



**HAL**  
open science

## Integrating Prior Knowledge in Post-hoc Explanations

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

► **To cite this version:**

Adulam Jeyasothy, Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Marcin Detyniecki. Integrating Prior Knowledge in Post-hoc Explanations. Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'2022), Jul 2022, Milan, Italy. pp.707-719, 10.1007/978-3-031-08974-9\_56 . hal-03728133

**HAL Id: hal-03728133**

**<https://hal.science/hal-03728133v1>**

Submitted on 20 Jul 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Integrating Prior Knowledge in Post-hoc Explanations

Adulam Jeyasothy<sup>1</sup>[0000-0002-6375-5879], Thibault Laugel<sup>2</sup>[0000-0002-5921-3225],  
Marie-Jeanne Lesot<sup>1</sup>[0000-0002-3604-6647], Christophe  
Marsala<sup>1</sup>[0000-0002-4022-9796], and Marcin Detyniecki<sup>1,2,3</sup>[0000-0001-5669-4871]

<sup>1</sup> Sorbonne Université, CNRS, LIP6, F-75005 Paris, France {adulam.jeyasothy,  
marie-jeanne.lesot, christophe.marsala}@lip6.fr

<sup>2</sup> AXA, Paris, France {thibault.laugel, marcin.detyniecki}@axa.com

<sup>3</sup> Polish Academy of Science, IBS PAN, Warsaw, Poland

**Abstract.** In the field of explainable artificial intelligence (XAI), post-hoc interpretability methods aim at explaining to a user the predictions of a trained decision model. Integrating prior knowledge into such interpretability methods aims at improving the explanation understandability and allowing for personalised explanations adapted to each user. In this paper, we propose to define a cost function that explicitly integrates prior 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 thus 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) to optimize it. The paper performs an experimental study on several benchmark data sets to characterize the counterfactual instances generated by KICE, as compared to reference methods.

**Keywords:** explainable artificial intelligence · prior knowledge · counterfactual explanation · compatibility

## 1 Introduction

Machine learning models are now present in many areas of everyday life. The XAI domain [1, 9] aims at answering questions raised by these models such as "Why do I get this prediction?" or "What do I need to do to get the prediction I want?". In particular, post-hoc methods focus on generating explanations for the predictions obtained from a trained classifier. They come in different formats such as feature importance (e.g LIME [16] and SHAP [10]) or counterfactual examples [21] (e.g. Growing Spheres [8] and FACE [15]). They also vary depending on their input data: model agnostic and data agnostic approaches consider that no knowledge is available either on the model, nor on data. This lack of knowledge makes it difficult to generate explanations adapted to the context. As a result, they may

then not be always understood by the user [17]. To tackle this issue, some studies focus on enriching the considered input and integrating the human in the loop by taking into account expert knowledge [11, 4, 19], so the explanation is more understandable.

In this context, this paper proposes a general formalization, through the definition of a cost function that integrates such prior knowledge in the search for explanation. It proposes to add to the classical penalty term, that aims at assessing the quality of a candidate explanation, a complementary compatibility term that takes as parameter the considered knowledge. We provide a discussion about this notion of compatibility for which two contradictory semantics can be considered. Indeed, the prior knowledge can be used either to find an explanation that is complementary to the knowledge or an explanation that follows the same language and is in agreement with the knowledge. It can be argued that an explanation in the knowledge language can increase the user’s confidence in the explanation, and a complementary explanation can enrich the user’s knowledge.

The paper discusses these cases when the considered knowledge provides information regarding features of interest. Then, it focuses on the case of explanation expressed within the knowledge language in the specific framework of counterfactual explanation: it considers that an explanation is compatible with the available knowledge if the features considered by the knowledge and the explanation are similar. In order to generate such a compatible explanation, it proposes a method called KICE which stands for Knowledge Integration in Counterfactual Explanation and presents the conducted illustrative experiments.

To summarize, the paper contribution consists in proposing a general framework for integrating expert knowledge into interpretability methods in Section 3, an instantiation of the framework in the context of counterfactual explanations and a new counterfactual explanation method to generate a knowledge-adaptive explanation in Section 4.

## 2 Background

This section provides background information regarding the framework to which the paper refers: it first describes some post-hoc explanation methods; then, it presents some existing approaches to integrate knowledge, after discussing different possible forms the latter can take.

