nexialog CONSULTING

Techniques d'interprétabilité des modèles de Machine Learning

Nexialog Consulting, Paris, France

2 mars 2022

Résumé

L'interprétabilité est l'un des obstacles techniques à franchir pour pleinement utiliser le Machine Learning dans le domaine bancaire. Dans ce papier, nous présentons différentes techniques permettant d'interpréter les algorithmes de Machine Learning. Nous exhibons ainsi chaque technique tout en présentant la théorie sous-jacente.

Introduction

Les algorithmes de Machine de Learning sont utilisés dans quasiment tous les domaines d'activité. Cette intégration s'est faite grâce à leur relative performance comparativement aux méthodes ou modèles classiques. Cependant, les acteurs de certains secteurs rigoureusement réglementés comme ceux de la Banque/Finance sont encore retissent quant à l'utilisation de ces algorithmes. Les raisons principales justifiant ce rechignement est le manque de compréhension par l'humain des mécanismes qui fondent les valeurs prédites des algorithmes de Machine Learning (ML). Plus précisément, l'interprètabilité est le degré ou la mesure dans laquelle un humain peut prédire de manière cohérente le résultat du modèle, Miller (2017) [1]. En effet, les algorithmes de Machine Learning ont longtemps été considérés comme des boîtes noires par conséquent inadaptés au domaine Bancaire où le régulateur exige une certaine transparence (explicabilité et interprétabilité) des modèles utilisés. Cependant, les avis du régulateur ne cessent d'évoluer sur le sujet. Les récentes discussions [2] de l'EBA (Eropean Banking Autority) en vue d'approuver davantage l'utilisation du Machine Learning dans de l'approche IRB (Internal Rating-Based approach) confirme cette tendance. De cette discussion ressort que l'un des défis majeurs que devrait relever les Banques afin d'utiliser pleinement ces algorithmes notamment dans le cadre réglementaire (IBA) est l'interprétabilité. La présente note vise à faire un état de l'art des différentes techniques permettant d'interpréter les algorithmes de Machine Learning. Il ambitionne de faire le lien entre la théorie sous-jacente à chaque technique et application métier. Ainsi, il présentera dans un premier temps le jeu de données qui servira de base d'application lequel (section 2). Par la suite, nous présenterons les fondements théoriques et l'implémentation de chaque outil.

1. Miller, Tim. "Explanation in artificial intelligence : Insights from the social sciences." arXiv Preprint arXiv :1706.07269. (2017)

2. https://www.eba.europa.eu/regulation-and-policy/model-validation/discussion-paper-machine-learning-irb-models

1 Notions et Prérequis

1.1 Nuance entre interprétabilité et explicabilité

Plusieurs techniques permettant de comprendre et de rendre prévisible pour un humain les résultats des prédictions d'un algorithme complexe seront abordées dans cette section. Mais avant, il convient de préciser deux notions qui prêtent parfois à confusion.

Interprétabilité

Comme évoquée en introduction, l'interprétabilité est la mesure dans laquelle un humain peut comprendre les causes d'une décision (d'une prédiction) d'un modèle.

Explicabilité

L'explicabilité peut se comprendre comme la facilité avec laquelle un humain peut reproduire les résultats d'un modèle.

1.2 Classification des techniques d'interprétabilité

On classifie généralement les techniques d'interprétabilité selon deux axes :

Agnostique ou spécifique

Une technique agnostique au modèle fonctionne en considérant le modèle comme une véritable boîte noire et s'applique donc théoriquement à tous les modèles. Cela ne signifie pas pour autant qu'une technique agnostique présentera les mêmes performances ou les mêmes garanties théoriques selon le modèle à interpréter.

A l'inverse il existe des techniques spécifiques au modèle étudié, on citera les modèles à base d'arbre de décision et les réseaux de neurones tout particulièrement. L'objectif de ce papier n'est que de présenter les techniques agnostiques.

Global ou Local

Une méthode peut tenter d'expliquer le rôle d'une variable dans l'ensemble des prédictions (globalement) ou dans une prédiction donnée (localement). L'avantage du premier vient du côté plus synthétique tandis que le second permet d'être plus précis.
De manière générale, le choix entre méthode globale ou locale dépend beaucoup du type d'interprétation que l'on cherche. Par exemple, dans le cadre du Credit Scoring, on choisira plutôt une méthode locale pour interpréter le score d'un individu donné. Dans le cas d'un modèle de stratégie d'investissement automatisée, on pourra être plus intéressé à l'exposition globale de la stratégie aux mouvements des différents instruments financiers qui forment les variables du modèles.

1.3 Notions et formalisation propres au Machine Learning

Un modèle de Machine Learning sera formalisé par une fonction $\hat{f} : X \mapsto \hat{y}$, construite par comme estimateur d'une fonction cible $f : X \mapsto y$, avec X le vecteur des variables d'entrée du modèle, y la cible à prédire et $\hat{y}$ la prédiction.
Les différentes variables d'entrée sont appelées variables explicatives ou prédicteurs. Le modèle $\hat{f}$ est construit par apprentissage, en minimisant une erreur de prédiction donnée L sur un jeu de données $(X, y) = (X_i, y_i)_{i=1..n}$ telle que :

$$\hat{f} = \arg\min_{g} L(y, g(X))$$

Parmi les fonctions d'erreurs communes, on citera :
- la Mean Absolute Error (MAE)

$$L(y, g(X)) = \sum_i |y_i - g(X_i)|$$

- la Root Mean Square Error (RMSE)

$$L(y, g(X)) = \sqrt{\sum_i (y_i - g(X_i))^2}$$

2 Données

Afin d'illustrer l'efficacité de chaque technique sur un exemple concret, nous allons implémenter les différents outils sur un jeu de données. Le jeu de données utilisé est celui de la base Pima. Cette base permet de décrire la survenance du diabète chez des femmes indiennes en fonction de certaines caractéristiques (diagnostic cliniques). C'est une base assez connue dans le domaine du ML (voir *uci* pour des descriptions plus détaillées). Le but n'est pas forcement de construire un modèle de manière aboutie, mais d'en estimer un de manière sommaire et ensuite tester les différents outils. L'algorithme qui sera construit est l'eXtreme Gradient Boosting (*XGBoost*) dont on sait qu'il n'est pas interprétable. Pour benchmarker ces outils, nous avons estimer un modèle logistique (GLM). Ainsi le résultat de chaque technique (amplitude et sens de l'influence d'une feature) sera comparé à celui de la GLM (signe du coefficient et l'ordre de grandeur du coefficient).

