Deriving Provably Correct Explanations for Decision Trees: The Impact of Domain Theories

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 are interested in identifying the complexity of computing local explanations of various types given a decision tree, when the Boolean conditions used in the tree are not independent. This is usually the case when decision trees are learned from instances described using numerical or categorical attributes. In such a case, considering the domain theory indicating how the Boolean conditions occurring in the tree are logically connected is paramount to derive provably correct explanations. However, the nature of the domain theory may have a strong impact on the complexity of generating explanations. In this paper, we identify the complexity of deriving local explanations (abductive or contrastive) given a decision tree in the general case, and under several natural restrictions about the domain theory.

1 Introduction

eXplainable AI (XAI) is a field that emerged a couple of years ago [Gunning, 2019] in response to the general need for explainability in AI, as well as the opacity of most Machine Learning (ML) models. In this paper, we deal with predictions achieved by binary classifiers, i.e., mappings from a set $\boldsymbol{X}$ of instances to the set $\mathcal{L} = \{0, 1\}$ of classes. More precisely, we focus on binary classifiers *represented by decision trees* [Breiman *et al.*, 1984; Quinlan, 1986]. We consider two types of provably correct local explanations suited to decision trees. On the one hand, *abductive explanations* (see e.g., [Ignatiev *et al.*, 2019]) aim to explain the classification of an instance $\boldsymbol{x} \in \boldsymbol{X}$ as achieved by the decision tree. On the other hand, *contrastive explanations* (see e.g., [Miller, 2019]) aim to explain why an input instance $\boldsymbol{x}$ has *not* been classified by the decision tree as expected by the explainee. Thus, abductive explanations are focused on the "Why?" question, whilst contrastive explanations are about the "Why not?" question [Ignatiev *et al.*, 2020b]. In both cases, explanations can be represented as subsets of the *characteristics* (i.e., the pairs attribute-value) used to represent the instance $\boldsymbol{x}$.

Decision trees are often considered as one of the leading forms of interpretable models, so that more opaque models can be distilled into decision trees to benefit from their improved interpretability [Ras *et al.*, 2022]. However, there is no clear and consensual definition of what "interpretable" means. In many papers, decision trees are said to be "intrinsically interpretable", because the paths in the trees can be directly read as classification rules. However, this characterization is not very satisfying. Indeed, a decision tree with many paths and/or with very long paths can hardly be considered interpretable (and bounding the number of paths or their depth would be arbitrary). Furthermore, the paths may contain many redundant characteristics [Izza *et al.*, 2022; Marques-Silva, 2023].

As a step towards a more rigorous definition of what "interpretable" means, [Audemard *et al.*, 2021] identifies the *computational interpretability* of an ML model as the set of XAI queries that are tractable for the model, i.e., solvable in polynomial time. Under this view, [Audemard *et al.*, 2021] shows that decision trees can be considered as more interpretable than many other ML models. In order to take advantage of such a setting, a set of relevant XAI queries must first be identified, which is a user-dependent issue. The objective is then to help the user decide to trust (or not to trust) the model and its predictions, by leveraging the answers he/she receives to his/her queries of interest. Thus, a central issue as to computational interpretability is to determine which XAI queries, among those of interest for the user, are tractable.

The results reported in the paper are part of this research direction. The tractability of eight explanation queries about decision trees is investigated. To be more precise, our goal is to determine the computational impact of leveraging a domain theory Σ in the task of generating abductive explanations and contrastive explanations for instances given a decision tree f. Such a theory Σ may have several origins: it may come from the encoding of the attributes used at start for learning the decision tree, it can be furnished by the explainee when he/she has knowledge about the precise meaning of the attributes and knows the extent to which they are logically dependent, it may also result from a data mining procedure run on the dataset considered for learning the tree. Whatever the case, Σ makes precise how the Boolean conditions used in f are logically connected. It is mandatory to take advantage of Σ to avoid the derivation of explanations that would be meaningless. We consider the general case when Σ is any theory, and also the more specific case when Σ is tractable.

Computation problem: deriving	Σ valid	any Σ	Σ tractable	Σ Horn	Σ Krom
One subset-minimal abductive explanation	$\surd$	$+$	$\surd$	$\surd$	$\surd$
All the subset-minimal abductive explanations	$\times$	$\times$	$\times$	$\times$	$\times$
One minimum-size abductive explanation	$+$	$+$	$+$	$+$	$+$
All the minimum-size abductive explanations	$\times$	$\times$	$\times$	$\times$	$\times$
One subset-minimal contrastive explanation	$\surd$	$+$	$\surd$	$\surd$	$\surd$
All the subset-minimal contrastive explanations	$\surd$	$\times$	$\times$	$\times$	$\surd$
One minimum-size contrastive explanation	$\surd$	$+$	$+$	$+$	$\surd$
All the minimum-size contrastive explanations	$\surd$	$\times$	$\times$	$\times$	$\surd$

Table 1: The complexity of deriving explanations given a constrained decision-function (f, Σ) when f is a decision tree. $\times$ means that the problem is provably intractable, $+$ means that the problem is intractable unless $\mathsf{P} = \mathsf{NP}$, and $\surd$ means that the problem is tractable.

What we mean here by "tractable theory" Σ is the existence of a polynomial-time algorithm for clausal entailment from Σ: we suppose that a polynomial-time algorithm exists, that takes as input Σ and any clause δ, and returns true if and only if $\Sigma \models \delta$ holds.

As to tractable theories, we focus on two specific families, the Krom one (i.e., CNF formulae consisting of binary clauses) and the Horn one (i.e., CNF formulae where each clause contains at most one positive literal). Krom theories are interesting because domain theories encoding numerical attributes or ordinal attributes are Krom theories. This is also the case of theories encoding categorical attributes under some open world assumption (i.e., when the domain of such an attribute is not supposed to be fully known). Horn theories are also interesting because they can be used for encoding hierarchical attributes.

