Intégration de connaissances dans les méthodes d'explications post-hoc

Adulam Jeyasothy, Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Marcin Detyniecki

Intégration de connaissances dans les méthodes d'explications post-hoc

Adulam Jeyasothy[1] Thibault Laugel[2] Marie-Jeanne Lesot[1] Christophe Marsala[1] Marcin Detyniecki[2,3]
[1] Sorbonne Université, CNRS, LIP6, F-75005 Paris, France
[2] AXA, Paris, France
[3] Polish Academy of Science, IBS PAN, Warsaw, Poland

Résumé :
Dans le domaine de l'intelligence artificielle explicable (XAI), les méthodes d'interprétabilité post-hoc intègrent des connaissances utilisateur afin d'améliorer la compréhension de l'explication et de proposer des explications personnalisées. Dans cet article, nous proposons de définir une fonction de coût qui intègre explicitement ces connaissances dans les objectifs d'interprétabilité : nous présentons un cadre général pour le problème d'optimisation des méthodes d'interprétabilité post-hoc, et montrons que les connaissances de l'utilisateur peuvent être intégrées à toute méthode en ajoutant un terme de compatibilité dans la fonction de coût. Nous instancions la formalisation proposée dans le cas des explications contre-factuelles et proposons une nouvelle méthode d'interprétabilité appelée Knowledge Integration in Counterfactual Explanation (KICE).

Mots-clés :
Intelligence artificielle explicable, Connaissances utilisateur, Explication contre-factuelle, Explication personnalisée

Abstract:
In the field of explainable artificial intelligence (XAI), post-hoc interpretability methods integrate user knowledge to improve the explanation understandability and allow for personalised explanations. In this paper, we propose to define a cost function that explicitly integrates such user knowledge into the interpretability objectives : we present a general framework for the optimization problem of post-hoc interpretability methods, and show that user knowledge can be integrated to any method by adding a compatibility term in the cost function. We instantiate the proposed formalization in the case of counterfactual explanations and propose a new interpretability method called Knowledge Integration in Counterfactual Explanation (KICE).

Keywords:
Explainable artificial intelligence, User knowledge, Counterfactual explanation, Personalised explanation

1 Introduction

Dans le domaine de l'XAI [1, 9], les méthodes post-hoc se concentrent sur la génération d'explications pour les prédictions obtenues par un classifieur qui a été entraîné par ailleurs. On les distingue classiquement selon leur format : importance des attributs (par exemple LIME [16] ou SHAP [10]) ou exemples contre-factuels [21] (par exemple Growing Spheres [8] ou FACE [15]). Elles varient également en fonction des données qu'elles prennent en entrée : les approches agnostiques par rapport au modèle et aux données (respectivement *model-agnostic* et *data-agnostic*) considèrent qu'aucune connaissance n'est disponible ni sur le modèle, ni sur les données. Cette absence de connaissances entraîne une difficulté à générer des explications adaptées au contexte, les explications obtenues ne sont alors pas toujours comprises par l'utilisateur [17]. Pour faire face à ce problème, certains travaux, détaillés dans la section 2, se concentrent sur l'enrichissement des données d'entrée considérées et l'intégration de l'humain dans la boucle (*Human in the loop*) en prenant en compte ses connaissances [11, 4, 19].

Dans ce contexte, cet article propose dans la section 3 une formalisation générale, à travers la définition d'une fonction de coût qui intègre des connaissances utilisateur dans la recherche d'explication. Il propose d'ajouter au terme classique de pénalité, qui vise à évaluer la qualité d'une explication candidate, un terme complémentaire de *compatibilité* qui prend comme paramètre la connaissance considérée. Nous discutons cette notion de compatibilité pour laquelle deux sémantiques contradictoires peuvent être considérées. En effet, la connaissance utilisateur peut être utilisée pour trouver une explication (i) complémentaire à la connaissance, (ii) ou, au contraire, exprimée dans le même langage c'est-à-dire en accord avec la connaissance. On peut argumenter qu'une explication dans le langage de la connaissance peut augmenter la confiance de l'utilisateur dans l'explication, et qu'une explication

complémentaire peut enrichir la connaissance de l'utilisateur.