Résultat de l'estimation du GLM

Afin que les coefficients des variables de la régression logistique soient directement comparables, les variables ont été normalisées (par la méthode min-max).

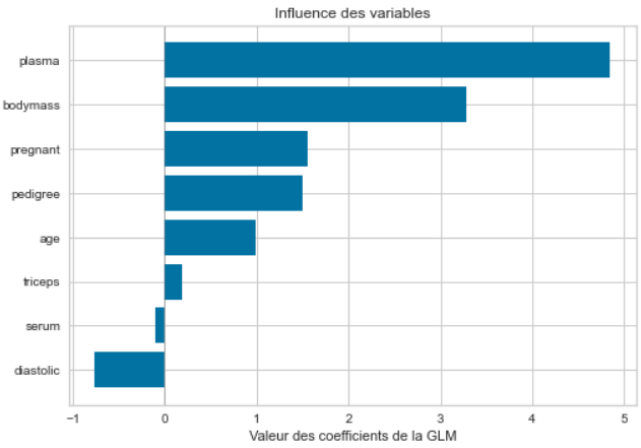

FIGURE 1 – Coefficient de la régression logistique

Deux variables contribuent négativement au risque d'apparition du diabète chez un individu dans la population étudiée. L'ordre d'importance des variables est clairement établie. Ainsi les variables **triceps** et **serum** sont les moins importantes selon la régression logistique.
Le GLM est estimé en tant que régression logistique avec régularisation L2 (avec un coefficient de régularisation de 1). Sa précision sur les données après entraînement est de 78%.

Résultats du XGBoost

L'algorithme de Gradient Boosting est entraîné avec un taux d'apprentissage de 0.1, 100 arbres de profondeur maximale 3. Il obtient un score de précision de 90% sur les données d'entraînement.

3 Permutation Features Importances

Introduit par Breiman (2001), la feature importance est l'un des premiers outils permettant de mesurer l'influence globale d'une variable dans un modèle. La feature importance est basée sur l'augmentation de l'erreur de prédiction du modèle après perturbation ou permutation des valeurs d'une variable, ce qui rompt la

relation initiale entre la variable et la valeur observée de la variable cible. En effet, une variable est importante dans un modèle si sa perturbation induit une augmentation significative de l'erreur de prédiction. A l'opposé, une variable est peu influence dans un modèle lorsque la perturbation de celle-ci a peu ou pas de conséquence sur la prédiction. Fisher, Rudin, and Dominici (2018) [3] ont résume le calcul de la feature importance en plusieurs étapes. Soient $\hat{f}$ le modèle, $\mathbf{X}$ la matrice des prédicteurs, $\mathbf{y}$ la variable cible et $\mathbf{L}(y, \hat{f})$ la mesure du performance (MAE, RMSE, MQQE, etc.).

— Estimation du modèle et calcul de son erreur $e_{\hat{f}} = L(y, \hat{f}(X))$.

— Pour $j \in \{1, ..., p\}$:

- On génère une nouvelle matrice X_{pert} par perturbation de la variable j dans la matrice X ;

- Estimation de l'erreur de prédiction $e_{pert} = L(y, \hat{f}(X_{pert}))$ basée sur la prédiction du modèle avec les prédicteurs perturbés ;

- Calcul de la feature importance comme un quotient $FI = \frac{e_{pert}}{e_{\hat{f}}}$ ou une différence $FI = e_{pert} - e_{\hat{f}}$.

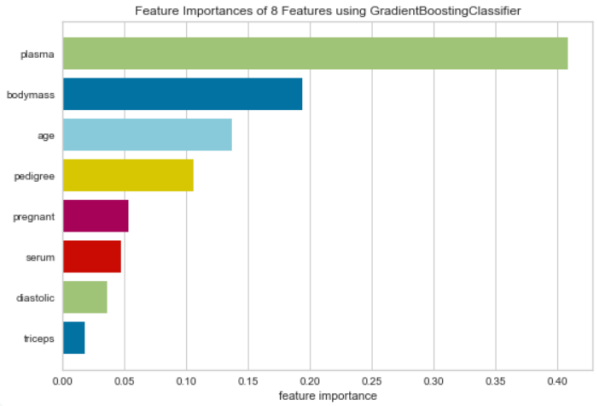

FIGURE 2 – Permutation Feature importance pour le GXBoost

La variable qui influence le plus le risque, pour une femme, d'avoir le diabète est la concentration de glucose dans le sang pendant (la variable plasma). L'épaisseur du pli cutané du triceps (triceps) a une faible importance sur la probabilité d'avoir la maladie dans la population d'étude.

Avantage

La *permutation feature importance* est théoriquement fondée tout type de modèle, très rapide à calculer et à interpréter.

Limite

La permutation Feature Importance ne donne pas le sens de l'influence de la variable.

4 Partial Dependence Plot (*PDP*)

Le PDP est une technique d'interprétation globale qui permet d'avoir l'effet marginal d'un ou deux prédicteurs sur la prédiction du modèle de Machine Learning. Il permet de voir comment un prédicteur influence en moyenne la prédiction, de déterminer le type de relation entre la variable cible et les prédicteurs : linéaire, simplement monotone ou plus complexe.

Soient S l'indice de la variable que l'on souhaite étudier et C représentant les autres variables. La fonction de dépendance partielle est définie dans ce cas par : $PDP_S(x_S) = \mathbb{E}_{x_C}\left[\hat{f}(x_S, x_C)\right] = \int \hat{f}(x_S, x_C) d\mathbb{P}(x_C)$ x_S sont les caractéristiques (au plus deux) pour lesquelles la fonction de dépendance partielle devrait être tracée et x_C les autres prédicteurs de la base utilisés dans l'estimation du modèle $\hat{f}$. Elle est estimée par la moyenne de la valeur de la fonction objective de chaque observation.

$$PDP_S(x_S) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_S, x_C^{(i)})$$

n la taille de la base. Dans le cadre d'un d'une modèle classification dont la sortie du modèle est une probabilité comme c'est généralement le cas en risque de crédit, le PDP permet de voir l'évolution de la probabilité de la classe prédite (ou tout simplement de l'évènement étudié) étant donnée x_S. Pour une variable catégorielle, l'estimation du PDP se fait modalité par modalité. Et pour estimer le PDP d'une catégorie, on contraint toutes les observations à être dans une même catégorie. Comme avantage, le PDP est assez intuitif. La fonction de dépendance partielle pour une valeur donnée d'un prédicteur n'est que la prédiction moyenne si nous forçons tous les individus à prendre cette valeur pour ce prédicteur. Il est relativement facile à implémenter et permet surtout une interprétation causale. Les principaux inconvénients sont le fait qu'il ne peut être calculé que pour deux variables à la fois et suppose une décorrélation entre les prédicteurs pour lesquels on calculs le PDP et les autres.

