On the Computation of Example-Based Abductive Explanations for Random Forests

Gilles Audemard[1] , **Jean-Marie Lagniez**[1] , **Pierre Marquis**[1,2] and **Nicolas Szczepanski**[1]
[1]Univ. Artois, CNRS, CRIL, F-62300 Lens, France
[2]Institut Universitaire de France
{audemard, lagniez, marquis, szczepanski}@cril.fr

Abstract

We show how to define and compute *example-based abductive explanations*. Such explanations are guaranteed to be 100% correct, fairly general, and persuasive enough since they cover sufficiently many reference instances furnished by the explainee. We prove that the latter coverage condition yields a complexity shift to the second level of the polynomial hierarchy. We present a CEGAR-based algorithm to derive such explanations, and show how to modify it to derive most anchored example-based abductive explanations, i.e., example-based abductive explanations that cover as many reference instances as possible. We also explain how to reduce example-based abductive explanations to get subset-minimal explanations. Experiments in the case of random forest classifiers show that our CEGAR-based algorithm is quite efficient in practice.

1 Introduction

The field of "eXplainable AI (XAI)" has got started in the recent past [Gunning, 2019] as a response to the need of understanding Machine Learning (ML) models that are opaque. The goal of XAI is to help users of a black-box ML model to determine whether the model itself and/or the predictions that can be made from it are trustable enough. The generation of local, post-hoc explanations is among the approaches that have been developed to reach this goal. Depending on how those explanations comply with their own expectations, users may decide to accept or to reject the predictions made.

There exists a huge diversity of methods for deriving explanations from instances and trained models (see e.g., [Molnar, 2019]). This diversity reflects the fact that various explanations (in terms of nature and format) make sense and that no consensus exists about what a "good" explanation should be. Thus, many criteria for evaluating explanations (and/or the XAI methods used to produce them) have been put forward (see e.g., [Nauta *et al.*, 2023; Amgoud and Ben-Naim, 2022; Vilone and Longo, 2021; Zhou *et al.*, 2021]). Because some of those criteria are antagonistic, trade-offs must be looked for. [Yang *et al.*, 2019] presents three main criteria: fidelity, generalizability, and persuasibility. Fidelity (also known as correctness or faithfulness) indicates to which extent explanations capture the actual behaviour of the model. Generalizability concerns the number of instances covered by the explanation that is considered: the larger this number the more general the explanation. Unlike the two other criteria that depend only on the model and on the instances to be explained, the persuasibility criterion also considers user satisfaction in the evaluation of an explanation.

Several families of XAI methods have been pointed out so far. Among them are *formal XAI methods* and *example-based XAI methods*. Formal XAI methods consist in associating ML models with circuits that have the same behaviour in terms of inputs/outputs [Narodytska *et al.*, 2018; Shih *et al.*, 2018a; 2019], so that XAI queries about the models can be delegated to the circuits. Formal XAI methods are, in essence, good at fidelity. In example-based XAI methods [Molnar, 2019; Kenny *et al.*, 2021; Poché *et al.*, 2023], explanations are examples. As such, example-based XAI methods are good at persuasibility. Indeed, studies of human reasoning have shown that the use of examples is fundamental to understand and explain: humans are prone to use examples as references [Miller, 2019]. Thus, example-based explanations have been widely used in the effort to improve interpretability.

In this paper, our goal is to take the best of both worlds (formal XAI methods and example-based XAI methods) to derive explanations for ML models that are not interpretable by design, but are convincing enough for justifying their use in safety-critical applications involving a binary classification issue. To be more precise, we present a new *model-specific* approach for deriving *abductive explanations* [Ignatiev *et al.*, 2019], i.e., explanations justifying why the decision made on a given input instance has been made (whatever the decision). Because sensitive applications are targeted, fidelity is paramount. Thus, our approach is relevant to formal XAI: it guarantees that the explanations that are generated are correct, in the sense that any instance covered by an explanation is classified in the same way as the instance that triggered the generation of the explanation. Generalizability is ensured by translating instances into the space of Boolean conditions used by the predictor [Audemard *et al.*, 2023], and by focusing on abductive explanations that are irredundant, i.e., subset-minimal. Those explanations do not contain characteristics we could get rid of them without questioning correctness. Reference instances, which are supposed to be

furnished by the explainee, are leveraged to ensure that the abductive explanations for an instance $\boldsymbol{x}$ that are generated are persuasive enough for him/her. This is done by focusing on explanations that cover a preset amount k (or a maximal number) of reference instances classified in the same way as $\boldsymbol{x}$ without covering any reference instance classified in a different way.

The contribution of this paper is as follows. We define example-based abductive explanations suited to Boolean classifiers based on (possibly dependent) attributes. We identify the computational complexity of deciding whether an instance has an example-based abductive explanation that is k-anchored, i.e., that covers at least k reference instances. We show that that the problem is at *the second level of the polynomial hierarchy* in the general case (and in the specific case of random forests). Then, in order to derive example-based abductive explanations, we take advantage of the Counter-Example Guided Refinement Abstraction (CEGAR) paradigm [Clarke *et al.*, 2003]. We present a CEGAR-based algorithm to derive most anchored example-based abductive explanations, i.e., example-based abductive explanations that cover as many reference instances as possible. We also explain how to reduce example-based abductive explanations to turn them into subset-minimal ones. Experiments are made showing the algorithms to compute example-based abductive explanations practical enough in the case of random forests, despite the high complexity of the problems they solve. Due to space limits, proofs are provided as a supplementary material, available at www.cril.fr/expekctation/. Additional empirical results and the code used in our experiments are also furnished in this supplementary material.

2 Preliminaries

We suppose the reader familiar with basic notions of propositional logic. We consider a set $X = \{x_1, \ldots, x_n\}$ of Boolean variables (representing the conditions used in a decision tree, a random forest or a boosted tree). The variables in X do not necessarily represent conditions that are logically independent. Indeed, they can come from the same numerical or categorical attributes used at start for learning the classifier (for example, we can find in X a variable $x_1 = (S > 30)$ related to a numerical attribute S but also a variable $x_2 = (S > 20)$ which is logically linked to it: x_1 cannot be true while x_2 would be false). A *domain theory*, in the form of a propositional formula (or a Boolean circuit) Σ over X, indicates the dependencies between the Boolean variables in X (for instance, we may have $\Sigma = x_1 \Rightarrow x_2$).