### 2.1 Post-hoc Explanations

Numerous post-hoc methods which explain the prediction of a trained classifier have been proposed, addressing different problems, see e.g. [6]. A large portion of them relies on generating explanations as a solution to an optimization problem. Among those, we are particularly interested in counterfactuals and surrogates approaches. The former [7, 8, 21, 15, 12] aim at answering the question: "What do I have to do to get the prediction I want?". The answer is defined as the closest instance to the studied instance belonging to the desired class. It can be

obtained by optimizing, under constraint, a cost function defined as the distance function between the considered instance and the explanation (usually the Euclidean distance [7, 8]). On the other hand, surrogate-based explanations [16, 14, 18] rather answer the question: "Why do I get this prediction?". They rely on an interpretable model, such as a decision tree, a linear regression or classification rules, that approximates the classifier under study to mimic its behavior. In the case of local surrogates we can cite LIME [16] that minimizes the fidelity to the black box classifier.

## 2.2 Representation of Knowledge

As discussed in the introduction, post-hoc explanations, as well as any explanation building method, can be enriched by available prior knowledge so as to improve their quality and intelligibility. The next section describes existing approaches, we discuss here several forms that can be considered to represent this knowledge; one can cite among others: class prototypes [20], distribution of data [15] or features-based knowledge [19]. We are particularly interested in the latter, which is detailed in this section.

A first type of expert knowledge can be expressed by information on the individual features: the user can for instance indicate the so-called actionable features [19], i.e. the features that can be modified. For instance, they can include the budget in a credit application, as opposed to the age [7]. A second type of expert knowledge on the features can provide information on their interactions through their covariation [3, 11] (e.g. "An increase in school level leads to an increase in age") or through a causality graph [11, 4]. In this paper, we consider knowledge provided by a set of expected features that can for instance be deduced from a rule-based model.

## 2.3 Integration of Knowledge

Beyond the question of the form the prior knowledge can take, the question of its exploitation and integration in the explanation must be considered. Existing works differ in the way they answer this question and the problem they address.

Considering the case where knowledge indicates which features are actionable and explanations take the form of counterfactual examples, Ustun et al. [19] restrict the search set to forbid the use of some features or directions. This allows to avoid proposing inapplicable explanation such as "In order to get the credit, you need to decrease your age".

Also considering the case of counterfactual explanations, Mahajan et al. [11] consider knowledge about causal interactions between features. They propose to integrate it in the definition of a new distance measure that quantifies the extent to which the candidate explanation satisfies the causal relationships. Unlike Ustun et al., they do not exclude candidates that do not respect these constraints, but penalize them, in a more flexible approach. This method makes it possible to avoid unrealistic explanations of the form "In order to get the credit, you need to increase your budget and to decrease your salary".

Frye et al. [4] are also interested in the integration of causal relationships, but exploit a different information they induce; they considers the case of explanations in the form of local feature importance vector. He focuses on the asymmetry brought by the causal link orientation and proposes to exploit it by weighting the cost function.

These three approaches integrate different types of expert knowledge into different forms of explanation. They consider that this knowledge is an additional constraint in the explanation generation. This constraint is represented in different forms: reduction of the search space, addition of penalty in the cost function or weight in the cost function. In the following section, we propose a general formalization for all methods integrating knowledge.

### 3 Proposed General Framework

The considered objective is to explain the prediction of a classifier  $f : \mathcal{X} \rightarrow \mathcal{Y}$  where  $\mathcal{X}$  denotes the input space, included in  $\mathbb{R}^d$ , and  $\mathcal{Y}$  the output space. We consider a domain where elements  $x \in \mathcal{X}$  are described by means of a set of features (for instance, age, salary, weight,...). In this paper, we consider continuous data, the case of categorical data will be considered in future work. The prior knowledge to be integrated in the generation of the explanation is denoted  $E$ . In the following, we consider that an explanation is related to the prediction  $f(x)$  for an element  $x \in \mathcal{X}$ .

