (Bank [MCR14]).

Pour l'ensemble des données d'apprentissage, nous répétons 10 expériences en échantillonnant au hasard deux sous-ensembles, 80% pour l'ensemble d'entraînement et 20% pour l'ensemble de test. Enfin, nous indiquons la moyenne des mesures du taux d'exactitude des prédictions (Accuracy) et d'équité de l'ensemble de test.

Étant donné que les différents objectifs d'optimisation des différentes taches d'équités entraînent des algorithmes différents, nous menons des expériences distinctes pour les deux mesures d'équité qui nous intéressent, la parité démographique (Tableau 1) et l'égalité des chances (Tableau 2). Plus précisément, pour la parité démographique, nous visons une règle-p de 90% pour tous les algorithmes et comparons ensuite le taux d'exactitude des prédictions. En optimisant la tache d'égalité des chances, les résultats sont plus difficiles à comparer. Afin de pouvoir comparer le taux d'exactitude des prédictions, nous avons fait de notre mieux pour obtenir, à chaque fois, un niveau de disparité inférieur à 0.03.

Comme base de référence, nous utilisons un algorithme classique d'arbre de boosting de gradient « inéquitable », Standard GTB, et un réseau neuronal profond, Standard NN.

De plus, pour évaluer si la complexité de l'architecture du réseau antagoniste a un impact sur la qualité des résultats, nous comparons une simple régression logistique antagoniste, FAGTB-1-Unit, avec un réseau neuronal profond complexe, FAGTB-NN.

En plus des algorithmes mentionnés ci-dessus, nous évaluons les algorithmes de « traitement en cours » de l'état de l'art suivants : Wadsworth2018 [WVP18][3], Zhang2018 [ZLM18][4], Kamishima [KAAS12][2] Feldman [FFM+14][2], Zafar-DI [ZVR+17][2] and Zafar-DM [ZVGG17][2].

Pour chaque algorithme et pour chaque ajustement de données, nous obtenons les meilleurs hyperparamètres par optimisation sur une grille de paramètres en validation croisée en 5-fold (spécifique à chacun d'entre eux). Pour rappel, pour le FAGTB, la valeur λ est utilisée pour pondérer les 2 fonctions de coût pendant la phase d'entraînement. Cette valeur dépend exclusivement de l'objectif principal : Par exemple, pour obtenir l'objectif de parité démographique avec une règle-p de 90%, nous choisissons un λ plus faible et donc moins important que pour un objectif de règle-p de 100%. Afin de mieux illustrer l'impact de cet hyperparamètre λ, nous présentons son impact sur le taux d'exactitude des prédictions (Accuracy) et la métrique de la règle-p dans la figure 2 pour l'ensemble de données UCI Adult. Pour cela, nous entraînons l'algorithme FAGTB-NN avec 10 valeurs différentes de λ et nous réalisons chaque expérience 10 fois. Dans le graphique, nous indiquons le taux d'exactitude des prédictions (Accuracy) et la mesure de l'équité de la règle-p, et enfin nous traçons une régression polynomiale de second ordre pour représenter l'effet général.

Pour le GTB standard, nous paramétrons le nombre d'arbres et la profondeur maximale des arbres. Par exemple, pour l'ajustement des paramètres sur l'ensemble de données Bank, une profondeur d'arbre de 3 avec un nombre d'arbres de 800 est suffisant. Pour le Standard NN, nous paramétrons le nombre de couches et de neurones cachées avec une fonction ReLU et nous appliquons une régularisation spécifique de décrochage (dropouts) pour éviter le sur-ajustement. De plus, nous utilisons une optimisation Adam avec une fonction de perte d'entropie croisée binaire. Pour l'ensemble de données UCI Adult par exemple, l'architecture se compose de 2 couches cachées de 16 et 8 neurones, respectivement, et d'activations ReLU. La couche de sortie comprend un seul neurone de sortie avec une activation sigmoïde.

