Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
KR meets Machine Learning and Explanation

# Explaining Classification Through Global Sufficient Reasons and its Complexity

**Marco Calautti**[1] , **Enrico Malizia**[2] , **Cristian Molinaro**[3]
[1]University of Milano
[2]University of Bologna
[3]University of Calabria
marco.calautti@unimi.it, enrico.malizia@unibo.it, cristian.molinaro@unical.it

**Abstract**

In recent years, explainable AI has become a major focus of research, driven by the need to better understand how AI systems reach their decisions in order to ensure trust and effective deployment. A central challenge in this field is the explanation of classifiers. Existing approaches typically distinguish between local explanations, which account for a classifier's decision on an individual input, and global explanations, which aim to characterize the classifier's behavior as a whole, independent of any particular input. This work concentrates on global explanations and characterizes classification decisions through "maximal" sufficient conditions that, when satisfied, guarantee that the classifier assigns the desired class to *any* input. We present a detailed analysis of the computational complexity of key problems in this setting across several important families of classifiers considered in the literature.

## 1 Introduction

In recent years, Explainable Artificial Intelligence (XAI) has emerged as a central topic in AI research, driven by the need to understand the behavior of automated decision-making systems in order to foster trust, accountability, and safe deployment. A fundamental problem in XAI is the provision of explanations for classifier decisions, which can be broadly categorized into *local* and *global* approaches. While local methods provide insights at the level of individual inputs, they do not capture general patterns or invariants in the model's behavior across the input space. In contrast, global explanations characterize the classifier's behavior as a whole, independent of any particular input, revealing a general behavior of the classifier across all possible inputs.

In both settings, a central issue is to analyze the complexity of the problems at hand, to understand how to approach the development of algorithmic solutions and their inherent limits. In XAI, there has been an extensive body of work addressing complexity issues (Bassan, Amir, and Katz 2024; Ordyniak, Paesani, and Szeider 2023; Arenas et al. 2023; Cooper and Marques-Silva 2023; Cooper and Amgoud 2023; Carbonnel, Cooper, and Marques-Silva 2023; Huang et al. 2022; Audemard et al. 2022; de Colnet and Marquis 2022; Barceló et al. 2020a; Marques-Silva et al. 2020).

In this work, we focus on global explanations for classifiers by considering the notion of a *global sufficient reason*. Intuitively, a global sufficient reason for a class $c$ with respect to a classifier $\mathcal{M}$ is a condition such that any instance (i.e., input) satisfying the condition is guaranteed to be classified as $c$ by $\mathcal{M}$. This notion has already been considered in the literature under different names, e.g., it coincides with the absolute explanation by Ignatiev, Narodytska, and Marques-Silva (2019b) and the (generalized) weak abductive explanation by Izza, Ignatiev, and Marques-Silva (2022)—we illustrate it in the following example.

**Example 1.** *Consider a bank that employs a classifier $\mathcal{M}$ to determine whether to approve a loan. The outcome is represented by $c = 1$ for an approved loan and $c = 0$ for a declined loan, based on four binary input features:*

1. *stable employment,*
2. *high income,*
3. *existing debt,*
4. *good credit history.*

*Each feature takes a binary value: a value of 1 indicates that the applicant possesses the corresponding characteristic, while 0 indicates that the characteristic is absent.*

*A global sufficient reason $\phi$ for loan approval ($c = 1$) w.r.t. $\mathcal{M}$ might be a logical condition of the form $(v_1 = 1) \wedge (v_3 = v_4)$, whose meaning is that a loan is guaranteed to be approved whenever the applicant has stable employment and their debt status equals their credit history. Concretely, this captures two reasonable approval scenarios:*

- *No existing debt and poor credit history ($v_3 = 0$, $v_4 = 0$): The applicant currently carries no debt, so past credit issues are less concerning; moreover, stable employment provides further assurance.*
- *Existing debt and good credit history ($v_3 = 1$, $v_4 = 1$): The applicant has debt but also a strong track record of repayment—the model trusts their ability to manage additional obligations; moreover, stable employment provides further assurance.*

*Such explanation conveys valuable insights to any user who is willing to obtain a loan from the bank: stable employment is an anchor for approval, and there's a compensatory relationship—debt is acceptable with good credit history, and poor credit history is acceptable without current debt.*

As different global sufficient reasons may convey different amount of information, we consider notions of "maximality" to identify the most informative global sufficient reasons.