#### 3.1 Proposed Optimization Problem

This section describes the proposed general optimization problem to generate explanation integrating user knowledge. Let  $\mathcal{E}$  be the set of explanations for a given type (e.g. counterfactual example or surrogate models), we propose the following optimization problem:

$$\operatorname{argmin}_{e \in \mathcal{E}} \operatorname{agg}(\operatorname{penalty}_x(e), \operatorname{incompatibility}_x(e, E)) \quad (1)$$

where  $\operatorname{agg}$ ,  $\operatorname{penalty}_x$  and  $\operatorname{incompatibility}_x$  are three functions described in the following. This optimization problem allows to generate a knowledge-based explanation for any type of explanation and type of knowledge. As discussed below, the choice of the three functions depends on the studied context: the motivations of the user, the type of explanation to be generated or the considered type of knowledge. In our approach, we use a subscript  $x$  for functions that are related to the studied instance  $x$ .

**Penalty Function** As recalled in Section 2.1, most existing approaches to generate explanations minimize a cost function. Let  $\operatorname{penalty}_x : \mathcal{E} \rightarrow \mathbb{R}$  be such a function: it takes as argument a candidate explanation  $e$ , and may depend on the studied instance  $x$  in the case of local explainer. It values the quality of the candidate explanation. For instance, it is defined as the distance between the

candidate and the considered instance in the case of counterfactual examples and the fidelity of explanation to the black box classifier in the case of surrogate approaches.

**Incompatibility Function** To generate an explanation complying with available knowledge, we propose to add a function to measure how compatible the candidate explanation is with the knowledge. This incompatibility function depends on the type of explanation and the type of considered knowledge, as well as the user objective. For instance, an explanation generated for a non-expert customer asking how to get his credit application accepted needs to focus on common knowledge actions such as increasing his salary. On the other hand, in the case where the explanation is destined to a domain expert, the incompatibility can represent the level of fidelity of the explanation with respect to the expert’s comprehension of the domain.

As mentioned in the introduction, we propose to distinguish two objectives: to propose an explanation that is complementary to the available knowledge, and to propose an explanation that compete with the knowledge language. We consider an explanation to be in the knowledge language if the features used in the explanation and the knowledge are similar, and an explanation to be complementary in the opposite case (i.e. when there is no redundancy). Below, we discuss these two possibilities in the case of surrogate-based explanations.

Let  $A_e$  denote the set of features used in the explanation generated by a surrogate  $e$  which is an interpretable model, and  $\bar{E}$  the set of features not considered in knowledge  $E$ . To build a surrogate-based explanation in the knowledge language, one possibility is to minimize the number of features it takes into account that are not part of  $E$ :

$$incompatibility_x(e, E) = Card(A_e \cap \bar{E})$$

On the contrary, where the explanation needs to be complementary to the knowledge, another possibility is to minimize the number of features present in both  $A_e$  and  $E$ :

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

These two definitions can be both relevant depending on the motivations and needs of the user, as discussed above. Such a case is further discussed in Section 4 for the case of counterfactual explanation.

**Aggregation Function** The aggregation function combines the penalty and incompatibility functions. There exists an abundant literature on aggregation operators, see e.g. [2, 5], which can be divided into three main categories: conjunctive, disjunctive or compromise. A conjunctive operator, for example the min function, returns a high value only if both penalty *and* incompatibility taken as input are high. A disjunctive operator can be represented by the max function, it requires *at least one* value to be high. Finally, trade-off functions such as weighted average result in the compensation of low values by high values.

The choice of this function can be made according to the user preferences, instead of necessarily considering the same function for everyone: one way to personalize the explanation to a user is to let him choose the aggregation he wants to perform.

### 3.2 An Example of Existing Method under this Formalism

To show how general the proposed framework is, we propose to express the state-of-the-art approach proposed by Ustun et al. [19] in the proposed formalism highlighting the definition of the three involved functions. In this approach, the user knowledge  $E$  is denoted  $A(x)$  and defined as the set of modifications that can be applied to instance  $x$ , it is integrated to generate an actionable counterfactual explanation. The latter is the solution to an optimization problem of the form:

$$a^* = \underset{a \in A(x)}{\operatorname{argmin}} \operatorname{cost\_fct}(a, x) \text{ with } f(x + a) \neq f(x)$$

where  $\operatorname{cost\_fct}$  values the distance between  $x$  and  $x + a$ . The considered explanation expresses the move between the studied instance and the closest instance in the opposite class. This optimization problem can be rewritten as follows: let  $\mathcal{E}_x = \{x' \in \mathbb{R}^d \mid f(x') \neq f(x)\}$ ,

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

where  $Z$  is arbitrarily large. Here  $e^*$  is the closest instance in the opposite class, and it enables to retrieve  $a$  as  $a = e^* - x$ .