Our results are synthesized in Table 1. Each line of this table corresponds to a computation problem, that consists in deriving one (or all) explanations of a specific type for an input instance $\boldsymbol{x}$ given a decision tree f and a domain theory Σ. Each column corresponds to an assumption about the underlying theory Σ: Σ valid (i.e., all the attributes used in f are considered as logically independent), any Σ, Σ tractable, Σ Horn, and Σ Krom. Each cell contains one of the following symbols: $\times$, $+$, or $\surd$. $\times$ means that the computation problem given by the line and the column is provably intractable, i.e., there is no polynomial-time algorithm to solve it. $+$ means that the computation problem given by the line and the column is likely to be intractable, i.e., there is no polynomial-time algorithm to solve the problem unless $\mathsf{P} = \mathsf{NP}$. Finally, $\surd$ indicates that the computation problem given by the line and the column is tractable, i.e., there exists a polynomial-time algorithm to solve the problem.

The results reported in Table 1 clearly show that the presence of a domain theory Σ heavily changes the picture as to the computational complexity of deriving abductive or contrastive explanations for an instance given a decision tree f. Especially, computing one subset-minimal abductive explanation (or one subset-minimal contrastive explanation) for an instance given a decision tree becomes NP-hard when Σ is unconstrained, while both problems are solvable in polynomial time when Σ is tractable. However, unlike what happens when no domain theory is considered (i.e., when Σ is valid) the tractability of Σ is not enough to ensure that computing all the subset-minimal contrastive explanations for an instance or computing one minimum-size contrastive explanations for an instance is feasible in polynomial time. Interestingly, imposing further restrictions to Σ may yield to additional tractability results. Thus, the presence of a Krom theory Σ does not lead to a complexity shift for the computation of abductive or contrastive explanations in comparison to the case when no domain theory is considered.

The rest of the paper is organized as follows. After some preliminaries (Section 2), we explain in Section 3 why explanations represented using the Boolean conditions that occur in f have been chosen (instead of explanations based on the characteristics used primarily for representing the instances of the dataset from which f has been learned). We also define in formal terms the types of local explanations one looks for (abductive or contrastive, subset-minimal or minimum-size) and we recall known complexity results (they concern the case when Σ is valid). Section 4 presents the complexity results we have identified. Section 5 concludes the paper. For space reasons, the proofs of the results presented in the paper are available online at www.cril.fr/expekctation/.

2 Preliminaries

Classification Let $\mathcal{A} = \{A_1, \ldots, A_n\}$ be a finite set of attributes, where each attribute is Boolean, categorical, or numerical. The *domain* D_i of A_i ($i \in [n]$) is $\{0, 1\}$ when A_i is Boolean, a finite set of values that are not ordered when A_i is categorical (for instance $D_i = \{red, yellow, green\}$), and (typically) $D_i = \mathbb{N}$ or $\mathbb{R}$ when A_i is numerical. Note that the type of an attribute A_i is a semantical piece of information that must be part of its description (as a meta-data). Especially, it cannot be inferred from the values in the corresponding domain D_i (numbers can be used to denote values, like 0 for red, 1 for $yellow$, and 2 for $green$, but it does not neces-

sarily make sense in this case to consider that $0 < 1 < 2$). Furthermore, in the general case, domains D_i are not provided in extension (this would not be possible for numerical attributes A_i) but inferred from datasets. Accordingly, for categorical attributes A_i, two assumptions can be made about D_i: a closed world assumption (D_i consists precisely of the values of A_i occurring in the dataset) or an open world assumption (the values of A_i occurring in the dataset form a proper subset of D_i). Again, in general, meta-data are required to figure out which assumption is reasonable (if A_i stands for the color of a traffic light and the three values *red*, *yellow*, *green* occurs in the dataset, considering the closed world assumption makes sense; if A_i denotes the color of a shirt, making the closed world assumption is more dubious).

An *instance* $\boldsymbol{x}$ over $\mathcal{A}$ is a vector from $D_1 \times \ldots \times D_n$. Every $\boldsymbol{x} = (v_1, \ldots, v_n)$ is also viewed logically as the conjunctively-interpreted set $t_{\boldsymbol{x}}$ of Boolean conditions (alias the characteristics of $\boldsymbol{x}$) $\{(A_i = v_i) : i \in [n]\}$. $\boldsymbol{X}$ is the set of all instances. A *classifier* f over $\mathcal{A}$ is a mapping from $\boldsymbol{X}$ to a finite set $\mathcal{L}$. A *binary classifier* f over $\mathcal{A}$ is a mapping from $\boldsymbol{X}$ to $\mathcal{L} = \{0, 1\}$. An instance $\boldsymbol{x} \in \boldsymbol{X}$ is *positive* when $f(\boldsymbol{x}) = 1$ and it is *negative* when $f(\boldsymbol{x}) = 0$.

A *decision tree* over $\mathcal{A}$ is a binary tree T, each of whose internal nodes is labeled with a Boolean condition over $A_i \in \mathcal{A}$, and each leaf is labeled by an element of $\mathcal{L}$. The value $T(\boldsymbol{x})$ of T on an input instance $\boldsymbol{x}$ is given by the label of the leaf reached from the root as follows: at each node go to the left (resp. right) child if the Boolean condition labelling the node is evaluated to 0 (resp. 1) for $\boldsymbol{x}$. The size of a decision tree is the number of nodes in it. A *stump* is a decision tree over $\mathcal{A}$ with a single internal node.

Example 1. *Figure 1 depicts a decision tree T over $\mathcal{A} = \{A_1, A_2\}$, where A_1 is a numerical attribute and A_2 is a Boolean attribute. The instance $\boldsymbol{x} = (45, 1)$ is such that $T(\boldsymbol{x}) = 1$.*

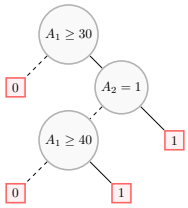

Figure 1: A simple decision tree classifier.