The main contribution of this work is a detailed analysis of the computational complexity of key decision problems associated with global sufficient reasons. Specifically, we study the problem IsSufficient of deciding whether a condition is a global sufficient reason, as well as the problem IsMaxSufficient of deciding whether a condition is a "maximal" global sufficient reason according to two common preorders, namely set-inclusion and cardinality. Our analysis concerns three widely studied classifier families of classifier: perceptrons, multi-layer perceptrons (MLPs), and binary decision diagrams (BDDs). Our results reveal significant differences across classifier families and provide interesting insights into (maximal) global sufficient reasons.

Specifically, for perceptrons, IsSufficient as well as IsMaxSufficient for the cardinality criterion can be decided in logarithmic space, highlighting their relative simplicity. For the set-inclusion criterion, we show membership in co-NP. In contrast, for MLPs and BDDs, the considered problems range from co-NP-completeness to $\Pi_2^{\mathrm{P}}$-completeness, demonstrating the intrinsic hardness of reasoning about global sufficient reasons in more expressive models. Our results complement the recent study on global *necessary* reasons by Calautti, Malizia, and Molinaro (2025), enabling us to draw meaningful connections between the two frameworks. In their work, the authors examined the problem IsNecessary, which asks whether a given condition is a global necessary reason, as well as IsMinNecessary, which concerns deciding whether a condition is a "minimal" global necessary reason. These problems were analyzed for the same families of classifiers and under the same criteria considered in this paper. Contrary to the commonly assumed duality between necessary and sufficient reasons, our findings reveal a clear asymmetry between the two notions. For BDDs, there is a dramatic jump in complexity: from NL-completeness of IsNecessary to co-NP-completeness of IsSufficient, and from NL-completeness of IsMinNecessary to $\Pi_2^{\mathrm{P}}$-completeness of IsMaxSufficient. For MLPs, we observe an increase in complexity, from $\mathrm{D}^{\mathrm{P}}$-completeness of IsMinNecessary to $\Pi_2^{\mathrm{P}}$-completeness of IsMaxSufficient.

By rigorously analyzing the complexity of (maximal) global sufficient reasons, this work contributes a theoretical foundation for global explainability in classification. Our findings suggest which classifier families allow for efficient reasoning, such as perceptrons, where memberships in L imply that the considered problem can be solved on highly-parallel machines—see, e.g., (Arora and Barak 2009). On the other hand, intractability results suggest the development of approximation algorithms for classifier families having intrinsically higher complexity, such as BDDs and MLPs. Furthermore, they provide the groundwork for solving such intractable problems by resorting to SAT solvers, which have proven to be very efficient at tackling computationally hard problems for computing classifiers' explanations (Marques-Silva 2022).

## 2 Preliminaries

**Classification** Throughout the paper, $n$ denotes the number of features of the instance domain, hence $n$ is assumed to be a (strictly) positive integer. An *$n$-instance* is an $n$-dimensional binary vector $\mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n$. We denote by $\mathbf{x}[i]$ the value $x_i$ from $\mathbf{x}$, for $1 \leq i \leq n$. An *$n$-feature (binary) classifier* $\mathcal{M}$ can be modeled by a function $\mathcal{M}\colon \{0,1\}^n \to \{0,1\}$ mapping $n$-instances $\mathbf{x}$ to the binary class $\mathcal{M}(\mathbf{x})$. Restricting the analysis to binary classifiers makes our framework cleaner, while still covering several relevant practical scenarios. Hardness results immediately apply to more general settings, such as those involving instances over the real numbers, while upper bounds apply to many scenarios where inputs are encoded using, e.g., one-hot encoding. Nonetheless, a complexity characterization for the case beyond binary features needs a dedicated analysis, which we leave as an interesting direction for future work.

We study the following key families of classifiers.

