Tractability of explaining classifier decisions

Martin Cooper, Joao Marques-Silva

Highlights

Tractability of explaining classifier decisions

Martin C. Cooper, João Marques-Silva

- Research highlight 1: a characterisation of tractable languages for the problem of finding a minimal explanation for a decision taken by a classifier. This highlights the importance of properties such as monotonicity or submodularity, as well as the asymmetry between explaining positive and negative decisions.
- Research highlight 2: a study of tractable languages for the problem of finding a minimum-cardinality explanation which indicate that the only non-trivial tractable class corresponds to modularity (i.e. decomposability into the sum of unary functions). For the problem of finding a set of diverse explanations, there are no non-trivial tractable classes.

Tractability of explaining classifier decisions

Martin C. Cooper[a,*], João Marques-Silva[b]

[a] *IRIT, University of Toulouse III, Toulouse, France*
[b] *IRIT, CNRS, Toulouse, France*

Abstract

Explaining decisions is at the heart of explainable AI. We investigate the computational complexity of providing a formally-correct and minimal explanation of a decision taken by a classifier. In the case of threshold (i.e. score-based) classifiers, we show that a complexity dichotomy follows from the complexity dichotomy for languages of cost functions. In particular, submodular classifiers allow tractable explanation of positive decisions, but not negative decisions (assuming P≠NP). This is an example of the possible asymmetry between the complexity of explaining positive and negative decisions of a particular classifier. Nevertheless, there are large families of classifiers for which explaining both positive and negative decisions is tractable, such as monotone or modular (e.g. linear) classifiers. We extend the characterisation of tractable cases to constrained classifiers (when there are constraints on the possible input vectors) and to the search for contrastive rather than abductive explanations. Indeed, we show that tractable classes coincide for abductive and contrastive explanations in the constrained or unconstrained settings. We show the intractability of returning a set of k diverse explanations even for linear classifiers and $k = 2$. Finding a minimum-cardinality explanation is tractable for the family of modular classifiers, i.e. when the score function is the sum of unary functions, but becomes intractable when any non-modular function is also allowed.

Keywords: machine learning, tractability, explanations, weighted constraint satisfaction
2000 MSC: 68Q25, 90C27

*corresponding author
Email addresses: cooper@irit.fr (Martin C. Cooper), Joao.Marques-Silva@irit.fr (João Marques-Silva)

1. Introduction: Explanations of decisions

Recent work has shown that it is possible to apply formal reasoning to explainable AI, thus providing formal guarantees of correctness of explanations [1, 2, 3, 4, 5, 6, 7, 8, 9][1]. However, scaleability quickly becomes an issue because testing the validity of an explanation may be NP-hard, or even #P-hard. As a result, more recent work focused on investigating classes of classifiers for which explanations can be found in polynomial time [11, 12, 13, 14, 15, 16, 17]. A natural question is thus which other classes of classifiers allow for formal explanations to be computed in polynomial time. This is our motivation for investigating the computational complexity of finding explanations of decisions taken by boolean classifiers. More concretely, the paper proposes conditions on the decision problems associated with classification functions, which enable finding in polynomial time a so-called abductive or contrastive explanation. Furthermore, the paper shows that several large classes of classifiers respect the proposed conditions.

We consider a boolean classification problem with two classes $\mathcal{K} = \{0, 1\}$, defined on a set of features (or attributes) $x_1, \ldots, x_n$, which will be represented by their indices $\mathcal{A} = \{1, \ldots, n\}$. The features can either be real-valued or categorical. For real-valued features, domains are (not necessarily finitely-bounded) intervals of the reals, whereas for categorical features, domains are finite sets. A concrete assignment to the features referenced by $\mathcal{A}$ is represented by an n-dimensional vector $\mathbf{a} = (a_1, \ldots, a_n)$, where a_j denotes the value assigned to feature j, represented by variable x_j, such that a_j is taken from the domain of x_j. The set of all n-dimensional vectors denotes the *feature space* $\mathbb{F}$.

Given a classifier with features $\mathcal{A}$, the corresponding *decision function* is a mapping from the feature space to the set of classes, i.e. $\kappa : \mathbb{F} \to \mathcal{K}$. For example, for a linear classifier, the decision function picks 1 if $\sum_i w_i x_i > t$, and 0 if $\sum_i w_i x_i \leq t$, for some constants w_i $(i = 1, \ldots, n)$ and t. Given $\mathbf{a} \in \mathbb{F}$, we consider the set of feature literals of the form $(x_i = a_i)$, where x_i denotes a variable and a_i a constant.

[1] There exist a wide range of explainable AI approaches offering no formal guarantees of correctness [10].

Definition 1. An *abductive explanation (AXp)* of the decision $\kappa(\mathbf{a}) = c$ is a subset-minimal set $\mathcal{P} \subseteq \mathcal{A}$, denoting feature literals, i.e. feature-value pairs (taken from $\mathbf{a}$), such that

$$\forall(\mathbf{x} \in \mathbb{F}).\ \Big(\Big(\bigwedge\nolimits_{j\in\mathcal{P}}(x_j = a_j)\Big)\ \rightarrow\ (\kappa(\mathbf{x}) = c)\Big) \tag{1}$$

is true.

Abductive explanations [3] are also referred to as sufficient reasons [18] or PI (prime implicant) explanations [1]. We can draw an analogy with prime implicants of propositional formulae: finding subset-minimal (prime) implicants rather than shortest implicants is interesting from a computational point of view since deciding the existence of an implicant of size less than k is Σ_2^P-complete [19].

Example 1. We consider as a running example the case of a bank which uses a function κ to decide whether to grant a loan to a couple represented by a feature vector $\mathbf{x} = (sal_1, sal_2, age_1, age_2)$, where sal_1, sal_2 are the salaries and age_1, age_2 the ages of the two people making up the couple. Suppose that $\kappa(\mathbf{x}) = 1$ if and only if $(\max(sal_1, sal_2) \geq sal_{\min}) \wedge (\min(age_1, age_2) \leq age_{\max})$. If $\mathbf{a}$ corresponds to a couple who both earn more than $sal_{\min}$ and both are younger than $age_{\max}$, then there are four abductive explanations for $\kappa(\mathbf{a}) = 1$: $\{1,3\}$, $\{1,4\}$, $\{2,3\}$ and $\{2,4\}$. For example, $\{1,3\}$ means that the first and third features (sal_1 and age_1) are sufficient to explain the decision. On the other hand, if $\mathbf{b}$ corresponds to a couple who both earn more than $sal_{\min}$ and both are older than $age_{\max}$, then the only abductive explanation for $\kappa(\mathbf{b}) = 0$ is $\{3,4\}$ (i.e. that they are both too old).

If AXp's contain many features, it may be interesting for the user to obtain a smaller set of features $\mathcal{P}$ with only a probabilistic guarantee that extensions return the same class c. Such sets $\mathcal{P}$ are known as relevant sets. Unfortunately finding one relevant set is NP-hard even for decision trees [20] and deciding the existence of a relevant set of size at most k is complete for the complexity class NP^{PP} [21]. In the search for tractable classes, it is therefore natural to restrict our attention to the non-probabilistic notion of explanation of Definition 1. One should note, however, that these negative complexity results do not preclude the practical efficiency of computing approximations to relevant sets for simple classifiers such as decision trees [22] and naive Bayes classifiers [12].

Our aim is to characterise those families of classifiers which allow tractable explaining. For threshold classifiers defined by an objective function which is the sum of functions from a finite language $\mathcal{L}$, we provide a characterisation of those languages $\mathcal{L}$ for which decisions can be explained (i.e. an AXp can be found) in polynomial time. This follows from the complexity dichotomy for languages of cost functions [23]. Examples of tractable languages include monotone and submodular functions. Indeed, over boolean domains, the characteristion implies that explainability is NP-hard for languages that are neither monotone nor submodular. Furthermore, for the important tractable classes of monotone or submodular functions, tractability does not require finiteness of the language $\mathcal{L}$ or decomposability of the objective function into the sum of functions from $\mathcal{L}$: minimising any function f which is either monotone or submodular can be achieved in polynomial time provided f itself can be evaluated in polynomial time [24].

Decisions may have an exponential number of AXp's, so it is natural to look beyond the basic problem of finding one AXp. In the presence of multiple explanations, one reasonable goal is to return a small set of diverse explanations, another is to return a minimum-cardinality explanation. For these two problems, we again provide characterisations of tractable cases for families of threshold classifiers.

Adding the criterion of diversity would appear to exclude the existence of tractable classes. Indeed, we show that finding a set of diverse explanations is NP-hard when asking for as little as two diverse solutions, even in the simplest case of a linear classifier over boolean domains.

Concision of explanations is an obvious criterion given the well known cognitive limitations of human users when processing information [25]. Finding a minimum-cardinality explanation is tractable for threshold classifiers whose objective function belongs to the language $\mathcal{L}_{\text{mod}}$ of modular functions, i.e. which are the sum of unary functions of each feature. We show that finding a minimum-cardinality explanation is NP-hard for any proper extension of $\mathcal{L}_{\text{mod}}$ and indeed for any non-modular language of $\{0,1\}$-valued functions over boolean domains.

We generalise our characterisation results to the case of contrastive explanations (minimal sets of features which if changed lead to a change of class) and also to the case when constraints are given on feature space.

The paper is structured as follows. After defining different families of classifiers in Section 2, we show the close relationship between finding one abductive explanation of a positive (respectively, negative) decision and the

problem TAUTOLOGY (resp., UNSAT) in Section 3 (resp., Section 4). We extend this analysis to cover the presence of constraints between features in Section 5 and contrastive explanations in Section 6. Section 7 provides a language dichotomy concerning the tractability of finding one abductive (or contrastive) explanation. In the presence of many alternative explanations, it can be interesting to either find a small set of diverse explanations or find a shortest explanation. These problems are almost always intractable, as shown in Section 8 and Section 9. In Section 10 we conclude and list some open problems. The main difference compared to the conference version of this paper [26] is Section 9 which is entirely new.

2. Definitions

In order to study the complexity of finding explanations, and in particular to identify tractable cases, we need to place restrictions on the classifier κ. Let $\mathcal{D}$ be a set of domains. For example, $\mathcal{D}$ may include all intervals of the real numbers and all finite subsets of the integers. Let $\mathcal{T}^{\mathcal{D}}$ represent the family of functions $\kappa : \Pi_{i=1}^{n} D_i \to \mathcal{K}$ where each domain D_i belongs to $\mathcal{D}$ (i.e. the feature space $\mathbb{F}$ is the Cartesian product of domains from $\mathcal{D}$). We call n the arity of κ. Recall that $\mathcal{K} = \{0, 1\}$.

Let $\mathcal{F}$ be a family of functions taking values in $\mathbb{R} \cup \{-\infty, \infty\}$. We say that $\kappa : \mathbb{F} \to \mathcal{K}$ is a *$\mathcal{F}$-threshold classifier* if it can be represented by an objective function $f : \mathbb{F} \to \mathbb{R} \cup \{-\infty, \infty\}$ belonging to $\mathcal{F}$ such that an input vector $\mathbf{x} \in \mathbb{F}$ is classified as positive ($\kappa(\mathbf{x}) = 1$) iff $f(\mathbf{x})$ is strictly greater than some threshold t, negative otherwise. Concentrating on threshold classifiers is not really a restriction, since any binary classifier $\kappa : \mathbb{F} \to \{0, 1\}$ can be viewed as a threshold classifier with $f = \kappa$ and threshold $t = 0$. It is the choice of the family of functions $\mathcal{F}$ which determines the complexity of explaining decisions.

If $\mathcal{F}$ is the set of real-valued linear functions, then $\mathcal{F}$-threshold classifiers are known as linear classifiers. Similarly, we can define larger families of threshold classifiers, such as monotone or submodular threshold-classifiers by restricting the objective function f to be monotone or submodular. A function f is *monotone* if $\forall \mathbf{x}, \mathbf{y},\ \mathbf{x} \preceq \mathbf{y}$ implies $f(\mathbf{x}) \leq f(\mathbf{y})$ (where $\preceq$ is componentwise comparison); f is *submodular* if $\forall \mathbf{x}, \mathbf{y},\ f(\min(\mathbf{x}, \mathbf{y})) + f(\max(\mathbf{x}, \mathbf{y})) \leq f(\mathbf{x}) + f(\mathbf{y})$, where min and max are applied componentwise [27]. All linear functions are submodular but only those linear functions whose coefficients are non-negative are monotone. Similarly, f is *antitone* if $\forall \mathbf{x}, \mathbf{y},\ \mathbf{x} \preceq \mathbf{y}$ im-

plies $f(\mathbf{x}) \geq f(\mathbf{y})$; f is *supermodular* if $\forall \mathbf{x}, \mathbf{y},\ f(\min(\mathbf{x}, \mathbf{y})) + f(\max(\mathbf{x}, \mathbf{y})) \geq f(\mathbf{x}) + f(\mathbf{y})$; f is *modular* if $\forall \mathbf{x}, \mathbf{y},\ f(\min(\mathbf{x}, \mathbf{y})) + f(\max(\mathbf{x}, \mathbf{y})) = f(\mathbf{x}) + f(\mathbf{y})$. It is worth pointing out that all these classes of functions (linear, modular, submodular, supermodular, monotone, antitone) are closed under addition. Modular functions are exactly those functions f that can be decomposed into a sum of unary functions $f(\mathbf{x}) = \sum_{i=1}^{n} f_i(x_i)$ [28]. By definition, modular functions are both submodular and supermodular and include linear functions as a special case.

Monotonicity [15] is a desirable property in applications where it is important to guarantee meritocratic fairness (do not favour a less-qualified candidate) [29]. It has been imposed even for classifiers as complex as neural networks [30].

Submodularity is a well-studied concept in Operations Research and Machine learning whose origins can be traced back to the optimal transport problem studied by Gaspard Monge in 1781 [31, 32]. Submodularity is a desirable property in settings where the feature-values have positive but decreasing marginal impact on the function's value [33]. Examples include predicting the demand for a product as a function of various options that might be included. Submodularity can be guaranteed by placing restrictions on common machine-learning models. For example, deep neural networks with weights that are strictly non-negative are submodular [34]. The learnability of submodular functions has been extensively studied in the machine learning community [35, 36, 37, 38].

It is well known that a submodular function over boolean domains can be minimized in polynomial time [24, 39, 40]. For example, if the objective function f is the sum of functions of pairs of variables, then minimizing f is equivalent to finding the minimum cut in a weighted graph [41]. A polynomial-time algorithm for minimizing a submodular function over any finite domains follows from the polynomial reduction to boolean domains obtained by replacing each variable x_i with domain $\{1, \ldots, d\}$ by $d-1$ boolean variables $x_{ir} = 1 \Leftrightarrow x_i \geq r$ $(r = 1, \ldots, d-1)$ [28].