L'article traite de ces cas lorsque les connaissances considérées fournissent des informations sur les attributs. Dans la section 4, il se concentre sur le cas des explications contre-factuelles exprimées dans le langage des connaissances : il considère qu'une explication est compatible avec les connaissances si elles utilisent toutes les deux les mêmes attributs. Afin de générer une telle explication compatible, l'article propose une méthode appelée KICE (Knowledge Integration in Counterfactual Explanation) et présente les expérimentations illustratives, réalisées dans la section 5.

2 État de l'art

Cette section décrit d'abord certaines méthodes post-hoc qui expliquent la prédiction réalisée par un classifieur donné pour une instance donnée. Elle présente ensuite des approches qui intègrent des connaissances, après avoir discuté des différentes formes que ces dernières peuvent prendre.

2.1 Explications post-hoc

De nombreuses méthodes post-hoc, qui expliquent la prédiction d'un classifieur entraîné par ailleurs, ont été proposées, voir par exemple [6] pour une vue d'ensemble. Une grande partie d'entre elles repose sur la génération d'explications comme solution d'un problème d'optimisation. Parmi celles-ci, nous nous intéressons particulièrement aux approches contre-factuelles et aux modèles de substitution. Les premières [7, 8, 21, 15, 12] visent à répondre à la question : "Que dois-je faire pour obtenir la prédiction souhaitée ?". La réponse est définie comme l'instance la plus proche de l'instance étudiée appartenant à la classe souhaitée. Elle peut être obtenue en optimisant, sous contrainte, une fonction de coût définie comme la distance entre l'instance considérée et l'explication (généralement la distance euclidienne [7, 8]). Les explications dans le cas des modèles de substitution (*surrogate*) [16, 14, 18] quant à elles répondent plutôt à la question : "Quel est le comportement du classifieur localement ?". Elles s'appuient sur un modèle interprétable, comme un arbre de décision, une régression linéaire ou des règles de classification, qui fournit une approximation du classifieur étudié pour imiter son comportement. Ainsi, LIME [16] minimise la fidélité locale de l'explication au classifieur, au voisinage de l'instance étudiée.

2.2 Représentation des connaissances

Les méthodes de construction d'explications peuvent être enrichies par des connaissances utilisateur afin d'améliorer leur qualité et leur intelligibilité. La section suivante décrit des approches existantes qui intègrent des connaissances, nous discutons ici des formes que celles-ci peuvent prendre ; on peut citer par exemple les prototypes de classe [20], des informations sur la distribution des données [15] ou les propriétés sur les attributs [19].

En ce qui concerne ces propriétés, un premier type de connaissance peut être exprimé par des informations individuellement sur chaque attribut : l'utilisateur peut par exemple indiquer l'ensemble des attributs dits actionnables [19], qui peuvent être modifiés. Un deuxième type de connaissances sur les attributs peut fournir des informations sur leurs interactions, par le biais de leur covariation [3, 11] ou d'un graphe de causalité [11, 4]. Dans cet article, nous considérons la connaissance représentée par un ensemble d'attributs.

2.3 Intégration des connaissances

Au-delà de la question de la forme que peuvent prendre les connaissances utilisateur, il faut se poser la question de leur exploitation et de leur intégration dans l'explication. Les travaux existants diffèrent dans la manière dont ils répondent à cette question et dans le

problème qu'ils abordent. Ils considèrent que ces connaissances constituent une contrainte supplémentaire dans la génération de l'explication. Cette contrainte est représentée sous différentes formes : réduction de l'espace de recherche, ajout de pénalité ou pondération dans la fonction de coût. Ainsi Ustun et al. [19] restreignent l'ensemble de recherche des explications contre-factuelles pour interdire l'utilisation de certains attributs ou directions. Mahajan et al. [11] proposent quant à eux d'intégrer les liens de causalité en définissant une nouvelle mesure de distance qui quantifie la mesure dans laquelle l'explication candidate satisfait les relations causales. Frye et al. [4] proposent plutôt d'intégrer ces relations causales en pondérant la fonction de coût dans le cas des explications locales sous forme de vecteur d'importance d'attributs. Dans la section suivante, nous proposons une formalisation générale pour toutes les méthodes intégrant des connaissances.