**Binary Decision Diagram (BDD)** A *(free) $n$-feature binary decision diagram*, or $n$-BDD for short, is defined by a rooted directed acyclic graph (DAG) $\mathcal{G} = (V, E, \lambda, \eta)$, where $\lambda$ and $\eta$ are node- and edge-labeling functions, respectively. We recall that in a rooted DAG, there is a *single* node without incoming edges, which is the root, and the nodes without outgoing edges are called sinks. The $n$-BDD $\mathcal{G}$ is such that:

- each sink of $\mathcal{G}$ is labeled with either 1 or 0;
- each internal node (i.e., a node that is not a sink) is labeled with a feature identifier, i.e., an element from $\{1, \ldots, n\}$;
- each internal node has two outgoing edges, one labeled with 1 and the other labeled with 0;
- no two internal nodes on a path of $\mathcal{G}$ originating from the root have the same label.

$\mathcal{G}$ classifies an $n$-instance $\mathbf{x}$ as $c$, denoted $\mathcal{G}(\mathbf{x}) = c$, iff there is a path $u_1, \ldots, u_m$ from the root of $\mathcal{G}$ to a sink of $\mathcal{G}$ such that $u_m$ is labeled with $c$, and, for each $i$ with $1 \leq i \leq m-1$, if $u_i$ is labeled with $j$, then the edge $(u_i, u_{i+1})$ of $\mathcal{G}$ is labeled with $\mathbf{x}[j]$; note that by definition, if $\mathcal{G}(\mathbf{x}) = c$ there always exists exactly one path witnessing this. We use BDD to denote the family of all $n$-BDDs, for all $n > 0$.

Applications of BDDs as classifiers include their use as a more succinct alternative to decision trees, e.g., see (Hu, Huguet, and Siala 2022; Cabodi et al. 2024).

**Perceptron** An *$n$-feature perceptron* $\mathcal{S}$, or $n$-perceptron for short, is defined by a pair $\mathcal{S} = (\mathbf{w}, b)$, where $\mathbf{w} = (w_1, \ldots, w_n) \in \mathbb{Q}^n$ and $b \in \mathbb{Q}$ are the perceptron's weights and bias, respectively. Then, $\mathcal{S}$ classifies an $n$-instance $\mathbf{x}$ as '1', denoted by $\mathcal{S}(\mathbf{x}) = 1$, iff $\mathbf{x} \cdot \mathbf{w} + b \geq 0$; otherwise $\mathcal{S}$ classifies $\mathbf{x}$ by '0', which is denoted as $\mathcal{S}(\mathbf{x}) = 0$. Equivalently, $\mathcal{S}$ can be seen as an object which receives as input the values $(x_1, \ldots, x_n)$, weighed via the weights $(w_1, \ldots, w_n)$, and outputs the value $\mathcal{S}(\mathbf{x}) = \mathit{step}(\mathbf{x} \cdot \mathbf{w} + b)$, where $\mathit{step}(\cdot)$ is the Heaviside step function, which is defined as $\mathit{step}(x) \coloneqq 0$ if $x < 0$, and $\mathit{step}(x) \coloneqq 1$ if $x \geq 0$. We use PRC to denote the family of all $n$-perceptrons, for all $n > 0$.

**Multilayer Perceptron (MLP)** An *$n$-feature multilayer perceptron (a.k.a. a neural network)* $\mathcal{N}$, or $n$-MLP for short, is defined by a tuple $\mathcal{N} = (\mathbf{W}^1, \ldots, \mathbf{W}^k, \mathbf{b}^1, \ldots, \mathbf{b}^k, f^1, \ldots, f^k)$, where $k > 0$ is the number of layers of $\mathcal{N}$. Moreover, for each layer $i$ with $1 \leq i \leq k$, assuming $d_i$ denotes the number of neurons on the $i$-th layer, and $d_0 = n$

is the size of the input of $\mathcal{N}$:

- $\mathbf{W}^i \in \mathbb{Q}^{d_{i-1} \times d_i}$ is the $i$-th layer weight matrix of $\mathcal{N}$;
- $\mathbf{b}^i \in \mathbb{Q}^{d_i}$ is the $i$-th layer bias vector of $\mathcal{N}$;
- $f^i \colon \mathbb{Q}^{d_i} \to \mathbb{Q}^{d_i}$ is the $i$-th layer ($d_i$-dimensional) activation function of $\mathcal{N}$.