Example 2. Consider again our example of a bank which uses a function κ to decide whether to grant a loan to a couple represented by the feature vector $\mathbf{x} = (sal_1, sal_2, age_1, age_2)$. Suppose that κ is a threshold classifier $\kappa(\mathbf{x}) = 1 \Leftrightarrow f(\mathbf{x}) > t$, where $f = \alpha f_1 + \beta f_2 + \gamma f_3$ and $f_1(\mathbf{x}) = \max(sal_1, sal_2) + \mu \min(sal_1, sal_2)$ (where $0 \leq \mu \leq 1$), and $f_2(\mathbf{x}) = 1$ iff $(\max(age_1, age_2) \geq age_{\min})$ (and $f_2(\mathbf{x}) = 0$ otherwise), and $f_3(\mathbf{x}) = 1$ iff $(\min(age_1, age_2) \leq$

$age_{\max}$) (and $f_3(\mathbf{x}) = 0$ otherwise), where $age_{\min}, age_{\max}$ and $\alpha, \beta, \gamma, \mu \geq 0$ are constants, with $age_{\min}$ being the age of majority and $age_{\max}$ retirement age.

It can be verified that f_1 and f_2 are both submodular and monotone, and that f_3 is both submodular and antitone. Thus (by additivity of submodularity), f is submodular but it is neither monotone nor antitone (assuming $\alpha, \beta, \gamma > 0$). On the other hand, f is monotone if $\gamma = 0$.

We say that κ is a *$\mathcal{F}$-multi-threshold classifier* if it can be represented by functions $f_i \in \mathcal{F}$ $(i = 1, \ldots, r)$ such that an input vector $\mathbf{x} \in \mathbb{F}$ is classified as positive ($\kappa(\mathbf{x}) = 1$) iff $(f_1(\mathbf{x}) > t_1) \wedge \ldots \wedge (f_r(\mathbf{x}) > t_r)$ for some constants t_i $(i = 1, \ldots, r)$. For example, if $\mathcal{F}$ is the set of real-valued linear functions, then for $\mathcal{F}$-multi-threshold classifiers the set of positive examples $\mathbf{x}$ is a polytope.

We are specifically interested in families of classifiers $\mathcal{T} \subseteq \mathcal{T}^{\mathcal{D}}$ which are closed under replacing arguments by constants (sometimes known as restriction, conditioning [42] or projection [43]) since this a necessary condition for the correctness of our polynomial-time algorithm. Fortunately, this is true for most families of functions of interest. For example, a linear/monotone/submodular threshold-classifier remains respectively linear/ monotone/submodular if any of its arguments are replaced by constants. For $\kappa \in \mathcal{T}^{\mathcal{D}}$ of arity n, $S \subseteq \{1, \ldots, n\}$ and $\mathbf{v}$ an assignment to the arguments indexed by S, let $\kappa_{\mathbf{v}} : \Pi_{i \notin S} D_i \to \mathcal{K}$ be the function obtained from κ by fixing the arguments in S to $\mathbf{v}$, i.e. for all $\mathbf{x} \in \Pi_{i \notin S} D_i$, $\kappa_{\mathbf{v}}(\mathbf{x}) = \kappa(\mathbf{v} \cup \mathbf{x})$ (where the partial assignments $\mathbf{v}$ and $\mathbf{x}$ are viewed as sets of literals). We say that $\mathcal{T}$ is *closed under fixing arguments* if for all $\kappa : \Pi_{i=1}^{n} D_i \to \mathcal{K}$ such that $\kappa \in \mathcal{T}$, for all $S \subseteq \{1, \ldots, n\}$ and for all $\mathbf{v} \in \Pi_{i \in S} D_i$, we have $\kappa_{\mathbf{v}} \in \mathcal{T}$.

We can view a boolean classifier κ as a decision problem: given an input vector $\mathbf{x}$, should it be classified 0 or 1? Thus we use the notation $\kappa \in \mathrm{P}$ to indicate that the value of *kappa* can be computed in polynomial time and $\mathcal{T} \subseteq \mathrm{P}$ to indicate that this is true for all $\kappa \in \mathcal{T}$.

In this paper we use the term *NP-hard* to refer to any problem Π for which a polynomial number of calls to an algorithm to solve Π is sufficient to solve any problem in NP.

3. Tractability of finding one abductive explanation

To obtain a polynomial-time algorithm, we require that a particular decision problem be solvable in polynomial time. For a family $\mathcal{T} \subseteq \mathcal{T}^{\mathcal{D}}$ of

boolean-valued functions, let TAUTOLOGY($\mathcal{T}$) be the following decision problem: given a function $\kappa \in \mathcal{T}$, is it true that $\kappa \equiv 1$, i.e. for all $\mathbf{x} \in \mathbb{F}$, $\kappa(\mathbf{x}) = 1$? To avoid exploring dead-end branches, our algorithm requires the answer to this question for functions obtained by fixing a subset of the arguments of a classifier, which is why we require that $\mathcal{T}$ be closed under fixing arguments.

Firstly we consider the more general case in which the only assumption we make is that all functions in $\mathcal{T}$ execute in polynomial time. In this case, TAUTOLOGY($\mathcal{T}$) $\in$ coNP (since a counter-example can be verified in polynomial time). If, furthermore, $\mathcal{T}$ is closed under fixing arguments, then using a greedy algorithm (as in Proposition 3.1 case (3) of [44]) we can deduce that n calls to an NP oracle are sufficient to find an abductive explanation. In the following, we investigate cases for which TAUTOLOGY($\mathcal{T}$) $\in$ P and hence for which finding an abductive explanation is also polynomial-time by a similar greedy algorithm.

We now state conditions which guarantee a polynomial-time algorithm to find one abductive explanation for large classes of classifiers. The algorithm initialises $\mathcal{P}$ to $\mathcal{A}$ and greedily deletes literals from $\mathcal{P}$ as long as this preserves property (1) of being an explanation.

Proposition 1. *If $\mathcal{T}$ is closed under fixing arguments and* TAUTOLOGY*($\mathcal{T}$) $\in$ P, then for any classifier $\kappa \in \mathcal{T}$ and any positively-classified input* $\mathbf{a}$*, an abductive explanation of $\kappa(\mathbf{a}) = 1$ can be found in polynomial time.*

Proof. An explanation is a set $\mathcal{P} \subseteq \{1, \ldots, n\}$ such that equation (1) holds. The algorithm is a simple greedy algorithm that initialises $\mathcal{P}$ to the trivial explanation $\{1, \ldots, n\}$ (corresponding to the complete assignment $\mathbf{a}$) and for each $i \in \mathcal{P}$ tests whether i can be deleted to leave a valid explanation $\mathcal{P} \setminus \{i\}$:

$\mathcal{P} \leftarrow \{1, \ldots, n\}$
for $i = 1, \ldots, n$:
    if $\mathcal{P} \setminus \{i\}$ is a valid explanation then $\mathcal{P} \leftarrow \mathcal{P} \setminus \{i\}$

Clearly, the final value $\tilde{\mathcal{P}}$ of $\mathcal{P}$ is an explanation. Furthermore, it is minimal because if $\mathcal{P} \setminus \{i\}$ was not a valid explanation for some $\mathcal{P} \supseteq \tilde{\mathcal{P}}$, then neither is $\tilde{\mathcal{P}} \setminus \{i\}$.

Let $\mathbf{v}$ be the partial assignment corresponding to the values a_j for $j \in \mathcal{P} \setminus \{i\}$. Testing whether $\mathcal{P} \setminus \{i\}$ is a valid explanation is equivalent to testing whether $\kappa_{\mathbf{v}} \equiv 1$ and hence can be performed in polynomial time since $\mathcal{T}$ is closed under fixing arguments and TAUTOLOGY($\mathcal{T}$) $\in$ P. The algorithm

needs to solve exactly n instances of TAUTOLOGY($\mathcal{T}$). It follows that one abductive explanation can be found in polynomial time. □

Proposition 1 can be seen as a special case of the complexity of finding maximal solutions to problems for which the instance-solution relation is in P (Proposition 3.1 of [44]).

As we will now see, Proposition 1 applies to a large range of classifiers, such as linear, submodular or monotone threshold-classifiers as well as multi-threshold classifiers.

Consider threshold classifiers of the form $\kappa(\mathbf{x}) = 1$ iff $f(\mathbf{x}) > t$, for some real-valued objective function $f \in \mathcal{F}$ and some constant t. Then

$$\kappa \equiv 1 \quad \Leftrightarrow \quad \min_{\mathbf{x} \in \mathbb{F}} f(\mathbf{x}) \ > \ t \tag{2}$$

Thus, if $\mathcal{T}$ is the set of $\mathcal{F}$-threshold classifiers, then TAUTOLOGY($\mathcal{T}$) $\in$ P if functions in $\mathcal{F}$ can be minimised in polynomial time. Examples of classes of functions that can be minimised in polynomial time are the objective functions of extended linear classifiers (referred to as XLCs) [12], monotone functions over real/integer intervals [15] and submodular functions over finite ordered domains [39, 28].

Now consider the case of multi-threshold classifiers of the form $\kappa(\mathbf{x}) = 1$ iff $\bigwedge_{i=1}^{r} f_i(\mathbf{x}) > t_i$, for some real-valued functions $f_i \in \mathcal{F}$ and some constants t_i $(i = 1, \ldots, r)$. Then

$$\kappa \equiv 1 \quad \Leftrightarrow \quad \bigwedge\nolimits_{i=1}^{r} (\min_{\mathbf{x} \in \mathbb{F}} f_i(\mathbf{x}) \ > \ t_i) \tag{3}$$

Thus, if $\mathcal{T}$ is the set of $\mathcal{F}$-multi-threshold classifiers, then again we have that TAUTOLOGY($\mathcal{T}$) $\in$ P if each function in $\mathcal{F}$ can be minimised in polynomial time. For example, f_1 could be monotone, f_2 submodular and the other f_i linear.

We end this section by showing that a polytime tautology test is not only a sufficient but also a necessary condition for tractabilty of finding an abductive explanation. Let AEXPL$^+$($\mathcal{T}$) be the problem of finding an abductive explanation of a positive decision taken by a classifier in $\mathcal{T}$. FP is the class of function problems that can be solved in polynomial time.

Theorem 1. *If $\mathcal{T} \subseteq$ P is closed under fixing arguments, then* AEXPL$^+$*($\mathcal{T}$)* $\in$ FP *iff* TAUTOLOGY*($\mathcal{T}$)* $\in$ P*.*

Proof. The 'if' part of the proof is Proposition 1. For the 'only if' part, suppose that $\mathcal{T}$ is closed under fixing arguments and $\text{AExpl}^+(\mathcal{T}) \in \text{FP}$. Let $\kappa \in \mathcal{T}$. Let $\mathbf{a}$ be an arbitrary choice of feature vector. Then κ is a tautology iff both $\kappa(\mathbf{a}) = 1$ and the empty set is an abductive explanation of $\kappa(\mathbf{a}) = 1$. Note that in the case that the empty set is an abductive explanation, it is necessarily the unique abductive explanation. Thus we can decide $\text{Tautology}(\mathcal{T})$ in polynomial time. □

4. Explanations of negative decisions

In the previous section we exclusively studied the problem of finding an explanation of a *positive* decision $\kappa(\mathbf{a}) = 1$. We show in this section that the complexity of this problem can change drastically if we require an explanation of a negative decision $\kappa(\mathbf{a}) = 0$. For a family $\mathcal{T} \subseteq \mathcal{T}^{\mathcal{D}}$ of boolean functions, let $\text{Unsat}(\mathcal{T})$ be the following decision problem: given a boolean function $\kappa \in \mathcal{T}$, is it true that $\kappa \equiv 0$, i.e. for all $\mathbf{x} \in \mathbb{F}$, $\kappa(\mathbf{x}) = 0$? By an entirely similar proof based on a greedy algorithm, we can deduce the following proposition which mirrors Proposition 1.

Proposition 2. *If $\mathcal{T}$ is closed under fixing arguments and $\text{Unsat}(\mathcal{T}) \in$ P, then for any classifier $\kappa \in \mathcal{T}$ and any negatively-classified input $\mathbf{a}$, an abductive explanation of $\kappa(\mathbf{a}) = 0$ can be found in polynomial time.*

A simple case in which all features are boolean is $\mathcal{T}_{\text{DNF}}$, the family of DNF classifiers. Since deciding the (un)satisfiability of a DNF is trivial, we have $\text{Unsat}(\mathcal{T}_{\text{DNF}}) \in \text{P}$ and so an abductive explanation of a negative decision can be found in polynomial time. On the other hand, by Theorem 1, and the co-NP-completeness of deciding whether a DNF is a tautology, an abductive explanation of a positive decision cannot be found in polynomial time (assuming P≠NP).

It is known that finding an AXp is NP-hard in the case of decision lists (DL) [9]. We can be more specific. Since DL's allow rules with both positive and negative conclusions, they provide a sufficiently rich language in which to express either positive decisions or negative decisions as a DNF. It follows that the two distinct problems of finding an abductive explanation of a positive decision or of a negative decision are both NP-hard for DL's.

We can mention the special case of classifiers (such as decision trees) which are equivalent to two mutually exclusive DNF's ϕ^+, ϕ^- representing respectively the positive cases and the negative cases (one term in ϕ^+ for each

positive leaf and one term in ϕ^- for each negative leaf in the case of decision trees). For such classifiers, it follows from Proposition 1 and Proposition 2 that finding an abductive explanation of any decision, whether positive or negative, is polynomial-time [45]. It is worth pointing out that in the case of decision trees, the resulting explanation is often significantly shorter than the corresponding term in ϕ^+ or ϕ^- [14].

Now consider threshold classifiers of the form $\kappa(\mathbf{x}) = 1$ iff $f(\mathbf{x}) > t$, for some real-valued objective function $f \in \mathcal{F}$ and some constant t. Then

$$\kappa \equiv 0 \quad \Leftrightarrow \quad \max_{\mathbf{x} \in \mathbb{F}} f(\mathbf{x}) \ \leq \ t \tag{4}$$

Thus, if $\mathcal{T}$ is the set of $\mathcal{F}$-threshold classifiers, then $\textsc{Unsat}(\mathcal{T}) \in \mathrm{P}$ if functions in $\mathcal{F}$ can be *maximised* in polynomial time. Examples of functions that can be maximised in polynomial time are linear, monotone, antitone (over real/integer intervals) or supermodular functions (over finite ordered domains). Note that submodular function maximisation cannot be achieved in polynomial time (assuming P≠NP) [46].

Thus, for a given family of classifiers (such as submodular threshold classifiers), the complexity of finding an explanation of a positive decision may be polynomial-time whereas the complexity of finding an explanation of a negative decision may be intractable.

We end this section with a theorem that is the equivalent of Theorem 1 for negative decisions. Let $\textsc{AExpl}^-(\mathcal{T})$ be the problem of finding an abductive explanation of a negative decision taken by a classifier in $\mathcal{T}$.

Theorem 2. *If* $\mathcal{T} \subseteq \mathrm{P}$ *is closed under fixing arguments, then* $\textsc{AExpl}^-(\mathcal{T}) \in \mathrm{FP}$ *iff* $\textsc{Unsat}(\mathcal{T}) \in \mathrm{P}$.