This expression, equivalent to the previous one, makes it possible to identify the *penalty*, *incompatibility* and *agg* functions. The penalty function equals  $\operatorname{cost\_fct}$  defined in [19]. The incompatibility function equals  $\mathbb{1}_{|x-e| \notin A(x)} \times Z$  and represents the presence or absence of the modified feature in the user knowledge; it takes only two values 0 or  $Z$ . An incompatible counterfactual explanation thus has a very high cost function, resulting in only compatible counterfactuals being considered. Finally, aggregation is performed by a weighted sum. However, as the incompatibility function is binary, only compatible counterfactuals are considered, so the resulting counterfactual is both compatible and of good quality.

## 4 Knowledge Integration in Counterfactual Explanation (KICE)

This section proposes a new method to generate counterfactual explanations taking into account prior knowledge, instantiating for counterfactuals the general framework introduced in the previous section. It then details the algorithm used to solve this problem. In the following, to value a distance we consider the  $l_2$  norm.

The principle of the expected counterfactual example can be illustrated in the following toy scenario; let us consider the case of a patient who risks diabetes if

he keeps his daily habits. Without knowledge integration, a plausible explanation is: "To predict a low risk, the patient has to decrease his blood sugar by 0.5g/L and increase his carbohydrate by 50g". However, the knowledge that the patient possesses is only reduced to less technical features, such as *sport duration* and *weight*: the proposed counterfactual explanation is thus not understandable to him. We aim to propose a counterfactual explanation more adapted to the user knowledge: "The patient has to run 30 minutes longer and lose 2kg".

#### 4.1 Proposed Cost Function: Instantiation of the Framework

We propose in this section an instantiation of the general framework in the counterfactual case and considering prior knowledge in the form of a set of features denoted  $E = \{a_i, i = 1, \dots, m\}$ , considered as a subset of features of the input space  $\mathcal{X}$ .

**Penalty Function** For the penalty function, we propose to use the classical counterfactual function which is the Euclidean square distance:

$$penalty_x(e) = \|x - e\|^2 \quad (2)$$

**Incompatibility Function** As specified in Section 3.1, the objective is to propose a counterfactual explanation in agreement with the user knowledge, meaning that ideally, the counterfactual modifications must be only performed according to the features appearing in  $E$ . However, for some instances, focusing only on a subset features raises the risk of never being able to meet the decision boundary of  $f$ , leading to no counterfactual explanation being generated. Therefore, we propose to relax this constraint by penalizing the modifications according to the features from  $\bar{E}$  that are not present in the knowledge. This allows, when the boundary cannot be found along the sole features of  $E$ , to make sure that a solution can still be found. Thus, we propose the incompatibility function that computes the Euclidean square distance only along absent features:

$$incompatibility_x(e, E) = \|x - e\|_{\bar{E}}^2 = \sum_{i \notin E} (x_i - e_i)^2 \quad (3)$$

Minimizing this incompatibility avoids generating counterfactual explanations that greatly modify the unknown features.

**Aggregation Function** The aggregation function combines the penalty function and the incompatibility function, we propose to use a compromise function:

$$agg(u, v) = u + \lambda v \quad (4)$$

with  $\lambda \in \mathbb{R}^+$  a hyper-parameter valued by the user.  $\lambda$  defines the weight given to the incompatibility function. In the case of categorical features, the  $l_2$  distance is replaced with an appropriate distance or a dissimilarity measure.



We then obtain the following optimization problem:

Given  $\mathcal{E} = \{x' \in \mathbb{R}^d \mid f(x') \neq f(x)\}$  and  $\lambda \in \mathbb{R}^+$ ,

$$e^* = \underset{e \in \mathcal{E}}{\operatorname{argmin}} \operatorname{cost}_{x,E}(e) \quad (5)$$

$$\text{with : } \operatorname{cost}_{x,E}(e) = \|x - e\|^2 + \lambda \|x - e\|_{\bar{E}}^2 \quad (6)$$

## 4.2 Algorithm Description

The algorithm we propose to solve optimization problem (5), named KICE for Knowledge Integration in Counterfactual Explanation, uses the principle of iterative generation of instances, inspired by the Growing Spheres algorithm [8], considering the additional term of compatibility to bias the generation. We consider a model and data agnostic case, in which no information about the data distribution nor the decision boundary is available. Therefore, we generate instances around the considered instance  $x$  (i.e. for which explanation related to  $f(x)$  should be found) in all directions: to find the point that minimizes the cost function, we generate points in increasingly larger spaces until we find one in the opposite class.