Pour FAGTB, afin d'accélérer la phase d'apprentissage, nous avons décidé de sacrifier une partie des performances en remplaçant l'optimisation unidimensionnelle γ_m par un taux d'apprentissage fixe spécifique pour le prédicteur du classifieur. Tous les hyperparamètres mentionnés ci-dessus, pour les arbres et les

2. `https://github.com/algofairness/fairness-comparison`
3. `https://github.com/equialgo/fairness-in-ml`
4. `https://github.com/IBM/AIF360`

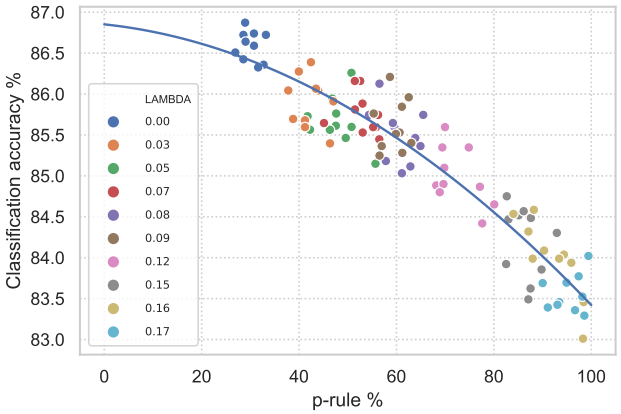

FIGURE 2 – Impact de l'hyperparamètre λ (ensemble de données Adult UCI) : des valeurs plus élevées de λ produisent des prédictions plus équitables, alors que pour des valeur de λ proche de 0 permet de se concentrer uniquement sur l'optimisation du classifieur de prédiction.

réseaux de neurones, sont sélectionnés conjointement. Notez que ces choix ont un impact sur la rapidité de la convergence pour chacun d'entre eux. Par exemple, si le classifieur de prédiction converge trop rapidement, cela peut entraîner des probabilités de prédiction biaisées pendant les premières itérations qui sont difficiles à corriger par la suite par l'adversaire. Pour FAGTB-NN, afin d'obtenir de meilleurs résultats, nous exécutons pour chaque itération de rétropagation de gradient du classifieur de prédiction plusieurs itérations d'entraînement du NN antagoniste. Cela produit un algorithme antagoniste plus persistant. Autrement, le classifieur prédicteur GTB pourrait être prédominant sur l'adversaire. À la première itération, nous commençons par entrainer un GTB biaisé et nous modélisons ensuite le NN antagoniste en fonction de ces prédictions biaisées. Cette approche permet d'avoir une meilleure initialisation de la pondération du NN antagoniste. Elle est plus adaptée au biais spécifique sur l'ensemble des données d'entraînement. Sans cette initialisation spécifique, nous avons rencontré certains cas où le classifieur de prédiction surpasse l'adversaire de manière disproportionnée et tend ainsi à dominé dès le commencement. Par rapport au FAGTB-NN, l'adversaire du FAGTB-1-Unit est moins complexe. Dans ce cas, les deux paramètres de l'adversaire sont choisis au hasard et pour chaque itération de rétropopagation de gradient, un seul est calculé pour l'adversaire.

Pour la parité démographique (Tableau 1), comme prévu, le GTB standard et le NN standard obtiennent les meilleurs taux d'exactitude des prédictions (Accuracy). Cependant, ils sont également les plus biaisés. Par exemple, l'algorithme standard de boosting de gradient obtient une règle-p de 32.6. En comparant les algorithmes d'atténuation, FAGTB-NN obtient le meilleur résultat avec le plus grand taux d'exactitude des prédictions tout en maintenant une égalité de règle-p raisonnablement élevée (90%). Le choix de l'architecture du réseau de neurones complexe de l'adversaire s'est avéré dans tous les cas meilleur qu'une simple régression logistique. Cela est particulièrement le cas pour l'ensemble de données COMPAS où, pour une règle-p similaire, la différence du taux d'exactitude des prédictions est considérable (2.7 points). Rappelons que pour la parité démographique, le classifieur antagoniste n'a qu'une seule variable d'entrée qui est la sortie du classifieur prédictif. Il semble nécessaire de pouvoir segmenter cette entrée de plusieurs manières afin de mieux saisir les informations pertinentes pour prédire l'attribut sensible. Le sacrifice sur le taux d'exactitude des prédictions est moins important pour les ensembles de données Bank et Default. La dépendance entre l'attribut sensible et l'étiquette cible est donc moins importante que pour l'ensemble de données COMPAS. Pour atteindre une règle-p de 90%, nous sacrifions 4.6 points de taux d'exactitude des prédictions (en comparant GTB et FAGTB-NN) pour COMPAS, 0.7 point pour Default et 0.6 point pour Bank.