Proof. The 'if' part of the proof is Proposition 2. For the 'only if' part, suppose that $\mathcal{T}$ is closed under fixing arguments and $\textsc{AExpl}^-(\mathcal{T}) \in \mathrm{FP}$. Let $\kappa \in \mathcal{T}$. Let $\mathbf{a}$ be an arbitrary choice of feature vector. Then κ is a unsatisfiable iff both $\kappa(\mathbf{a}) = 0$ and the empty set is an abductive explanation of $\kappa(\mathbf{a}) = 0$. Thus we can decide $\textsc{Unsat}(\mathcal{T})$ in polynomial time. □

5. Explanation of classifiers with constrained features

It may be that some constraints exist between features, so that not all vectors in $\mathbb{F}$ are possible. For example, $gender = male$ and $pregnant = yes$ are incompatible, and clearly we must have $years\_of\_employment \leq age$.

Constraints may also exist due to the semantics of the encoding of features. For example, if a real-valued attribute such as salary or age is encoded as a set S of boolean features corresponding to non-overlapping ranges of values, then there is a constraint $\textsc{AtMostOne}(S)$ on the set of features S. This affects the definition of an abductive explanation. Suppose that there are constraints on the possible feature vectors $\mathbf{x}$ given by a predicate $C(\mathbf{x})$. In the context of constraints C, an abductive explanation [26, 47] of a decision $\kappa(\mathbf{a}) = c$ is now a subset-minimal set $\mathcal{P} \subseteq \mathcal{A}$ of feature literals such that

$$\forall(\mathbf{x} \in \mathbb{F}).\left(C(\mathbf{x}) \wedge \bigwedge\nolimits_{j \in \mathcal{P}}(x_j = a_j)\right) \;\rightarrow\; \kappa(\mathbf{x}) = c \tag{5}$$

Example 3. Consider a medicine that doctors are allowed to prescribe to everybody who has the flu except to pregnant women. An abductive explanation why Alice (who is pregnant) was not prescribed the medicine is that she is pregnant; there is no need to mention that she is a woman given the constraint that there are no pregnant men. There are two abductive explanations why Bob was prescribed the medicine: (1) that he is not pregnant and he had the flu, (2) that he is a man and he had the flu. Note that the rule for prescribing the medicine can be stated without mentioning gender: prescribe to people who have the flu but are not pregnant. The abductive explanations remain the same. In particular, the explanation (2) for Bob being prescribed the medicine mentions gender even though this feature is not mentioned in the rule. If we did not take into account the constraint that men cannot be pregnant, then the explanation (2) would not be valid.

We have the following equivalence which follows from equations (1), (5) and the logical equivalence $(C \wedge A) \rightarrow B \;\equiv\; A \rightarrow (B \vee \neg C)$

Proposition 3. *A set of literals P is an abductive explanation of the decision $\kappa(a) = c$ under constraints C if and only if it is an abductive explanation of the unconstrained $(\kappa(a) = c) \vee \neg C$.*

Consider a threshold classifier with objective function f under constraints C. We can reduce to the unconstrained case by introducing the function g where

$$g(\mathbf{x}) = \begin{cases} 0 & \text{if } C(\mathbf{x}) \\ \infty & \text{if } \neg C(\mathbf{x}) \end{cases} \tag{6}$$

Then an abductive explanation for $f(\mathbf{a}) > t$ under constraints C is an abductive explanation of $f(\mathbf{a}) + g(\mathbf{a}) > t$ (in the unconstrained setting). We

saw in Section 3 that finding an abductive explanation of a positive decision taken by a threshold classifier is polynomial-time if the objective function can be minimised in polynomial time. Thus, for example, if $f + g$ is submodular over finite domains, then an abductive explanation can be found in polynomial time. Assume in the following that f is finite-valued and g is defined as in equation (6). A necessary condition for $f + g$ to be submodular is that g be both min-closed and max-closed [48], where *min-closed* means $\mathcal{C}(\mathbf{x}) \wedge \mathcal{C}(\mathbf{y}) \;\Rightarrow\; \mathcal{C}(\min(\mathbf{x}, \mathbf{y}))$ and *max-closed* means $\mathcal{C}(\mathbf{x}) \wedge \mathcal{C}(\mathbf{y}) \;\Rightarrow\; \mathcal{C}(\max(\mathbf{x}, \mathbf{y}))$ [49]. Over finite domains, the class of monotone objective functions can be extended to a maximal tractable class of constrained minimisation problems by adding min-closed constraints and the class of antitone objective functions can be extended to a maximal tractable class by adding max-closed constraints [28].

As we have already seen, explanations of positive and negative decisions may have very different complexities. Indeed, an abductive explanation for $f(\mathbf{a}) \leq t$ under constraints C is an abductive explanation of $f(\mathbf{a}) - g(\mathbf{a}) \leq t$ (in the unconstrained setting). The sign of g has changed so that the inequality is satisfied whenever g is infinite. As we saw in Section 4, an abductive explanation of a negative decision of a threshold classifier can be found in polynomial time if the objective function can be maximised in polynomial time. Thus, for example, if $f - g$ is a supermodular function (over finite domains), then an abductive explanation can be found in polynomial time. A necessary condition for $f - g$ to be supermodular is that g be both min-closed and max-closed [48]. For the class of monotone functions f, the maximisation of $f - g$ is tractable if the relations C (corresponding to the functions g) are max-closed, and for the class of antitone functions f, the maximisation of $f - g$ is tractable if the relations C are min-closed [28]. This allows us to identify the tractable families of constrained threshold-classifiers listed in Table 1.

Other combinations of classifiers and constraints which imply tractable explaining can also be deduced from Proposition 3, Theorem 1 and Theorem 2. For example, explaining modular classifiers remains tractable when we add constraints of the form $\textsc{AtMostOne}(S_i)$ $(i = 1, \ldots, r)$ where $S_1, \ldots, S_r$ is a laminar (e.g. non-overlapping) family of sets of features [50]. Similarly, explainability remains tractable for modular classifiers in the presence of non-overlapping AllDifferent constraints (since the corresponding constrained optimisation problem satisfies the joint-winner property [51]).

decision	objective function f	constraints $\mathcal{C}$
positive	submodular	max and min-closed
positive	monotone	min-closed
positive	antitone	max-closed
negative	supermodular	max and min-closed
negative	monotone	max-closed
negative	antitone	min-closed

Table 1: Examples of tractable families of constrained threshold-classifiers over finite domains.

6. Contrastive explanations

Abductive explanations are answers to the question 'Why is $\kappa(\mathbf{a}) = c$?' A contrastive explanation [52, 53] is an answer to a different question: 'Why is it not the case that $\kappa(\mathbf{a}) \neq c$'? It gives a set of features which if changed in the feature vector $\mathbf{a}$ can lead to a change of class. Contrastive explanations tend to be smaller than abductive explanations and hence can be easier to interpret by a human user [52].

Definition 2. Given that $\kappa(\mathbf{a}) = c$, a *contrastive explanation (CXp)* is a subset-minimal set $\mathcal{S} \subseteq \mathcal{A}$ such that

$$\exists(\mathbf{x} \in \mathbb{F}).\ \Big(\Big(\bigwedge\nolimits_{j \notin \mathcal{S}}(x_j = a_j)\Big)\ \wedge\ \kappa(\mathbf{x}) \neq c\Big) \tag{7}$$

If $\kappa \equiv c$, then there is no contrastive explanation of $\kappa(\mathbf{a}) = c$.

Example 4. Consider the classifier studied in Example 1: a bank uses a function κ, given by $\kappa(\mathbf{x}) = 1$ if and only if $(\max(sal_1, sal_2) \geq sal_{\min}) \wedge (\min(age_1, age_2) \leq age_{\max})$, to decide whether to grant a loan to a couple represented by a feature vector $\mathbf{x} = (sal_1, sal_2, age_1, age_2)$. If $\mathbf{a}$ corresponds to a couple who both earn more than $sal_{\min}$ and both are younger than $age_{\max}$, then the contrastive explanations of the decision $\kappa(\mathbf{a}) = 1$ are $\{1, 2\}$ and $\{3, 4\}$. If $\mathbf{b}$ corresponds to a couple who both earn more than $sal_{\min}$ but both are older than $age_{\max}$, then the contrastive explanations of the decision $\kappa(\mathbf{b}) = 0$ are $\{3\}$ and $\{4\}$.

Let $\textsc{Invalid}(\mathcal{T})$ be the following decision problem: given a boolean function $\kappa \in \mathcal{T}$, does there exist $\mathbf{x} \in \mathbb{F}$ such that $\kappa(\mathbf{x}) = 0$. Similarly, let $\textsc{Sat}(\mathcal{T})$ be the problem: given a boolean function $\kappa \in \mathcal{T}$, does there exists $\mathbf{x} \in \mathbb{F}$

such that $\kappa(\mathbf{x}) = 1$. The following proposition is the contrastive equivalent of Proposition 1 and Proposition 2.

Proposition 4. *Suppose $\mathcal{T}$ is closed under fixing arguments. If* $\textsc{Invalid}(\mathcal{T}) \in \mathrm{P}$*, then for any classifier $\kappa \in \mathcal{T}$ and any $\mathbf{a}$ such that $\kappa(\mathbf{a}) = 1$, a contrastive explanation of $\kappa(\mathbf{a}) = 1$ can be found in polynomial time. If* $\textsc{Sat}(\mathcal{T}) \in \mathrm{P}$*, then for any classifier $\kappa \in \mathcal{T}$ and any $\mathbf{a}$ such that $\kappa(\mathbf{a}) = 0$, a contrastive explanation of $\kappa(\mathbf{a}) = 0$ can be found in polynomial time.*

Proof. We say that $\mathcal{S}$ can lead to a class change if equation (7) holds. The algorithm is analogous to the algorithm for abductive explanations. It requires n tests of equation (7) to find a contrastive explanation:

$\mathcal{S} \leftarrow \{1, \ldots, n\}$
if $\mathcal{S}$ cannot lead to a class change then report that no CXp exists ;
for $i = 1, \ldots, n$:
    if $\mathcal{S} \setminus \{i\}$ can lead to a class change then $\mathcal{S} \leftarrow \mathcal{S} \setminus \{i\}$

Testing whether $\mathcal{S}$ can lead to a class change from 1 is a test of invalidity (after fixing features in $\mathcal{A} \setminus \mathcal{S}$), whereas testing whether $\mathcal{S}$ can lead to a class change from 0 is a test of satisfiability (after fixing features in $\mathcal{A} \setminus \mathcal{S}$). Thus, the above algorithm finds a contrastive explanation of $\kappa(\mathbf{a}) = c$ in polynomial time if $\textsc{Invalid}(\mathcal{T}) \in \mathrm{P}$ (in the case $c = 1$) or $\textsc{Sat}(\mathcal{T}) \in \mathrm{P}$ (in the case $c = 0$). □

For threshold classifiers of the form $\kappa(\mathbf{x}) = 1$ iff $f(\mathbf{x}) > t$, invalidity corresponds to $\min_{\mathbf{x} \in \mathbb{F}} f(\mathbf{x}) \leq t$ and satisfiability corresponds to $\max_{\mathbf{x} \in \mathbb{F}} f(\mathbf{x}) > t$. Thus, if $\mathcal{T}$ is the set of $\mathcal{F}$-threshold classifiers, then $\textsc{Invalid}(\mathcal{T}) \in \mathrm{P}$ if functions in $\mathcal{F}$ can be minimised in polynomial time and $\textsc{Sat}(\mathcal{T}) \in \mathrm{P}$ if functions in $\mathcal{F}$ can be maximised in polynomial time.

Let $\textsc{CExpl}^+(\mathcal{T})$ (respectively, $\textsc{CExpl}^-(\mathcal{T})$) be the problem of finding a contrastive explanation of a positive (negative) decision taken by a classifier in $\mathcal{T}$ or determining that no contrastive explanation exists. The following theorem follows from Proposition 4 and the fact that deciding the existence of a contrastive explanation of $\kappa(\mathbf{a}) = c$ is equivalent to deciding $\neg(\kappa \equiv c)$.

Theorem 3. *If $\mathcal{T} \subseteq \mathrm{P}$ is closed under fixing arguments, then* $\textsc{CExpl}^+(\mathcal{T}) \in \mathrm{FP}$ *iff* $\textsc{Invalid}(\mathcal{T}) \in \mathrm{P}$*, and* $\textsc{CExpl}^-(\mathcal{T}) \in \mathrm{FP}$ *iff* $\textsc{Sat}(\mathcal{T}) \in \mathrm{P}$*.*

In the context of constraints C, a contrastive explanation of a decision $\kappa(\mathbf{a}) = c$ is now a subset-minimal set $\mathcal{S} \subseteq \mathcal{A}$ of feature literals such that

$$\exists(\mathbf{x} \in \mathbb{F}).\left(\left(\bigwedge\nolimits_{j \notin \mathcal{S}}(x_j = a_j)\right) \wedge \kappa(\mathbf{x}) \neq c \wedge C(\mathbf{x})\right) \tag{8}$$

Using the logical equivalence $\neg B \wedge C \equiv \neg(B \vee \neg C)$, we have the following proposition.

Proposition 5. *A set of literals P is a contrastive explanation of the decision $\kappa(a) = c$ under constraints C if and only if it is a contrastive explanation of the unconstrained classifier $\kappa \vee \neg C$.*

In the case of constrained threshold classifiers, with objective function f and threshold t, let g be as defined by equation (6). Then testing invalidity under constraints C is equivalent to determining whether $\min_{\mathbf{x} \in \mathbb{F}}(f(\mathbf{x}) + g(\mathbf{x})) \leq t$ and testing satisfiability is equivalent to determining whether $\max_{\mathbf{x} \in \mathbb{F}}(f(\mathbf{x}) - g(\mathbf{x})) > t$. It follows that the tractable cases for finding contrastive explanations or abductive explanations are identical. Examples are shown in Table 1, where, in both cases, the decision corresponds to the original decision (i.e. the value of $\kappa(\mathbf{a})$).

In fact, from Theorem 1, Theorem 2, Theorem 3, Proposition 3 and Proposition 5, we can deduce the following theorem which says that tractable classes of finding abductive or contrastive explanations coincide. It follows from the fact that $\textsc{Invalid}(\mathcal{T}) \in \mathrm{P}$ iff $\textsc{Tautology}(\mathcal{T}) \in \mathrm{P}$ and that $\textsc{Sat}(\mathcal{T}) \in \mathrm{P}$ iff $\textsc{Unsat}(\mathcal{T}) \in \mathrm{P}$ (since a problem is in P iff its complement is in P).