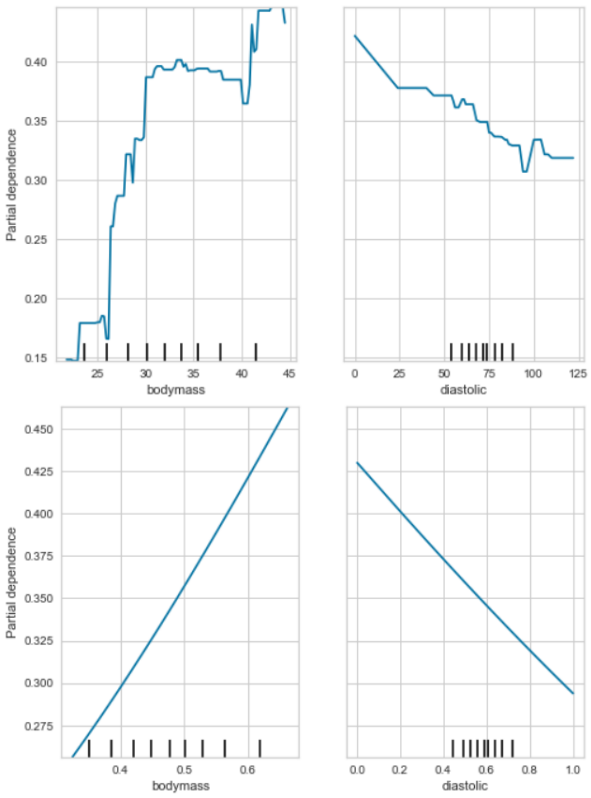

FIGURE 3 – Partial Dependence Plot XGBoost vs GLM

A la lecture du graphique précédent, nous pouvons faire deux constats (i) les deux variables n'ont pas la même importance ; (ii) l'indice de masse corporelle *bodymass* augmente le risque d'avoir le diabète tandis que la pression sanguine (diatolic) a une corrélation négative avec la probabilité pour une femme d'avoir le diabète.
Le graphique de dependance partielle est cohérent avec le résultat de la GLM. On peut voir que par l'amplitude des variations que la variable *bodymass* est plus importante que la variable *diatolic*. Aussi, le graphique montre que *bodymass* globalement influence positivement le risque d'avoir le diabète dans la population étudiée.

4.1 Individual Conditional Expectation (ICE)

L'ICE est l'équivalent local du PDP. L'ICE montre pour chaque observation i l'évolution de la prédiction en fonction de la ou les caractéristiques considérées x_S.

$$ICE_{S,i}(x_S) = \hat{f}(x_S, x_C^{(i)})$$

Il permet ainsi de visualiser la dépendance de la prédiction à une caractéristique donnée pour chaque observation prise séparément, ce qui donne une ligne par instance, contre une ligne globale dans les PDPs. La moyenne des courbes ICE redonne la courbe PDP associée à la même variable x_S.
Ce graphique est intéressant pour comprendre le sens de l'influence et la magnitude d'une variable dans la prédiction du modèle. Une étude des résultats du PDP/ICE sur modèle interprétable du type GLM afin de voir le niveau de concordance avec les coefficients (signe et amplitude) pourra être menée dans un autre papier.

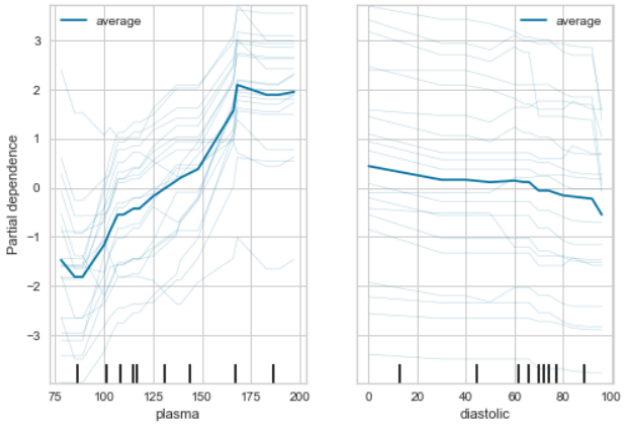

FIGURE 4 – ICE XGBoost

4.2 Accumulated Local Effects (ALE)

L'inconvénient principal des méthode PDP (et ICE) réside dans l'hypothèse de départ de décorrélation entre la variable étudié et les autres. Cette hypothèse n'est en pratique quasiment jamais vérifiée. De ce fait, l'estimation des courbes passe par l'évaluation de points irréalistes : il est par exemple irréaliste voire impossible d'avoir une personne très jeune ayant été enceinte pendant de très nombreux mois. Cela peut introduire un biais important, mais qui peut être globalement corrigé :

Intuition et principe

La première solution à ce problème consiste à calculer les points de courbes PDP seulement à partir de données existantes. Cela revient théoriquement à faire la moyenne conditionnellement à une valeur de x_S, c'est le M-Plot :
$M_S(x_S) = \mathbb{E}_{x_C}\left[\hat{f}(x_S, x_C)|X_S = x_S\right]$
Le problème de cette méthode est que l'effet représenté par la courbe n'est plus celui de la variable x_S seule mais également une partie des effets des variables qui lui sont corrélées. Le biais est donc théoriquement corrigé mais toujours présent dans la pratique de l'interprétation des courbes. La méthode ALE permet de reprendre le principe du PDP sans le biais induit par la corrélation entre les variables sans prendre en compte l'effet des variables corrélées comme le M-Plot. Le principe permettant d'éliminer l'effet des autres variables est l'utilisation de différentielles de prédictions plutôt que les prédictions elles-mêmes pour obtenir des "Local Effects" : $LE_S(x_S) = \mathbb{E}_{x_C}\left[\frac{d\hat{f}(x_S,x_C)}{dx_S}|X_S = x_S\right]$
En intégrant et en centrant on obtient les "Accumulated Local Effects" (C est choisi pour que la moyenne des ALE soit nulle) : $ALE_S(x_S) = \int_{z_0}^{x_S} \mathbb{E}_{x_C}\left[\frac{d\hat{f}(z,x_C)}{dx_S}|X_S = z\right] dz$ - C