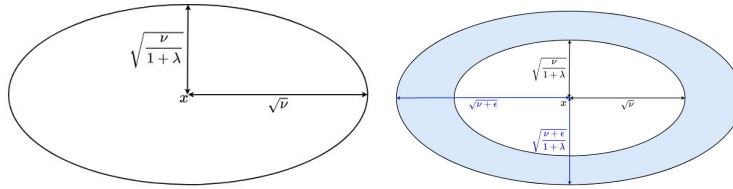
The equation  $\operatorname{cost}_{x,E}(e) = \nu$  defines the equation of an ellipse with center  $x$  and radius  $\sqrt{\frac{\nu}{1+\lambda}}$  with respect to the features in  $\bar{E}$  and  $\sqrt{\nu}$  with respect to the features in  $E$ . To make the ellipses grow, the value taken by  $\nu$  is iteratively increased. As we cannot cover all possible values for  $\nu$ , a hyper-parameter  $\epsilon > 0$  is used to generate elliptical layers defined by radius  $\nu$  and  $\nu + \epsilon$  and generate uniformly in these layers.

In order to do so, we use a modified version of the HLG algorithm [13] which consists in generating instances uniformly in the spherical layer  $SL(x, a_0, a_1)$  which represents the set of points at a distance greater than  $a_0$  and less than  $a_1$  from  $x$ . In the HLG algorithm the rescaling step consists in generating a random variable  $U = \{u_i\} \sim \mathcal{U}([0, 1])$ , and then calculating  $a_0 + a_1 U^{\frac{1}{\dim(x)}}$ . We distinguish the features  $E$  and  $\bar{E}$ , and perform the following operations:  $a_0 + a_1 u_i$  for  $i \in E$  and  $\frac{a_0}{\sqrt{1+\lambda}} + \frac{a_1}{\sqrt{1+\lambda}} u_i$  for  $i \in \bar{E}$ . The instances are then indeed generated in the desired ellipse.

The proposed KICE algorithm covers the space by generating instances iteratively: in the first step,  $n$  instances are generated in the ellipse of center  $x$  and radius  $\sqrt{\frac{\nu}{1+\lambda}}$  with respect to the features in  $\bar{E}$  and  $\sqrt{\nu}$  with respect to the features in  $E$ . This ellipse is represented in the left part of Figure 1. If none of these instances belong to the opposite class, we generate instances in the ellipsoidal layer between  $\nu$  and  $\nu + \epsilon$ . This layer is represented in blue on Figure 1.

## 5 Experiments

This section presents the experiments conducted to evaluate the KICE algorithm: the purpose of these experiments is to show that the proposed method



**Fig. 1.** (left) Generation zone for the initial step, (right) Iteration zone.

finds the expected counterfactual instance, thus it allows to achieve a trade-off between the quality and the compatibility of an explanation.