Boolean functions By $\mathcal{F}_n$ we denote the class of all Boolean functions from $\{0, 1\}^n$ to $\{0, 1\}$, and we use $X_n = \{x_1, \cdots, x_n\}$ to denote the set of input Boolean variables. A Boolean vector $\boldsymbol{x} \in \{0, 1\}^n$ represents an interpretation over X_n, i.e., a mapping from X_n to $\{0, 1\}$. $\boldsymbol{x}$ is a *model* of f if $f(\boldsymbol{x}) = 1$. Otherwise, $\boldsymbol{x}$ is a *counter-model* of f. $[f]$ denotes the set of all models of f.

We refer to f as a propositional formula when it is described using the Boolean connectives $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation), together with the constants 1 (true) and 0 (false). f is *satisfiable* if it has a positive instance, and it is *unsatisfiable* otherwise. f is *valid* when it has no negative instance. If f and g are two propositional formulae over X_n, f *entails* g, noted $f \models g$, if and only if $[f] \subseteq [g]$ holds and f and g are *equivalent*, noted $f \equiv g$, if and only if $[f] = [g]$. The class of decision trees over X_n is denoted $\mathtt{DT}_n$. A *literal* ℓ_i is a variable $x_i \in X_n$ (a positive literal) or its negation $\neg x_i$ (a negative literal), also denoted $\overline{x}_i$. The complementary literal $\sim \ell_i$ of literal ℓ_i is $\overline{x}_i$ if $\ell_i = x_i$ is a positive literal, and x_i if $\ell_i = \overline{x}_i$ is a negative literal. L_{X_n} is the set of all literals over X_n. A *term* t is a conjunction of literals, and a *clause* c is a disjunction of literals. In the following, we shall often treat instances as terms, and terms as sets of literals. A term t is an *implicant* of f if and only if $t \models f$ holds and t is a *prime implicant* of f if and only if t is an implicant of f and no proper subset of t is an implicant of f. A clause c is an *implicate* of f if and only if $f \models c$ holds, and c is a *prime implicate* of f if and only if c is an implicate of f and no proper subset of c is an implicate of f. A DNF *formula* is a disjunction of terms and a CNF *formula* is a conjunction of clauses. The set of variables occurring in a formula f is denoted $Var(f)$.

For an assignment $\boldsymbol{z} \in \{0, 1\}^n$, the corresponding canonical term is

$$t_{\boldsymbol{z}} = \bigwedge_{i=1}^{n} x_i^{z_i} \text{ where } x_i^0 = \overline{x}_i \text{ and } x_i^1 = x_i$$

A term t *covers* an assignment $\boldsymbol{x}$ if $t \subseteq t_{\boldsymbol{x}}$.

3 Representing and Computing Explanations

In this section, we present formal definitions for the notions of local explanations we are interested in. Those explanations are based on the Boolean conditions used in the decision tree that has been learned and not on the characteristics corresponding to the set of attributes used to learn the tree. We start by motivating this choice.

Two spaces of characteristics When every Boolean condition occurring in a decision tree T over a set $\mathcal{A}$ of attributes is viewed as a Boolean variable from a set of variables X_n, the decision tree T can be viewed as a Boolean function f over X_n. Accordingly, *two spaces of characteristics* can be used to describe the instances and their explanations when the model used is a decision tree (and more generally, when it is a tree-based classifier, e.g., a random forest [Breiman, 2001], or a boosted tree [Freund and Schapire, 1997; Schapire and Freund, 2014; Friedman, 2001]). Indeed, instances and explanations can be represented as *sets of characteristics based on the initial set of attributes*, but also as *sets of characteristics based on the Boolean conditions used in f*.

Example 2 (Example 1 cont'ed)**.** *T can be viewed as a Boolean function f over a set $X_3 = \{x_1, x_2, x_3\}$ of Boolean attributes where $x_1 = (A_1 \geq 40)$, $x_2 = (A_1 \geq 30)$, and $x_3 = (A_2 = 1)$. The instance $\boldsymbol{x} = (45, 1)$ over $\mathcal{A}$ corresponds to the instance $(1, 1, 1)$ over X_3.*

It turns out that considering the sets of characteristics based on the Boolean conditions used in f is preferable from an XAI perspective since it leads to explanations (abductive or contrastive) that are *more general* than those defined when the set of characteristics based on the initial set of attributes

is considered [Audemard *et al.*, 2023], in the sense that they cover more instances. The point is that generalizability is valuable for explanations since it allows the explainee to anticipate the outcome of the model in situations that may differ from the explained one [Yang *et al.*, 2019].

Example 3 (Example 1 cont'ed)**.** *As a matter of illustration, let us consider a very simple loan granting scenario. Suppose that the decision tree classifier T, depicted on Figure 1, is used to determine whether the requested loan must be granted or not to the applicant. Two attributes are used primarily to describe instances: A_1 (numerical) gives the annual incomes of the applicant, and A_2 (Boolean) indicates whether the applicant has reimbursed a previous loan.*

Alice wants to get a loan. Alice's annual incomes are equal to $\$45\ k$ and she has reimbursed a previous loan. Thus, Alice corresponds to the instance $\boldsymbol{x} = (45, 1)$. Since $T(\boldsymbol{x}) = 1$, Alice will get the loan. The unique subset-minimal abductive explanation for $\boldsymbol{x}$ given T in the space of characteristics considered at start is $\{(A_1 = 45)\}$. Using words, the abductive explanation provided to Alice is "you got the loan since your annual incomes are equal to $\$45\ k$". In the space of characteristics of the predictor, two subset-minimal abductive explanations for $\boldsymbol{x}$ given f can be derived, namely $\{(A_1 \geq 40)\}$ and $\{(A_1 \geq 30), (A_2 = 1)\}$. Those explanations are better than the previous one $\{(A_1 = 45)\}$ since they correspond to more general classification rules and they reflect in a much more accurate way the behaviour of the predictor. Using words, "you got the loan since your annual incomes are greater than or equal to $\$40\ k$, but also because your annual incomes are greater than or equal to $\$30\ k$ and you have reimbursed a previous loan".