An *instance* $\boldsymbol{x}$ over X is an n-uple of Boolean values (noted 0 and 1) that satisfies Σ. Thus, every $x \in X$ can also be viewed as a Boolean attribute. $\boldsymbol{X}$ is the set of all instances. Requiring that every $\boldsymbol{x} \in \boldsymbol{X}$ satisfies Σ ensures that only n-uples corresponding to feasible instances are considered (for instance, if $n = 2$ and x_1 and x_2 are as above, $(1, 0)$ is not an instance because it violates Σ). $t_{\boldsymbol{x}}$ is the set of literals over X making precise the characteristics of $\boldsymbol{x}$ (i.e., if $\boldsymbol{x}_i = 1$, then $t_{\boldsymbol{x}}$ contains the positive literal x_i and if $\boldsymbol{x}_i = 0$, then $t_{\boldsymbol{x}}$ contains the negative literal $\overline{x_i}$). We say that a term t, i.e., a (conjunctively-interpreted) set of literals over X, *covers* an instance $\boldsymbol{x} \in \boldsymbol{X}$ if and only if $t \subseteq t_{\boldsymbol{x}}$. The empty term is equivalent to $\top$, the Boolean constant always true. For every literal ℓ over X, we denote by $\sim \ell$ the complementary literal. Thus, when $\ell = x_i$, we have $\sim \ell = \overline{x_i}$, and when $\ell = \overline{x_i}$, we have $\sim \ell = x_i$. $\ell_i^1 = x_i$ and $\ell_i^2 = \overline{x_i}$ are the two literals over the variable x_i, and $var(\ell_i^1) = var(\ell_i^2) = x_i$.

A *binary classifier* f over X is a mapping from $\boldsymbol{X}$ to $\{0, 1\}$, associating a Boolean label $f(\boldsymbol{x})$ with any input instance $\boldsymbol{x}$. f can be represented as a propositional formula or a Boolean circuit over X. When $f(\boldsymbol{x}) = 1$, $\boldsymbol{x}$ is a *positive instance*, and when $f(\boldsymbol{x}) = 0$, $\boldsymbol{x}$ is a *negative instance*. Stated differently, we have $\boldsymbol{x} \in C_f$ if and only if $f(\boldsymbol{x}) = 1$ ($C_f \subseteq \boldsymbol{X}$ is the *concept* characterized by f).

We consider a set R_C of labelled instances $\boldsymbol{x} \in \boldsymbol{X}$ and we assume that the class associated with every $\boldsymbol{x}$ in R_C is the right class of $\boldsymbol{x}$ for the target concept C that f is expected to capture. The elements of R_C are *reference instances (alias anchors)*. That is, whenever $\boldsymbol{x} \in R_C$ is labeled as positive, the explainee is sure that $\boldsymbol{x} \in C$, while when $\boldsymbol{x} \in R_C$ is labeled as negative, the explainee is sure that $\boldsymbol{x} \notin C$. Thus, we can split R_C into two disjoint subsets, R_C^+, containing the elements of R_C labeled as positive, and R_C^-, containing the elements of R_C labeled as negative. Note that R_C is not necessarily a subset of the training set used to learn f (we do not assume that the explainee is aware of the dataset used to train the classifier). Thus, there may exist instances $\boldsymbol{x}$ belonging to R_C that are labelled differently in the training set used to learn f. In addition, there may exist instances $\boldsymbol{x}$ belonging to R_C^+ (resp. to R_C^-) that are such that $f(\boldsymbol{x}) = 0$ (resp. $f(\boldsymbol{x}) = 1$)).

The purpose of abductive explanations [Ignatiev *et al.*, 2019] is to explain why the instance $\boldsymbol{x}$ that is considered as input has been classified by f in the way it has been classified, thus addressing the "Why?" question.

Definition 1. *Given a binary classifier f over X, a set of Boolean attributes connected via a domain theory Σ, and an instance $\boldsymbol{x} \in \boldsymbol{X}$, an* abductive explanation *t for $\boldsymbol{x}$ given f and Σ is a term $t \subseteq t_{\boldsymbol{x}}$ such that for every instance $\boldsymbol{x}' \in \boldsymbol{X}$ satisfying $t \subseteq t_{\boldsymbol{x}'}$, we have $f(\boldsymbol{x}') = f(\boldsymbol{x})$.*

Equivalently, when $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$) an abductive explanation t for $\boldsymbol{x}$ given f and Σ is an implicant t of $\Sigma \Rightarrow f$ (resp. $\Sigma \Rightarrow \neg f$) that covers $\boldsymbol{x}$. Subset-minimal[1] abductive explanations for $\boldsymbol{x}$ given f and Σ are also referred to as *sufficient reasons* [Gorji and Rubin, 2022] (when Σ is valid, one recovers the notion of sufficient reasons introduced in [Darwiche and Hirth, 2020], also called PI-explanations [Shih *et al.*, 2018b]).

[1]Unlike [Ignatiev *et al.*, 2020], the notion of abductive explanations considered here does not require explanations to be minimal w.r.t. set inclusion. When Σ is valid (i.e., all the attributes in X are logically independent), the notion of abductive explanations we use thus corresponds to the notion of weak abductive explanation from [Huang *et al.*, 2021]. However, both notions do not coincide in the general case since our notion of abductive explanation takes a domain theory into account.

3 Example-Based Abductive Explanations

Anchored abductive explanations Let us start by defining formally a notion of "sufficiently anchored" explanation for the case f is a binary classifier over X, where "sufficiently" is captured by considering a minimal number k of reference instances that must be covered.

Definition 2. *Let f be a binary classifier over X, a set of Boolean attributes connected via a domain theory Σ. Let $\boldsymbol{x} \in \boldsymbol{X}$ be an instance such that $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$), R_C be a set of reference instances, and k be a non-negative integer. A k-*anchored abductive explanation *t for $\boldsymbol{x}$ given f and Σ is an abductive explanation for $\boldsymbol{x}$ given f and Σ that covers at least k instances $\boldsymbol{x}'$ from R_C^+ (resp. R_C^-) and no instance from R_C^- (resp. R_C^+).*

Let us illustrate the previous definition using a simple example (that will serve as a running example throughout the paper).