Theorem 4. *In the unconstrained or constrained setting, if $\mathcal{T} \subseteq \mathrm{P}$ is closed under fixing arguments, $\textsc{AExpl}^{+}(\mathcal{T}) \in \mathrm{FP}$ iff $\textsc{CExpl}^{+}(\mathcal{T}) \in \mathrm{FP}$, and $\textsc{AExpl}^{-}(\mathcal{T}) \in \mathrm{FP}$ iff $\textsc{CExpl}^{-}(\mathcal{T}) \in \mathrm{FP}$.*

7. A language dichotomy for threshold classifiers

In this section we consider threshold classifiers over finite (i.e. categorical) domains whose objective function can be decomposed into functions of bounded arity. If σ is a list of indices from $\{1, \ldots, n\}$ (i.e. features) and $x \in \mathbb{F}$ is a feature vector, then we use the notation $\mathbf{x}[\sigma_i]$ to denote the projection of $\mathbf{x}$ on these indices. We assume that the objective function f is the sum of functions f_i each with a corresponding scope σ_i (the list of featues on which it is applied):

$$f(\mathbf{x}) \;=\; \sum\nolimits_{i=1}^{m} f_i(\mathbf{x}[\sigma_i]) \tag{9}$$

Given a set (language) $\mathcal{L}$ of functions, we denote by $\mathcal{T}_\mathcal{L}$ the set of threshold classifiers whose objective function f is the sum of functions $f_i \in \mathcal{L}$. Recall that AExpl$^+(\mathcal{T}_\mathcal{L})$ is the problem of finding an abductive explanation of a positive decision taken by a classifier in $\mathcal{T}_\mathcal{L}$.

Cost Function Networks (CFNs) (also known as Valued Constraint Satisfaction Problems) are defined by sets of functions f_i (and their associated scopes) over finite domains whose sum f (given by equation (9)) is an objective function to be minimized [54]. CFNs are a generic framework covering many well-studied optimisation problems. For example, Bayesian networks can be transformed into CFNs after taking logarithms of probabilities [54]. Let CFN($\mathcal{L}$) denote the problem of determining, given an objective function f of the form given in equation (9) where each $f_i \in \mathcal{L}$, together with a real constant t, whether

$$\min f(\mathbf{x}) \ \leq \ t.$$

A technical point is that, due to the necessarily bounded precision of the values of functions, this is equivalent to the problem of determining, given f and $t \in \mathbb{R}$, whether $\min f(\mathbf{x})$ is strictly less than t.

The complexity of CFN($\mathcal{L}$) has been extensively studied for finite languages (i.e. languages $\mathcal{L}$ such that $|\mathcal{L}|$ is finite). It is now known that there is a dichotomy: depending on the language $\mathcal{L}$, CFN($\mathcal{L}$) is either in P or is NP-complete. This result was known for languages of finite-valued cost-functions [55] and the dichotomy for the more general case, in which costs can be infinite, follows from the recently-discovered language dichotomy for constraint satisfaction problems [56, 57, 23, 58]. The following proposition will lead us to a similar dichotomy for explaining decisions.

Proposition 6. *Let $\mathcal{L}$ be a set of non-negative functions closed under fixing arguments. Then* AExpl$^+(\mathcal{T}_\mathcal{L}) \in$ FP *if and only if CFN($\mathcal{L}$)* $\in$ P.

Proof. If $\mathcal{L}$ is closed under fixing arguments, then so is $\mathcal{T}_\mathcal{L}$. The 'if' part of the proof follows directly from Proposition 1 and the subsequent discussion in Section 3, so we concentrate on the 'only if' part.

By Theorem 1 we know that if AExpl$^+(\mathcal{T}_\mathcal{L}) \in$ FP then Tautology$(\mathcal{T}_\mathcal{L})$ $\in$ P. Tautology$(\mathcal{T}_\mathcal{L})$ is the problem of determining, for a function f expressible as the sum of functions $f_i \in \mathcal{L}$ (as in equation (9)) and a constant t, whether $f(x) > t$ for all $x \in \mathbb{F}$. This is the complement of CFN($\mathcal{L}$) which is the problem of determining whether $\min_{x\in\mathbb{F}} f(x) \leq t$. Hence, if Tautology$(\mathcal{T}_\mathcal{L}) \in$ P then CFN($\mathcal{L}$) $\in$ P, which completes the proof. □

We now consider constrained classifiers. Let Γ be a language of constraint relations. For each constraint relation in Γ we can construct a corresponding $\{0, \infty\}$-valued function g, as given by equation (6). Let $\mathcal{C}_\Gamma$ denote the set of all such $\{0, \infty\}$-valued functions for relations in Γ. Then $\mathcal{L} \cup \mathcal{C}_\Gamma$ can be viewed as a language of cost functions. Let $\textsc{ConAExpl}^+(\mathcal{T}_\mathcal{L}, \Gamma)$ (respectively, $\textsc{ConAExpl}^-(\mathcal{T}_\mathcal{L}, \Gamma)$) denote the problem of finding one abductive explanation of a positive (negative) decision taken by a classifier in $\mathcal{T}_\mathcal{L}$ under a finite set of constraints from Γ.

Proposition 7. *Let $\mathcal{L}$ be a set of non-negative functions closed under fixing arguments and Γ a finite set of constraint relations. Then $\textsc{ConAExpl}^+(\mathcal{T}_\mathcal{L}, \Gamma) \in \mathrm{FP}$ if and only if CFN($\mathcal{L} \cup \mathcal{C}_\Gamma$) $\in \mathrm{P}$.*

Proof. We know from the discussion in Section 5 that $\textsc{ConAExpl}^+(\mathcal{T}_\mathcal{L}, \Gamma)$ is equivalent to $\textsc{AExpl}^+(\mathcal{T}_{\mathcal{L} \cup \mathcal{C}_\Gamma})$. Thus the result follows immediately from Proposition 6. □

We now consider finding explanations for negative decisions. Although, as we will show, there is again a dichotomy, it is not the same since in this case we are studying a (constrained) maximisation problem rather than a (constrained) minimisation problem. Given a finite language $\mathcal{L}$ of real-valued functions, all bounded above by $B \in \mathbb{R}$, let $\mathcal{L}_{\text{inv}}$ denote the set $\{B - f : f \in \mathcal{L}\}$. Clearly, maximising a sum of functions from $\mathcal{L}$ is equivalent to minimising a sum of functions from $\mathcal{L}_{\text{inv}}$.

Proposition 8. *Let $\mathcal{L}$ be a set of non-negative finite-valued functions closed under fixing arguments. Then $\textsc{AExpl}^-(\mathcal{T}_\mathcal{L}) \in \mathrm{FP}$ if and only if CFN($\mathcal{L}_{inv}$) $\in \mathrm{P}$.*

Proof. The 'if' part follows from Proposition 2 and the subsequent discussion in Section 4. For the 'only if' part, we know from Theorem 2 that if $\textsc{AExpl}^-(\mathcal{T}_\mathcal{L})$ is in FP then $\textsc{Unsat}(\mathcal{T}_\mathcal{L})$ is in P. $\textsc{Unsat}(\mathcal{T}_\mathcal{L})$ is the problem of determining, for a function f expressible as the sum of m functions $f_i \in \mathcal{L}$ and a constant t, whether $f(x) \leq t$ for all $x \in \mathbb{F}$. This is equivalent to determining whether $mB - f(x) \geq mB - t$ for all $x \in \mathbb{F}$. This is the complement of the problem of determining whether $\min(mB - f) < t'$ (for $t' = mB - t$). This is precisely $\mathrm{CFN}(\mathcal{L}_{\text{inv}})$. Hence, if $\textsc{Unsat}(\mathcal{T}_\mathcal{L}) \in \mathrm{P}$, then $\mathrm{CFN}(\mathcal{L}_{\text{inv}}) \in \mathrm{P}$, which completes the proof. □

We now generalise this result to constrained classifiers.

Proposition 9. *Let $\mathcal{L}$ be a set of non-negative functions closed under fixing arguments and Γ a finite set of constraint relations. Then* ConAExpl$^-(\mathcal{T}_\mathcal{L}$, $\Gamma) \in$ FP *if and only if CFN($\mathcal{L}_{inv} \cup \mathcal{C}_\Gamma$) $\in$ P.*

Proof. ConAExpl$^-(\mathcal{T}_\mathcal{L}, \Gamma)$ is equivalent to ConAExpl$^+(\mathcal{T}_{\mathcal{L}_{\text{inv}}}, \Gamma)$. Thus the result follows immediately from Proposition 7. □

Given the known P/NP-complete dichotomy for CFN($\mathcal{L}$) for finite languages $\mathcal{L}$, discussed above, we can immediately deduce the following theorem.

Theorem 5. *Let $\mathcal{L}$ be a finite language of non-negative functions closed under fixing arguments and Γ a finite set of constraint relations with $\mathcal{T}_\mathcal{L}, \Gamma \subseteq$ P. Then each of the problems* AExpl$^+(\mathcal{T}_\mathcal{L})$, ConAExpl$^+(\mathcal{T}_\mathcal{L}, \Gamma)$, AExpl$^-(\mathcal{T}_\mathcal{L})$ *and* ConAExpl$^-(\mathcal{T}_\mathcal{L}, \Gamma)$ *is either in* FP *or is* NP*-hard.*

Indeed, by Theorem 4, we have an identical dichotomy result for contrastive explanations.

Corollary 1. *Let $\mathcal{L}$ be a finite language of non-negative functions closed under fixing arguments and Γ a finite set of constraint relations with $\mathcal{T}_\mathcal{L}, \Gamma \subseteq$ P. Then each of the problems* CExpl$^+(\mathcal{T}_\mathcal{L})$, ConCExpl$^+(\mathcal{T}_\mathcal{L}, \Gamma)$, CExpl$^-(\mathcal{T}_\mathcal{L})$ *and* ConCExpl$^-(\mathcal{T}_\mathcal{L}, \Gamma)$ *is either in* FP *or is* NP*-hard.*

In the special but important case of Boolean domains, the characterisation of tractable cost-function languages [28] tells us that the only tractable cases are those identified in Table 1. In the unconstrained case over non-boolean domains, the only tractable languages are those that admit a binary symmetric fractional polymorphism [55]. Binary symmetric fractional polymorphisms can be viewed as componentwise closure operations which include as special cases both monotonicity and submodularity. An interesting point is that there is a common algorithm, which can be seen as a linear programming relaxation of an integer program, that solves the corresponding minimisation problem for all such languages [55].

8. Diversity of explanations

We have concentrated up until now on the problem of finding a single explanation. This is because the problem of finding all explanations has the obvious disadvantage that the number of explanations may be exponential. For example, in a first-past-the-post election in which a A wins with $m \geq k$

out of the $n = 2k - 1$ votes cast, and each vote is considered as a feature, there are C_m^k abductive explanations for this victory; for a candidate B who lost with only $p \leq k$ votes, there are C_{n-p}^{k-p} contrastive explanations for why they did not win.

Rather than providing a single explanation to the user or listing all explanations, we can envisage providing a relatively small number of *diverse* explanations. A similar strategy of finding a number of diverse good-quality solutions to a Weighted Constraint Satisfaction Problem has been used successfully in computational protein design [59], among other examples [60, 61, 62].

An obvious measure of diversity of a set of explanations $\{S_1, \ldots, S_k\}$ is the minimum Hamming distance $|S_i \Delta S_j|$ between pairs of distinct explanations S_i, S_j, where Δ is the symmetric difference operator between two sets. This leads to the following computational problem.

k-DIV-AEXPL$^+$: Given a binary classifier $\kappa : \mathbb{F} \to \{0, 1\}$, a positively-classified input $\mathbf{a}$ and an integer m, find k abductive explanations $S_1, \ldots, S_k$ of $\kappa(\mathbf{a}) = 1$ such that for all i, j such that $1 \leq i < j \leq k$, $|S_i \Delta S_j| \geq m$.

The definitions for negatively-classified inputs $\mathbf{a}$ (k-DIV-AEXPL$^-$) and/or for contrastive explanations (k-DIV-CEXPL$^+$, k-DIV-CEXPL$^-$) are entirely similar. Since Hamming distance is a submodular function, one might hope that there would be interesting tractable classes. Unfortunately, since we are, in a sense, maximising this distance rather than minimising it, these four problems turn out to be NP-hard even in the simplest non-trivial case.

Proposition 10. *Even in the case of $k = 2$ and for a linear classifier κ over domains of size 2, the following four problems are* NP*-hard: (a) k-*DIV-AEXPL$^+$*, (b) k-*DIV-AEXPL$^-$*, (c) k-*DIV-CEXPL$^+$*, (d) k-*DIV-CEXPL$^-$*.*

Proof. (a) Without loss of generality, we suppose that the domains D_i ($i = 1, \ldots, n$) are all $\{0, 1\}$ and $\kappa(\mathbf{x}) = 1$ iff $\sum_{i=1}^n \alpha_i x_i > t$. We prove NP-hardness for the particular case in which $\mathbf{a} = (1, \ldots, 1)$ and the values $t, \alpha_1, \ldots, \alpha_n$ are strictly positive integers which satisfy the following inequalities:

$$\alpha_1 \leq \cdots \leq \alpha_m < \alpha_{m+1} \leq \cdots \leq \alpha_n \tag{10}$$

$$\sum_{i=1}^{m} \alpha_i \; + \; 2 \sum_{i=m+1}^{n} \alpha_i \; = \; 2(t+1) \tag{11}$$

To solve 2-DIV-AEXPL$^+$ we require sets $S_1, S_2 \subseteq \{1, \ldots, n\}$ satisfying (1) $|S_1 \Delta S_2| \geq m$ and (2) S_1, S_2 are minimal (for inclusion) sets such that the minimum value of $\sum_{i=1}^{n} \alpha_i x_i$ is at least $t+1$ for inputs $\mathbf{x}$ with $x_i = a_i = 1$ for all $i \in S_j$ $(j = 1, 2)$. Since the values α_i are positive, the minimum is attained when $x_i = 0$ for all $i \notin S_j$, and so this is equivalent to

$$\sum_{i \in S_j} \alpha_i \ \geq \ t+1 \qquad (j = 1, 2) \tag{12}$$

Summing these two inequalities (for $j = 1, 2$) gives

$$\sum_{i \in S_1} \alpha_i \ + \ \sum_{i \in S_2} \alpha_i \ \geq \ 2(t+1) \tag{13}$$

Since, by (10), we have $\alpha_r < \alpha_s$ for $r \leq m < s$, and $|S_1 \Delta S_2| \geq m$, we know that the left hand side of equation (13) is at most equal to the left hand side of equation (11), which is equal to $2(t+1)$. It follows that we actually have equality in inequality (13) and $S_1 \Delta S_2 = \{1, \ldots, m\}$ and $S_1 \cap S_2 = \{m+1, \ldots, n\}$. Equality in (13) implies that we must also have equality in the inequalities (12) for $j = 1, 2$. Equality implies minimality for subset inclusion since all weights α_i are strictly positive. Denoting $t+1-\sum_{i=m+1}^{n} \alpha_i$ by T and $S_j \cap \{1, \ldots, m\}$ by P_j (for $j = 1, 2$), we can deduce that we require a partition P_1, P_2 of $\{1, \ldots, m\}$ such that

$$\sum_{i \in P_1} \alpha_i \ = \ T \ = \sum_{i \in P_2} \alpha_i$$

This is precisely the partition problem which is well known to be NP-complete [63]. It follows that k-DIV-AEXPL$^+$ is NP-hard.