Consider now Bob, who also wants to get a loan. Bob has reimbursed a previous loan, but his annual incomes are equal to $\$20\ k$, only. Bob corresponds to the instance $\boldsymbol{x}' = (20, 1)$. Since $T(\boldsymbol{x}') = 0$, Bob will not get the loan. Using the definition provided in [Ignatiev et al.*, 2020a], the unique subset-minimal contrastive explanation for $\boldsymbol{x}'$ given T is $\{A_1\}$. Using words, "in order to get the loan, you have to change your annual incomes". This is correct, but insufficient since Bob surely expects to know to which extent his annual incomes must be updated in order to get the loan. The contrastive explanation $\{\overline{(A_1 \geq 30)}\}$ for $\boldsymbol{x}'$ given f, represented in the space of characteristics of the predictor, is a better explanation. Indeed, it indicates that "in order to get the loan, you have to make your annual incomes at least equal to $\$30\ k$".*

To take advantage of their generality, we focus on explanations represented in the space of characteristics of the decision tree. Accordingly, from now on, any decision tree is considered as a Boolean function f based on the Boolean conditions labelling its decision nodes. By construction, f may be based on Boolean conditions that are *not logically independent*. This is the case when the Boolean conditions in f come from the same (non-Boolean) attribute A_i used to describe instances at start.

Example 4 (Example 1 cont'ed)**.** *In our running example, the Boolean conditions $(A_1 \geq 30)$ and $(A_1 \geq 40)$ are not independent, since no instance may satisfy $(A_1 \geq 40)$ while not satisfying $(A_1 \geq 30)$. As a consequence, some propositional constraints forming a domain theory Σ and indicating how the Boolean conditions used in f are logically connected must be taken into account when computing explanations. Here, $\Sigma = x_1 \Rightarrow x_2 = (A_1 \geq 40) \Rightarrow (A_1 \geq 30)$ (or any formula equivalent to it) would be convenient.*[1]

Because feasible instances reduce to those satisfying Σ, leveraging Σ is mandatory to avoid the derivation of abductive explanations that are unnecessarily specific [Gorji and Rubin, 2022]. It is also necessary to prevent from generating contrastive explanations that would correspond to instances that are impossible [Yu et al.*, 2022], for example, the contrastive explanation for $\boldsymbol{x}'$ (associated with Bob) given by $\{\overline{(A_1 \geq 40)}\}$ that would correspond to the (impossible) contrastive instance given by $\{(A_1 \geq 40), (A_2 = 1), \overline{(A_1 \geq 30)}\}$.*

Abductive explanations and contrastive explanations Since we are interested in deriving abductive explanations and contrastive explanations that take account for a domain theory, we first need to recall the notion of constrained decision-function.

Definition 1 ([Gorji and Rubin, 2022])**.** *Let $X_n = \{x_1, \ldots, x_n\}$ be a set of Boolean variables. A* constrained decision-function *over X_n is a pair (f, Σ) where $f \in F_n$ and Σ is a propositional formula over X_n. Σ indicates how the Boolean variables from X_n are logically connected.*

Abductive explanations and contrastive explanations given a constrained decision-function can be defined as follows [Audemard *et al.*, 2023]:

Definition 2. *Let (f, Σ) be a constrained decision-function over X_n and $\boldsymbol{x} \in [\Sigma]$ be an instance s.t. $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$).*

- *An* abductive explanation *for $\boldsymbol{x}$ given (f, Σ) is a set $t \subseteq t_{\boldsymbol{x}}$ such that $t \wedge \Sigma \models f$ (resp. $t \wedge \Sigma \models \overline{f}$).*
- *A* subset-minimal abductive explanation *for $\boldsymbol{x}$ given (f, Σ) is an abductive explanation t for $\boldsymbol{x}$ given (f, Σ) such that no proper subset of t is an abductive explanation for $\boldsymbol{x}$ given (f, Σ).*
- *A* minimum-size abductive explanation *for $\boldsymbol{x}$ given (f, Σ) is an abductive explanation t for $\boldsymbol{x}$ given (f, Σ) such that no abductive explanation t' for $\boldsymbol{x}$ given (f, Σ) such that $|t'| < |t|$ exists.*

Subset-minimal abductive explanations for $\boldsymbol{x}$ given (f, Σ) are called sufficient reasons in [Gorji and Rubin, 2022]. When Σ is valid, such explanations correspond to PI-explanations [Shih *et al.*, 2018], also known as sufficient reasons [Darwiche and Hirth, 2020] and as abductive explanations [Ignatiev *et al.*, 2020a].

Example 5 (Example 1 cont'ed)**.** *Given the constrained decision-function (f, Σ) where f is represented by the decision tree T given in Figure 1 and $\Sigma = (A_1 \geq 40) \Rightarrow (A_1 \geq 30)$, $\{x_1 = (A_1 \geq 40)\}$ and $\{x_2 = (A_1 \geq 30), x_3 =$*

[1] For the sake of clarity, in the following we write the Boolean conditions in f using the attributes considered at start in $\mathcal{A}$ and not using the corresponding Boolean variables in X_n.

$(A_2 = 1)\}$ *are the two subset-minimal abductive explanations for the instance* $(1, 1, 1)$ *over* X_3 *corresponding to Alice.* $\{x_1 = (A_1 \geq 40)\}$ *is the sole minimum-size abductive explanation for* $(1, 1, 1)$.