Example 1. *Suppose that $X = \{a, b, c, d\}$, $\Sigma = \top$, and that $f = (a \wedge \overline{b}) \vee (\overline{c} \wedge \overline{d}) \vee (a \wedge \overline{c} \wedge d) \vee (\overline{b} \wedge c \wedge d)$. f is represented by the Karnaugh map [Karnaugh, 1953] given in Figure 1. Suppose also that $R_C^+ = \{(0,0,1,1), (1,0,1,0), (1,1,0,0), (1,1,0,1)\}$ and that $R_C^- = \{(0,0,0,0), (0,0,1,0), (0,1,1,0)\}$. Those reference instances are provided as superscripts in the Karnaugh map. Observe that though $(0,0,0,0) \in R_C^-$, we have $f((0,0,0,0)) = 1$. Let $\boldsymbol{x} = (1,0,0,0)$. We have $f(\boldsymbol{x}) = 1$. $\boldsymbol{x}$ has three subset-minimal abductive explanations given f and Σ, namely $\{\overline{c}, \overline{d}\}$, $\{a, \overline{b}\}$, and $\{a, \overline{c}\}$.*

- *$\{\overline{c}, \overline{d}\}$ covers one element of R_C^+ ($(1,1,0,0)$) and one element of R_C^- ($(0,0,0,0)$). While it is an abductive explanation for $\boldsymbol{x}$ given f and Σ and it covers one element of R_C^+, the fact that it also covers one element of R_C^- prevents $\{\overline{c}, \overline{d}\}$ from being a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, whatever k.*
- *$\{a, \overline{b}\}$ covers one element of R_C^+ ($(1,0,1,0)$) and no element of R_C^-, thus $\{a, b\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.*
- *Finally, $\{a, \overline{c}\}$ covers two elements of R_C^+ ($(1,1,0,0)$ and $(1,1,0,1)$) and no element of R_C^-, thus $\{a, \overline{c}\}$ is a 2-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.*

Accordingly, $\{a, \overline{c}\}$ can be viewed as a better explanation than $\{a, \overline{b}\}$ since it covers more instances than $\{a, \overline{b}\}$ that are classified in the same way as $\boldsymbol{x}$. $\{a, \overline{b}\}$ can be viewed as a better explanation than $\{\overline{c}, \overline{d}\}$ because $\{\overline{c}, \overline{d}\}$ covers an instance classified by f in a different way than $\boldsymbol{x}$.

Obviously enough, every k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ also is a k'-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, for every $k' \leq k$. Furthermore, every instance $\boldsymbol{x}$ has an abductive explanation given f and Σ that does not cover any instance from R_C classified by f in a different way: $t_{\boldsymbol{x}}$ is such an abductive explanation. Hence, every $\boldsymbol{x}$ has a 0-anchored abductive explanation given f and Σ. Note nevertheless that the set of abductive explanations for $\boldsymbol{x}$ given f and Σ and the set of 0-anchored abductive explanations for $\boldsymbol{x}$ given f and Σ do not coincide in general: every 0-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ is an abductive explanation for $\boldsymbol{x}$ given f and Σ, but the two sets are not always equal. Indeed, on the previous example, $\{\overline{c}, \overline{d}\}$ is an abductive explanation for $\boldsymbol{x}$ given f and Σ but it is not a 0-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ since it covers $(0,0,0,0)$ and $(0,0,0,0) \in R_C^-$ while $f(\boldsymbol{x}) = 1$.

f ($c\,d$ \ $a\,b$)	00	01	11	10
00	$1^{\mathbf{0}}$	0	$1^{\mathbf{1}}$	$0^{\mathbf{0}}$
01	1	0	0	$0^{\mathbf{0}}$
11	$1^{\mathbf{1}}$	$1^{\mathbf{1}}$	0	0
10	$\underline{1}$	1	1	$1^{\mathbf{1}}$

Figure 1: A Karnaugh map for f. Each cell corresponds to an instance from $\boldsymbol{X}$, labelled by 1 when it is positive, and by 0 when it is negative. Instances from R_C are provided as superscripts. The instance $\boldsymbol{x} = (1,0,0,0)$ to be explained is underlined and red coloured.

As soon as $k > 0$, the existence of a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ is not ensured in general. Thus, on the previous example, $\boldsymbol{x}$ has no 3-anchored abductive explanation given f and Σ.

In the general case, deciding whether an instance $\boldsymbol{x}$ has a k-anchored abductive explanation given f and Σ is at the second level of the polynomial hierarchy:[2]

Proposition 1. *Given a domain theory about X (represented by a propositional formula or a Boolean circuit Σ), a binary classifier f over X (represented by a propositional formula or a Boolean circuit), an instance $\boldsymbol{x} \in \boldsymbol{X}$, a set $R_C \subseteq \boldsymbol{X}$ of reference instances and an integer $k > 0$, deciding whether an instance $\boldsymbol{x} \in \boldsymbol{X}$ has a k-anchored abductive explanation given f and Σ is Σ_2^p-complete in the general case (and this complexity holds in the restricted case when f is represented by a random forest).*

Most anchored abductive explanations Instead of considering that the value of the bound k to be used has been provided by the explainee, we can determine its maximal value via an optimization process. In that case, one is interested in generating abductive explanations for $\boldsymbol{x}$ among the most anchored ones:

Definition 3. *Given a domain theory about X (represented by a propositional formula or a Boolean circuit Σ), a binary classifier f over X (represented by a propositional formula*

[2] In the supplementary material, we identify some conditions that makes the problem "only" NP-complete. Those conditions are satisfied when f is a decision tree.

or a Boolean circuit), and an instance $\boldsymbol{x} \in \boldsymbol{X}$, a set R_C of reference instances, and a non-negative integer k, a most anchored abductive explanation *t for $\boldsymbol{x}$ given f and Σ is a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ such that k is maximal (i.e., no $k+1$-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ exists).*

A straightforward observation is that since every instance $\boldsymbol{x}$ has a 0-anchored abductive explanation given f and Σ, every instance $\boldsymbol{x}$ also has a most anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.

Example 2 (Example 1 cont'd)**.** *On the previous example, $\boldsymbol{x}$ has a unique most anchored abductive explanation given f and Σ, namely $\{a, \overline{c}\}$, and this explanation is 2-anchored. It can be observed that $\{a, \overline{c}\}$ is a subset-minimal abductive explanation for $\boldsymbol{x}$ given f and Σ.*

However, in the general case, it is not ensured that the most anchored abductive explanations for $\boldsymbol{x}$ given f and Σ are among the subset-minimal abductive explanations for $\boldsymbol{x}$ given f and Σ. This comes from the condition stating that no instance from R_C classified by f in a different way than $\boldsymbol{x}$ can be covered by a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.

Example 3 (Example 1 cont'd)**.** *Considering the previous example, if $(1,0,0,1)$ was added to R_C^-, none of the subset-minimal abductive explanations for $\boldsymbol{x}$ given f and Σ (i.e., $\{a, \overline{c}\}$, $\{a, \overline{b}\}$, and $\{\overline{c}, \overline{d}\}$) would be a 0-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ: after such an update of R_C^-, the most anchored abductive explanations for $\boldsymbol{x}$ given f and Σ would be $\{a, \overline{c}, \overline{d}\}$ and $\{a, \overline{b}, \overline{d}\}$ (those explanations are 1-anchored abductive explanations for $\boldsymbol{x}$ given f and Σ) but they are not among the subset-minimal abductive explanations for $\boldsymbol{x}$ given f and Σ.*

Obviously enough, as a by-product of Proposition 1, the computation of a most anchored abductive explanation for $\boldsymbol{x}$ given f and Σ is Σ_p^2-hard in the general case. Indeed, once a most anchored abductive explanation t for $\boldsymbol{x}$ given f and Σ has been computed, one can decide in polynomial time for any k whether $\boldsymbol{x}$ has a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ: it is enough to count the number max of instances of R_C^+ covered by t when $f(\boldsymbol{x}) = 1$ (resp. the number max of instances of R_C^- covered by t when $f(\boldsymbol{x}) = 0$) and to compare max with k to determine whether $\boldsymbol{x}$ has a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. This is the case if and only if $max \geq k$.