3 Formalisation générale proposée

L'objectif considéré est d'expliquer la prédiction réalisée pour une instance $x \in \mathcal{X}$ par un classifieur f, d'une famille de classifieurs $\mathcal{F}$, $f : \mathcal{X} \rightarrow \mathcal{Y}$ où $\mathcal{X}$ désigne l'espace d'entrée de dimension d, inclus dans $\mathbb{R}^d$, et $\mathcal{Y}$ l'espace de sortie. La connaissance à intégrer dans la génération de l'explication est notée E.

3.1 Problème d'optimisation proposé

Cette section présente le problème d'optimisation qui permet de générer tout type d'explication enrichi par tout type de connaissance. Puis, nous détaillons les choix des trois fonctions qui dépendent du contexte étudié : les motivations de l'utilisateur, le type d'explication à générer et le type de connaissance considéré.

Fonction de coût. Soit $\mathcal{E}$ l'ensemble des explications pour un type d'explication donné (par exemple les exemples contre-factuels ou les modèles de substitution), nous proposons la formalisation suivante du problème d'optimisation général :

$$\underset{e \in \mathcal{E}}{\operatorname{argmin}}\, agg(pen_x(e, f), incomp_x(e, E)) \quad (1)$$

où agg, pen_x et $incomp_x$ sont trois fonctions décrites dans les sous-sections suivantes.

Fonction de pénalité. Comme rappelé dans la section 2.1, la plupart des approches existantes pour générer des explications minimisent une fonction de coût notée $pen_x : \mathcal{E} \times \mathcal{F} \longrightarrow \mathbb{R}$ qui définit la qualité de l'explication candidate : elle prend en argument une explication candidate e, le classifieur étudié f et peut dépendre de l'instance étudiée x dans le cas d'une méthode locale. Elle peut par exemple être définie comme la distance entre le candidat e et l'instance considérée x dans le cas d'exemples contre-factuels ou la fidélité de l'explication au classifieur dans le cas d'approches par substitution.

Fonction d'incompatibilité. Pour générer une explication conforme aux connaissances, nous proposons d'ajouter au terme classique de pénalité, un terme complémentaire $incomp_x(e, E)$ qui prend comme paramètre la connaissance considérée E. Cette fonction dépend du type d'explication, du type de connaissances considérées et de l'objectif de l'utilisateur. Comme évoqué dans l'introduction, nous proposons de distinguer deux objectifs : proposer une explication complémentaire aux connaissances, et proposer une explication dans le langage des connaissances. Nous considérons qu'une explication est dans le langage des connaissances si les attributs utilisés dans l'explication et les connaissances sont similaires, et qu'une explication est complémentaire dans le cas contraire (il n'y a pas de redondance). Nous discutons ci-dessous ces deux possibilités dans le cas d'explications basées sur des modèles de substitution.

Soit A_e l'ensemble des attributs utilisés dans l'explication générée par un modèle de substitution e qui est un modèle interprétable, et $\overline{E}$ l'ensemble des attributs non considérés dans

la connaissance E. Pour construire une explication de substitution dans le langage des connaissances, une possibilité est de minimiser le nombre d'attributs qu'elle prend en compte et qui ne font pas partie de E :

$$incomp_x(e, E) = Card(A_e \cap \overline{E})$$

Au contraire, lorsque l'explication est complémentaire aux connaissances, une proposition consiste à minimiser le nombre d'attributs présents à la fois dans A_e et E :

$$incomp_x(e, E) = Card(A_e \cap E)$$

Ces deux définitions peuvent être pertinentes suivant les cas d'usage considérés, comme discuté plus en détail dans la section 4 qui considère le cas des explications contre-factuelles.

Fonction d'agrégation. La fonction d'agrégation combine les fonctions de pénalité et d'incompatibilité. Il existe un grand nombre d'opérateurs d'agrégation (cf. par exemple [2, 5]), qui peuvent être divisés en trois catégories principales : conjonctif, disjonctif ou compromis. Pour imposer que la pénalité *et* l'incompatibilité soient faibles, on peut les agréger par la fonction max. Une explication associée à un classifieur qui n'est pas en accord avec les connaissances a alors une mauvaise incompatibilité donc une fonction de coût élevée. Pour imposer qu'*au moins une* des deux valeurs soit faible, on peut les agréger par la fonction min. Si on considère les explications pour un classifieur qui n'est pas en accord avec les connaissances, l'incompatibilité est élevée et donc le coût se résume à la fonction de pénalité. Enfin, les fonctions de compromis telles que la moyenne pondérée permettent de compenser des valeurs faibles par des valeurs élevées. Le choix de cette fonction peut être fait en fonction des préférences de l'utilisateur, ce qui permet d'augmenter la personnalisation des explications.