Estimation

Pour estimer les ALE, nous utiliserons des différences simples plutôt que des différentielles. On calculera les différences à partir d'intervalles de x_S délimités par des valeurs z_k formant des voisinages N_k. Il s'agira pour calculer un point $ALE_S(x_S)$ de :

1. Considérer l'indice $k(x_S)$ tel que l'intervalle $N_{k(x_S)} = [z_{k(x_S)-1}, z_{k(x_S)}]$ contienne x_S
2. Pour chaque intervalle $N_k, k \leq k(x_S)$, calculer le Local Effect, c'est-à-dire la moyenne pour chaque point de donnée dans N_k des différences de prédictions en z_k et z_{k-1} : $LE_{S,k}(x_S) = \frac{1}{n_k} \sum_{i, x_C^{(i)} \in N_k} \hat{f}(z_k, x_C^{(i)}) - \hat{f}(z_{k-1}, x_C^{(i)})$
3. Sommer les Local Effects pour $k = 1..k(x_S)$ pour obtenir l'Accumulated Local Effect non centré : $ALE'_S(x_S) = \sum_{k=1}^{k(x_S)} LE_{S,k}(x_S)$
4. Centrer pour obtenir l'ALE final : $ALE_S(x_S) = ALE'_S(x_S) - \frac{1}{n} \sum_i ALE'_S(x_S^{(i)})$

Pour illustrer l'utilité de la méthode ALE, prenons le cas de la prédiction du nombre de vélos vendus à partir notamment de la température. On remarque que le PDP sous estime la diminution de vélo vendus induit par de très hautes températures par rapport à la courbe ALE. Cela est sans doute dû à la corrélation entre température et saison et au fait que le PDP évalue, lorsque la température est haute, des points de donnée irréalistes attestant d'une grande température en hiver.

Les inconvénients de la méthode ALE sont qu'elle nécessite de plus de données que le PDP pour être robuste, de par le fait que les moyenne se font sur de petits intervalles et qu'elle est plus lourde à implémenter et calculer.

5 Modèle de substitution (Surrogate Models)

La substitution est une technique d'interprétabilité basée sur des modèles interprétables entraînés pour approximer les prédictions du modèle du Machine Learning. On peut appliquer un modèle de substitution global ou local.

5.1 Global Surrogate

L'idée est d'approximer la fonction de prédiction du modèle de Machine Learning f par une fonction (la plus proche) de prédiction d'un modèle interprétable (LM, GLM, Arbre de décision, etc.). Le processus peut se décomposer en plusieurs étapes :

1. Construction du modèle de ML non interpétable ;
2. Sélection d'un jeu de données X. Ce jeu de données peut être celui qui a servi à la construction du modèle précédent (en partie ou en totalité) ;
3. Faire la prédiction du modèle de ML ($\hat{y}$) sur le jeu de données X ;
4. Entraîner le modèle interprétable (ou de substitution) sur le jeu de données X en prenant $\hat{y}$ comme variable cible ;
5. Évaluer les performances du modèle interprétable c'est-à-dire mesurer à quel point le modèle de substitution reproduit les prédictions du modèle de ML.
6. Interpréter le modèle de substitution.

L'une des mesures de performances habituelles utilisée pour évaluer la capacité du modèle interprétable à répliquer les prédictions du modèle de ML est le R^2. Elle est définie par :

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (\hat{y}_i^* - \hat{y}_i)^2}{\sum_{i=1}^{n} (\hat{y}_i - \bar{\hat{y}})^2}$$

$\hat{y}^*$: Le vecteur des prédictions du modèle de substitution ;
$\hat{y}$: Le vecteur des prédictions du modèle de ML ;
$\bar{\hat{y}} = \frac{1}{n} \sum_{i=1}^{n} \hat{y}_i$: La moyenne des prédictions du modèle de ML ;
R^2 : La part de la variabilité captée par le modèle de substitution permet de juger de la pertinence de ce dernier. Ainsi, un R^2 *proche* de 1 signifie que le modèle substitution peut être utiliser pour expliquer les prédictions du modèle de ML. Il faut cependant retenir que si $R^2 = 1$, alors il faudra directement estimer le modèle interprétable sans utiliser un algorithme plus complexe car ce dernier ne présente aucun intérêt.

L'utilisation de modèle de substitution global permettra rarement d'obtenir les meilleures qualités d'interprétation, sinon cela veut dire que le modèle d'origine peut être tout simplement remplacé par le modèle de substitution. Cependant cette méthode présente l'avantage de pouvoir quantifier la qualité d'interprétation obtenue ainsi de la flexibilité : il s'agira de trouver le juste milieu entre le choix du modèle de substitution offrant le type d'interprétation le plus pertinent selon le cas d'usage et celui offrant l'interprétation la plus globale via le calcul du R^2.

5.2 Local Surrogate Models (LIME)

C'est la version locale des modèles de substitution globale. LIME permet ainsi d'expliquer les prédictions individuelles du modèle de ML non interprétable. Cette technique a été proposée et concrètement implémentée par Ribeiro, Marco Tulio et al (2016). Au lieu de construire un modèle de substitution global, LIME se concentre sur la construction de modèles de substitution locaux pour expliquer les prédictions individuelles. L'idée posée par les auteurs est assez intuitive. Il s'agit d'entraîner le modèle de substitution interprétable sur un nouveau jeu de données formé des échantillons localement perturbés.
Pour l'étude de l'interprétation locale au point x_i, on simule des perturbations x_{ik} autour de x_i pour lequels on calcule les prédictions y_{ik} du modèle non interprétable. On entraîne ensuite le modèle de substitution sur le jeu de données $(x_{ik}, y_{ik})_{k=1..n}$ ainsi obtenu.