4 Computing Example-Based Explanations

We now show how to derive a subset-minimal most anchored abductive explanation for a given instance $\boldsymbol{x}$ given f and Σ. Our algorithm is based on linear search where more and more anchored explanations are successively looked for. Thus, the algorithm consists in looking first for an a-anchored abductive explanation with $a = 1$, and if such an explanation t is found, to remove redundant literals from it to derive a subset-minimal explanation t_{smin}; then one computes from t_{smin} the largest integer i such that t_{smin} is an i-anchored abductive explanation, and the algorithm resumes with $a = i + 1$. Of course, an algorithm to compute a k-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ for a fixed k can be easily derived as a by-product of the latter algorithm (it is enough to stop the linear search whenever $a \geq k$). If subset-minimality is not requested, one can also not run the code that remove redundant literals, at least at the last step (for the previous steps, the shifts from a to $i + 1$ that are made possible via subset-minimization prove computationally useful in general for improving the linear search).

Let us first explain how to compute an a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, where a is any positive integer. Our approach relies on a two-phase procedure reminiscent to the well-established Counter-Example Guided Refinement Abstraction (CEGAR) paradigm [Clarke *et al.*, 2003]. Given $a \in \mathbb{N}^*$, our approach iteratively generates a candidate t and then tests whether t actually is an a-anchored abductive explanation given f and Σ. Each candidate t is derived from a model of a CNF formula Φ generated from $\boldsymbol{x}$, R_C, and k. By construction, when $\boldsymbol{x}$ is such that $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$), a candidate t is a term satisfying $t \subseteq t_{\boldsymbol{x}}$ such that at least a instances of R_C^+ (resp. R_C^-) are covered while avoiding any instances from R_C^- (resp. R_C^+) . For each candidate t generated, the verification process then checks whether t is an abductive explanation for $\boldsymbol{x}$ given f and Σ. Since each test corresponds to an instance of a coNP-problem, one uses an NP oracle to achieve each of them. The generation step is repeated until every candidate has been considered but none of them has been retained (this shows that no a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ exists) or a candidate that qualifies as an a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ has been found.

The CNF formula Φ used to characterize the candidates t is based on additional Boolean variables. For each $\ell_i \in t_{\boldsymbol{x}}$ such that $var(\ell_i) = x_i$ (an element of $X = \{x_1, \ldots, x_n\}$), a variable s_{ℓ_i} is introduced. For any model ω of Φ, s_{ℓ_i} is set to 1 in ω precisely when the literal $\ell_i \in t_{\boldsymbol{x}}$ belongs to the candidate t associated with ω. Let $R_C^{\boldsymbol{x}}$ be the subset of R_C consisting of instances classified by f in the same way as $\boldsymbol{x}$. We have $R_C^{\boldsymbol{x}} = R_C^+$ when $\boldsymbol{x}$ is a positive instance and $R_C^{\boldsymbol{x}} = R_C^-$ when $\boldsymbol{x}$ is a negative instance. New variables associated with elements of $R_C^{\boldsymbol{x}}$ are also considered. To be more precise, a variable $p_{\boldsymbol{x}'}$ is introduced for each reference instance $\boldsymbol{x}'$ from $R_C^{\boldsymbol{x}}$. For any model ω of Φ, $p_{\boldsymbol{x}'}$ is set to 1 in ω precisely when the candidate t associated with ω covers both $\boldsymbol{x}$ and $\boldsymbol{x}'$.

The CNF formula Φ consists of the conjunction of three CNF formulae (1), (2), and (3).

$$\bigwedge_{\boldsymbol{x}' \in R_C^{\boldsymbol{x}}} \bigwedge_{\ell_i \in t_{\boldsymbol{x}} \setminus t_{\boldsymbol{x}'}} \neg s_{\ell_i} \vee \neg p_{\boldsymbol{x}'} \tag{1}$$

$$\texttt{CNF}\Big(\sum_{\boldsymbol{x}' \in R_C^{\boldsymbol{x}}} p_{\boldsymbol{x}'} \geq a\Big) \tag{2}$$

$$\bigwedge_{\boldsymbol{x}' \in R_C \setminus R_C^{\boldsymbol{x}}} \bigvee_{\ell_i \in t_{\boldsymbol{x}} \setminus t_{\boldsymbol{x}'}} s_{\ell_i} \tag{3}$$

The CNF formula (1) is composed of binary clauses indicating how the variables s_{ℓ_i} ($i \in [n]$) and $p_{\boldsymbol{x}'}$ ($\boldsymbol{x}' \in R_C^{\boldsymbol{x}}$) are

connected. Basically, whenever $p_{\boldsymbol{x}'}$ is set to 1, every literal ℓ_i belonging to $t_{\boldsymbol{x}}$ but not to $t_{\boldsymbol{x}'}$ must be set to 0 in order to ensure that the candidates t to be considered cover both $\boldsymbol{x}$ and $\boldsymbol{x}'$. This is ensured by considering the clause $\neg s_{\ell_i} \vee \neg p_{\boldsymbol{x}'}$ within (1), that forces s_{ℓ_i} to be set to 0 when $p_{\boldsymbol{x}'}$ is set to 1.

The CNF formula (2) ensures that any candidate t covers at least a instances from $R_C^{\boldsymbol{x}}$. Primarily, the cardinality constraint $\sum_{\boldsymbol{x}' \in R_C^{\boldsymbol{x}}} p_{\boldsymbol{x}'} \geq a$ is considered, and this constraint is turned into a CNF formula using state-of-the-art encoding techniques, as presented in [Roussel and Manquinho, 2021].

Finally, for being a candidate, t must not cover any instance $\boldsymbol{x}'$ from $R_C \setminus R_C^{\boldsymbol{x}}$. We have that t does not cover $\boldsymbol{x}'$ when there is at least one literal $\ell_i \in t_{\boldsymbol{x}}$ such that $\sim \ell_i \in t_{\boldsymbol{x}'}$ and ℓ_i belongs to t. To implement this, for each $\boldsymbol{x}'$ from $R_C \setminus R_C^{\boldsymbol{x}}$ a clause consisting of all the selector variables s_{ℓ_i} associated with the literals ℓ_i such that $\ell_i \in t_{\boldsymbol{x}}$ and $\sim \ell_i \in t_{\boldsymbol{x}'}$ must be satisfied.