3.2 Un exemple de méthode existante sous ce formalisme

Pour montrer la généralité du cadre proposé, nous proposons d'exprimer l'approche de l'état de l'art présentée par Ustun et al. [19] dans ce formalisme en mettant en évidence la définition des trois fonctions présentées précédemment.

Dans cette approche, la connaissance de l'utilisateur E, notée $A(x)$, est définie comme l'ensemble des modifications qui peuvent être appliquées à l'instance x. Elle est intégrée pour générer une explication contre-factuelle dite actionnable définie comme :

$$a^* = \underset{a \in A(x)/f(x+a) \neq f(x))}{\operatorname{argmin}} (dist(a, x))$$

où $dist$ évalue la distance entre x et $x + a$. Le déplacement a entre l'instance étudiée x et l'instance la plus proche dans la classe opposée $a + x$ définit l'explication.

Ce problème d'optimisation peut être réécrit de la façon suivante, en notant $\mathcal{E}_x = \{x' \in \mathbb{R}^d | f(x') \neq f(x)\}$,

$$e^* = \underset{e \in \mathcal{E}_x}{\operatorname{argmin}}\, dist(e, x) + \mathbb{1}_{|x-e| \notin A(x)} \times Z$$

où Z est arbitrairement grand. Ici e^* est l'instance la plus proche de x dans la classe opposée, et elle permet de retrouver a tel que $a = e^* - x$.

Cette expression, équivalente à la précédente, permet d'identifier les fonctions pen_x, $incomp_x$ et agg. La fonction de pénalité est égale à $dist$. La fonction d'incompatibilité est égale à $\mathbb{1}_{|x-e| \notin A(x)} \times Z$ et représente la présence ou l'absence d'attribut modifié dans l'ensemble d'attributs considéré par l'utilisateur ; elle ne prend que deux valeurs 0 ou Z. Une explication contre-factuelle incompatible a donc une fonction de coût très élevée, ce qui fait que seuls les contre-factuels compatibles sont considérés. Enfin, l'agrégation est effectuée par une somme pondérée. Cependant, comme la fonction d'incompatibilité est binaire, seuls les contre-factuels compatibles sont pris en compte, de sorte que le contre-factuel résultant est à la fois compatible et de bonne qualité.

4 Knowledge Integration in Counterfactual Explanation (KICE)

Cette section propose une nouvelle méthode pour générer des explications contre-factuelles tenant compte des connaissances, en instanciant le cadre général introduit dans la section précédente. Elle détaille ensuite l'algorithme utilisé pour résoudre ce problème.

4.1 Fonction de coût proposée

Nous considérons pour l'instanciation une connaissance sous la forme d'un ensemble d'attributs noté E.

Fonction de pénalité. Pour la fonction de pénalité, nous proposons d'utiliser la fonction classique pour les exemples contre-factuels qui est le carré de la distance euclidienne :

$$pen_x(e, f) = \| x - e \|^2 \tag{2}$$

Fonction d'incompatibilité. Comme spécifié dans la section 3.1, l'objectif est de proposer une explication contre-factuelle en accord avec les connaissances utilisateur, ce qui signifie idéalement que les modifications contre-factuelles sont effectuées uniquement en fonction des attributs présents dans E. Cependant, dans certains cas, se concentrer uniquement sur un sous-ensemble d'attributs empêche de trouver un exemple contre-factuel : il se peut que les déplacements selon les attributs de E ne permettent pas de rencontrer la frontière de décision. Plutôt que d'interdire les modifications selon les attributs $\overline{E}$, nous proposons de pénaliser les modifications en fonction des attributs $\overline{E}$. Cela permet, lorsque la frontière ne peut être trouvée selon les attributs de E, de s'assurer qu'une solution existe. Ainsi, nous proposons la fonction d'incompatibilité qui calcule le carré de la distance euclidienne uniquement selon les attributs de $\overline{E}$:

$$\begin{aligned} incomp_x(e, E) &= \| x - e \|^2_{\overline{E}} \\ &= \sum_{i \notin E} (x_i - e_i)^2 \end{aligned} \tag{3}$$

Minimiser cette incompatibilité permet d'éviter de générer des exemples contre-factuels qui modifient fortement les attributs non présents dans les connaissances.