(b) We consider the same linear classifier κ as in case (a), except that equation (11) is replaced by $\sum_{i=1}^{m} \alpha_i = 2t$, and this time we consider the vector $\mathbf{a} = (0, \ldots, 0)$ which is classified negatively by κ. To solve k-DIV-AEXPL$^-$, we require two sets S_1, S_2 such that (1) $|S_1 \Delta S_2| \geq m$ and (2) S_1, S_2 are minimal (for inclusion) sets such that $\sum_{i \notin S_j} \alpha_i \leq t$ $(j = 1, 2)$. Given equation (10), this can only be attained when $S_1 \Delta S_2 = \{1, \ldots, m\}$ and $S_1 \cap S_2 = \{m+1, \ldots, n\}$, so that $\sum_{i \notin S_1} \alpha_i = \sum_{i \notin S_2} \alpha_2 = t$. Thus, we need to find two sets $P_j = \{1, \ldots, n\} \setminus S_j$ $(j = 1, 2)$ which partition $\{1, \ldots, m\}$ and such that

$$\sum_{i \in P_1} \alpha_i \ = \ t \ = \sum_{i \in P_2} \alpha_i$$

Thus, again we have a polynomial reduction from the partition problem. Hence k-DIV-AEXPL$^-$ is NP-hard.

(c) Consider the same linear classifier κ as in case (b), but this time $\mathbf{a} = (1, \ldots, 1)$. To solve k-DIV-CEXPL$^+$, we require two sets $S_1, S_2 \subseteq \{1, \ldots, n\}$ such that $\sum_{i \notin S_j} \alpha_i \leq t$ $(j = 1, 2)$ and $|S_1 \Delta S_2| \geq m$. Since this is exactly the same problem encountered in case (b), we can again deduce NP-hardness.

(d) Consider the same linear classifier κ as in case (a), but with $\mathbf{a} = (0, \ldots, 0)$. To solve k-DIV-CEXPL$^-$, we require $S_1, S_2 \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in S_j} \alpha_i \geq t + 1$ $(j = 1, 2)$ and $|S_1 \Delta S_2| \geq m$. Since this is exactly the problem encountered in case (a), we can again deduce NP-hardness.

□

It is worth pointing out that the NP-hardness of finding a diverse set of k explanations is robust to changes in the definition of diversity of a set of k explanations, since Proposition 10 applies even in the case $k = 2$. For example, defining a set of k explanations to be diverse if the *average* Hamming distance is bounded below by a constant is equivalent to our definition when $k = 2$. Indeed, this is true for all measures of diversity which coincide with a minimum Hamming distance in the case of a set of two explanations.

It is well known that the partition problem is one of the easiest NP-hard problems to solve in practice [64]. Thus, Proposition 10 precludes (assuming P≠NP) a worst-case polynomial-time algorithm for finding a diverse set of explanations, but leaves the door open to the existence of practically-efficient algorithms.

9. Minimum-cardinality explanations

In this section we show that finding a minimum-cardinality explanation is in P for modular classifiers (i.e. classifiers whose objective function is the sum of unary functions of each feature). We then show that finding a minimum-cardinality explanation is NP-hard for threshold classifiers whose objective function is the sum of simple non-modular functions. Indeed, over boolean domains, the problem is NP-hard as soon as any $\{0, 1\}$-valued non-modular binary function is allowed. Over arbitrary domains, the problem is NP-hard for any proper extension of modular classifiers.

Many decisions have more than one explanation. Many researchers have identified parsimony as an important criterion for choosing between explanations [65, 66]. From the point of view of a human user, smaller explanations tend to be easier to understand and more meaningful. As an example, consider a classifier κ on n boolean features x_i $(i = 1, \ldots, n)$. Suppose that $\kappa(\mathbf{x}) \equiv x_1 \vee (\overline{x_2} \wedge \ldots \wedge \overline{x_n})$ and we want to explain the decision $\kappa(1, 0, \ldots, 0) = 1$. Clearly $\{1\}$ is an AXp (abductive explanation), but so is $\{2, \ldots, n\}$. We will therefore now study the complexity of finding the smallest explanation. We know that, on the one hand, deciding whether there exists an AXp of size less than k is Σ_2^{P}-complete for multilayer perceptron classifiers [66], but is quasi-linear time for threshold classifiers with modular objective functions [12]. An interesting point is that the symmetry we have observed between AXp's and CXp's no longer holds when looking for smallest explanations: deciding whether there exists a CXp of size less than k lies at the first (rather than the second) level of the polynomial hierarchy, as we now show.

Proposition 11. *The problem of deciding whether there exists a CXp of size less than k for a decision $\kappa(\mathbf{v}) = c$, where $\kappa \in \mathrm{P}$, belongs to* NP*.*

Proof. Let $\mathbf{v}$ be the feature vector whose decision $\kappa(\mathbf{v})$ is to be explained. A certificate consists of a set of features $\mathcal{S}$ and a feature vector $\mathbf{x} \in \mathbb{F}$. We can clearly verify in polynomial time that (1) $|\mathcal{S}| < k$, (2) $\mathbf{x}$ and $\mathbf{v}$ only differ on features in $\mathcal{S}$ and (3) $\kappa(\mathbf{x}) \neq \kappa(\mathbf{v})$. □

As in Section 7, we consider languages $\mathcal{L}$ of real-valued functions and denote by $\mathcal{T}_{\mathcal{L}}$ the set of threshold classifiers κ defined by $\kappa(\mathbf{x}) \equiv f(\mathbf{x}) > t$ for a threshold t, where f is the sum of functions from $\mathcal{L}$ applied to components of the feature vector $\mathbf{x}$.

We denote by MinCardAXp$^+(\mathcal{T}_{\mathcal{L}})$ (respectively, MinCardAXp$^-(\mathcal{T}_{\mathcal{L}})$) the problem of finding a minimum-cardinality AXp of a decision $\kappa(\mathbf{v}) = 1$ (respectively, $\kappa(\mathbf{v}) = 0$), given a classifier $\kappa \in \mathcal{T}_{\mathcal{L}}$ and a positively (respectively, negatively) classified feature vector $\mathbf{v}$. We use similar notation for contrastive explanations (CXp's). Let $\mathcal{L}_{\mathrm{mod}}$ be the language of modular functions. We first state a positive result concerning modular objective functions which is a minor generalisation of known results [12, 66].

Theorem 6. *Suppose that the objective function f of a threshold classifier $\kappa \in \mathrm{P}$ is modular and that domain-sizes are bounded by a constant. Then*

minimum-cardinality AXp's and CXp's can be found for either positive or negative decisions in log-linear time.

Proof. Consider first a positive decision to be explained, corresponding to $f(\mathbf{v}) > t$, where f is modular. It is well known that all modular functions are expressible as the sum of unary functions [28], so

$$f(\mathbf{x}) \;=\; \sum_{i=1}^{n} f_i(x_i)$$

for some unary functions f_i. For $i = 1, \ldots, n$, let $u_i \in D_i$ be such that $f_i(u_i)$ is minimum, and let $\delta_i = f_i(v_i) - f_i(u_i)$. In $O(n \log n)$ time we can order the values δ_i in decreasing order. Without loss of generality and for notational convenience, assume that $\delta_1 \geq \ldots \geq \delta_n$. Let k be minimal such that

$$f(\mathbf{v}) \;-\; \sum_{i=k+1}^{n} \delta_i \;>\; t$$

By our choice of k, $\{1, \ldots, k\}$ is an AXp, and since $\delta_i \geq \delta_j$ $(i \leq k < j)$ it is of minimum-cardinality.

To find a minimum-cardinality CXp of $f(\mathbf{v}) > t$, it is sufficient to choose the minimal j such that

$$f(\mathbf{v}) \;-\; \sum_{i=1}^{j} \delta_i \;\leq\; t$$

Our choice of j ensures that $\{1, \ldots, j\}$ is a CXp and $\delta_1 \geq \ldots \geq \delta_n$ ensures that it is of minimum cardinality.

Since $-f$ is modular if and only if f is modular, and a negative decision $f(\mathbf{v}) \leq t$ can be interpreted as a positive decision $-f(\mathbf{v}) \geq -t$, we can easily adapt the above proof to explanations of negative decisions. □

Thus it is tractable to find smallest explanations for classifiers in $\mathcal{T}_{\mathcal{L}}$ when $\mathcal{L} = \mathcal{L}_{\text{mod}}$. Unfortunately, as we will show in this section, there is little chance of finding another language $\mathcal{L}$ for which finding smallest explanations is tractable.

9.1. Boolean domains

We first define some simple functions over boolean domains:

$$f_{\text{OR}}(u,v) = \begin{cases} 1 & \text{if } u = 1 \ \vee \ v = 1 \\ 0 & \text{otherwise.} \end{cases}$$

$$f_{\text{AND}}(u,v) = \begin{cases} 1 & \text{if } u = v = 1 \\ 0 & \text{otherwise.} \end{cases}$$

$$f_{\text{CUT}}(u,v) = \begin{cases} 1 & \text{if } u = 0 \ \wedge \ v = 1 \\ 0 & \text{otherwise.} \end{cases}$$

$$f_{\text{NEQ}}(u,v) = \begin{cases} 1 & \text{if } u \neq v \\ 0 & \text{otherwise.} \end{cases}$$

We note that f_{OR} and f_{AND} are monotone, and that f_{OR}, f_{CUT} and f_{NEQ} are submodular. Indeed, modulo domain inversion (i.e. interchanging 0 and 1 in each domain), these are the only $\{0,1\}$-valued functions g of arity 2 over boolean domains which are not modular but for which $\text{CFN}(\{g\}) \in \text{P}$ [67].

Proposition 12. MinCardAXp$^+(\mathcal{T}_\mathcal{L})$ *and* MinCardCXp$^-(\mathcal{T}_\mathcal{L})$ *are* NP*-hard if* $f_{\text{OR}} \in \mathcal{L}$*.*

Proof. The proof is by reduction from Minimum Vertex Cover which is a well-known NP-hard problem. Let $G = \langle V, E \rangle$ be a graph with $V = \{1, \ldots, n\}$ and $m = |E|$. Let

$$f(\mathbf{x}) \ = \ \sum_{\{i,j\} \in E} f_{\text{OR}}(x_i, x_j)$$

Let $t = m - 1$ and $\mathbf{v} = (1, \ldots, 1)$. Clearly $f(\mathbf{v}) = m > t$, so $\mathbf{v}$ is positively classified by the corresponding threshold classifier. The AXp's of $f(\mathbf{v}) > t$ are easily seen to be subset-minimal $S \subseteq V$ such that for all $\{i,j\} \in E$, $i \in S$ or $j \in S$ (since each copy of f_{OR} has to return 1 whatever the values assigned to features not in S). Thus, a minimum-cardinality AXp is a smallest vertex cover of the graph G. This completes the reduction from Minimum Vertex Cover to MinCardAXp$^+(\mathcal{T}_\mathcal{L})$, which is clearly polynomial.

Now consider $\mathbf{w} = (0, \ldots, 0)$ which is negatively classified since $f(\mathbf{w}) = 0 \leq t = m - 1$ (assuming without loss of generality that $m > 0$). The

minimum-cardinality CXp's of $f(\mathbf{w}) \leq t$ are again the smallest vertex covers of the graph G. Thus we have a polynomial reduction from MINIMUM VERTEX COVER to MINCARDCXP$^-(\mathcal{T}_\mathcal{L})$. □

Proposition 13. MINCARDAXP$^+(\mathcal{T}_\mathcal{L})$ *and* MINCARDCXP$^-(\mathcal{T}_\mathcal{L})$ *are* NP*-hard if* $f_{\text{AND}} \in \mathcal{L}$.

Proof. The proof is by reduction from CLIQUE, a well-known NP-complete problem, which consists in determining whether a graph contains a clique of size k, i.e. whether the complete graph on k vertices is a subgraph. Let $G = \langle V, E \rangle$ be a graph with $V = \{1, \ldots, n\}$ and $m = |E|$. Let

$$f(\mathbf{x}) \;=\; \sum_{\{i,j\} \in E} f_{\text{AND}}(x_i, x_j)$$

Let $t = k(k-1)/2 - 1$ and $\mathbf{v} = (1, \ldots, 1)$. Suppose, without loss of generality, that $m \geq k(k-1)/2$ and $k > 1$, otherwise CLIQUE is trivial. Clearly $f(\mathbf{v}) = m > t$, so $\mathbf{v}$ is positively classified by the corresponding threshold classifier. Let S be a minimum-cardinality AXp of $f(\mathbf{v}) > t$. We claim that G contains contains a clique of size k if and only if $|S| = k$.

In one direction, suppose that $C = \langle V_C, E_C \rangle$ is a subgraph of G which is a clique of size k. Then the minimum value of f over all feature vectors which agree with $\mathbf{v}$ on features in V_C is $k(k-1)/2 > t$. Furthermore this minimum value of f is at most $(k-1)(k-2)/2 \leq k(k-1)/2 - 1 = t$ for proper subsets of V_C (since we assume $k > 1$). Hence V_C is an AXp of $f(\mathbf{v}) > t$.

In the other direction, suppose that S is an AXp of $f(\mathbf{v}) > t$ and $|S| = k$. Then the minimum value of f over all feature vectors which agree with $\mathbf{v}$ on features in S is at most $k(k-1)/2$ (due to $|S| = k$) and at least $t + 1 = k(k-1)/2$ since S is an AXp. It follows that this minimum value of f must be equal to $k(k-1)/2$ which is only attained when the induced subgraph of G on vertex set S is a clique of size k.

Thus, there is a minimum-cardinality AXp of size k iff there is a clique of size k in the graph G. This completes the reduction from CLIQUE to MINCARDAXP$^+(\mathcal{T}_\mathcal{L})$, which is clearly polynomial.

Now consider $\mathbf{w} = (0, \ldots, 0)$ which is negatively classified since $f(\mathbf{w}) = 0 \leq t = k(k-1)/2 - 1$. By a similar argument to the above, there is a minimum-cardinality CXp of $f(\mathbf{w}) \leq t$ of size k iff there is a clique of size k in the graph G. Hence, there is a polynomial reduction from CLIQUE to MINCARDCXP$^-(\mathcal{T}_\mathcal{L})$. □

Proposition 14. MINCARDAXP$^+(\mathcal{T}_\mathcal{L})$ *is* NP*-hard if* $f_{\text{NEQ}} \in \mathcal{T}_\mathcal{L}$.

Proof. The proof is by reduction from MINIMUM VERTEX COVER, as in the proof of Proposition 12 but, this time, there is a feature corresponding to each vertex and two features corresponding to each edge of the graph. Let $G = \langle V, E \rangle$ be a graph with $V = \{1, \ldots, n\}$ and $m = |E|$.