Example 4 (Example 1 cont'd)**.** *To avoid too heavy notations, let us index the instances from $R_C^{\boldsymbol{x}} = R_C^+$ as follows:*

$$\underbrace{(0,0,1,1)}_{1}, \underbrace{(1,0,1,0)}_{2}, \underbrace{(1,1,0,0)}_{3}, \underbrace{(1,1,0,1)}_{4}.$$

Suppose that $a = 1$. The formula Φ for the running example is then composed of the following clauses:

$$\neg s_a \vee \neg p_1 \quad \neg s_{\overline{c}} \vee \neg p_1 \quad \neg s_{\overline{d}} \vee \neg p_1 \quad \neg s_{\overline{c}} \vee \neg p_2$$
$$\neg s_{\overline{b}} \vee \neg p_3 \quad \neg s_{\overline{b}} \vee \neg p_4 \quad \neg s_{\overline{d}} \vee \neg p_4$$

$$\texttt{CNF}(p_1 + p_2 + p_3 + p_4 \geq 2)$$

$$s_a \qquad s_a \vee s_{\overline{c}} \qquad s_a \vee s_{\overline{b}} \vee s_{\overline{c}}$$

By construction, every model ω of Φ characterizes a candidate t that consists of the literals ℓ_i of $t_{\boldsymbol{x}}$ such that $\omega(s_{\ell_i}) = 1$. t covers at least a elements of $R_C^{\boldsymbol{x}}$ and no element of $R_C \setminus R_C^{\boldsymbol{x}}$. If there is no such model ω, i.e., if Φ is unsatisfiable, then no a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ exists. In the remaining case, in order to determine whether t qualifies as a true a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, we need to verify whether $t \models \Sigma \Rightarrow f$ holds when $f(\boldsymbol{x}) = 1$ (and whether $t \models \Sigma \Rightarrow \neg f$ holds when $f(\boldsymbol{x}) = 0$). This is equivalent to examining whether the formula $\Gamma = t \wedge \Sigma \wedge \neg f$ is unsatisfiable when $f(\boldsymbol{x}) = 1$ (and whether $\Gamma = t \wedge \Sigma \wedge f$ is unsatisfiable when $f(\boldsymbol{x}) = 0$). If this verification condition holds, i.e., if Γ is unsatisfiable, then t is an a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. Accordingly, the algorithm may stop if a is considered by the explainee as sufficiently large (i.e., a has reached a preset bound k).

Clearly enough, the resulting explanation t is not guaranteed to be subset-minimal (it is not in general). If we are interested in deriving from t a subset-minimal a-anchored abductive explanation t_{smin} for $\boldsymbol{x}$ given f and Σ, a simple greedy algorithm can be used. This greedy algorithm consists in considering the literals ℓ_i of t in a specific order and to test iteratively whether $(t \setminus \{\ell_i\}) \wedge \Sigma \wedge \neg f$ is still unsatisfiable when $f(\boldsymbol{x}) = 1$ (and whether $(t \setminus \{\ell_i\}) \wedge \Sigma \wedge f$ is still unsatisfiable when $f(\boldsymbol{x}) = 0$). If this is the case and $t \setminus \{\ell_i\}$ does not cover any instance from $R_C \setminus R_C^{\boldsymbol{x}}$, literal ℓ_i can be definitely removed from t (i.e., in the algorithm, t is replaced by $t \setminus \{\ell_i\}$), otherwise it is kept and it will be kept when the next literals of t will be processed. At the end, when all the literals belonging to t at start have been processed, the resulting term t_{smin} is a subset-minimal a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. Note that there is no need to test that whether t_{smin} covers at least a elements of $R_C^{\boldsymbol{x}}$ since this is necessarily the case (t_{smin} is a subset of t, so that every instance covered by t also is covered by t_{smin}). Though it is guaranteed that t_{smin} is a subset-minimal a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, it turns out that t_{smin} can be more than a-anchored, thanks to the removal of redundant literals that took place when computing it.

From t_{smin}, we can compute in linear time the number i of elements of $R_C^{\boldsymbol{x}}$ that are covered by t_{smin}. Thus, when the goal is to compute a most anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, the algorithm may resume with $a = i + 1$ as the next bound to be tested. Whenever $i > a$, the corresponding shift in the linear search is useful to get rid of computationally expensive, yet useless steps.

Example 5 (Example 1 cont'd)**.** *Suppose that the model ω of Φ has been found, such that ω satisfies $s_a, \neg s_{\overline{b}}$, $s_{\overline{c}}, s_{\overline{d}}, \neg p_1, \neg p_2, p_3, \neg p_4$. The corresponding candidate $t = \{a, \overline{c}, \overline{d}\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.*

Considering the literals of t in sequence as folllows a, $\overline{c}$, $\overline{d}$, the greedy algorithm tests first whether $\{\overline{c}, \overline{d}\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. This is not the case since this term covers the instance $(0,0,0,0)$ from $R_C \setminus R_C^{\boldsymbol{x}}$. Thus, at the next iteration, the greedy algorithm tests whether $\{a, \overline{d}\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. This is not the case since this term covers the model $(1,1,1,0)$ of $\neg f$ while $f(\boldsymbol{x}) = 1$. At the following iteration, the greedy algorithm tests whether $\{a, \overline{c}\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. The test succeeds, so the subset-minimal explanation $t_{smin} = \{a, \overline{c}\}$ is returned.

Finally, one can realize that t_{smin} actually is a 2-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ since it covers not only the instance $(1,1,0,0)$ from R_C^+ (just as the candidate $t = \{a, \overline{c}, \overline{d}\}$ one started with), but also the instance $(1,1,0,1)$ from R_C^+. Thus, once t_{smin} has been identified, there is no need to resume the search while looking for a 2-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, since such an explanation has already been identified. The next step is thus to consider whether a 3-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, which is not the case for the running example (so the algorithm stops and t_{smin} is returned).