Definition 3. *Let* (f, Σ) *be a constrained decision-function over* X_n *and* $\boldsymbol{x} \in [\Sigma]$ *be an instance.*

- *A* contrastive explanation *for* $\boldsymbol{x}$ *given* (f, Σ) *is a set* $c \subseteq t_{\boldsymbol{x}}$ *such that the vector* $\boldsymbol{x}_c \in \{0, 1\}^n$ *that coincides with* $\boldsymbol{x}$ *except on the characteristics of* c *(*$\boldsymbol{x}_c$ *is a so-called contrastive instance) is such that* $\boldsymbol{x}_c \in [\Sigma]$ *and* $f(\boldsymbol{x}_c) \neq f(\boldsymbol{x})$.
- *A* subset-minimal contrastive explanation *for* $\boldsymbol{x}$ *given* (f, Σ) *is a contrastive explanation* c *for* $\boldsymbol{x}$ *given* (f, Σ) *such that no proper subset of* c *is a contrastive explanation for* $\boldsymbol{x}$ *given* (f, Σ).
- *A* minimum-size contrastive explanation *for* $\boldsymbol{x}$ *given* (f, Σ) *is a contrastive explanation* c *for* $\boldsymbol{x}$ *given* (f, Σ) *such that no contrastive explanation* c' *for* $\boldsymbol{x}$ *given* (f, Σ) *such that* $|c'| < |c|$ *exists.*

When Σ is valid, subset-minimal contrastive explanations are also referred to as necessary reasons [Darwiche and Ji, 2022] or contrastive explanations [Ignatiev *et al.*, 2020a].

Example 6 (Example 1 cont'ed)**.** *Given the constrained decision-function* (f, Σ) *presented before,* $\{\overline{x_2} = \overline{(A_1 \geq 30)}\}$ *is the unique subset-minimal contrastive explanation for the instance* $(0, 0, 1)$ *over* X_3 *corresponding to Bob (thus, it is also the unique minimum-size contrastive explanation for* $(0, 0, 1)$*). Indeed, the corresponding contrastive instance* $(0, 1, 1)$ *is feasible (it satisfies* Σ*) and such that* $f((0, 1, 1)) = 1$.

Clearly enough, every instance $\boldsymbol{x}$ has an abductive explanation given (f, Σ) that can be obtained without any computational effort, since $t_{\boldsymbol{x}}$ is such an explanation. Furthermore, provided that $\boldsymbol{x}$ is known, every contrastive explanation c for $\boldsymbol{x}$ given (f, Σ) entirely defines a corresponding contrastive instance $\boldsymbol{x}_c$, and vice-versa, an instance $\boldsymbol{x}_c \in [\Sigma]$ such that $f(\boldsymbol{x}_c) \neq f(\boldsymbol{x})$ entirely defines a contrastive explanation c for $\boldsymbol{x}$ given (f, Σ). Finally, it is obvious that minimum-size abductive (resp. contrastive) explanations form a subset (in general, a proper subset) of the set of subset-minimal abductive (resp. contrastive) explanations.

Beyond allowing to avoid the generation of abductive explanations that are unnecessarily specific and the generation of contrastive explanations that are impossible, the domain theory Σ can also be exploited to simplify explanations:

Definition 4. *Let* (f, Σ) *be a constrained decision-function over* X_n *and* $\boldsymbol{x} \in [\Sigma]$ *be an instance. Let* $e \subseteq t_{\boldsymbol{x}}$ *be an explanation for* $\boldsymbol{x}$ *given* (f, Σ) *(it can be abductive or contrastive).* e *is said to be* simplified *w.r.t.* Σ *if and only if* $\forall \ell \in e, (e \setminus \{\ell\}) \wedge \Sigma \not\equiv e \wedge \Sigma$. *A* simplification *of* e *w.r.t.* Σ *is any subset* $s \subseteq e$ *such that* $s \wedge \Sigma \equiv e \wedge \Sigma$ *and* s *is simplified w.r.t.* Σ.

Example 7 (Example 1 cont'ed)**.** *Given the constrained decision-function* (f, Σ) *presented before, the abductive explanation for* $(1, 1, 1)$ *(associated with Alice) given by* $\{x_1 = (A_1 \geq 40), x_2 = (A_1 \geq 30)\}$ *can be simplified into the subset-minimal explanation* $\{x_1 = (A_1 \geq 40)\}$. $\{x_1 = (A_1 \geq 40)\}$ *is the sole simplification of* $\{x_1 = (A_1 \geq 40), x_2 = (A_1 \geq 30)\}$ *w.r.t.* Σ.

While the problem of deciding whether an explanation for $\boldsymbol{x}$ given (f, Σ) is simplified w.r.t. Σ is NP-complete in the general case, it is easy to show that testing whether an explanation e is simplified w.r.t. Σ is tractable whenever Σ is tractable for clausal entailment.[2] In this case, generating a simplification s of e w.r.t. Σ can be achieved in (deterministic) polynomial time using a greedy algorithm based on successive clausal entailment tests. Computing such a simplification s can be useful when dealing with abductive explanations e that are not subset-minimal, but not when abductive explanations that are subset-minimal are considered. Indeed, it turns out that subset-minimal abductive explanations are always simplified (thus, this is also the case for minimum-size abductive explanations):

Proposition 1. *Let* (f, Σ) *be a constrained decision-function over* X_n *and let* $\boldsymbol{x} \in [\Sigma]$ *be an instance. Let* $t \subseteq t_{\boldsymbol{x}}$*. If* t *is a subset-minimal abductive explanation for* $\boldsymbol{x}$ *given* (f, Σ) *then* t *is simplified w.r.t.* Σ.