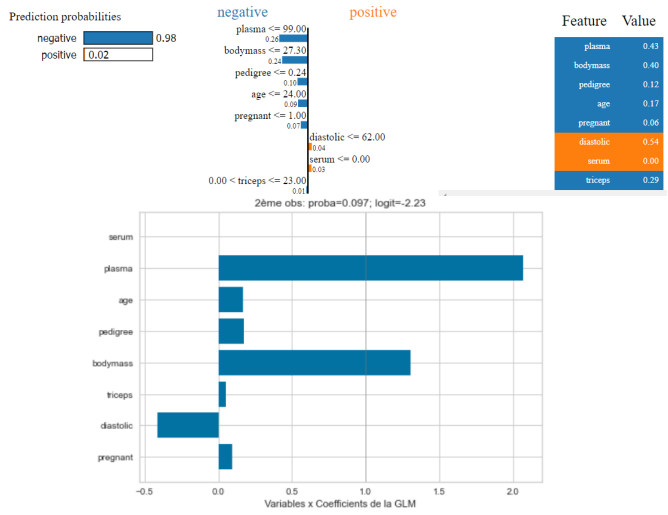

FIGURE 5 – LIME vs GLM estimée pour la 2ème observation de la base

Le premier graphique (au dessus) montre l'estimation du LIME pour une observation de la base (choisie aléatoirement). Le modèle interprétable utilisé est la régression Ridge. Pour rappel, le modèle de ML non interprétable fitté est le XGBoost. La lecture montre que cette observation a plus de 90% d'être classée négative au diabète. Cette classification est induite par 2 des 8 prédicteurs. Il est difficile d'analyser profondément et séparément le sens d'influence des prédicteurs sans avoir au préalable fait une analyse des interaction (ce qui est secondaire dans le contexte de ce papier). Le résultat du LIME est corroboré par ceux de la régression logistique. En effet, le second graphique présente le produit du coefficient (issue de la GLM) par la valeur prise par l'observation. Ainsi hormis les variables *serum* et *diatolic* qui respectivement n'ont aucunes contributions (la valeur prise par l'individu est nulle) et réduit le risque de diabète, toutes autres variables sont des facteurs de risques.

Avantages

Adapté aux données non structurées (textes, images) pour lesquels on peut facilement simuler des perturbations, LIME tient sa popularité de l'hypothèse que l'on peut peut localement faire une approximation linéaire ou par un arbre de décision simple pour beaucoup de modèles.

Limites

Le point faible de LIME tient dans son instabilité. Une approximation locale peut très bien fonctionner pour certains points de données mais être fausse pour d'autres ; pour ces points-ci, il peut suffire d'un changement de jeu de données perturbés pour obtenir des interprétations différentes. De plus, pour qu'une approximation locale soit pertinente, il faut bien définir le voisinage dans lequel on autorise les perturbations, étape assez difficile qui peut également engendrer de l'instabilité si le voisinage est mal défini.

6 Shapley Values

La valeur de shapley est basée sur la théorie des jeux coopératifs. La valeur de shapley est calculée de manière individuelle. Dans ce contexte de jeu coopératif, il faut préciser les termes suivants :

— Le Jeu : prédiction liée à une observation donnée ;
— Les Joueurs : ensemble des valeurs prises par l'observation sur les différentes variables ;
— Le Gain : valeur prédite de l'observation moins la prédiction moyenne pour toutes les observations.

Nous allons présenter de manière plus détaillée le calcul du Shapley value.

6.1 Formalisation

6.1.1 Cas des modèles linéaires

L'idée du shapley value est de quantifier l'effet de chaque variable sur la prédiction d'un point. Cette quantification est simple dans le cas du modèle linéaire. En effet, considérons la fonction de prédiction du modèle linéaire $\hat{f}$.

$$\hat{f}(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$$

x étant une observation du jeu de données, les x_j, $j = 1...p$ sont les valeurs prises par cette observation et $(\beta_0, ..., \beta_p) \in \mathbb{R}^{p+1}$ le vecteur des paramètres du modèle.

Soit ϕ_j la contribution de la j-ème variable à la prédiction $\hat{f}(x)$.

$$\phi_j(\hat{f}) = \beta_j x_j - \mathbb{E}(\beta_j X_j)$$

$\mathbb{E}(\beta_j X_j)$ est l'estimation de l'effet moyen de la variable j. Ainsi, la contribution est la différence entre l'effet de la variable et l'effet moyen. Pour avoir la contribution totale de cette observation, il suffit de sommer les contributions de chaque variable.

$$\begin{aligned}\sum_{j=1}^{p} \phi_j(\hat{f}) &= \sum_{j=1}^{p} (\beta_j x_j - \mathbb{E}(\beta_j X_j)) \\ &= (\beta_0 + \sum_{j=1}^{p} \beta_j x_j) - (\beta_0 + \sum_{j=1}^{p} \mathbb{E}(\beta_j X_j)) \\ &= \hat{f}(x) - \mathbb{E}(\hat{f}(X))\end{aligned}$$

La contribution de l'individu est bien la différence entre la valeur prédite pour ce dernier et la prédiction moyenne. Cependant, il faut remarquer que la présence de pondérations des variables (paramètre du modèle) ainsi que la linéarité du modèle simplifie énormément le calcul de la contribution d'un individu. Qu'en est-il du calcul de cette contribution pour un modèle quelque en particulier non-linéaire ?

6.1.2 Cas général

Soit *val* une fonction de valeur (qui sera explicitée ultérieurement) des variables dans leur univers S. De

manière analogue au cas du modèle linéaire, la contribution de la variable j est donnée par :

$$\phi_j(val) = \sum_{S \subseteq \{1...p\} \setminus \{j\}} \frac{|S|!(p-|S|-1)!}{p!} \left(val(S \cup \{j\}) - val(S)\right)$$

S : un sous-ensemble des variables du jeu de données ;
$|S|$: le cardinal de l'ensemble S ;
p : le nombre de variables ;
x : le vecteur de valeurs des variables pour une observation.
$val_x(S)$: la prédiction des valeurs des variables dans S marginalisées par rapport aux variables n'appartenant pas à S.

$$val_x(S) = \int \hat{f}(x_1, ..., x_p) d\mathbb{P}_{x \notin S} - \mathbb{E}_X(\hat{f}(X))$$

Ainsi, la contribution totale de l'individu est :

$$\begin{aligned}\sum_{j=1}^{p} \phi_j(val) &= \sum_{j=1}^{p} \sum_{S \subseteq \{1...p\} \setminus \{j\}} \frac{|S|!(p-|S|-1)!}{p!} \left(val(S \cup \{j\}) - val(S)\right) \\ &= \hat{f}(x) - \mathbb{E}_X(\hat{f}(X))\end{aligned}$$