Since we deal with *binary* classifiers, here $d_k = 1$. We assume that the activation function of the non-output neurons (i.e., the function $f^i$, with $1 \leq i \leq k-1$) is the ReLU function[1] $relu(x) \coloneqq max(0, x)$, whereas the activation function $f^k$ of the single output neuron is the Heaviside step function (see, e.g., Barceló et al.; Marques-Silva 2020b; 2022).

Given an $n$-instance $\mathbf{x}$, we inductively define

$$\mathbf{h}^i \coloneqq f^i(\mathbf{h}^{i-1}\mathbf{W}^i + \mathbf{b}^i), \quad \text{for each } i, \text{ with } 1 \leq i \leq k,$$

where $\mathbf{h}^0 \coloneqq \mathbf{x}$. Then, $\mathcal{N}$ classifies an $n$-instance $\mathbf{x}$ as $c$, denoted by $\mathcal{N}(\mathbf{x}) = c$, iff $\mathbf{h}^k = c$. We use MLP to denote the family of all $n$-MLPs, for all $n > 0$.

We point out that, in this paper, we do not deal with the task of training classifiers. Instead, we are interested in explaining the behavior of (already learned) classifiers—explanations in this setting are often called "post-hoc" explanations. For this reason, the classifiers will be assumed to be given as input with all the (already trained) parameters characterizing them.

**Computational Complexity** We briefly recall the complexity classes that we encounter. PTIME and L are the classes of all decision problems that can be decided in polynomial time and logarithmic space, respectively, by a deterministic Turing machine (the space constraint is over the work tape). NP is the class of all decision problems that can be decided in polynomial time by a nondeterministic Turing machine. The class $\Sigma_2^{\mathrm{P}}$ is the class of all decision problems that can be decided in polynomial time by a nondeterministic Turing machine equipped with an oracle for a problem in NP. co-NP and $\Pi_2^{\mathrm{P}}$ are the complement classes of NP and $\Sigma_2^{\mathrm{P}}$, respectively, where 'yes' and 'no' answers are interchanged. We recall that L and PTIME are closed under complement. The inclusion relationships (which are all currently believed to be strict) between the above complexity classes are: $\mathrm{L} \subseteq \mathrm{PTIME} \subseteq \mathrm{NP}, \text{co-NP} \subseteq \Sigma_2^{\mathrm{P}}, \Pi_2^{\mathrm{P}}$.

We refer the reader to any textbook on the topic, such as (Arora and Barak 2009), for a broader introduction.

**Explanation Language** In this work, we consider the notion of a *global sufficient reason* as a way to explain classifiers' behavior. Intuitively, for a classifier and a class of interest, the idea is to provide a condition that whenever satisfied (by any instance) ensures the classifier assigns the class; more details will be given in Section 3. To express global sufficient reasons, we use the logical language of conditions proposed in (Calautti, Malizia, and Molinaro 2025) to express global *necessary* reasons; we will discuss the reasons behind our choice after defining the language.

Let $\mathcal{L}[n]$ denote the set of all expressions, called *conditions*, of the form $\bigwedge_{i=1}^{m} \ell_i$, where $m \geq 0$, each $\ell_i$ is a *literal* of the form $t \Diamond t'$, with $\Diamond \in \{=, \neq\}$, and $t, t'$ are *terms* from the set $\{0, 1\} \cup \{v_i \mid 1 \leq i \leq n\}$, where each $v_i$ is a Boolean variable, i.e., over the values $\{0, 1\}$, associated to the $i$-th feature. Obviously, a literal is a condition (where $m = 1$). Notice that a condition can be empty (i.e., $m = 0$); we use $\top$ to denote such a condition. Given an $n$-instance $\mathbf{x}$ and a condition $\phi \in \mathcal{L}[n]$, let $\phi[\mathbf{x}]$ denote the condition obtained from $\phi$ by replacing every $v_i$ in $\phi$ with $\mathbf{x}[i]$. We say that $\mathbf{x}$ *satisfies* $\phi$, denoted by $\mathbf{x} \models \phi$, iff $\phi[\mathbf{x}]$ is true under the usual semantics of comparison operators and Boolean logical connectives—in such a case, we also say that $\mathbf{x}$ is a *model* of $\phi$. When $\phi = \top$, every $n$-instance trivially satisfies $\phi$. We further define $[\![\phi]\!] = \{\mathbf{x} \in \{0, 1\}^n \mid \mathbf{x} \models \phi\}$, i.e., the set of all $n$-instances satisfying $\phi$. We say that a condition $\phi \in \mathcal{L}[n]$ is *consistent* if $[\![\phi]\!] \neq \emptyset$, otherwise it is *inconsistent*. Moreover, for two conditions $\phi, \psi \in \mathcal{L}[n]$, we say that $\phi$ *entails* $\psi$, denoted as $\phi \models \psi$, iff $[\![\phi]\!] \subseteq [\![\psi]\!]$. Finally, we write $\phi \equiv \psi$ iff $\phi \models \psi$ and $\psi \models \phi$. Notice that any condition has $O(n^2)$ literals.