A contrario, subset-minimal contrastive explanations are not necessarily simplified. Furthermore, simplifying a contrastive explanation c for $\boldsymbol{x}$ given (f, Σ) (i.e., computing a simplification of c w.r.t. Σ) may result in a set of literals that no longer is a contrastive explanation for $\boldsymbol{x}$ given (f, Σ). In such a case, applying a simplification process would be counter-productive since it would question the status of the explanation one starts with.

Example 8 (Example 1 cont'ed)**.** *Given the constrained decision-function* (f, Σ) *presented before, the subset-minimal contrastive explanation for* $(1, 1, 1)$ *(associated with Alice) given by* $\{x_1 = (A_1 \geq 40), x_2 = (A_1 \geq 30)\}$ *is not simplified w.r.t.* Σ *since* $x_1 \wedge \Sigma \equiv x_1 \wedge x_2 \wedge \Sigma$ *holds. As shown above,* $\{x_1 = (A_1 \geq 40)\}$ *is the unique simplification of* $\{x_1 = (A_1 \geq 40), x_2 = (A_1 \geq 30)\}$ *w.r.t.* Σ*. However,* $c = \{x_1 = (A_1 \geq 40)\}$ *is not a contrastive explanation for* $\boldsymbol{x}$ *given* (f, Σ) *because the corresponding contrastive instance* $(0, 1, 1)$ *that coincides with* $(1, 1, 1)$ *except on* x_1 *satisfies* Σ *but verifies* $f((0, 1, 1)) = f((1, 1, 1))$.

4 The Impact of Domain Theories

In the following, we focus on the issue of deriving abductive explanations and contrastive explanations when f *is a decision tree*. We first recall known results for the case when no domain theory connecting the Boolean conditions that occur in f is available (or, equivalently, Σ is a valid formula). In such a case, it has been shown that:

- As to abductive explanations:
  - An instance $\boldsymbol{x}$ over X_n may have exponentially many abductive explanations given f, and even exponentially many subset-minimal abductive explanations, and exponentially many minimum-size abductive explanations in n [Audemard *et al.*, 2022b; 2022a].

[2] For the sake of completeness, a proposition stating the result in formal terms and its proof is provided as a supplementary material.

  - Computing a subset-minimal abductive explanation for $\boldsymbol{x}$ given f can be done in time polynomial in the size of f and n [Izza *et al.*, 2020], but it is unlikely that we can enumerate subset-minimal abductive explanations for $\boldsymbol{x}$ given f in output polynomial time [de Colnet and Marquis, 2022].
  - Computing a minimum-size abductive explanation for $\boldsymbol{x}$ given f is NP-hard [Barceló *et al.*, 2020].
- As to contrastive explanations:
  - An instance $\boldsymbol{x}$ over X_n may have exponentially many contrastive explanations,[3] but only polynomially-many subset-minimal contrastive explanations in n [Huang *et al.*, 2021; Audemard *et al.*, 2022b].
  - Computing all the subset-minimal contrastive explanations for $\boldsymbol{x}$ given f can be done in time polynomial in the size of f and n [Huang *et al.*, 2021; Audemard *et al.*, 2022b].
  - As a direct consequence, computing all the minimum-size contrastive explanations for $\boldsymbol{x}$ given f can be done in time polynomial in the size of f and n.

Let us now show how the presence of a domain theory connecting the Boolean conditions used in f impacts the complexity of deriving local explanations.

Σ is any theory We first consider the general case when Σ is any propositional formula. In such a case, the presence of Σ can make the derivation of some explanations computationally harder. Obviously, the case when no domain theory is available (i.e., Σ is valid) is a specific case of the general case (when Σ is unconstrained). As a consequence, all hardness results obtained for the case when no domain theory is available still hold in the general case:

- As to abductive explanations:
  - An instance $\boldsymbol{x}$ over X_n may have exponentially many abductive explanations given f and Σ, and even exponentially many minimum-size abductive explanations in n.
  - Computing a minimum-size abductive explanation for $\boldsymbol{x}$ given f and Σ is NP-hard.
- As to contrastive explanations:
  - An instance $\boldsymbol{x}$ over X_n may have exponentially many contrastive explanations.

Let us now look at the remaining issues. We have derived the following proposition:

Proposition 2. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a* CNF *formula, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing a subset-minimal abductive explanation for $\boldsymbol{x}$ given (f, Σ) is* NP*-hard, even if f reduces to a stump.*

[3] Just because f may have exponentially many models and exponentially many counter-models.

Subsequently, subset-minimal abductive explanations cannot be enumerated in output polynomial time unless P = NP.

Similarly, for subset-minimal contrastive explanations, we have that:

Proposition 3. *Let (f, Σ) be a constrained decision-function over X_n , where f is a decision tree, Σ is a* CNF *formula, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing a subset-minimal contrastive explanation for $\boldsymbol{x}$ given (f, Σ) (or just a contrastive explanation for $\boldsymbol{x}$ given (f, Σ)) is* NP*-hard, and this holds even if f reduces to a stump.*

Unless P = NP, the previous result prevents from the polynomial-time generation of all subset-minimal contrastive explanations for $\boldsymbol{x}$ given (f, Σ), which is feasible when Σ is valid. Actually, the result can be proved *unconditionally* due to the number of subset-minimal contrastive explanations for $\boldsymbol{x}$ given (f, Σ). Indeed, it can be the case that the *minimum-size* contrastive explanations for $\boldsymbol{x}$ given (f, Σ) are exponentially numerous in the number of features used, and this holds not only in the general case when Σ is any theory, but also in the specific case when Σ is a tractable theory. Indeed:

Proposition 4. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a* CNF *formula, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. The number of minimum-size contrastive explanations for $\boldsymbol{x}$ given (f, Σ) can be exponential in n, and this is the case even when Σ is a Horn* CNF *formula or a* CNF *formula representing a set of domain constraints for categorical attributes where each domain contains at least 3 elements.*