Fonction d'agrégation. La fonction d'agrégation combine la fonction de pénalité et la fonction d'incompatibilité, nous proposons d'utiliser une fonction de compromis par pondération :

$$agg(u, v) = u + \lambda v \tag{4}$$

où $\lambda \in \mathbb{R}^+$ un hyperparamètre défini par l'utilisateur.

Fonction de coût. Nous obtenons alors le problème d'optimisation suivant : soit $\mathcal{E} = \{x' \in \mathbb{R}^d \mid f(x') \neq f(x)\}$ et $\lambda \in \mathbb{R}^+$,

$$\begin{aligned} e^* &= \underset{e \in \mathcal{E}}{\operatorname{argmin}}\, cost_{x,E}(e) \\ \text{où } cost_{x,E}(e) &= \| x - e \|^2 + \lambda \| x - e \|^2_{\overline{E}} \end{aligned} \tag{5}$$

4.2 Algorithme KICE

Pour résoudre le problème d'optimisation (5), nous proposons un algorithme nommé KICE pour Knowledge Integration in Counterfactual Explanation. Nous considérons un cas agnostique, dans lequel aucune information sur la distribution des données ni sur la frontière de décision n'est disponible. KICE génère des instances uniformément autour de l'instance étudiée x. Pour trouver le point qui minimise la fonction de coût, il génère des points dans des espaces de plus en plus grands jusqu'à en trouver un que le classifieur f prédit dans une autre classe. Ce principe de génération itérative d'instances est inspiré de l'algorithme Growing Spheres [8], et considère le terme supplémentaire de compatibilité pour biaiser la génération.

Ces espaces de génération sont définis par la fonction de coût. L'équation $cost_{x,E}(e) = \nu$ définit l'équation d'une ellipse de centre x et de rayon $\sqrt{\frac{\nu}{1+\lambda}}$ selon les attributs $\overline{E}$ et $\sqrt{\nu}$ selon les attributs E. Pour considérer des ellipses de

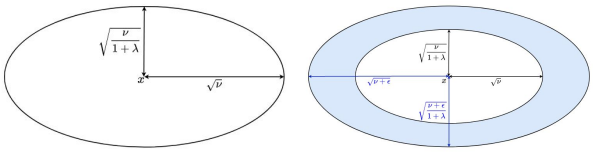

FIGURE 1 – (gauche) Zone de génération pour l'étape initiale, (droite) zone de génération pour les itérations de l'algorithme proposé KICE.

plus en plus grandes, la valeur de ν est augmentée par pas de ϵ, hyper-paramètre utilisé pour générer uniformément dans des couches ellipsoïdales définies par les rayons ν et $\nu + \epsilon$.

Pour ce faire, nous utilisons une version modifiée de l'algorithme HLG [13] qui permet de générer des instances uniformément dans la couche sphérique $SL(x, a_0, a_1)$, qui désigne l'ensemble des points à une distance comprise entre a_0 et a_1 de x. Notre modification consiste à distinguer les attributs E et $\overline{E}$ afin de générer n instances dans une couche ellipsoïdale, comme représenté sur la partie gauche de la figure 1. Si aucune de ces instances n'appartient à la classe opposée, nous générons des instances dans la couche ellipsoïdale entre ν et $\nu + \epsilon$. Cette couche est représentée en bleu à droite sur la figure 1.

5 Expérimentations

Cette section présente les expérimentations menées pour évaluer l'algorithme KICE : le but est de montrer que la méthode proposée trouve l'instance contre-factuelle attendue, ce qui permet de réaliser un compromis entre la qualité et la compatibilité d'une explication.