Now, if the verification condition is not met, indicating that the candidate t is not a true a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ, we need to search for an alternative candidate. To be sure that the spurious candidate term t will not be considered again during the generation step, one needs to block it. This could be achieved by simply adding to Φ a clause composed of the complementary literals to those satisfied by the model ω of Φ that characterizes t. However, this approach would be rather inefficient as it requires the addition to Φ of very large clauses, and each clause eliminates only a single model of Φ. A more robust approach consists

in adding to Φ a constraint asserting that any subset of the candidate term t cannot be a valid a-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. Thus, the next candidates must include at least one literal not belonging to t. This can be ensured by adding to Φ a single clause given by (4):

$$\bigvee_{\ell_i \in t_{\boldsymbol{x}} \setminus t} s_{\ell_i} \tag{4}$$

Example 6 (Example 1 cont'd)**.** *Assume that $a = 1$ and that at the first iteration, the model ω_1 of Φ satisfying $s_a, \neg s_{\overline{b}}, \neg s_{\overline{c}}, \neg s_{\overline{d}}, \neg p_1, p_2, p_3, p_4$ has been computed. The associated candidate $t_1 = \{a\}$ is s.t. $\Gamma = t_1 \wedge \top \wedge \neg f \equiv a \wedge \neg f$ is satisfiable, as $\{a, b, c, \neg d\}$ is a model of Γ. Thus, the verification condition does not hold. $t_1 = \{a\}$ is deemed a spurious candidate, prompting the addition to Φ of the clause $s_{\overline{b}} \vee s_{\overline{c}} \vee s_{\overline{d}}$.*

At the second iteration, suppose that the model ω_2 of $\Phi \wedge (s_{\overline{b}} \vee s_{\overline{c}} \vee s_{\overline{d}})$ has been found, where ω_2 satisfies $s_a, \neg s_{\overline{b}}, s_{\overline{c}}, \neg s_{\overline{d}}, \neg p_1, \neg p_2, p_3, p_4$. The associated candidate is $t_2 = \{a, \overline{c}\}$. Since $t_2 \wedge \neg f$ is unsatisfiable, the verification condition holds, showing that $t_2 = \{a, \overline{c}\}$ is a 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ. The derivation from t_2 of a subset-minimal 1-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ via the greedy algorithm leads to keep t_2 as a whole (t_2 does not contain any redundant literal). Then we can count the number of elements of $R_C^{\boldsymbol{x}}$ that are covered by t_2 and realize that t_2 actually is a subset-minimal 2-anchored abductive explanation for $\boldsymbol{x}$ given f and Σ.

5 Experiments

We made some experiments is order to figure out how our algorithm for deriving subset-minimal most anchored abductive explanations scales up. We focused on binary classifiers f represented by random forests.

Experimental setup The empirical protocol we considered was as follows. We have focused on 14 datasets issued from three well-known repositories, namely OpenML[1] (openml.org), UCI[2] (archive.ics.uci.edu/ml/), and UCR[3] (timeseriesclassification.com). Multi-class datasets (*balance* and *arrowhead*) have been converted into datasets for binary classification using the one-vs-one approach. Ten random forests have been learned per dataset using `scikit-learn`. All the hyperparameters have been set to their default values (100 trees per forest). Categorical features have been one-hot encoded. Numerical features, have been binarized on-the-fly by the random forest learning algorithm. A 10-fold cross validation process has been achieved. All the experiments have been conducted on a computer equipped with Intel(R) XEON E5-2637 CPU @ 3.5 GHz and 128 Gib of memory.

Table 1 summarizes information about the datasets used and the random forests learned. The first four columns give respectively the dataset name (with a superscript indicating the repository from which it comes), the number of features ($\#F$), the number of instances ($\#I$) and the average number of Boolean conditions ($\#V$) used in the 10 random forests. The fifth column ($\%L$) indicates the percentage of instances of the dataset that are labelled as negative. Finally, the last column gives the mean $F1$-score of the forests that have been learned.

Dataset	$\#F$	$\#I$	$\#V$	$\%L$	$F1\text{-}score$
balance[2]	4	337	28.0	85	90.72(±2.09)
compas[1]	11	6172	69.5	54	61.03(±2.23)
breasttumor[1]	37	286	114.9	58	62.6(±10.63)
divorce[2]	54	170	116.5	51	97.59(±3.9)
cleveland[1]	22	303	663.3	54	80.47(±6.83)
wine[3]	234	111	851.2	51	97.31(±4.22)
arrowhead[3]	249	146	879.4	45	85.39(±10.56)
australian[1]	38	690	1564.0	56	85.92(±5.69)
biodegradation[1]	41	1055	5730.7	66	90.98(±2.1)
dexter[1]	20000	600	7892.9	50	93.38(±2.76)
spambase[2]	57	4601	15005.5	61	96.24(±0.48)
gisette[1]	5000	7000	24464.6	50	97.54(±0.6)
mnist38[1]	784	13966	32638.6	51	98.67(±0.41)
christine[1]	1636	5418	43587.5	50	71.34(±2.16)

Table 1: Dataset characteristics and classification performance of the random forests learned.

Our goal was to evaluate the algorithm for computing a subset-minimal most anchored abductive explanation, presented in Section 4. For each dataset and each random forest learned, a set R_C of reference instances has been selected uniformly at random from the training set. The number of reference instances retained in R_C varied to consider 5%, 10% or 20% of the instances. Several encodings of random forests into `CNF` formulae exist (e.g., [Audemard *et al.*, 2020; Izza and Marques-Silva, 2021]), we used the one presented in [Audemard *et al.*, 2022]. A domain theory has been considered connecting the non-independent Boolean conditions used in the forests. Ten instances picked up uniformly at random in the corresponding test set have been considered. Thus, a total number of 100 instances per dataset has been considered. For each of them, the algorithm presented in Section 4 has been run in order to compute a most anchored abductive explanation and the number k of reference instances it covers. The SAT solver `glucose` [Audemard and Simon, 2009] has been used to address the various (un)satisfiability tests encountered at each run. A timeout (TO) of 60s has been considered per instance.

Experimental results Table 2 synthesizes the empirical results. The three main columns gather the results obtained for the three percentages used (5%, 10%, and 20%). Each main column is divided into six parts. The number of instances $\boldsymbol{x}$ for which the time limit has been reached during the search for a 1-anchored explanation is given in columns $k_{=0}^{TO}$. The number of instances $\boldsymbol{x}$ for which our algorithm has proved that no 1-anchored explanation exists is given by columns $k_{=0}^{\overline{TO}}$. The number of instances $\boldsymbol{x}$ for which our algorithm has found during the search an explanation that is at least 1-anchored is given by columns $k_{>0}$. Accordingly, we have $k_{=0}^{TO} + k_{=0}^{\overline{TO}} + k_{>0} = 100$. Columns $k_{>0}^{AVG}$ indicate the mean number of reference instances covered by the most anchored explanation that has been derived before the time limit, provided that this number is > 0. Columns $time$ give the mean runtime (in seconds) used to compute a most anchored explanation when found before the time limit, and columns TO indicate the number of timeout (out of 100) that have been