A feature vector $\mathbf{x}$ is composed of $n + 2m$ features: a feature x_i ($i \in V$) corresponding to each vertex and two features x_{ij}, x_{ji} ($i < j \wedge \{i, j\} \in E$) corresponding to each edge. Let

$$f(\mathbf{x}) \;=\; \sum_{i<j\wedge\{i,j\}\in E} (f_{\text{NEQ}}(x_i, x_j)+f_{\text{NEQ}}(x_i, x_{ij})+f_{\text{NEQ}}(x_j, x_{ij})+2f_{\text{NEQ}}(x_{ij}, x_{ji}))$$

Let $t = 4m-1$ and $\mathbf{v} = (1, \ldots, 1, 0, \ldots, 0, 1, \ldots, 1)$ (i.e. $\mathbf{v}$ assigns 1 to each x_i ($i \in V$), 0 to each x_{ij} ($i < j \wedge \{i, j\} \in E$) and 1 to each x_{ji} ($i < j \wedge \{i, j\} \in E$)). We can easily see that $f(\mathbf{v}) = 4m > t$, so $\mathbf{v}$ is positively classified by the threshold classifier corresponding to the objective function f and threshold t. We will show that for each minimum-cardinality AXp of $f(\mathbf{v}) > t$, there is a corresponding minimum vertex cover of G and vice versa.

Consider the term in f corresponding to edge $\{i, j\}$:

$$T_{ij} \;=\; f_{\text{NEQ}}(x_i, x_j) + f_{\text{NEQ}}(x_i, x_{ij}) + f_{\text{NEQ}}(x_j, x_{ij}) + 2f_{\text{NEQ}}(x_{ij}, x_{ji})$$

Since the three boolean features x_i, x_j, x_{ij} cannot all be different, we can easily deduce that the maximum value of T_{ij} is 4 and hence the maximum value of $f(\mathbf{x})$ is $4m$ (the value of $f(\mathbf{v})$). Consider the quadruplet of features $(x_i, x_j, x_{ij}, x_{ji})$, assigned $(1, 1, 0, 1)$ when $\mathbf{x} = \mathbf{v}$. An AXp S must fix features so that each term T_{ij} is guaranteed to have a value of 4 whatever the values assigned to the features not in S. This implies that S must contain x_{ij} and x_{ji}. Furthermore, S must contain at least one of x_i and x_j, since changing both of x_i,x_j reduces the value of T_{ij} from 4 to 2. Let $C = \{i \mid (1 \leq i \leq n) \wedge (x_i \in S)\}$. Then C is a vertex cover of G since, as we have just seen, S contains at least one of x_i and x_j for each edge $\{i, j\} \in E$.

Hence, for each minimum-cardinality AXp S, there is a minimum vertex cover of the graph G. Conversely, it is easy to see that each minimum vertex cover of G is an AXp of $f(\mathbf{v}) > t = 4m - 1$. Thus, finding a minimum-cardinality AXp for this f and t is equivalent to finding a minimum vertex cover of G. This reduction is clearly polynomial. □

Proposition 15. MINCARDAXP$^+(\mathcal{T}_\mathcal{L})$ *is* NP*-hard if* $f_{\text{CUT}} \in \mathcal{L}$.

Proof. This is a corollary of Proposition 14 since $f_{\text{NEQ}}(u,v) = f_{\text{CUT}}(u,v) + f_{\text{CUT}}(v,u)$ provides a polynomial reduction from MinCardAXp$^+(\mathcal{T}_{\{f_{\text{NEQ}}\}})$ to MinCardAXp$^+(\mathcal{T}_{\mathcal{L}})$ if $f_{\text{CUT}} \in \mathcal{L}$. □

As pointed out at the beginning of this section, there is an asymmetry between finding smallest AXp's and smallest CXp's. Indeed, for AXp's we want to minimise the number of features which stay the same, whereas for CXp's we want to minimise the number of features which change. In order to cover all cases for contrastive explanations, we need to study two more functions over boolean domains:

$$f_{\text{EQ}}(u,v) = \begin{cases} 1 & \text{if } u = v \\ 0 & \text{otherwise.} \end{cases}$$

$$f_{\text{NCUT}}(u,v) = \begin{cases} 1 & \text{if } u = 1 \ \vee \ v = 0 \\ 0 & \text{otherwise.} \end{cases}$$

The following two propositions show that each of these two functions are again sufficient to provoke NP-hardness (for the case of a smallest CXp of a negative decision).

Proposition 16. MinCardCXp$^-(\mathcal{T}_{\mathcal{L}})$ *is* NP*-hard if* $f_{\text{EQ}} \in \mathcal{L}$.

Proof. The proof is by a reduction from Clique. Let $G = \langle V, E\rangle$ be a graph with $V = \{1, \ldots, n\}$ and $m = |E|$. We consider the following function f on $n + 2m$ features: x_i $(i \in V)$ and x_{ij}, x_{ji} $(i < j \wedge \{i,j\} \in E)$.

$$f(\mathbf{x}) \ = \sum_{i<j \wedge \{i,j\} \in E} (f_{\text{EQ}}(x_i, x_j) + f_{\text{EQ}}(x_i, x_{ij}) + f_{\text{EQ}}(x_j, x_{ij}) + 2f_{\text{EQ}}(x_{ij}, x_{ji}))$$

Let $t = 3m + k(k-1) - 1$ (where $k > 1$) and $\mathbf{w} = (0, \ldots, 0, 1, \ldots, 1, 1, \ldots, 1)$ (i.e. each x_i is assigned 0, each x_{ij} is assigned 1 and x_{ji} assigned 1). Clearly $f(\mathbf{w}) = 3m \leq t$, so $\mathbf{w}$ is negatively classified. Consider the term

$$T_{ij} \ = \ f_{\text{EQ}}(x_i, x_j) + f_{\text{EQ}}(x_i, x_{ij}) + f_{\text{EQ}}(x_j, x_{ij}) + 2f_{\text{EQ}}(x_{ij}, x_{ji})$$

It is easily verified that the value of T_{ij} is either 1, 3 or 5. The value of T_{ij} is 3 when $\mathbf{x} = \mathbf{w}$ and only increases (from 3 to 5) if both x_i and x_j change from 0 to 1 *or* both of x_{ij} and x_{ji} change from 1 to 0. Observe that flipping just x_i alone does not change the value of T_{ij}. So, in the case in which x_i and

x_j are both flipped, this has no side-effects on the values of T_{ih} for $h \neq j$ or T_{hj} for $h \neq i$. Given any CXp X, we can replace any pair of features x_{ij},x_{ji} by the pair x_i,x_j: the resulting set will still be a CXp thanks to the lack of side-effects on terms other than T_{ij}. The resulting CXp will be of the same size or even smaller (if x_i or x_j were already in X). Thus, without loss of generality we can consider just CXp's consisting of a subset of the features x_i $(i = 1, \ldots, n)$ corresponding to the vertices of G. Since a CXp of $f(\mathbf{x}) \leq t$ requires an increase of $t - f(\mathbf{w}) = k(k-1)$ in $f(\mathbf{x})$, it follows that CXp's of size k correspond to cliques of size k in G. Hence there is a polynomial reduction from Clique to MinCardCXp$^-(\mathcal{T}_\mathcal{L})$. □

Proposition 17. MinCardCXp$^-(\mathcal{T}_\mathcal{L})$ *is* NP*-hard if* $f_{\text{NCUT}} \in \mathcal{L}$.

Proof. This is a corollary of Proposition 16 since $f_{\text{EQ}}(u, v) = f_{\text{NCUT}}(u, v) + f_{\text{NCUT}}(v, u) - 1$ provides a polynomial reduction from MinCardCXp$^-(\mathcal{T}_{\{f_{\text{EQ}}\}})$ to MinCardCXp$^-(\mathcal{T}_\mathcal{L})$ if $f_{\text{NCUT}} \in \mathcal{L}$. □

Theorem 7. *Suppose that* $\mathcal{L} \subseteq$ P *is a language of* $\{0, 1\}$*-valued functions over boolean domains that is closed under fixing arguments. Then each of the problems* MinCardAXp$^+(\mathcal{T}_\mathcal{L})$, MinCardAXp$^-(\mathcal{T}_\mathcal{L})$, MinCardCXp$^+(\mathcal{T}_\mathcal{L})$ *and* MinCardCXp$^-(\mathcal{T}_\mathcal{L})$ *belong to* P *if all functions in* $\mathcal{L}$ *are modular, otherwise these four problems are* NP*-hard.*

Proof. We know from Theorem 6 that if all functions in $\mathcal{L}$ are modular then each of MinCardAXp$^+(\mathcal{T}_\mathcal{L})$, MinCardAXp$^-(\mathcal{T}_\mathcal{L})$, MinCardCXp$^+(\mathcal{T}_\mathcal{L})$ and MinCardCXp$^-(\mathcal{T}_\mathcal{L})$ belong to P.

It is well known that if a function f is not modular then there is a pair of vectors $\mathbf{a}, \mathbf{b}$ which are a witness of non-modularity (i.e. $f(\mathbf{a}) + f(\mathbf{b}) \neq f(\min(\mathbf{a}, \mathbf{b})) + f(\max(\mathbf{a}, \mathbf{b}))$) and such that $\mathbf{a}, \mathbf{b}$ differ on only two variables. Since $\mathcal{L}$ is closed under fixing arguments, it follows that if $\mathcal{L}$ contains a function f which is not modular, then it contains a non-modular function on just two variables (the variables on which $\mathbf{a}$ and $\mathbf{b}$ differ). Thus it suffices to prove NP-hardness for languages containing a non-modular function on two variables.

There are exactly 16 distinct $\{0, 1\}$-valued functions g on two boolean variables, since each of the four values $g(a, b)$ $(a, b \in \{0, 1\})$ may be 0 or 1. Ten of these functions are not modular. Three of these functions are neither submodular, monotone or antitone, and hence by the argument in the proof of Proposition 6 and the characterisation of tractable languages of

CFN($\mathcal{L}$) over boolean domains [67], finding one AXp (let alone a minimum-cardinality AXp) of a positive decision is NP-hard if $g \in \mathcal{L}$. The seven remaining functions are either one of f_{AND}, f_{OR}, f_{NEQ} or f_{CUT} or equivalent to one of these functions after inversion of all domains. For these seven non-modular functions, NP-hardness of MinCardAXp$^+(\mathcal{T}_{\mathcal{L}})$ follows from Proposition 12, Proposition 13, Proposition 14 and Proposition 15.

Given a language $\mathcal{L}$ of $\{0,1\}$-valued functions, let $\mathcal{L}'$ represent the language of $\{0,1\}$-valued functions $1-g$ for $g \in \mathcal{L}$. Consider a minimum-cardinality AXp of $f(\mathbf{v}) \leq t$ where f is the sum of M functions from $\mathcal{L}$. Let $t' = M - t - 1$ and $f' = M - f$. Clearly, f' is the sum of M functions from $\mathcal{L}'$. Now $f(\mathbf{v}) \leq t \Leftrightarrow M - f'(\mathbf{v}) \leq M - t' - 1 \Leftrightarrow f'(\mathbf{v}) \geq t' + 1 \Leftrightarrow f'(\mathbf{v}) > t'$. Thus the minimum-cardinality AXp's of the negative decision $f(\mathbf{v}) \leq t$, where $f \in \mathcal{L}$, are the minimum-cardinality AXp's of the positive decision $f'(\mathbf{v}) > t'$, where $f' \in \mathcal{L}'$. Knowing that $\mathcal{L}$ is modular iff $\mathcal{L}'$ is modular, we can deduce that the dichotomy (proved for minimum-cardinality AXp's of positive decisions in the previous paragraph) also holds for minimum-cardinality AXp's of negative decisions.

For the case of CXp's, we consider first negative decisions. All non-modular $\{0,1\}$-valued functions on two boolean variables are either one of f_{AND}, f_{OR}, f_{NEQ}, f_{CUT}, f_{EQ}, f_{NCUT} or are equivalent to one of these functions after inversion of all domains. By the characterisation of tractable languages of CFN($\mathcal{L}$) over boolean domains [67] we can deduce from Theorem 4 and the argument in the proof of Proposition 8 that finding one CXp (which requires maximising the objective function f) is NP-hard if $f_{\text{NEQ}} \in \mathcal{L}$ or $f_{\text{CUT}} \in \mathcal{L}$ since these functions are neither supermodular, monotone nor antitone (the only tractable languages for the maximisation of sums of finite-valued cost functions over boolean domains). We can then deduce the NP-hardness of MinCardCXp$^-(\mathcal{T}_{\mathcal{L}})$ if $\mathcal{L}$ contains a non-modular function since the cases in which one of f_{AND}, f_{OR}, f_{EQ}, f_{NCUT} belongs to $\mathcal{L}$ are covered by Proposition 12, Proposition 13, Proposition 16 and Proposition 17.

By a similar argument to that in the case of AXp's, the dichotomy for minimum-cardinality CXp's of positive decisions follows from the dichotomy for minimum-cardinality CXp's of negative decisions. □

9.2. Arbitrary finite domains

We now consider arbitrary finite values and domains of any finite size. On the other hand, we assume that all unary functions are available. Recall that $\mathcal{L}_{\text{mod}}$ is the language of modular functions and that all modular

functions are expressible as the sum of unary functions. We know that MinCardAXp$^+(\mathcal{T}_{\mathcal{L}_{\text{mod}}}) \in$ P by Theorem 6. We now show that this $\mathcal{L}_{\text{mod}}$ a maximal tractable language (assuming P $\neq$ NP).

Theorem 8. *Consider threshold classifiers whose objective function is a sum of functions from a language $\mathcal{L}$ of functions which is a proper superset of $\mathcal{L}_{\text{mod}}$ (and hence contains all unary functions). Then* MinCardAXp$^+(\mathcal{T}_{\mathcal{L}})$, MinCardAXp$^-(\mathcal{T}_{\mathcal{L}})$, MinCardCXp$^+(\mathcal{T}_{\mathcal{L}})$ *and* MinCardCXp$^-(\mathcal{T}_{\mathcal{L}})$ *are* NP*-hard.*

Proof. For each of the four combinations (an abductive/contrastive explanation of a positive/negative decision) we use the same construction to demonstrate a polynomial reduction from either Minimum Vertex Cover or Clique. The main difference between the cases lies only in the instance $\mathbf{w}$ whose decision is to be explained and the threshold. We therefore give a detailed proof for MinCardAXp$^+(\mathcal{L})$ and then indicate how this proof can be modified for the other three cases.