On peut remarquer de par sa formulation que la valeur de shapley possède plusieurs propriétés. Entre autre la symétrie, l'additivité, etc. Cette propriété d'additivité permet de garantir son utilisation pour les algorithmes basés sur le principe du bagging. [3]

6.2 Estimation

Pour tous les individus, calculer les ϕ_j des différentes variables peut devenir problématique dès lors que le nombre de variables devient important. Pour palier ce problème computationnel, Strumbelj et al. (2014) [6] ont proposé une approximation du Shapley par Monte-Carlo.
On considère toujours un point de donnée x et sa composante x_j à expliquer. Il s'agira de tirer aléatoirement un autre point de donnée z_m et une permutation aléatoire p_m dans $\{1, 2\}^J$ (avec J le nombre de variables et donc la taille de x et z_m). On construira un vecteur x^m avec les composantes de x et z_m mélangées, choisies selon la permutation p_m :

— Si $p_{mk} = 1$ alors $x_k^m = x_k$
— Si $p_{mk} = 2$ alors $x_k^m = z_k$

Soient x_{+j}^m et x_{-j}^m des vecteurs x^m à qui ont a respectivement forcé la présence et l'absence de la composante x_j. On obtient une estimation grossière de la Shapley value en faisant la différence :

$$\hat{\phi_j}^m = \hat{f}(x_{+j}^m) - \hat{f}(x_{-j}^m)$$

On réitère cette opération M fois et on fait la moyenne des $\hat{\phi_j}^m$ pour obtenir l'estimateur final :

$$\hat{\phi_j} = \frac{1}{M} \sum_{m=1}^{M} \left(\hat{f}(x_{+j}^m) - \hat{f}(x_{-j}^m)\right) = \frac{1}{M} \sum_{m=1}^{M} \hat{\phi_j}^m$$

3. Le bagging est une méthode d'apprentissage ensembliste, abbréviation de bootstrap aggregating, qui consiste à aggréger plusieurs modèles entraînés chacun sur des échantillons bootstrap différents des données. C'est le principe sous-jacent du célèbre Random Forest

6.3 Interprétation

La valeur de shapley $\phi_j^{(i)}$ pour l'individu i et la variable j est la contribution de la valeur x_{ij} à la prédiction $\hat{y}_i$ par rapport à la prédiction moyenne dans le jeu de données.

6.4 Avantages

Pour une observation, la différence entre la prédiction pour une variable et la prédiction moyenne est répartie équitablement entre les valeurs des variables. Son avantage le plus important est qu'elle est basée sur une théorie solide et donc peut-être utilisée dans un cadre réglementaire, Christoph Molnar (2021).

6.5 Limites

Le calcul de la shapley nécessite un temps machine assez important.

7 SHAP (SHapley Additive exPlanations)

SHAP est la version globale de la valeur de shapley. Le but de SHAP est d'expliquer la prédiction d'une observation en calculant la contribution de chacune des variables à cette prédiction. Cet outil (SHAP) s'accompagne de nombreuses méthodes d'interprétation globale basées sur des agrégations de valeurs de Shapley. Il y a plusieurs variantes du SHAP (i) **KernelSHAP** qui est une approche alternative d'estimation des valeurs de shapley basée sur les noyaux, (ii) **TreeSHAP** une approche d'estimation efficace pour les modèles basés sur des arbres.

7.1 SHAP Feature Importance

Ce graphique représente les valeurs absolues des valeurs de shapley pour chaque variable. C'est une alternative à la permutation feature importance. En effet, pour chaque variable, il suffit de calculer :

$$\frac{1}{n} I_j = \sum_{i=1}^{n} |\phi_j^{(i)}|$$

Il existe cependant une grande différence entre les deux mesures d'importance : la permutation feature importance est basée sur la diminution des performances du modèle. Le SHAP est basé sur l'ampleur des attributions des variables.

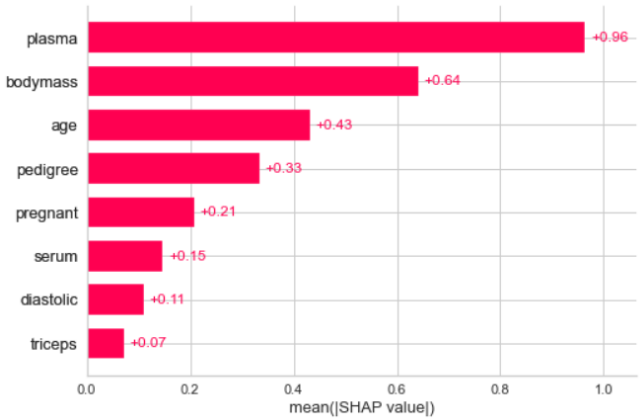

FIGURE 6 – SHAP Feature Importance du modèle

Variables	Valeur
pregnant	1
diastolic	66
triceps	29
bodymass	26.6
pedigree	0.351
age	31
plasma	85
serum	0

TABLE 1 – Les caractéristiques de l'observation 2 de la base

Au vue des caractéristiques de l'individu et de l'importance intrinsèques de chaque variable, le sens et le niveau de contribution des variables nous semble assez pertinent. Cette femme de 31 ans a taux de concentration de glucose dans le sang (*plasma*) très inférieure à la moyenne (120), un **IMC** ou encore *bodymass* normal et n'a eu qu'une seule grossesse.

Le graphique précédent nous donne la contribution marginale de chaque variable à la prédiction.
Il faut noter que l'on somme les valeurs absolues et que donc, comme pour la méthode Permutation Feature Importance, la méthode SHAP ne donne pas le sens mais seulement l'amplitude de l'impact d'une variable. SHAP est une méthode globale mais nous pouvons avoir une vision plus détaillée localement en utilisant les Shapley Values. Prenons le cas du second individu de la base (par souci de comparabilité avec le LIME).