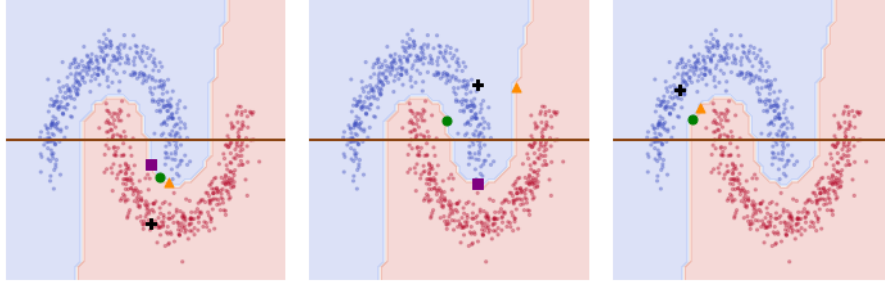
### 5.1 Experimental Protocol

Experiments are conducted on three classical benchmark tabular datasets: half-moons, Boston and breast cancer. As a pre-treatment, dataset values are normalized and in the case of the Boston dataset, the regression value is transformed into a binary class by means of a discretization step: the price is "expensive" if it is greater than \$21,000, and "cheap" otherwise. Datasets are split into train and test data subsets (80%-20%). Within the considered post-hoc explanation framework, the classifier choice does not matter, we apply SVM with a Gaussian kernel, that obtains good accuracy on the three datasets: half-moons (0.99), Boston (0.98) and breast cancer (0.93). Regarding prior knowledge, we consider the user disposes of less features than the classifier. In order to build somehow realistic knowledge, we train a decision tree with low depth on the train data (more precisely maximal depth is set to half the number of data features), the set of features  $E$  is then the ones that occur in this tree. Counterfactual instances  $e^*$  are then generated for each instance  $x$  of the test data set using KICE, with the value of  $\lambda$  being chosen to observe interesting results, respectively 4, 1 and 6 for the datasets half-moons, Boston and breast cancer.

We compare the proposed method KICE to two competitors. The first one is Growing Spheres [8], which solves the reference counterfactual optimization problem that only minimizes the Euclidean distance and denote  $e_{ref}$  the generated point. This corresponds to an extreme case of aggregation of Eq. 5 where the incompatibility term is ignored i.e.  $\lambda = 0$ . A second competitor is proposed by imposing to strictly comply with the knowledge. Its associated cost function thus minimizes the Euclidean distance according to the knowledge features only, a naive way of integrating knowledge in the explanation.

We denote  $e_{user}$  the counterfactual instance that solves the associated problem:

$$e_{user} = \operatorname{argmin}_{e \in \mathcal{E}} \|x - e\|_E^2$$



**Fig. 2.** Examples of obtained results  $e_{ref}$ ,  $e_{user}$  and  $e^*$  for three instances  $x$  (+:  $x$ ,  $\blacktriangle$ :  $e_{ref}$ ,  $\blacksquare$ :  $e_{user}$ ,  $\bullet$ :  $e^*$ )

We can write this optimization problem in the proposed formulation given in Eq. (6) with  $\lambda$  arbitrarily large:

$$e_{user} = \underset{e \in \mathcal{E}}{\operatorname{argmin}} \|x - e\|^2 + \lambda \|x - e\|_E^2$$

## 5.2 Illustrative Examples

First, we illustrate the behavior of the methods with examples in two dimensions denoted  $X_0$  and  $X_1$ . Figure 2 shows the counterfactual examples obtained for three different instances of the half-moons dataset. The figure shows the training set, the blue and red regions represent the predicted classes (darker points are the training examples); the decision boundary of the trained SVM classifier is shown in white, it achieves 0.99 accuracy. The considered knowledge system is a rule on a single feature, it is represented by the brown horizontal line :  $E = \{X_1\}$ .

In the figure, we observe that the counterfactual instance  $e_{ref}$  is the closest point belonging to the opposite class. As expected,  $e_{user}$  is further away than  $e_{ref}$  and only modifies feature  $X_1$  which is the knowledge feature: we observe that  $e^*$  is a compromise between  $e_{ref}$  and  $e_{user}$ . It requires less modifications according to  $X_0$  than  $e_{ref}$ , hence it is more compatible. It is also closer to the studied instance than  $e_{user}$ . In the graph, on the right, we notice that there is no  $e_{user}$ . In this case, there is no counterfactual instance modifying only feature  $X_1$ . The proposed method is then useful because it allows to obtain a more compatible counterfactual than  $e_{ref}$ .

## 5.3 Evaluation of the KICE Method

In this section, we further compare the proposed method with its presented competitors and study quantitatively the gain in terms of the proposed cost function offered by KICE.