5.1 Protocole expérimental

Les expérimentations sont menées sur trois jeux de données tabulaires classiques : Half-moons, Boston et Breast cancer dont les valeurs sont normalisées et divisées en données d'entraînement et de test (80%-20%). Pour les données Boston, la tâche de régression est transformée en une tâche de classification binaire par seuillage de la valeur à prédire. Dans le cadre de l'explication post-hoc considérée, le choix du classifieur n'a pas d'importance, nous appliquons un SVM avec noyau gaussien, qui obtient une bonne précision sur les trois jeux de données.

En ce qui concerne la connaissance utilisateur, nous considérons que l'utilisateur dispose de moins d'attributs que le classifieur. Afin de construire une connaissance réaliste, nous générons un arbre de décision avec une profondeur égale à la moitié du nombre d'attributs des données. L'ensemble E contient les attributs qui apparaissent dans cet arbre. Des instances contre-factuelles e^* sont ensuite générées pour chaque instance x de l'ensemble de test en utilisant KICE, la valeur de λ étant choisie pour observer des résultats intéressants, respectivement 4, 3 et 6 pour les ensembles de données Half-moons, Boston et Breast cancer.

Nous comparons la méthode proposée KICE à deux méthodes concurrentes qui correspondent aux valeurs extrêmes de λ : le cas où $\lambda = 0$, c'est-à-dire où le terme d'incompatibilité est ignoré, qui conduit à l'exemple contre-factuel de référence qui minimise uniquement la distance euclidienne, noté e_{ref}. Un second concurrent est proposé en imposant de respecter strictement les connaissances. Sa fonction de coût associée minimise la distance euclidienne en fonction des seuls attributs de la connaissance, une manière naïve d'intégrer la connaissance dans l'explication. On note e_{user} l'exemple contre-factuel qui résout le problème associé à $cost_{x,E}(e) = \| x - e \|_E^2$. Il correspond à la formulation proposée dans l'Eq. (6) avec λ arbitrairement grand.

5.2 Exemples illustratifs

Tout d'abord, nous illustrons le comportement des méthodes avec des exemples en deux dimensions. La figure 2 montre les exemples contre-factuels obtenus pour trois instances du jeu de données Half-moons. La figure montre les données d'entraînement, les régions bleues

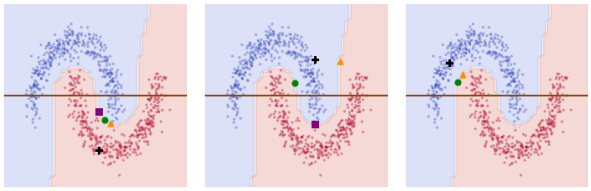

FIGURE 2 – Exemples de résultats obtenus e_{ref}, e_{user} et e^* pour trois instances (+ : x, ▲ : e_{ref}, ■ : e_{user}, ● : e^*)

et rouges représentent les classes prédites ; la frontière de décision du classifieur SVM entraîné est indiquée en blanc, il atteint une précision de 0.99. La connaissance considérée est une règle sur un seul attribut, représentée par la ligne horizontale.

Sur la figure, on observe, comme attendu, que l'exemple contre-factuel e_{ref} est le point le plus proche appartenant à la classe opposée. De plus, e_{user} est plus éloigné de x que e_{ref} et ne modifie que l'attribut X_1 qui est l'attribut défini dans la connaissance. On observe que e^* constitue un compromis entre e_{ref} et e_{user}. Il nécessite moins de modifications selon X_0 que e_{ref}, donc il est plus compatible. Il est également plus proche de x que e_{user}. Sur le dernier graphe de la figure 2, on remarque l'absence de e_{user} : dans ce cas, il n'existe pas de contre-factuel modifiant uniquement l'attribut X_1. KICE permet d'obtenir un contre-factuel plus compatible que e_{ref}.

5.3 Évaluation de la méthode KICE

Nous appliquons les trois méthodes décrites dans la section précédente sur les données de test des différents jeux de données. Pour certaines instances, e_{user} n'est pas défini (comme illustré dans la section précédente), cela concerne respectivement 20%, 0% et 11% des instances pour Half-moons, Boston et Breast cancer. Pour les autres instances, le tableau 1 montre la moyenne et l'écart-type des fonctions de pénalité, d'incompatibilité et de coût pour les trois approches.