As thoroughly discussed in (Calautti, Malizia, and Molinaro 2025), the language $\mathcal{L}[n]$, which allows to encode conjunctions of (in)equalities, has been designed to go beyond simple conjunctions of feature-value pairs, allowing to support a set of logical constructs that help better explain classifiers' behaviour without hindering interpretability. For example, conditions can express *relationships* among features by means of (in)equalities between feature variables, which is deemed as an important feature for explaining classifiers—e.g., $v_3 = v_4$ in Example 1 expresses that debt status equals credit history, which cannot be expressed as a simple conjunction of feature-value pairs. Indeed, (in)equalities allow for a controlled form of disjunction, such as specifying alternatives between values assigned to features—e.g., $v_3 = v_4$ expresses the two alternatives $v_3 = v_4 = 0$ or $v_3 = v_4 = 1$.

## 3 Global Sufficient Reasons

In this section, we present the notion of a *global sufficient reason* as a way to explain classifiers' behavior. Intuitively, for a classifier and a class of interest, the idea is to provide a condition that whenever satisfied (by any instance) ensures the classifier assigns the class. Additionally, in order to identify the most informative global sufficient reasons, we will be interested in "maximal" ones, as defined later on. This section also introduces the (decision) problems whose complexity will be analyzed in the rest of the paper.

As mentioned, for a classifier $\mathcal{M}$, and a class $c \in \{0, 1\}$, a global sufficient reason is a condition that, when it is satisfied by an instance, the latter is classified with $c$ by $\mathcal{M}$.

**Definition 1** (Global Sufficient Reasons)**.** Let $\mathcal{M}$ be an $n$-feature classifier, and let $c \in \{0, 1\}$ be a class. A *global sufficient reason* for $c$ w.r.t. $\mathcal{M}$ is a condition $\phi \in \mathcal{L}[n]$ where

$$\forall \mathbf{x} \in \{0, 1\}^n, \ (\mathbf{x} \models \phi) \rightarrow (\mathcal{M}(\mathbf{x}) = c)\,.$$

For an $n$-feature classifier $\mathcal{M}$ and a class $c \in \{0, 1\}$, with an abuse of notation, we let $[\![\mathcal{M}, c]\!]$ be the set of all $n$-instances $\mathbf{x}$ such that $\mathcal{M}(\mathbf{x}) = c$. Clearly, a condition $\phi$ is a global sufficient reason for $c$ w.r.t. $\mathcal{M}$ iff $[\![\phi]\!] \subseteq [\![\mathcal{M}, c]\!]$.

Different global sufficient reasons might convey different amounts of information. As an example, an inconsistent

[1] As for the impact of this choice on our results, our upper bounds hold as far as evaluating the MLP over an instance is feasible in polynomial time, while the lower bounds immediately hold for many other activation functions that simply generalize ReLUs.

condition $\phi$, i.e., $[\![\phi]\!] = \emptyset$, is always a global sufficient reason, but it does not provide useful information. In this regard, we point out that every global sufficient reason $\phi$ for a class $c$ w.r.t. a classifier $\mathcal{M}$ "under-approximates" the assignment of $c$ by $\mathcal{M}$ in that $[\![\phi]\!] \subseteq [\![\mathcal{M}, c]\!]$. Thus, a criterion to identify the most informative global sufficient reasons should select the ones for which such under-approximation is as large as possible. Such most informative global sufficient reasons are the "maximal" $\phi$ such that $[\![\phi]\!] \subseteq [\![\mathcal{M}, c]\!]$. Formally, maximality is defined w.r.t. an arbitrary preorder $\preccurlyeq$, i.e., a reflexive and transitive binary relation over $\mathcal{L}[n]$; $\phi \prec \phi'$ denotes that $\phi \preccurlyeq \phi'$ and $\phi' \not\preccurlyeq \phi$. Then, maximal global sufficient reasons are naturally defined as follows.