Dans la figure 3, nous traçons la distribution des probabilités prédites pour chaque attribut sensible S pour 3 modèles différents : Un modèle inéquitable avec $\lambda = 0$, et 2 modèles FAGTB équitables avec $\lambda = 0.06$ et $\lambda = 0.15$, respectivement. Pour le modèle inéquitable, la distribution diffère surtout pour les probabilités les plus faibles. Le deuxième graphique montre une amélioration mais il reste quelques différences. Pour le dernier, les distributions sont sensiblement alignées.

Zhang2018 [ZLM18] a introduit un terme de projection qui garantit que le prédicteur ne se déplace jamais dans une direction qui pourrait aider l'adversaire. Bien que cette approche soit intéressante, nous avons remarqué que ce terme n'améliore pas les résultats pour la parité démographique. Noter que l'algorithme de Wadsworth2018 [WVP18] suit la même approche mais sans terme de projection et obtient des résultats similaires.

Pour la tache d'égalité des chances, l'optimisation min-max est plus difficile à réaliser que la parité démographique. Les mesures d'équité D_{FPR} et D_{FNR} ne sont pas exactement comparables, nous n'avons donc pas réussi à obtenir le même niveau d'équité. Cependant, nous constatons que le FAGTB-NN atteint un meilleure taux d'exactitude des prédictions avec un

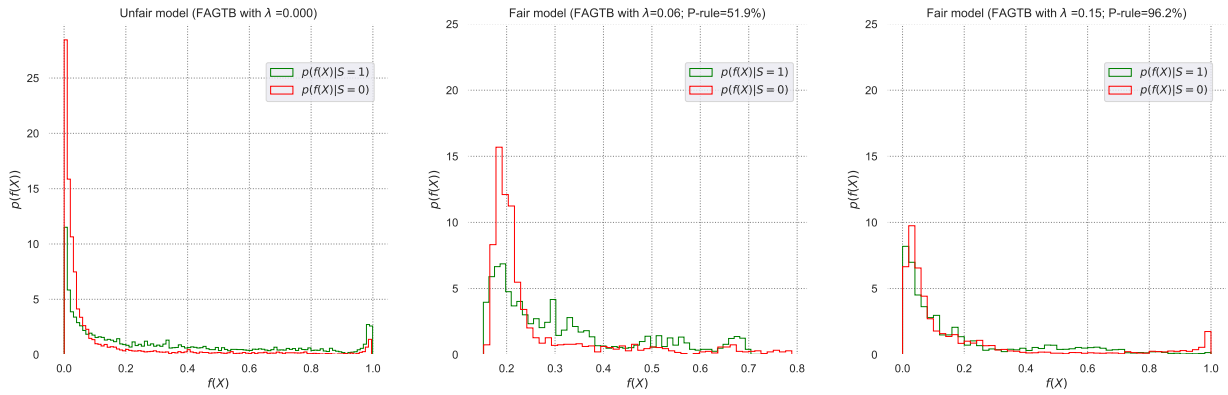

FIGURE 3 – Distributions des probabilités prédites en fonction de l'attribut sensible S (ensemble de données Adult UCI)

TABLE 1 – Résultats pour la parité démographique

	Adult		COMPAS		Default		Bank	
	Acc.	P-rule	Acc.	P-rule	Acc.	P-rule	Acc.	P-rule
Standard GTB	86.8%	32.6%	69.1%	61.2%	82.9%	77.2%	90.8%	48.1%
Standard NN	85.3%	31.4%	67.5%	71.1%	82.1%	63.3%	90.3%	58.6%
FAGTB-1-Unit	84.4%	90.4%	61.8%	90.1%	81.5%	90.1%	90.1%	90.0%
FAGTB-NN	84.9%	90.3%	64.5%	90.0%	82.2%	90.2%	90.2%	90.0%
Wadsworth2018 [WVP18]	83.1%	89.7%	63.9%	90.1%	81.8%	90.0%	90.2%	90.1%
Zhang2018 [ZLM18]	83.3%	90.0%	64.1%	89.8%	81.4%	90.0%	90.0%	90.0%
Zafar-DI [ZVRG15]	82.2%	89.8%	63.9%	89.7%	80.7%	89.8%	89.2%	90.1%
Kamishima [KAAS12]	82.3%	89.9%	63.8%	90.0%	81.1%	90.0%	89.6%	89.9%
Feldman [FFM+14]	-	-	61.4%	90.1%	72.2%	90.2%	-	-