Dataset	5%						10%						20%					
	$k_{=0}^{TO}$	$k_{=0}^{\overline{TO}}$	$k_{>0}$	$k_{>0}^{AVG}$	$time$	TO	$k_{=0}^{TO}$	$k_{=0}^{\overline{TO}}$	$k_{>0}$	$k_{>0}^{AVG}$	$time$	TO	$k_{=0}^{TO}$	$k_{=0}^{\overline{TO}}$	$k_{>0}$	$k_{>0}^{AVG}$	$time$	TO
balance	0	8	92	4.9(±2.0)	0.1(±0.0)	0	0	2	98	9.2(±3.9)	0.1(±0.0)	0	0	0	100	17.9(±6.6)	0.1(±0.0)	0
compas	0	50	50	6.3(±6.6)	0.2(±0.1)	0	0	52	48	6.6(±4.8)	0.3(±0.1)	0	0	68	32	8.6(±6.0)	0.6(±0.1)	0
breasttumor	0	34	66	1.7(±0.9)	0.2(±0.3)	0	0	24	76	2.5(±1.6)	0.3(±0.3)	0	0	12	88	3.6(±2.5)	0.4(±0.5)	0
divorce	0	0	100	5.3(±1.1)	0.1(±0.0)	0	0	0	100	14.8(±0.6)	0.1(±0.0)	0	0	0	100	32.8(±3.2)	0.2(±0.0)	0
cleveland	0	6	94	4.1(±1.6)	1.5(±1.0)	0	0	4	96	7.9(±3.2)	2.9(±2.2)	0	0	2	98	13.6(±6.0)	6.2(±5.2)	0
wine	0	26	74	1.9(±0.8)	0.2(±0.1)	0	0	14	86	2.6(±1.1)	0.3(±0.1)	0	0	6	94	3.8(±1.7)	0.6(±0.5)	0
arrowhead	0	18	82	2.8(±1.3)	0.3(±0.1)	0	0	18	82	6.2(±2.1)	0.5(±0.2)	0	0	18	82	11.5(±3.3)	6.3(±4.8)	0
australian	0	2	98	6.1(±2.1)	44.1(±19.6)	54	0	2	98	10.2(±4.2)	53.0(±14.2)	74	0	0	100	16.6(±8.3)	55.7(±12.9)	86
biodegradation	2	6	92	2.5(±1.4)	40.8(±22.7)	52	4	6	90	3.1(±1.6)	51.3(±15.6)	67	4	6	90	4.3(±2.1)	59.8(±1.3)	90
dexter	0	30	70	2.5(±1.9)	14.3(±16.4)	6	0	28	72	3.2(±2.3)	33.1(±23.1)	22	0	26	74	4.3(±3.2)	51.4(±16.5)	56
spambase	18	8	74	1.6(±0.9)	60.0(±0.0)	92	28	8	64	1.9(±1.3)	60.0(±0.0)	92	24	6	70	2.5(±1.6)	58.7(±7.8)	92
gisette	0	18	82	2.6(±0.7)	60.0(±0.0)	82	0	14	86	2.5(±0.7)	60.0(±0.0)	86	0	10	90	2.0(±0.6)	60.0(±0.0)	90
mnist38	17	0	83	1.0(±0.2)	60.0(±0.0)	100	12	0	88	1.1(±0.3)	60.0(±0.0)	100	19	0	81	1.1(±0.3)	60.0(±0.0)	100
christine	5	90	5	1.0(±0.0)	60.0(±0.0)	10	14	86	0	− − −−	− − −−	14	73	27	0	− − −−	− − −−	73

Table 2: Empirical results.

reached before the normal termination of the algorithm (i.e., when a subset-minimal most anchored explanation for $\boldsymbol{x}$ has been computed).

The datasets in Table 2 are sorted according to their difficulty, assessed by the value of $\#V$ in Table 1, that appears strongly correlated to the performance of our algorithm, as reflected by columns $time$ and TO. Looking at columns TO, it turns out that the optimal value of k has systematically been found (i.e., no timeout) for half the datasets, i.e., up to the *arrowhead* dataset whatever the percentage used, and the time needed to determine this optimal value was very short (columns $time$). Furthermore, with a single exception, our algorithm has been able to point out useful information about at least 72% of the instances (even when the algorithm did not terminate normally). Indeed, for many instances $\boldsymbol{x}$ (their numbers being given in columns $k_{=0}^{\overline{TO}}$), the algorithm succeeded in showing that no k-anchored explanation (with $k > 0$) exists for $\boldsymbol{x}$. For many other instances $\boldsymbol{x}$ (their numbers being given in columns $k_{>0}$), the algorithm succeeded in showing that a k-anchored explanation (with $k > 0$) exists for $\boldsymbol{x}$. It can be checked that the sum $k_{=0}^{\overline{TO}} + k_{>0}$ exceeds 72 for each dataset, whatever the percentage used, except for *christine* when the percentage used was 20%. Finally, our experiments has shown that the datasets used exhibit a significant discrepancy as to the number of instances that only have 0-anchored explanations.

6 Other Related Work

Our notion of anchored explanations should not be confused with the notion of dataset-based abductive explanations (aka sample-based explanations) introduced recently in [Cooper and Amgoud, 2023]. The (weak) dataset-based abductive explanations for $\boldsymbol{x}$ given f as pointed out in [Cooper and Amgoud, 2023] correspond to the abductive explanations for $\boldsymbol{x}$ given f in the sense of Definition 1 provided that the sole instances that are feasible are those in R_C. A strong point of (weak) dataset-based abductive explanations is that they can be identified, derived and minimized w.r.t. set-inclusion, in polynomial time. Furthermore, their computation does not require to have a representation of the classifier f available (an oracle for computing f is enough, thus black-box classifiers can be taken into account). The main downside of such explanations is that their correctness is not guaranteed. For instance, considering Example 1 again, a subset-minimal dataset-based abductive explanation for $\boldsymbol{x} = (1, 0, 0, 0)$ is $\{a\}$. Indeed, when the two instances $(1, 1, 1, 1)$ and $(1, 1, 1, 0)$ are considered as impossible because they do not belong to R_C, one may assume that the corresponding decisions for them is 1. However, this does not comply with the decisions produced by the classifier. Indeed, $f((1, 1, 1, 1)) = f((1, 1, 1, 0)) = 0$ showing that $\{a\}$ does not properly explain the behaviour of the classifier f when it predicts a positive decision for $\boldsymbol{x}$ since $f(\boldsymbol{x}) = 1$.

7 Conclusion and Perspectives

We have defined example-based abductive explanations suited to binary classifiers. We proved that deciding whether a k-anchored abductive explanation for an instance exists is at the second level of the polynomial hierarchy when $k > 0$, which precludes the existence of efficient algorithms for generating such explanations. Nevertheless, we designed a CEGAR-based algorithm to derive subset-minimal most anchored abductive explanations. To evaluate its performance in practice, we focused on binary classifiers represented by random forests. Empirical results showed that despite the intrinsically high complexity of the problem it solves, our CEGAR-based algorithm is practical enough for "mildly hard" datasets, i.e., those leading to random forests based on up to a thousand Boolean conditions.