**Definition 2** (Maximal Global Sufficient Reason)**.** Let $\mathcal{M}$ be an $n$-feature classifier, and let $c \in \{0, 1\}$ be a class. A global sufficient reason $\phi \in \mathcal{L}[n]$ for $c$ w.r.t. $\mathcal{M}$ is $\preccurlyeq$*-maximal* iff there is no global sufficient reason $\phi' \in \mathcal{L}[n]$ for $c$ w.r.t. $\mathcal{M}$ such that $\phi \prec \phi'$.

We consider two concrete common preorders (see, e.g., Cooper and Marques-Silva 2023), which we use to compare conditions w.r.t. their models:

- Model cardinality ($\leq$): Given two conditions $\phi$ and $\phi'$, we write $\phi \leq \phi'$ iff $|[\![\phi]\!]| \leq |[\![\phi']\!]|$.
- Model subset ($\subseteq$): Given two conditions $\phi$ and $\phi'$, we write $\phi \subseteq \phi'$ iff $[\![\phi]\!] \subseteq [\![\phi']\!]$ (or, equivalently, $\phi \models \phi'$).

As a simple example, for the two conditions $\phi := (v_1 = 1 \wedge v_2 = v_3)$ and $\phi' := (v_1 = 1)$ over $\mathcal{L}[3]$, we have that both $\phi \leq \phi'$ and $\phi \subseteq \phi'$ hold. Obviously, $\phi'$ makes a more general statement than $\phi$ by not requiring $v_2$ and $v_3$ to assume the same value, and we consider such statements more informative (as long as they are global sufficient reasons).

As customary in complexity analysis, we focus on decision problems: for a family $\mathsf{C} \in \{\mathsf{PRC}, \mathsf{BDD}, \mathsf{MLP}\}$ of classifiers, and a preorder $\preccurlyeq \in \{\leq, \subseteq\}$, we study the following problem:

| | |
|---|---|
| Problem : | IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$ |
| Input : | An $n$-feature classifier $\mathcal{M} \in \mathsf{C}$, a class $c \in \{0, 1\}$, and a condition $\phi \in \mathcal{L}[n]$. |
| Question : | Is $\phi$ a $\preccurlyeq$-maximal global sufficient reason for $c$ w.r.t. $\mathcal{M}$? |

As we will see in the following, to study the complexity of the problem above, we will also need to focus on the complexity of deciding whether a condition is a global sufficient reason (i.e., not necessarily maximal) for a class $c$ w.r.t. a classifier $\mathcal{M}$. We hence define the following problem (notice how this problem is not parametric in the preorder):

| | |
|---|---|
| Problem : | IsSufficient$[\mathsf{C}]$ |
| Input : | An $n$-feature classifier $\mathcal{M} \in \mathsf{C}$, a class $c \in \{0, 1\}$, and a condition $\phi \in \mathcal{L}[n]$. |
| Question : | Is $\phi$ a global sufficient reason for $c$ w.r.t. $\mathcal{M}$? |

**Remarks:** In the rest of the paper, to streamline the presentation, we implicitly assume, unless specified otherwise, that classifiers are over $n > 0$ features, instances and classes are from $\{0, 1\}^n$ and $\{0, 1\}$, respectively, and conditions and literals are from $\mathcal{L}[n]$. Finally, if $\phi$ is a global sufficient reason for a class $c$ w.r.t. to a classifier $\mathcal{M}$, and $\mathcal{M}$ and $c$ are clear from the context, we may refer to $\phi$ simply as a global sufficient reason, without mentioning $\mathcal{M}$ and $c$.

## 4 Complexity of Global Sufficient Reasons

In this section, we study the complexity of IsSufficient$[\mathsf{C}]$ and IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$, for each family of classifiers $\mathsf{C} \in \{\mathsf{PRC}, \mathsf{BDD}, \mathsf{MLP}\}$, and preorder $\preccurlyeq \in \{\leq, \subseteq\}$. A summary of the complexity results obtained in this paper is reported in Table 1. *Full proofs of all results can be found in the supplementary material*.