Σ is a tractable theory Let us now consider the case when Σ is a tractable theory. It turns out that supposing that Σ is tractable for clausal entailment changes the picture concerning the complexity of deriving a subset-minimal abductive explanation:

Proposition 5. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a tractable theory, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing a subset-minimal abductive explanation for $\boldsymbol{x}$ given (f, Σ) can be done in time polynomial in the size of the input.*

Focusing now on the generation of subset-minimal contrastive explanations, it is valuable to consider a further restriction on the tractable theory at hand, namely that Σ is a Krom CNF formula (i.e., Σ is given as a conjunction of binary clauses). Such theories are known as tractable for a while [Even *et al.*, 1976; Aspvall *et al.*, 1979]. Interestingly, when Σ is a Krom CNF formula, the computation of all the subset-minimal contrastive explanations for $\boldsymbol{x}$ given (f, Σ) can be done in time polynomial in the size of the input. As a direct consequence, the computation of all the minimum-size contrastive explanations for $\boldsymbol{x}$ given (f, Σ) can also be achieved in time polynomial in the size of the input. So in the case when Σ is a Krom CNF formula, the results obtained for the case when Σ is valid still hold.

Proposition 6. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a Krom* CNF *formula, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing all the subset-minimal contrastive explanations for $\boldsymbol{x}$ given (f, Σ) can be done in time polynomial in the size of the input.*

Notably, the theories Σ obtained by encoding numerical and/or ordinal attributes are Krom CNF formulae. This is also the case of theories encoding categorical attributes, provided that an open world assumption is made. What we mean here is that if $D_i = \{v_1^i, \ldots, v_{p_i}^i\}$ is the set of values of a categorical attribute A_i where the values v_j^i ($j \in [p_i]$) are those encountered in the dataset used to learn the decision tree f, then Σ is equivalent to the Krom CNF formula $\bigwedge_{v_j^i, v_k^i \in D_i | v_j^i \neq v_k^i} (\overline{(A_i = v_j^i)} \vee \overline{(A_i = v_k^i)})$, i.e., the values in D_i are mutually exclusive. So when considering decision trees f based on numerical and/or ordinal features and/or categorical features under an open world assumption, the generation of all the subset-minimal contrastive explanations for an instance $\boldsymbol{x}$ given (f, Σ) can be achieved in polynomial time. And, as a consequence, the generation of all the minimum-size contrastive explanations for an instance $\boldsymbol{x}$ given (f, Σ) can be achieved in polynomial time as well.

We now focus on domain theories Σ that are tractable but that do not reduce to Krom CNF formulae. Among them are theories encoding categorical attributes, provided that a closed world assumption is made, i.e., one supposes that the domain of the categorical attribute A_i that is considered is limited to its active (aka running) domain, i.e., to the values appearing in the dataset used to learn f. When A_i is such an attribute with domain $D_i = \{v_1^i, \ldots, v_{p_i}^i\}$, the corresponding domain theory Σ can be stated as the CNF formula $\bigwedge_{v_j^i, v_k^i \in D_i | v_j^i \neq v_k^i} (\overline{(A_i = v_j^i)} \vee \overline{(A_i = v_k^i)} \wedge \bigvee_{v_j^i \in D_i} v_j^i$. The conjunct $\bigvee_{v_j^i \in D_i} v_j^i$ has been added to the domain theory considered when D_i is interpreted under the open world assumption in order to specify that no other values that those listed in D_i are possible for A_i. Though Σ is not a Krom CNF formula when D_i contains more than 2 values because of the conjunct that has been added, Σ is tractable for clausal entailment. Indeed, the clauses occurring in Σ are precisely its prime implicates, thus to test whether a clause is a logical consequence of Σ, it is enough to test whether it is a logical consequence of one of the clauses from Σ.

Among the tractable theories of interest are also those consisting of Horn CNF formulae. Such Horn theories can be used for encoding hierarchical features, for instance the fact that every plane geometry object that satisfies the property "rectangle" and the property "diamond" must have the property "square" as well, and vice-versa. Using symbols, $\Sigma = ((A_i = \mathit{rectangle}) \wedge (A_i = \mathit{diamond})) \Leftrightarrow (A_i = \mathit{square})$ could be considered. This formula is equivalent to the Horn CNF formula $(\overline{(A_i = \mathit{rectangle})} \vee \overline{(A_i = \mathit{diamond})} \vee (A_i = \mathit{square})) \wedge (\overline{(A_i = \mathit{square})} \vee (A_i = \mathit{rectangle})) \wedge (\overline{(A_i = \mathit{square})} \vee (A_i = \mathit{diamond}))$, but not to any Krom CNF formula. Because of Proposition 4, we already know that, in general, the derivation of all subset-minimal contrastive explanations for an instance $\boldsymbol{x}$ given (f, Σ) cannot be achieved in polynomial time when Σ is a Horn CNF formula (so the result extends to tractable theories in the general case). Thus, we need to focus on computationally easier problems, namely the generation of one subset-minimal contrastive explanation and the generation of one minimum-size contrastive explanation.

As to the generation of one subset-minimal contrastive explanation, we have obtained the following tractability result:

Proposition 7. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a tractable theory, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing one subset-minimal contrastive explanation for $\boldsymbol{x}$ given (f, Σ) can be done in time polynomial in the size of the input.*

This tractability result does not extend to the case one wants to derive one minimum-size contrastive explanation:

Proposition 8. *Let (f, Σ) be a constrained decision-function over X_n, where f is a decision tree, Σ is a tractable theory, and let $\boldsymbol{x} \in [\Sigma]$ be an instance. Computing one minimum-size contrastive explanation for $\boldsymbol{x}$ given (f, Σ) is* NP*-hard and this holds even if Σ is a pure Horn* CNF *formula and f is a stump.*

5 Conclusion

In this paper, we have shown that leveraging a domain theory indicating how the Boolean conditions occurring in a decision tree are logically connected may have a strong impact on the complexity of generating provably correct explanations. As shown in Table 1, none of the explanation problems that are tractable when the Boolean conditions used in the tree are independent (i.e., when Σ is valid) remain tractable when no assumptions are made about the domain theory. Accordingly, as it is the case in general for families of threshold classifiers [Cooper and Marques-Silva, 2023], the presence of a domain theory may question the computational intelligibility of the ML model under consideration.