Nous remarquons, comme attendu, que les

		$pen_x(e, f)$	$incomp_x(e, E)$	$cost_{x,E}(e)$
$\mathcal{D}_1$	e_{ref}	**0.32 ± 0.21**	0.14 ± 0.13	0.86 ± 0.56
	e_{user}	1.48 ± 1.3	**0.0 ± 0.0**	1.48 ± 1.3
	e^*	0.42 ± 0.29	0.08 ± 0.11	**0.73 ± 0.52**
$\mathcal{D}_2$	e_{ref}	**1.48 ± 1.75**	0.7 ± 1.03	3.57 ± 4.72
	e_{user}	2.26 ± 2.71	**0.0 ± 0.0**	2.26 ± 2.71
	e^*	1.72 ± 2.09	0.13 ± 0.19	**2.12 ± 2.54**
$\mathcal{D}_3$	e_{ref}	**8.82 ± 9.22**	7.27 ± 8.25	52.41 ± 58.63
	e_{user}	22.42 ± 24.87	**0.0 ± 0.0**	22.42 ± 24.87
	e^*	10.74 ± 9.85	1.25 ± 1.33	**18.24 ± 16.83**

TABLEAU 1 – Résultats obtenus par les trois méthodes considérées sur $\mathcal{D}_1$ = half moons, $\mathcal{D}_2$ = Boston et $\mathcal{D}_3$ = Breast cancer : pen_x définie dans l'équation (2), $incomp_x$ définie dans l'équation (4) et $cost$ définie dans l'équation (6)

exemples contre-factuels proposés ont une pénalité supérieure à celle de e_{ref} mais inférieure à celle de e_{user}. De plus, la valeur d'incompatibilité est beaucoup plus faible que celle de e_{ref}. Le coût pour e^* est le plus faible. Les écarts types sont élevés, car les instances sont à différentes distances de la frontière.

Pour vérifier que KICE minimise la fonction de coût par opposition aux deux autres sur toutes les instances, la figure 3 montre la valeur de la fonction de coût obtenue par e^* (qui est définie comme la minimisant) par rapport à la valeur qu'elle prend pour e_{ref} (à gauche) et e_{user} (à droite), pour chacune des instances test des données Half-moons. Les figures montrent, comme attendu, que tous les points sont au-dessus de la ligne $y = x$.

Sur le graphique de droite, les points sont plus dispersés, mais ils restent au-dessus de la ligne. On remarque que les exemples contre-factuels générés ont une fonction de coût plus proche de celle de e_{ref} que de celle de e_{user}. Un seul attribut est considéré par la connaissance E, il est difficile d'obtenir un exemple contre-factuel proche en modifiant uniquement cet attribut. Il est alors nécessaire de s'éloigner de x pour obtenir une explication compatible à 100%. La dispersion des points dépend également de la valeur de λ.

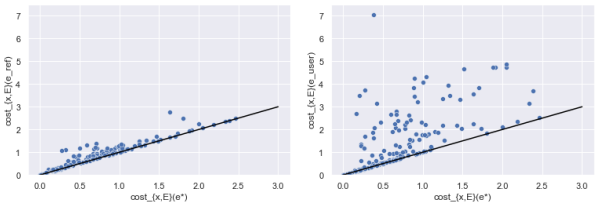

FIGURE 3 – Fonction de coût $cost$ définie dans l'équation (6) de e_{ref}, e_{user} et e^* pour 80% des données de test pour lesquelles les trois exemples contre-factuels sont définis

6 Conclusion et perspectives

Dans cet article, une formalisation générale est proposée pour aider à définir un problème d'optimisation permettant d'intégrer les connaissances dans les explications post-hoc agnostiques sans information sur le type de connaissances ou le type d'explication. Nous proposons une instanciation de ce cadre pour les explications contre-factuelles et la méthode KICE permettant de résoudre ce problème en minimisant les modifications selon les attributs inconnus. Les futurs travaux comprendront une étude des explications contre-factuelles générées pour différentes valeurs de λ. Ils viseront également à explorer d'autres instanciations du cadre proposé pour différents modèles et types de connaissance, ainsi que sur des expérimentations réelles incluant des utilisateurs réels.