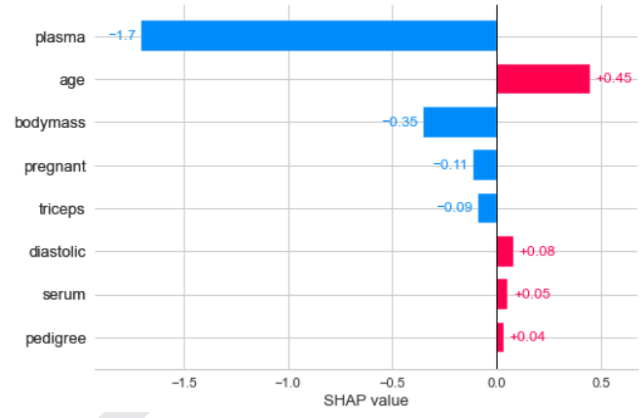

FIGURE 7 – SHAP Feature Importance individuelle (2ème observation de la base)

7.2 SHAP Summary Plot

Le graphique suivant combine *feature importance* et *feature effects* et permet d'avoir une interprétation globale ainsi que le sens de l'impact des variables. Chaque point du summary représente la Shapley Value pour une observation. Leur position sur l'axe des abscisses indique la Shapley value (la contribution) et leur couleur indique la valeur x_{ij} de la variable explicative (le joueur).

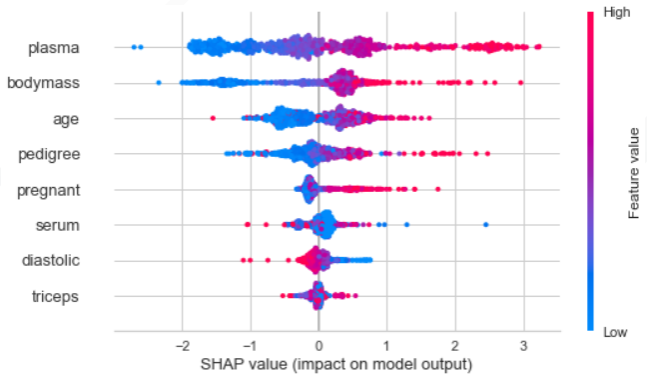

FIGURE 8 – Distribution des valeurs de shapley par variable

Les variables sont ordonnées par ordre d'importance dans le modèle. Ce résultat en terme d'ordre d'importance est conforme à celui de la permutation feature importance.
Pour la variable *plasma* par exemple on retrouve une grande amplitude sur l'axe horizontal représentative d'une grande Feature Importance. De plus le fait d'avoir une majorité de points bleus dans les Shapley Values négatives, rouges dans les valeurs positives et violet dans les valeurs proches de zéro montre que le

plasma est positivement corrélé au probabilité d'obésité.
Pour la variable *pregnant*, seuls les points rouges s'étalent vers les valeurs positives alors que les bleus et violet s'entassent autour de zéro. On en déduit qu'un grand nombre de grossesses est lié à de plus grand risque d'obésité mais qu'un faible nombre de grossesse ne participe à diminuer ce risque par rapport à la moyenne.
Enfin on peut confirmer par rapport aux méthode précédentes la corrélation inverse avec la variable *diastolic* et l'absence d'effet réel de la variable *triceps* avec des points très rapprochés de zéro. Cependant, avec une plus grande amplitude, le fait d'avoir des points rouges aux extrêmes et des points bleus au centre pourrait être interprété comme le fait qu'avoir une peau de triceps épaisse a un impact soit positif, soit négatif, alors que le fait d'avoir une peau fine n'a pas d'impact.

7.3 SHAP Dependence Plot

C'est la représentation des couples $\{(x_j^{(i)}, \phi_j^{(i)})\}_{i=1}^n$. Autrement dit, ce graphique présente la répartition des valeurs de shapley d'une variable en fonction des valeurs de celle-ci. Il permet de visualiser la forme de la dépendance (ou de liaison) du modèle à la variable en question plus précisément que le graphique précédent, mais une variable à la fois.

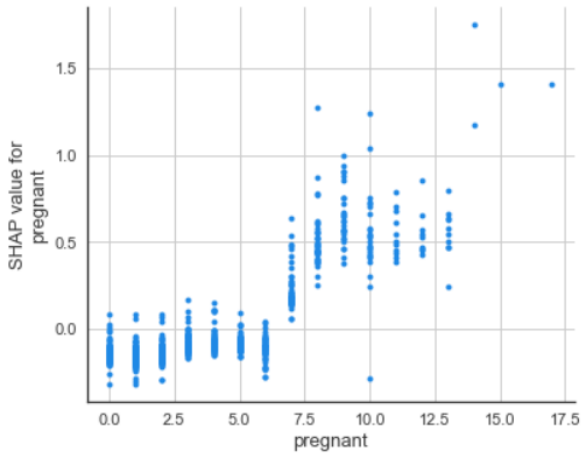

FIGURE 9 – SHAP dependence plot

Le graphique pour la variable (*pregnant*) montre une stagnation autour de zéro pour des valeurs de 0 à 6 grossesses puis une croissance de la probabilité prédite avec les nombre élevés de grosseses ce qui confirme l'interprétation précédente.
On observe de plus des alignements verticaux de points : cela indique la présence d'interactions avec une ou plusieurs autres variables. Étudions par exemple l'interaction avec la variable *age*.

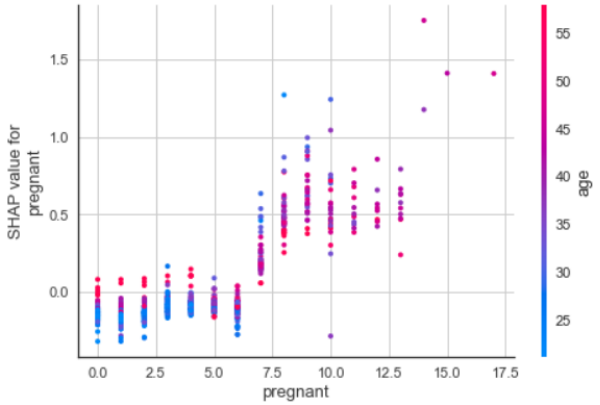

FIGURE 10 – SHAP interaction value

Le graphique ci-dessus montre une interaction entre l'âge et le temps de grossesse :

- Pour les femmes avec 2 grossesses au plus, on remarque que la contribution de la grossesse est d'autant plus négative que l'individu est jeune.
- Pour les femmes avec au moins 6 grossesses, on commence à observer la relation inverse, à savoir que *pregnant* augmente d'autant plus le risque de diabète que lorsque l'individu est jeune.