In the case of decision trees, ensuring that Σ is tractable is enough to preserve the results about the computation of abductive explanations that hold when Σ is valid. Interestingly, knowledge compilation techniques can be exploited to "render tractable" propositional formulae Σ that are not tractable at start [Darwiche and Marquis, 2002]. Such algorithms can be useful in practice to make tractable domain theories provided by explainees provided that their compiled forms remain small enough (which cannot be guaranteed in the general case). Contrastingly, ensuring only that Σ is tractable changes significantly the picture concerning the computation of contrastive explanations given a decision tree. As a valuable exception, all the explanation problems that are tractable when Σ is valid remain tractable when Σ is a Krom CNF formula. The practical significance of this result comes notably from the fact that the domain theories Σ obtained by encoding numerical and/or categorical attributes under an open world assumption are Krom CNF formulae.

A more intensive use of the domain theory under consideration can be exploited to focus on some specific abductive explanations [Cooper and Amgoud, 2023] (so-called "coverage-based prime-implicant explanations") given a classifier, at the expense of a complexity shift (the identification of such explanations becoming Π_2^p-complete in the general case). A perspective for further research will be to investigate to which extent imposing specific conditions on the domain theory used and focusing on decision tree classifiers is enough to remove at least one source of complexity, and even characterize tractable restrictions of the problem.

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

[Aspvall *et al.*, 1979] B. Aspvall, M. Plass, and R. Tarjan. A linear-time algorithm for testing the truth of certain quantified Boolean formulas. *Information Processing Letters*, 8:121–123, 1979. Erratum: Information Processing Letters 14(4): 195 (1982).

[Audemard *et al.*, 2021] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, and P. Marquis. On the computational intelligibility of boolean classifiers. In *Proc. of KR'21*, pages 74–86, 2021.

[Audemard *et al.*, 2022a] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, and P. Marquis. On preferred abductive explanations for decision trees and random forests. In *Proc. of IJCAI'22*, pages 643–650, 2022.

[Audemard *et al.*, 2022b] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, and P. Marquis. On the explanatory power of boolean decision trees. *Data Knowl. Eng.*, 142:102088, 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*, pages 117–124, 2023.

[Barceló *et al.*, 2020] P. Barceló, M. Monet, J. Pérez, and B. Subercaseaux. Model interpretability through the lens of computational complexity. In *Proc. of NeurIPS'20*, 2020.

[Breiman *et al.*, 1984] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

[Breiman, 2001] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[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.

[Cooper and Marques-Silva, 2023] M.C. Cooper and J. Marques-Silva. Tractability of explaining classifier decisions. *Artif. Intell.*, 316:103841, 2023.

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

[Darwiche and Ji, 2022] A. Darwiche and C. Ji. On the computation of necessary and sufficient explanations. In *Proc. of AAAI'22*, pages 5582–5591, 2022.

[Darwiche and Marquis, 2002] A. Darwiche and P. Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.

[de Colnet and Marquis, 2022] A. de Colnet and P. Marquis. On the complexity of enumerating prime implicants from decision-dnnf circuits. In *Proc. of IJCAI'22*, pages 2583–2590, 2022.

[Even *et al.*, 1976] S. Even, A. Itai, and A. Shamir. On the complexity of timetable and integral multi-commodity flow problems. *SIAM J. on Comp*, 5(4):691–703, 1976.

[Freund and Schapire, 1997] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, 1997.

[Friedman, 2001] J. H. Friedman. Greedy function approximation: A gradient boosted machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[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, and J. Marques-Silva. On efficiently explaining graph-based classifiers. In *Proc. of KR'21*, pages 356–367, 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.*, 2020a] A. Ignatiev, N. Narodytska, N. Asher, and J. Marques-Silva. From contrastive to abductive explanations and back again. In *Proc. of AIxIA'20*, pages 335–355, 2020.

[Ignatiev *et al.*, 2020b] Alexey Ignatiev, Nina Narodytska, Nicholas Asher, and João Marques-Silva. On relating 'why?' and 'why not?' explanations. *CoRR*, abs/2012.11067, 2020.

[Izza *et al.*, 2020] Y. Izza, A. Ignatiev, and J. Marques-Silva. On explaining decision trees. *CoRR*, abs/2010.11034, 2020.

[Izza *et al.*, 2022] Y. Izza, A. Ignatiev, and J. Marques-Silva. On tackling explanation redundancy in decision trees. *J. Artif. Intell. Res.*, 75:261–321, 2022.

[Marques-Silva, 2023] J. Marques-Silva. Disproving XAI myths with formal methods - initial results. *CoRR*, abs/2306.01744, 2023.

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

[Quinlan, 1986] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[Ras *et al.*, 2022] G. Ras, N. Xie, M. van Gerven, and D. Doran. Explainable deep learning: A field guide for the uninitiated. *J. Artif. Intell. Res.*, 73:329–396, 2022.

[Schapire and Freund, 2014] R.E. Schapire and Y. Freund. *Boosting: Foundations and Algorithms*. MIT Press, 2014.

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

[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.

[Yu *et al.*, 2022] J. Yu, A. Ignatiev, P.J. Stuckey, N. Narodytska, and J. Marques-Silva. Eliminating the impossible, whatever remains must be true. *CoRR*, abs/2206.09551, 2022.