MinCardAXp$^+(\mathcal{T}_{\mathcal{L}})$. Since $\mathcal{L}_{\text{mod}} \subset \mathcal{L}$, there is a function $g \in \mathcal{L}$ which is not modular. Thus, there are vectors $\mathbf{u} = (u_1, \ldots, u_r)$, $\mathbf{v} = (v_1, \ldots, v_r)$ such that

$$g(\max(\mathbf{u}, \mathbf{v})) \; + \; g(\min(\mathbf{u}, \mathbf{v})) \; \neq \; g(\mathbf{u}) \; + \; g(\mathbf{v}) \tag{14}$$

where max and min are applied component-wise. It is well known that submodularity (and supermodularity) can be tested by comparing only vectors which differ in only two arguments [27]. So, we can assume without loss of generality, and by permuting arguments if necessary, that $u_1 > v_1$, $u_2 < v_2$ and $u_i = v_i$ $(i = 3, \ldots, r)$ where r is the arity of g. Let $\alpha = g(\max(\mathbf{u}, \mathbf{v})) = g(u_1, v_2, u_3, \ldots, u_r)$, $\beta = g(\mathbf{u})$, $\gamma = g(\mathbf{v})$ and $\delta = g(\min(\mathbf{u}, \mathbf{v})) = g(v_1, u_2, u_3, \ldots, u_r)$. Thus, from Equation 14 we can identify two distinct cases:

1. $\alpha + \delta \; < \; \beta + \gamma$ (i.e. g is not supermodular)
2. $\alpha + \delta \; > \; \beta + \gamma$ (i.e. g is not submodular)

For any real value ρ and any domain value a, let f_a^ρ be the unary function given by $f_a^\rho(a) = \rho$, $f_a^\rho(x) = 0$ (for $x \neq a$). Let $\overline{f}_A^\rho$ be the unary function defined by $\overline{f}_A^\rho(x) = 0$ if $x \in A$ and $\overline{f}_A^\rho(x) = \rho$ if $x \notin A$. We will use the unary functions $\overline{f}_A^\rho$ to assign large values (greater than the threshold t) to domain values outside of subdomains A of interest, so that such values $b \notin A$ have

no effect on determining whether a set of features is an AXp. This reduces the problem to size-2 domains, but since these domains (namely $\{u_1, v_1\}$ and $\{u_2, v_2\}$) may be different, we need a gadget involving two copies of g and some unary functions $f_a^\rho(a) = \rho$ (for different values of a and ρ) to reduce to boolean domains. Our gadget will effectively simulate f_{OR} which will allow us to use, as in the proof of Proposition 12, a reduction from Minimum Vertex Cover. Let $A_1 = \{u_1, v_1\}$, $A_2 = \{u_2, v_2\}$ and $A_i = \{u_i\}$ for $i = 3, \ldots, r$.

First, consider the case $\alpha + \delta < \beta + \gamma$. Let μ be an arbitrary value greater than the threshold t. Define the arity-$(r+1)$ function h as follows:

$$\begin{aligned} h(x_1, x_1', x_2, x_3, \ldots, x_r) \;=\; & g(x_1, x_2, \ldots, x_r) \;+\; g(x_1', x_2, \ldots, x_r) \\ & + f_{u_1}^{\gamma}(x_1) + f_{u_1}^{\gamma}(x_1') + f_{v_1}^{\alpha}(x_1) + f_{v_1}^{\alpha}(x_1') + f_{v_2}^{2\beta}(x_2) \\ & + f_{u_2}^{\alpha+\beta+\gamma-\delta}(x_2) + \left(\sum_{i=1}^{r} \overline{f}_{A_i}^{\mu}(x_i) \right) + \overline{f}_{A_1}^{\mu}(x_1') \end{aligned}$$

For $x_1, x_1' \in A_1$, $x_2 \in A_2$ and $x_i = u_i$ $(i = 3, \ldots, r)$, the last two terms in the definition of h contribute 0 and we have the following values for h:

$$\begin{aligned} h(v_1, v_1, u_2, u_3, \ldots, u_r) &= 3\alpha + \beta + \gamma + \delta \\ h(v_1, u_1, u_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \\ h(v_1, v_1, v_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \\ h(v_1, u_1, v_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \\ h(u_1, v_1, u_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \\ h(u_1, u_1, u_2, u_3, \ldots, u_r) &= \alpha + 3\beta + 3\gamma - \delta \\ h(u_1, v_1, v_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \\ h(u_1, u_1, v_2, u_3, \ldots, u_r) &= 2(\alpha + \beta + \gamma) \end{aligned}$$

Note that $\alpha + 3\beta + 3\gamma - \delta > 2(\alpha + \beta + \gamma)$ and $3\alpha + \beta + \gamma + \delta < 2(\alpha + \beta + \gamma)$ since $\alpha + \delta < \beta + \gamma$.

We will demonstrate a reduction from Minimum Vertex Cover in which for each edge $e = \{i, j\}$ of a graph $G = \langle V, E \rangle$ there is a copy of h. So, to understand this reduction, consider first a single copy of h: $h(x_i, x_j, x_e, x_{e3}, x_{e4}, \ldots, x_{er})$ on variables x_i, x_j corresponding to the vertices i, j, variable x_e corresponding to the edge $e = \{i, j\}$ and 'padding' variables $x_{e3}, \ldots, x_{er}$. The AXp's of $h(u_1, u_1, v_2, u_3, \ldots, u_r) \geq 2(\alpha + \beta + \gamma)$ are $\{i\}$, $\{j\}$ and $\{e\}$, since the only way that the minimum value of $h(x_i, x_j, x_e, x_{e3}, \ldots, x_{er})$

can be less than $2(\alpha+\beta+\gamma)$ is if all three of the variables x_i, x_j, x_e are allowed to vary from their respective values u_1, u_1, v_2. Recall that the large value μ incurred by other assignments means that we only need to consider $(x_i, x_j, x_e) \in A_1 \times A_1 \times A_2$.

So, given a graph $G = \langle V, E \rangle$, define the objective function f as follows.

$$f(\mathbf{x}) \;=\; \sum_{e=\{i,j\} \in E} h(x_i, x_j, x_e, x_{e3}, x_{e4}, \ldots, x_{er})$$

where the feature vector $\mathbf{x}$ contains a feature x_i for each $i \in V = \{1, \ldots, n\}$, a feature for each edge $e \in E$ together with $r-2$ features $x_{e3}, \ldots, x_{er}$ for each edge $e \in E$. Let $t = 2(\alpha+\beta+\gamma)m - \epsilon$, where $m = |E|$ and ϵ is the smallest non-zero difference between values taken by the function g. Let $\mathbf{w}$ be the feature vector that assigns the value u_1 to each feature x_i ($i \in V$), the value v_2 to each feature x_e ($e \in E$) and the value u_k to each feature x_{ek} ($e \in E$, $k = 3, \ldots, r$). Then $f(\mathbf{w}) = 2(\alpha+\beta+\gamma)m > t$. By the argument in the previous paragraph, any AXp S of $f(\mathbf{w}) > t$ must contain for each edge $e = \{i, j\}$ of G, at least one of i, j, or e. Since S is minimal, it cannot contain e and i (or e and j) since the feature e could be deleted to leave an AXp. If S is a minimum-cardinality AXp of $f(\mathbf{w}) > t$ which contains e and neither of i or j, then we can replace e by i, i.e. $(S \setminus \{e\}) \cup \{i\}$ is also necessarily a minimum-cardinality AXp. So, from any minimum-cardinality AXp we can obtain a minimum-cardinality set S which contains either i or j for each edge $\{i, j\}$ of G. We have therefore demonstrated a polynomial reduction from Minimum Vertex Cover.

To complete the proof we need to consider the case $\alpha+\delta > \beta+\gamma$. We can use the same construction and, in particular, the same functions h and f. Now we have $\alpha+3\beta+3\gamma-\delta < 2(\alpha+\beta+\gamma)$ and $3\alpha+\beta+\gamma+\delta > 2(\alpha+\beta+\gamma)$ since $\alpha+\delta > \beta+\gamma$. This means that when explaining $h(v_1, v_1, v_2, u_3, \ldots, u_r) \geq 2(\alpha+\beta+\gamma)$ we again have the three AXp's $\{i\}$, $\{j\}$ and $\{e\}$, since it suffices to fix any one of the first three features to avoid the low-scoring assignment $(u_1, u_1, u_2, u_3, \ldots, u_r)$. The objective function f is identical, but the instance to explain changes compared to the above proof: $\mathbf{w}'$ assigns the value v_1 to each feature x_i ($i \in V$), the value v_2 to each feature x_e ($e \in E$) and the value u_k to each feature x_{ek} ($e \in E$, $k = 3, \ldots, r$). As above, from any minimum-cardinality AXp of $f(\mathbf{w}') > t$ we can obtain a minimum vertex cover of G, and we again have a polynomial reduction from Minimum Vertex Cover.

MinCardAXp$^-(\mathcal{T}_\mathcal{L})$. We use an identical construction as above in the MinCardAXp$^+$ case, except for the subtle difference that the value μ assigned to feature-values that we wish to ignore is now a large negative value so that they have no effect on an explanation of a negative decision.

Let $\tilde{t} = 2(\alpha+\beta+\gamma)m$. In case (1) (i.e. $\alpha+\delta < \beta+\gamma$), for any minimum-cardinality AXp of the negative decision $f(\mathbf{w}') \leq \tilde{t}$, there is a minimum vertex cover of G and vice-versa. Similarly, in case (2) (i.e. $\alpha+\delta > \beta+\gamma$), for any minimum-cardinality AXp of the negative decision $f(\mathbf{w}) \leq \tilde{t}$, there is a minimum vertex cover of G and vice-versa. Hence, we have a polynomial reduction from Minimum Vertex Cover.

MinCardCXp$^+(\mathcal{T}_\mathcal{L})$. In the case of contrastive explanations of a positive decision, we demonstrate a polynomial reduction from Clique. The function f and the instances to explain are identical to those in the case of MinCardAXp$^+$, but the value of the thresholds are different.

In case (1) (i.e. $\alpha+\delta < \beta+\gamma$), the instance to explain is $\mathbf{w}$, defined above, i.e. $\mathbf{w}$ assigns the value u_1 to each feature x_i ($i \in V$), the value v_2 to each feature x_e ($e \in E$) and the value u_k to each feature x_{ek} ($e \in E$, $k = 3, \ldots, r$). $f(\mathbf{w}) = 2(\alpha+\beta+\gamma)m$. We say that the features *associated* with an edge $e = \{i, j\}$ of G are x_i, x_j and x_e. A decrease in the value of $f(\mathbf{w})$ occurs when (and only when) each of the three features x_i, x_j, x_e associated with an edge $e = \{i, j\}$ change from u_1, u_1, v_2 to v_1, v_1, u_2. The resulting decrease in $f(\mathbf{w})$ is $2(\alpha+\beta+\gamma)-(3\alpha+\beta+\gamma+\delta) = \beta+\gamma-\alpha-\delta$. So, to provoke a decrease of exactly $h(\beta+\gamma-\alpha-\delta)$, the features associated with exactly h edges must be changed. Let $H = \langle V_H, E_H \rangle$ be the subgraph of G corresponding to these h edges. Let $t_c = 2(\alpha+\beta+\gamma)m - \frac{k(k-1)}{2}(\beta+\gamma-\alpha-\delta)$. A CXp of $f(\mathbf{w}) > t_c$ corresponds to a subgraph $H = \langle V_H, E_H \rangle$ of G such that $|E_H| = \frac{k(k-1)}{2}$. The total number of features associated with the edges of H is $|V_H| + |E_F|$. It follows that there is a CXp of $f(\mathbf{w}) > t_c$ of size $\frac{k(k+1)}{2} = k + \frac{k(k-1)}{2}$ iff G has a subgraph H with $\frac{k(k-1)}{2}$ edges and k vertices (i.e. a clique of size k). We have thus demonstrated a polynomial reduction from Clique.

In case (2) (i.e. $\alpha+\delta > \beta+\gamma$), the feature vector to explain is $\mathbf{w}'$ which assigns the value v_1 to each feature x_i ($i \in V$), the value v_2 to each feature x_e ($e \in E$) and the value u_k to each feature x_{ek} ($e \in E$, $k = 3, \ldots, r$). The threshold is now $t'_c = 2(\alpha+\beta+\gamma)m - \frac{k(k-1)}{2}(\alpha+\delta-\beta-\gamma)$. As in the proof of case (1), the minimum-cardinality CXp's of the positive decision $f(\mathbf{w}') > t'_c$, correspond to cliques H of G of size k. Hence, again we have a polynomial

reduction from Clique.

MinCardCXp$^-(\mathcal{T}_\mathcal{L})$. In the case of contrastive explanations of a *negative* decision, we again demonstrate a polynomial reduction from Clique. We use the same function f as in the MinCardAXp$^-$ case (i.e. identical to the function f used in the MinCardAXp$^+$ case except that the value μ assigned to feature-values we wish to ignore is a large negative value).

In case (1) (i.e. $\alpha + \delta < \beta + \gamma$), the feature vector to explain is $\mathbf{w}'$. The threshold is now $\tilde{t}_c = 2(\alpha + \beta + \gamma)m - \frac{k(k-1)}{2}(\beta + \gamma - \alpha - \delta) - \epsilon$. The minimum-cardinality CXp's of the negative decision $f(\mathbf{w}') \leq \tilde{t}_c$ correspond to cliques H of G of size k. Hence, again we have a polynomial reduction from Clique.

In case (2) (i.e. $\alpha + \delta > \beta + \gamma$), the feature vector to explain is $\mathbf{w}$. The threshold is now $\tilde{t}'_c = 2(\alpha + \beta + \gamma)m - \frac{k(k-1)}{2}(\alpha + \delta - \beta - \gamma) - \epsilon$. The minimum-cardinality CXp's of the negative decision $f(\mathbf{w}) \leq \tilde{t}'_c$ correspond to cliques H of G of size k. Hence, again we have a polynomial reduction from Clique. □

10. Discussion and Conclusion

We have investigated the complexity of finding subset-minimal abductive or contrastive explanations for different families of classifiers. Explaining a decision is, in general, (co)NP-hard, but there are interesting tractable classes, i.e. classes of (constrained) threshold-based classifiers for which one explanation can be found in polynomial time. Interestingly, these classes coincide for abductive and contrastive explanations, but not for positive and negative decisions.

We have identified a strong link with the tractability of cost-function networks. However, since, as yet, there is no known characterisation of the complexity of cost-function languages over infinite domains, the existence of a language dichotomy concerning the complexity of finding an explanation for decisions taken by classifiers with real-valued features is still an open problem.

Instead of searching for one explanation, we may want to find many explanations. Unfortunately, the fact that a greedy algorithm can find one explanation in polynomial time provides no guarantee that explanations can be enumerated with polynomial delay. For linear classifiers, there is a polynomial-delay algorithm for enumerating abductive explanations [12],

and it is an open question is whether this is true for other families of classifiers. It is known to be false for monotone classifiers (assuming P≠NP) [15].

Since the number of explanations may be exponential, we investigated the problem of finding a small but diverse set of explanations. Unfortunately, this problem is NP-hard even for linear classifiers. Another approach is to find the best explanation according to some simple criterion such as cardinality. The problem of finding a smallest explanation is tractable for the class of modular classifiers (which includes linear classifiers) but NP-hard for any proper extension of this class. Although our NP-hardness results coincide for abductive and contrastive explanations, it is known that the complexity of finding minimum-cardinality abductive explanations lies at the second level of the polynomial hierarchy [66] whereas finding minimum-cardinality contrastive explanations is at the first level (Proposition 11). Thus, in the case of minimum-cardinality abductive explanations, the complexity landscape among the NP-hard languages could reveal richer structure and is an interesting open problem.