Comparaison de notre approche avec différents algorithmes équitables de l'état de l'art sur le taux d'éxactitude de prédictions et l'équité (règle-p) pour les ensembles de données UCI Adult, COMPAS, Defaut et Bank.

niveau d'équité raisonnable. Concrètement, nous obtenons pour les 4 ensembles de données et pour les deux mesures d'équités (D_{FPR} et D_{FNR}) des valeurs inférieures à 0.02, sauf pour l'ensemble de données Bank où D_{FNR} est égal à 0.07. Pour cet ensemble de données, la plupart des algorithmes de l'état de l'art donnent un D_{FNR} compris entre 0.06 et 0.08. Il s'avère difficile d'obtenir un taux de faux négatifs (FNR) faible car la part totale de l'objectif cible moyen $E(Y)$ est très faible (proche de 11.7%) sur cet ensemble de données. Une manière possible de résoudre ce problème de déséquilibre de la classe cible pourrait être d'ajouter un poids spécifique directement dans la fonction de perte. Nous remarquons également que la différence de résultats entre FAGTB-1-Unit et FAGTB-NN est beaucoup plus importante, une des raisons possibles étant qu'une régression logistique en adversaire ne peut pas conserver une quantité suffisante d'informations pour prédire l'attribut sensible.

5 Conclusion

Dans ce travail, nous avons développé une nouvelle approche pour entraîner des algorithmes de boosting de gradient équitable. En comparaison par rapport aux autres algorithmes de l'état de l'art, notre méthode s'avère plus efficace en termes de taux d'exactitude des prédictions tout en obtenant un niveau d'équité similaire.

Références

[ALMK16] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. ProPublica, May 23, 2016, 2016.

TABLE 2 – Résultats pour l'égalité des chances

	Adult			COMPAS			Default			Bank		
	Acc.	D_{FPR}	D_{FNR}	Acc.	D_{FPR}	D_{FNR}	Acc.	D_{FPR}	D_{FNR}	Acc.	D_{FPR}	D_{FNR}
Standard GTB	86.8%	0.06	0.07	69.1%	0.12	0.20	82.9%	0.02	0.04	90.8%	0.04	0.06
Standard NN	85.3%	0.07	0.10	67.5%	0.09	0.15	82.1%	0.02	0.05	90.3%	0.05	0.08
FAGTB-1-Unit	86.3%	0.02	0.02	65.1%	0.03	0.12	82.1%	0.00	0.01	89.7%	0.02	0.07
FAGTB-NN	86.4%	0.02	0.02	66.2%	0.01	0.02	82.5%	0.00	0.01	90.3%	0.01	0.07
Wadsworth2018 [WVP18]	84.9%	0.02	0.03	65.4%	0.02	0.03	81.2%	0.01	0.02	89.4%	0.01	0.07
Zhang2018 [ZLM18]	84.8%	0.03	0.03	64.9%	0.03	0.02	81.9%	0.00	0.01	89.8%	0.00	0.07
Zafar-DM [ZVGG17]	83.9%	0.03	0.09	64.3%	0.09	0.17	81.0%	0.00	0.03	89.5%	0.01	0.08
Kamishima [KAAS12]	82.6%	0.06	0.24	63.6%	0.08	0.11	80.5%	0.00	0.04	89.3%	0.00	0.08
Feldman [FFM+14]	80.6%	0.07	0.05	61.1%	0.03	0.03	71.8%	0.02	0.02	87.1%	0.05	0.06

Comparaison de notre approche avec différents algorithmes communs équitables en termes de taux d'exactitude des prédictions et d'équité (D_{FPR}, D_{FNR}) pour les ensembles de données UCI Adult, COMPAS, Default et Bank.