		$penalty_x(e)$	$incompatibility_x(e, E)$	$cost_{x,E}(e)$
Half-moons	$e_{ref}$	<b>0.32 ± 0.21</b>	0.14 ± 0.13	0.86 ± 0.56
	$e_{user}$	1.48 ± 1.3	<b>0.0 ± 0.0</b>	1.48 ± 1.3
	$e^*$	0.42 ± 0.29	0.08 ± 0.11	<b>0.73 ± 0.52</b>
Boston	$e_{ref}$	<b>1.48 ± 1.75</b>	0.7 ± 1.03	3.57 ± 4.72
	$e_{user}$	2.26 ± 2.71	<b>0.0 ± 0.0</b>	2.26 ± 2.71
	$e^*$	1.72 ± 2.09	0.13 ± 0.19	<b>2.12 ± 2.54</b>
Breast cancer	$e_{ref}$	<b>8.82 ± 9.22</b>	7.27 ± 8.25	52.41 ± 58.63
	$e_{user}$	22.42 ± 24.87	<b>0.0 ± 0.0</b>	22.42 ± 24.87
	$e^*$	10.74 ± 9.85	1.25 ± 1.33	<b>18.24 ± 16.83</b>

**Table 1.** Results with the three considered approaches on the three datasets for metrics:  $penalty$  defined in Equation (2),  $incompatibility$  defined in Equation (3) and  $cost$  defined in Equation (6)

We apply the three methods described in the previous section on the test set of the different datasets. Among the test set, some instances do not find counterfactual examples with the method that modifies only the features in  $E$  ( $e_{user}$ ). For the datasets half-moons, Boston and breast cancer, this concerns respectively 20%, 0% and 11% of the instances. For the rest of the instances, Table 1 shows the mean and standard deviation values for the penalty, incompatibility and cost functions for the three approaches.

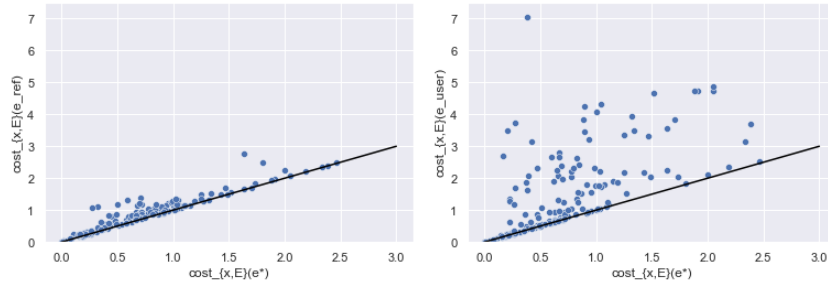
We notice that, as expected, the proposed counterfactual examples have a penalty greater than that of  $e_{ref}$  but lower than that of  $e_{user}$ . Moreover, the incompatibility function is much smaller than that of  $e_{ref}$ . Finally,  $e^*$  cost function is the lowest. We notice that the standard deviations are high, which is due the fact that the instances are at different distances from the boundary.

To verify that KICE minimizes the cost function as opposed to the other two on all instances, Figure 3 shows the value of the cost function obtained by  $e^*$  (that is defined as minimizing it) as compared to the value it takes for  $e_{ref}$  (left) and  $e_{user}$  (right), for each of the test instances of the half-moons dataset. The figures show that, as expected, all points are above the line  $y = x$ .

On the right hand graph, the points are more scattered but they remain above the line. We notice that the generated counterfactual instances are closer to the cost function of  $e_{ref}$  than to  $e_{user}$ . Only one feature is consider by the knowledge, it is difficult to get a close counterfactual instance only by modifying this feature. It is then necessary to move away from the studied instance to get 100% compatible one. It is also possible to increase  $\lambda$  so that the points are less scattered.

## 6 Conclusion and Perspectives

In this paper, a general framework is proposed to help defining an optimization problem to integrate prior knowledge in post-hoc model-agnostic explanations. Using this framework, we proposed a new method, KICE, to generate counter-



**Fig. 3.** Cost functions  $cost$  as defined in Equation (6) of  $e_{ref}$ ,  $e_{user}$  and  $e^*$  for the 80% of the test set for which all three counterfactual instances are defined.

factual explanations in the knowledge language by minimizing the modifications according to the unknown features.

Future works will include a study of the counterfactuals explanations generated for different values of  $\lambda$ . Thus, we will study the  $\lambda$  associated with the expected counterfactual examples. Another direction will focus on exploring and proposing instantiations of the framework for different models and applications, as well as real world experiments including real users.

## References

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 Neural Information Processing Systems*. 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 Information Processing and Management of Uncertainty in Knowledge-Based Systems*. 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 Neural Information Processing Systems. 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 Fairness, Accountability, and Transparency. 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 Fairness, Accountability, and Transparency. 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)