En termes plus concrets, on peut en déduire qu'un petit nombre de grossesses "augmente plus" (on devrait dire diminue moins) le risque de diabète chez les femmes âgées que les jeunes alors qu'un grand nombre de grossesses augmente plus le risque de diabète chez les jeunes femmes que les plus âgées.

Limites

SHAP fournit de puissants outils d'interprétation, accompagnés d'une garantie théorique assez forte. La principale limite de SHAP repose sur sa complexité de calcul qui peut parfois être un vrai frein à son utilisation.

8 Anchors

Les Anchors sont une technique d'interprétabilité qui cherche à trouver des ensembles de règles (de type *if ... then ...*) résumant au mieux le comportement du modèle étudié. L'objectif est de délimiter des régions locales les plus grandes possible pour lesquelles les prédictions sont le plus homogènes possibles. On obtient par exemple des règle sous le format suivant :

- Si $X_1 > a$
- Si $X_2 = b$
- Si ...

... alors la prédiction est $y = y_0$ (avec $p\%$ de précision)

8.1 Algorithme

Il existe plusieurs variantes, mais on peut considérer par souci de compréhension que la construction des Anchors se fait selon la démarche suivante :

- Choisir une précision p cible

— Choisir un point de donnée x_i dont on souhaite interpréter la prédiction

— Déterminer une première condition sur une variable qui segmente les données entre celles qui ont la même prédiction que x_i et les autres. On supposera qu'on obtient la règle $X_1 > a \Rightarrow f(X) = \hat{f}(x_i)$

— Estimer la précision p_1 de cette règle

— Si $p_1 < p$ alors on ajoute une condition à la règle. On suppose que l'on obtient $X_1 > a \wedge X_2 = b \Rightarrow f(X) = \hat{f}(x_i)$

— Estimer la précision p_2 de cette nouvelle règle

— Répéter l'opération jusqu'à ce que $p_k > p$

Chaque condition est obtenue en générant plusieurs "conditions candidates" et en choisissant celle qui optimise la précision. La précision est calculé en générant des perturbation à partir de x_i selon un algorithme d'apprentissage par renforcement (Multi-Armed-Bandit) que l'on n'explicitera pas ici.

Une notion importante à définir pour la méthode des Anchors est le coverage.
Le coverage est une mesure de la portée de la règle sur l'ensemble des données. C'est simplement le pourcentage parmi les données disponibles qui vérifie la règle R. Lors de la construction itérative de la règle, à chaque fois qu'une condition est ajoutée la précision augmente mais le coverage diminue. L'enjeu principal dans le choix de p est d'obtenir une règle avec une précision élevée sans trop réduire le coverage.

8.2 Avantages et inconvénients

L'avantage d'Anchors et de pouvoir fournir des explications très concrètes, facile à interpréter même sans aucune compétence scientifique.
Un bon exemple de ce que apporter Anchors est sa capacité à prendre en compte des interactions entre variables que d'autres méthodes auraient beaucoup de mal à identifier. Considérons un modèle devant prédire si un commentaire est positif ou négatif. Parmi les variables, nous avons la présence ou non du mot "bad" et celle du mot "not". Les règles Anchors peuvent indiquer que si une phrase contient à la fois "bad" et "not" alors le commentaire est sûrement positif, alors que LIME par exemple nous donnera un coefficient négatif pour bad et un positif pour not sans expliciter vraiment comment l'interaction entre les deux mots fonctionne.

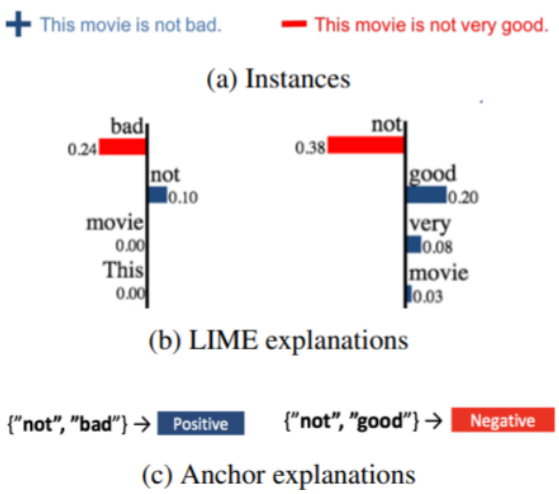

FIGURE 11 – Comparaison des interprétation de LIME et Anchors

En revanche l'inconvénient de Anchors réside souvent dans la difficulté à trouver des règles avec à la fois une précision et un coverage intéressants. On peut comparer ce dilemme à celui de la méthode Global Surrogate, dans le sens ou l'on doit chercher le juste milieu entre sa pertinence de l'interprétation et son pouvoir d'explication global. En effet, si une règle ne couvre en réalité que deux ou trois observations, elle ne représente plus une interprétation si intéressante que cela.

9 Conclusion

L'interprétabilité est un aspect métier important dans le processus de construction d'un modèle de Machine Learning dans des domaines assez réglementés. Nous avons présenter dans ce papier différentes méthodes simples (permutation feature importance) et complexes (basées sur la valeur de Shapley). Cette dernière est théoriquement fondée et très répandue dans la littérature. Le benchmark a permis de montrer la cohérence des méthodes abordées.

Références

[1] Christoph Molnar (2021), Interpretable Machine Learning ; https://christophm.github.io/interpretable-ml-book/

[2] Miller, Tim. (2017) "Explanation in artificial intelligence : Insights from the social sciences." arXiv Preprint arXiv :1706.07269.

[3] Aaron Fisher , Cynthia Rudin and Francesca Dominici (2018). All Models are Wrong, but Many are Useful : Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously.

[4] Alex Goldstein, Adam Kapelnery, Justin Bleichz and Emil Pitkinx (2014) Peeking Inside the Black Box : Visualizing Statistical Learning with Plots of Individual Conditional Expectation.

[5] Ribeiro, Marco Tulio et al (2016), "Why should I trust you? : Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM.

[6] Strumbelj, E. and Kononenko (2014), I. Explaining prediction models and individual predictions with feature contributions. Knowledge and information systems, 41(3) :647– 665.

[7] EBA (2021) EBA DISCUSSION PAPER ON MACHINE LEARNING FOR IRB MODELS.

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors :High-precision model-agnostic explanations," inPro-ceedings of the AAAI conference on artificial intelli-gence, vol. 32, 2018.