[CHKV19] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with fairness constraints : A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*, pages 319–328. ACM, 2019.

[DG17] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[DHP+11] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. Fairness Through Awareness. 2011.

[FFM+14] Michael Feldman, Sorelle Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. pages 1–28, 2014.

[GPAM+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[HPS16] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of Opportunity in Supervised Learning. pages 1–22, 2016.

[KAAS12] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In Peter A. Flach, Tijl De Bie, and Nello Cristianini, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 35–50, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[LKC16] Gilles Louppe, Michael Kagan, and Kyle Cranmer. Learning to Pivot with Adversarial Networks. (Nips), 2016.

[MCR14] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62, 06 2014.

[WVP18] Christina Wadsworth, Francesca Vera, and Chris Piech. Achieving Fairness through Adversarial Learning : an Application to Recidivism Prediction. (July), 2018.

[YL09] I-Cheng Yeh and Che-hui Lien. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Syst. Appl.*, 36(2) :2473–2480, March 2009.

[ZLM18] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating Unwanted Biases with Adversarial Learning. *Association for the Advancement of Artificial Intelligence*, jan 2018.

[ZLZA17] Chongsheng Zhang, Changchang Liu, Xiangliang Zhang, and George Almpanidis. An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications*, 82, 04 2017.

[ZVGG17] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. Fairness Beyond Disparate Treatment & Disparate Impact. pages 1171–1180, 2017.

[ZVR+17] Muhammad Bilal Zafar, Isabel Valera, Manuel Rodriguez, Krishna Gummadi, and Adrian Weller. From parity to preference-based notions of fairness in classification. pages 229–239, 2017.

[ZVRG15] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. Fairness Constraints : Mechanisms for Fair Classification. 54, 2015.

Algorithm 1 Fair Adversarial Gradient Tree Boosting

Input : Ensemble d'entraînement $(x_i, s_i, y_i)_{i=1}^n$, un nombre d'itérations M, un taux d'apprentissage antagoniste α, une fonction de perte différenciable $\mathcal{L}_F$ pour le classifieur de sortie et $\mathcal{L}_A$ pour l' antagoniste.

Initialisation : Calculer la valeur constante :

$$F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{n} \mathcal{L}_{F_i}(\gamma)$$

Initialiser les paramètres θ_A du réseau neuronal $A(x)$

Pour $m = 1$ **to** $M - 1$ **:**

(a) Calculer les pseudo-résidus :

$$r_{im} = -\left[\frac{\partial \mathcal{L}_{F_i}(F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$$
Pour $i = 1, \ldots, n$

(b) Calculer les pseudo-résidus de l'adversaire à partir de l'entrée $F_{m-1}(x_i)$:

$$t_{im} = -\left[\frac{\partial \mathcal{L}_{A_i}(F(x_i; \theta_A))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$$
Pour $i = 1, \ldots, n$

(c) Calculer la fonction de perte de l'entraînement :

$$u_{im} = r_{im} - \lambda * t_{im}$$

(d) Entraîner un classifieur $h_m(x)$ aux pseudo-résidus sur l'entraînement $\{(x_i, u_{im})\}_{i=1}^n$

(e) Calculer le coefficient γ_m en résolvant le problème d'optimisation unidimensionnel :

$$\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{n} \mathcal{L}_{F_i}\left(F_{m-1}(x_i) + \gamma * h_m(x_i)\right) - \lambda * \mathcal{L}_{A_i}(F_{m-1}(x_i) + \gamma * h_m(x_i); \theta_A).$$

(f) Mettre à jour le modèle d'apprentissage :

$$F_m(x_i) = F_{m-1}(x_i) + \gamma_m * h_m(x_i)$$

(g) Ajuster l'adversaire A sur les nouvelles sorties sur l'entraînement $\{(F_m(x_i), s_i)\}_{i=1}^n)$

$$\theta_A := \theta_A - \alpha * \frac{\partial \mathcal{L}_{A_i}(F_m(x_i); \theta_A)}{\partial \theta_A}$$