Acknowledgements

This work was supported by the AI Interdisciplinary Institute ANITI, funded by the French program "Investing for the Future – PIA3" under Grant agreement ANR-19-PI3A-0004.

References

[1] A. Shih, A. Choi, A. Darwiche, A symbolic approach to explaining bayesian network classifiers, in: IJCAI, 2018, pp. 5103–5111.

[2] A. Shih, A. Choi, A. Darwiche, Compiling bayesian network classifiers into decision graphs, in: AAAI, 2019, pp. 7966–7974. doi:10.1609/aaai.v33i01.33017966.

[3] A. Ignatiev, N. Narodytska, J. Marques-Silva, Abduction-based explanations for machine learning models, in: AAAI, 2019, pp. 1511–1519.

[4] A. Ignatiev, N. Narodytska, J. Marques-Silva, On relating explanations and adversarial examples, in: NeurIPS, 2019, pp. 15857–15867.

[5] A. Darwiche, A. Hirth, On the reasons behind decisions, in: ECAI, 2020, pp. 712–720. doi:10.3233/FAIA200158.

[6] A. Darwiche, Three modern roles for logic in AI, in: PODS, 2020, pp. 229–243. doi:10.1145/3375395.3389131.

[7] A. Ignatiev, Towards trustable explainable AI, in: IJCAI, 2020, pp. 5154–5158. doi:10.24963/ijcai.2020/726.

[8] Y. Izza, J. Marques-Silva, On explaining random forests with SAT, in: Z. Zhou (Ed.), IJCAI, 2021, pp. 2584–2591. doi:10.24963/ijcai.2021/356.

[9] A. Ignatiev, J. P. M. Silva, SAT-based rigorous explanations for decision lists, in: C. Li, F. Manyà (Eds.), SAT 2021, Vol. 12831 of LNCS, Springer, 2021, pp. 251–269. doi:10.1007/978-3-030-80223-3_18.

[10] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM Comput. Surv. 51 (5) (2019) 93:1–93:42. doi:10.1145/3236009.

[11] G. Audemard, F. Koriche, P. Marquis, On tractable XAI queries based on compiled representations, in: KR, 2020, pp. 838–849. doi:10.24963/kr.2020/86.

[12] J. Marques-Silva, T. Gerspacher, M. C. Cooper, A. Ignatiev, N. Narodytska, Explaining naive Bayes and other linear classifiers with polynomial time and delay, in: Larochelle et al. [68].

[13] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J. Lagniez, P. Marquis, On the computational intelligibility of boolean classifiers, in: Bienvenu et al. [69], pp. 74–86. doi:10.24963/kr.2021/8.

[14] X. Huang, Y. Izza, A. Ignatiev, J. Marques-Silva, On efficiently explaining graph-based classifiers, in: Bienvenu et al. [69], pp. 356–367. doi:10.24963/kr.2021/34.

[15] J. Marques-Silva, T. Gerspacher, M. C. Cooper, A. Ignatiev, N. Narodytska, Explanations for monotonic classifiers, in: M. Meila, T. Zhang (Eds.), ICML 2021, Vol. 139 of Proceedings of Machine Learning Research, PMLR, 2021, pp. 7469–7479.

[16] Y. Izza, A. Ignatiev, J. Marques-Silva, On explaining decision trees, CoRR abs/2010.11034 (2020). arXiv:2010.11034.

[17] X. Huang, Y. Izza, A. Ignatiev, M. C. Cooper, N. Asher, J. Marques-Silva, Tractable explanations for d-dnnf classifiers, in: AAAI 2022, AAAI Press, 2022, pp. 5719–5728.

[18] A. Darwiche, A. Hirth, On the reasons behind decisions, in: G. D. Giacomo, A. Catalá, B. Dilkina, M. Milano, S. Barro, A. Bugarín, J. Lang (Eds.), ECAI, Vol. 325 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2020, pp. 712–720. doi:10.3233/FAIA200158.

[19] C. Umans, The minimum equivalent DNF problem and shortest implicants, J. Comput. Syst. Sci. 63 (4) (2001) 597–611. doi:10.1006/jcss.2001.1775.

[20] M. Arenas, P. Barceló, M. Romero, B. Subercaseaux, On computing probabilistic explanations for decision trees, CoRR abs/2207.12213 (2022). arXiv:2207.12213, doi:10.48550/arXiv.2207.12213.

[21] S. Wäldchen, J. MacDonald, S. Hauch, G. Kutyniok, The computational complexity of understanding binary classifier decisions, J. Artif. Intell. Res. 70 (2021) 351–387. doi:10.1613/jair.1.12359.

[22] Y. Izza, A. Ignatiev, N. Narodytska, M. C. Cooper, J. Marques-Silva, Provably precise, succinct and efficient explanations for decision trees, CoRR abs/2205.09569 (2022). arXiv:2205.09569, doi:10.48550/arXiv.2205.09569.

[23] V. Kolmogorov, A. A. Krokhin, M. Rolínek, The complexity of general-valued CSPs, SIAM J. Comput. 46 (3) (2017) 1087–1110. doi:10.1137/16M1091836.

[24] J. B. Orlin, A faster strongly polynomial time algorithm for submodular function minimization, Math. Program. 118 (2) (2009) 237–251. doi:10.1007/s10107-007-0189-2.

[25] N. Cowan, The magical number 4 in short-term memory: A reconsideration of mental storage capacity, Behavioral and brain sciences 24 (1) (2001) 87–114.

[26] M. C. Cooper, J. Marques-Silva, On the tractability of explaining decisions of classifiers, in: L. D. Michel (Ed.), CP 2021, Vol. 210 of LIPIcs,

Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, pp. 21:1–21:18. doi:10.4230/LIPIcs.CP.2021.21.

[27] S. Fujishige, Submodular Functions and Optimisation, 2nd Edition, Vol. 58 of Annals of Discrete Mathematics, Elsevier, 2005.

[28] D. A. Cohen, M. C. Cooper, P. Jeavons, A. A. Krokhin, The complexity of soft constraint satisfaction, Artif. Intell. 170 (11) (2006) 983–1016.

[29] M. Joseph, M. J. Kearns, J. Morgenstern, A. Roth, Fairness in learning: Classic and contextual bandits, in: Lee et al. [70], pp. 325–333.

[30] X. Liu, X. Han, N. Zhang, Q. Liu, Certified monotonic neural networks, in: Larochelle et al. [68].

[31] G. Monge, Mémoire sur la théorie des déblais et des remblais, Imprimerie Royale, 1781.

[32] R. E. Burkard, B. Klinz, R. Rudolf, Perspectives of Monge properties in optimization, Discret. Appl. Math. 70 (2) (1996) 95–161.

[33] M. Balcan, N. J. A. Harvey, Learning submodular functions, in: L. Fortnow, S. P. Vadhan (Eds.), STOC, ACM, 2011, pp. 793–802. doi:10.1145/1993636.1993741.

[34] B. W. Dolhansky, J. A. Bilmes, Deep submodular functions: Definitions and learning, in: Lee et al. [70], pp. 3396–3404.

[35] M. X. Goemans, N. J. A. Harvey, S. Iwata, V. S. Mirrokni, Approximating submodular functions everywhere, in: C. Mathieu (Ed.), SODA, SIAM, 2009, pp. 535–544. doi:10.1137/1.9781611973068.

[36] V. Feldman, P. Kothari, J. Vondrák, Representation, approximation and learning of submodular functions using low-rank decision trees, in: S. Shalev-Shwartz, I. Steinwart (Eds.), COLT, Vol. 30 of JMLR Workshop and Conference Proceedings, JMLR.org, 2013, pp. 711–740.

[37] V. Feldman, J. Vondrák, Optimal bounds on approximation of submodular and XOS functions by juntas, SIAM J. Comput. 45 (3) (2016) 1129–1170. doi:10.1137/140958207.

[38] M. Balcan, N. J. A. Harvey, Submodular functions: Learnability, structure, and optimization, SIAM J. Comput. 47 (3) (2018) 703–754. doi:10.1137/120888909.

[39] Y. T. Lee, A. Sidford, S. C. Wong, A faster cutting plane method and its implications for combinatorial and convex optimization, in: FOCS, 2015, pp. 1049–1065.

[40] D. Chakrabarty, Y. T. Lee, A. Sidford, S. C. Wong, Subquadratic submodular function minimization, in: STOC, 2017, pp. 1220–1231.

[41] D. A. Cohen, M. C. Cooper, P. Jeavons, A. A. Krokhin, A maximal tractable class of soft constraints, J. Artif. Intell. Res. 22 (2004) 1–22.

[42] A. Darwiche, P. Marquis, A knowledge compilation map, J. Artif. Intell. Res. 17 (2002) 229–264. doi:10.1613/jair.989.

[43] M. H. Anthony, Discrete mathematics of neural networks, Vol. 8 of SIAM monographs on discrete mathematics and applications, SIAM, 2001.

[44] Z. Chen, S. Toda, The complexity of selecting maximal solutions, Inf. Comput. 119 (2) (1995) 231–239. doi:10.1006/inco.1995.1087.

[45] X. Huang, Y. Izza, A. Ignatiev, M. C. Cooper, N. Asher, J. Marques-Silva, Tractable explanations for d-dnnf classifiers, in: AAAI, AAAI Press, 2022, pp. 5719–5728. doi:10.1609/aaai.v36i5.20514.

[46] N. Creignou, S. Khanna, M. Sudan, Complexity classifications of Boolean constraint satisfaction problems, Vol. 7 of SIAM monographs on discrete mathematics and applications, SIAM, 2001.

[47] N. Gorji, S. Rubin, Sufficient reasons for classifier decisions in the presence of domain constraints, in: AAAI, AAAI Press, 2022, pp. 5660–5667. doi:10.1609/aaai.v36i5.20507.

[48] M. C. Cooper, S. de Givry, M. Sánchez-Fibla, T. Schiex, M. Zytnicki, T. Werner, Soft arc consistency revisited, Artif. Intell. 174 (7-8) (2010) 449–478.

[49] P. Jeavons, M. C. Cooper, Tractable constraints on ordered domains, Artif. Intell. 79 (2) (1995) 327–339. doi:10.1016/0004-3702(95)00107-7.

[50] M. C. Cooper, S. Zivný, Tractable triangles and cross-free convexity in discrete optimisation, J. Artif. Intell. Res. 44 (2012) 455–490. doi:10.1613/jair.3598.

[51] M. C. Cooper, S. Zivný, Hybrid tractability of valued constraint problems, Artif. Intell. 175 (9-10) (2011) 1555–1569. doi:10.1016/j.artint.2011.02.003.

[52] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artif. Intell. 267 (2019) 1–38.

[53] A. Ignatiev, N. Narodytska, N. Asher, J. Marques-Silva, From contrastive to abductive explanations and back again, in: M. Baldoni, S. Bandini (Eds.), AIxIA 2020, Vol. 12414 of Lecture Notes in Computer Science, Springer, 2020, pp. 335–355. doi:10.1007/978-3-030-77091-4_21.

[54] M. C. Cooper, S. de Givry, T. Schiex, Graphical models: Queries, complexity, algorithms (tutorial), in: STACS, 2020, pp. 4:1–4:22.

[55] J. Thapper, S. Zivny, The complexity of finite-valued CSPs, J. ACM 63 (4) (2016) 37:1–37:33.

[56] A. A. Bulatov, A dichotomy theorem for nonuniform CSPs, in: FOCS, 2017, pp. 319–330. doi:10.1109/FOCS.2017.37.

[57] D. Zhuk, A proof of CSP dichotomy conjecture, in: FOCS, 2017, pp. 331–342. doi:10.1109/FOCS.2017.38.

[58] A. A. Krokhin, S. Zivný, The complexity of valued CSPs, in: A. A. Krokhin, S. Zivný (Eds.), The Constraint Satisfaction Problem: Complexity and Approximability, Vol. 7 of Dagstuhl Follow-Ups, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017, pp. 233–266. doi:10.4230/DFU.Vol7.15301.9.

[59] M. Ruffini, J. Vucinic, S. de Givry, G. Katsirelos, S. Barbe, T. Schiex, Guaranteed diversity & quality for the weighted CSP, in: ICTAI 2019, IEEE, 2019, pp. 18–25. doi:10.1109/ICTAI.2019.00012.

[60] E. Hebrard, B. Hnich, B. O'Sullivan, T. Walsh, Finding diverse and similar solutions in constraint programming, in: M. M. Veloso, S. Kambhampati (Eds.), AAAI, AAAI Press / The MIT Press, 2005, pp. 372–377.

[61] J. Horan, B. O'Sullivan, Towards diverse relaxations of over-constrained models, in: ICTAI 2009, IEEE Computer Society, 2009, pp. 198–205. doi:10.1109/ICTAI.2009.89.

[62] L. Ingmar, M. G. de la Banda, P. J. Stuckey, G. Tack, Modelling diversity of solutions, in: AAAI 2020, AAAI Press, 2020, pp. 1528–1535.

[63] R. M. Karp, Reducibility among combinatorial problems, in: R. E. Miller, J. W. Thatcher (Eds.), Proceedings of a symposium on the Complexity of Computer Computations, The IBM Research Symposia Series, Plenum Press, New York, 1972, pp. 85–103. doi:10.1007/978-1-4684-2001-2_9.

[64] E. L. Schreiber, R. E. Korf, M. D. Moffitt, Optimal multi-way number partitioning, J. ACM 65 (4) (2018) 24:1–24:61. doi:10.1145/3184400.

[65] R. Boumazouza, F. C. Alili, B. Mazure, K. Tabia, ASTERYX: A model-agnostic sat-based approach for symbolic and score-based explanations, in: G. Demartini, G. Zuccon, J. S. Culpepper, Z. Huang, H. Tong (Eds.), CIKM '21, ACM, 2021, pp. 120–129. doi:10.1145/3459637.3482321.

[66] P. Barceló, M. Monet, J. Pérez, B. Subercaseaux, Model interpretability through the lens of computational complexity, in: Larochelle et al. [68].

[67] D. A. Cohen, M. C. Cooper, P. Jeavons, A complete characterization of complexity for boolean constraint optimization problems, in: M. Wallace (Ed.), CP 2004, Vol. 3258 of LNCS, Springer, 2004, pp. 212–226. doi:10.1007/978-3-540-30201-8_18.

[68] H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), NeurIPS 2020, 2020.

[69] M. Bienvenu, G. Lakemeyer, E. Erdem (Eds.), Proceedings of the 18th International Conference on Principles of Knowledge Representation and Reasoning, KR, 2021. doi:10.24963/kr.2021.

[70] D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (Eds.), NIPS 2016, 2016.