Références

[1] Burkart, N., Huber, M.F. : A Survey on the Explainability of Supervised Machine Learning. Journal of Artificial Intelligence Research **70**, 245–317 (2021)

[2] Calvo, T., Mayor, G., Mesiar, R. (eds.) : Aggregation Operators : New Trends and Applications, vol. 97. Springer (2002)

[3] Drescher, M., Perera, A.H., Johnson, C.J., Buse, L.J., Drew, C.A., Burgman, M.A. : Toward rigorous use of expert knowledge in ecological research. Ecosphere **4**(7) (2013)

[4] Frye, C., Rowat, C., Feige, I. : Asymmetric Shapley values : incorporating causal knowledge into model-agnostic explainability. In : Proc. of Advances in NeurIPS. vol. 33 (2020)

[5] Grabisch, M., Marichal, J., Mesiar, R., Pap, E. : Aggregation Functions. No. 127 in Encyclopedia of Mathematics and its Applications, Cambridge Univ. Press (2009)

[6] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D. : A Survey of Methods for Explaining Black Box Models. ACM Comput. Surv. **51**(5) (2018)

[7] Lash, M.T., Lin, Q., Street, N., Robinson, J.G., Ohlmann, J. : Generalized Inverse Classification. In : Proc. of the SIAM Int. Conf. on Data Mining. p. 162–170 (2017)

[8] Laugel, T., Lesot, M-J., Marsala, C., Renard, X., Detyniecki, M. : Comparison-based Inverse Classification for Interpretability in Machine Learning. In : Proc. of Int. Conf. on IPMU pp. 100–111. Springer (2018)

[9] Linardatos, P., Papastefanopoulos, V., Kotsiantis, S. : Explainable AI : A Review of Machine Learning Interpretability Methods. Entropy **23**(1) (2021)

[10] Lundberg, S.M., Lee, S.I. : A unified approach to interpreting model predictions. In : Proc. of the 31st Int. Conf. on NeurIPS pp. 4768–4777 (2017)

[11] Mahajan, D., Tan, C., Sharma, A. : Preserving Causal Constraints in Counterfactual Explanations for Machine Learning Classifiers. NeurIPS workshop (2019)

[12] Mothilal, R.K., Sharma, A., Tan, C. : Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations. In : Proc. of the 2020 Conf. on FAT. ACM (2020)

[13] Muller, M.E. : A Note on a Method for Generating Points Uniformly on N-Dimensional Spheres. Commun. ACM **2**(4), 19–20 (1959)

[14] Peltola, T. : Local Interpretable Model-agnostic Explanations of Bayesian Predictive Models via Kullback-Leibler Projections. In : Proc. of the 2nd Workshop on Explainable Artificial Intelligence (XAI 2018) at IJCAI/ECAI 2018 (2018)

[15] Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T., Flach, P. : FACE : Feasible and Actionable Counterfactual Explanations. In : Proc. of the AAAI/ACM Conf. on AI, Ethics, and Society (2020)

[16] Ribeiro, M.T., Singh, S., Guestrin, C. : "Why should I trust you ?" Explaining the predictions of any classifier. In : Proc. of the 22nd ACM SIGKDD Int. Conf. on knowledge discovery and data mining. pp. 1135–1144 (2016)

[17] Rudin, C. : Stop Explaining Black Box Machine Learning Models for High Stakes decisions and Use Interpretable Models Instead. Nature Machine Intelligence **1**, 206–215 (2019)

[18] Sokol, K., Hepburn, A., Santos-Rodriguez, R., Flach, P. : bLIMEy : Surrogate Prediction Explanations Beyond LIME. In : Proc. of the HCML@NeurIPS (2019)

[19] Ustun, B., Spangher, A., Liu, Y. : Actionable Recourse in Linear Classification. In : Proc. of the Conf. on FAT. p. 10–19. Association for Computing Machinery (2019)

[20] Van Looveren, A., Klaise, J. : Interpretable Counterfactual Explanations Guided by Prototypes. In : Proc. of European Conf. on Machine Learning (2021)

[21] Wachter, S., Mittelstadt, B., Russell, C. : Counterfactual Explanations without Opening the Black Box : Automated Decisions and the GDPR. Harvard journal of law & technology **31**, 841–887 (2018)