Several perspectives for further research can be pointed out. Thus, we plan to improve our CEGAR-based algorithm. A way to do it consists, during the generation phase, in looking for candidates t such that the number of literals of $t_{\mathbf{x}}$ belonging to t is as high as possible. That way, clause (4) added at the refinement phase would restrict the remaining part of the search space in a more drastic way. Candidates t could be generated using a time-efficient local search approach (in that case the maximality of the number of literals of $t_{\mathbf{x}}$ belonging to t would not be ensured) or using a MaxSAT solver (in that case, the optimality would be guaranteed at the price of a generation phase that would be computationally more demanding). In addition, it would be interesting to evaluate the empirical performance of our CEGAR-based algorithm when other binary classifiers than random forests are used.

Acknowledgements

Many thanks to the anonymous reviewers for their comments and insights. This work has benefited from the support of the AI Chair EXPEKCTATION (ANR-19-CHIA-0005-01) and of the France 2030 MAIA Project (ANR-22-EXES-0009) of the French National Research Agency (ANR). It was also partially supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

[Amgoud and Ben-Naim, 2022] L. Amgoud and J. Ben-Naim. Axiomatic foundations of explainability. In *Proc. of IJCAI'22*, pages 636–642, 2022.

[Audemard and Simon, 2009] G. Audemard and L. Simon. Predicting learnt clauses quality in modern SAT solvers. In *Proc. of IJCAI'09*, pages 399–404, 2009.

[Audemard *et al.*, 2020] G. Audemard, F. Koriche, and P. Marquis. On tractable XAI queries based on compiled representations. In *Proc. of KR'20*, pages 838–849, 2020.

[Audemard *et al.*, 2022] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, and P. Marquis. Trading complexity for sparsity in random forest explanations. In *Proc. of AAAI'22*, pages 5461–5469. AAAI Press, 2022.

[Audemard *et al.*, 2023] G. Audemard, J.-M. Lagniez, P. Marquis, and N. Szczepanski. On contrastive explanations for tree-based classifiers. In *Proc. of ECAI'23*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, pages 117–124. IOS Press, 2023.

[Clarke *et al.*, 2003] E. M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. Counterexample-guided abstraction refinement for symbolic model checking. *J. ACM*, 50(5):752–794, 2003.

[Cooper and Amgoud, 2023] M. C. Cooper and L. Amgoud. Abductive explanations of classifiers under constraints: Complexity and properties. In *Proc. of ECAI'23*, pages 469–476, 2023.

[Darwiche and Hirth, 2020] A. Darwiche and A. Hirth. On the reasons behind decisions. In *Proc. of ECAI'20*, pages 712–720, 2020.

[Gorji and Rubin, 2022] N. Gorji and S. Rubin. Sufficient reasons for classifier decisions in the presence of domain constraints. In *Proc. of AAAI'22*, pages 5660–5667, 2022.

[Gunning, 2019] D. Gunning. DARPA's explainable artificial intelligence (XAI) program. In *Proc. of IUI'19*, 2019.

[Huang *et al.*, 2021] X. Huang, Y. Izza, A. Ignatiev, M. C. Cooper, N. Asher, and J. Marques-Silva. Efficient explanations for knowledge compilation languages. *CoRR*, abs/2107.01654, 2021.

[Ignatiev *et al.*, 2019] A. Ignatiev, N. Narodytska, and J. Marques-Silva. Abduction-based explanations for machine learning models. In *Proc. of AAAI'19*, pages 1511–1519, 2019.

[Ignatiev *et al.*, 2020] A. Ignatiev, N. Narodytska, N. Asher, and J. Marques-Silva. On relating 'why?' and 'why not?' explanations. *CoRR*, abs/2012.11067, 2020.

[Izza and Marques-Silva, 2021] Y. Izza and J. Marques-Silva. On explaining random forests with SAT. In *Proc. of IJCAI'21*, pages 2584–2591, 2021.

[Karnaugh, 1953] G. Karnaugh. The map method for synthesis of combinational logic circuits. *AIEE Transactions on Communications and Electronics*, 72:593–599, 1953.

[Kenny *et al.*, 2021] E. M. Kenny, C. Ford, M. S. Quinn, and M. T. Keane. Explaining black-box classifiers using *post-hoc* explanations-by-example: The effect of explanations and error-rates in XAI user studies. *Artif. Intell.*, 294:103459, 2021.

[Miller, 2019] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[Molnar, 2019] Ch. Molnar. *Interpretable Machine Learning - A Guide for Making Black Box Models Explainable*. Leanpub, 2019.

[Narodytska *et al.*, 2018] N. Narodytska, S. Prasad Kasiviswanathan, L. Ryzhyk, M. Sagiv, and T. Walsh. Verifying properties of binarized deep neural networks. In *Proc. of AAAI'18*, pages 6615–6624, 2018.

[Nauta *et al.*, 2023] M. Nauta, J. Trienes, S. Pathak, E. Nguyen, M. Peters, Y. Schmitt, J. Schlötterer, M. van Keulen, and Ch. Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. *ACM Comput. Surv.*, 55(13s), 2023.

[Poché *et al.*, 2023] A. Poché, L. Hervier, and M. C. Bakkay. Natural example-based explainability: A survey. In *Proc. of XAI'23*, volume 1902 of *Communications in Computer and Information Science*, pages 24–47. Springer, 2023.

[Roussel and Manquinho, 2021] O. Roussel and V. M. Manquinho. Pseudo-boolean and cardinality constraints. In *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 1087–1129. IOS Press, 2021.

[Shih *et al.*, 2018a] A. Shih, A. Choi, and A. Darwiche. Formal verification of Bayesian network classifiers. In *Proc. of PGM'18*, pages 427–438, 2018.

[Shih *et al.*, 2018b] A. Shih, A. Choi, and A. Darwiche. A symbolic approach to explaining bayesian network classifiers. In *Proc. of IJCAI'18)*, pages 5103–5111, 2018.

[Shih *et al.*, 2019] A. Shih, A. Choi, and A. Darwiche. Compiling Bayesian networks into decision graphs. In *Proc. of AAAI'19*, pages 7966–7974, 2019.

[Vilone and Longo, 2021] G. Vilone and L. Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Inf. Fusion*, 76:89–106, 2021.

[Yang *et al.*, 2019] F. Yang, M. Du, and X. Hu. Evaluating explanation without ground truth in interpretable machine learning. *CoRR*, abs/1907.06831, 2019.

[Zhou *et al.*, 2021] J. Zhou, A.H. Gandomi, F. Chen, and A. Holzinger. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics*, 10:593, 2021.