We start by carrying out some observations on the problem IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$, for $\mathsf{C} \in \{\mathsf{PRC}, \mathsf{BDD}, \mathsf{MLP}\}$, and $\preccurlyeq \in \{\subseteq, \leq\}$. In particular, we observe that the complement of IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$, i.e., the problem of deciding if a given condition $\phi$ is *not* a $\preccurlyeq$-maximal global sufficient reason for a class $c$ w.r.t. a classifier $\mathcal{M}$, can be easily decided by the following nondeterministic procedure (cf. Algorithm 1): check that $\phi$ is not a global sufficient reason, in which case halt and accept, otherwise, guess another condition $\psi$, and check that (i) $\psi$ is a global sufficient reason, and (ii) $\phi \prec \psi$.

With $n$ features, the size of any condition $\psi \in \mathcal{L}[n]$ is polynomial w.r.t. $n$, and thus the algorithm (which is nondeterministic) runs in polynomial time, assuming it has access to oracles for the following two decision problems: (i) decide whether a condition is a global sufficient reason (i.e., the problem IsSufficient$[\mathsf{C}]$), and (ii) decide, for two given conditions $\phi$ and $\psi$, whether $\phi \prec \psi$; the latter is formally defined as follows, for $\preccurlyeq \in \{\subseteq, \leq\}$:

| | |
|---|---|
| Problem : | Precedence$[\preccurlyeq]$ |
| Input : | Two conditions $\phi, \psi$. |
| Question : | Is $\phi \prec \psi$? |

Thus, for the upper bounds of IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$, we can rely on Algorithm 1, whenever the complexity of IsSufficient$[\mathsf{C}]$ and Precedence$[\preccurlyeq]$ is known.

Notice that the complexity of IsSufficient$[\mathsf{C}]$ depends on the specific family of classifiers considered, whereas the complexity of Precedence$[\preccurlyeq]$ does not. We hence focus first on the latter problem, and we will study the former,

| IsSufficient | | | IsMaxSufficient | | | $\preccurlyeq$ |
|---|---|---|---|---|---|---|
| PRC | BDD | MLP | PRC | BDD | MLP | |
| in L | co-NP | co-NP | in L | in $\Pi_2^{\mathrm{P}}$ | in $\Pi_2^{\mathrm{P}}$ and (co-)NP-hard | $\leq$ |
| | | | in co-NP | $\Pi_2^{\mathrm{P}}$ | $\Pi_2^{\mathrm{P}}$ | $\subseteq$ |

Table 1: Summary of the complexity of IsSufficient$[\mathsf{C}]$ and of IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$, for each family of classifiers $\mathsf{C} \in \{\mathsf{PRC}, \mathsf{BDD}, \mathsf{MLP}\}$, and preorder $\preccurlyeq \in \{\leq, \subseteq\}$. All non-"in" entries are completeness results.

**Algorithm 1:** A generic algorithm deciding whether a condition is *not* a $\preccurlyeq$-maximal global sufficient reason

**Input:** An $n$-feature classifier $\mathcal{M}$, a class $c \in \{0,1\}$, and a condition $\phi \in \mathcal{L}[n]$
**Output:** accept, if $\phi$ is *not* a $\preccurlyeq$-maximal global sufficient reason for $c$ w.r.t. $\mathcal{M}$; reject, otherwise

**Procedure** `NotMaxSufficient` $(\mathcal{M}, c, \phi)$**:**
1 **if** $\phi$ *is not a global sufficient reason for* $c$ *w.r.t.* $\mathcal{M}$ **then**
2 **return** accept
3 $\psi \leftarrow$ **guess** *a condition from* $\mathcal{L}[n]$
4 **if** $\psi$ *is a global sufficient reason and* $\phi \prec \psi$ **then**
5 **return** accept
6 **return** reject

together with IsMaxSufficient$[\mathsf{C}, \preccurlyeq]$ in the next sections, where we will consider each family of classifiers in turn.

## 4.1 The Complexity of PRECEDENCE$[\preccurlyeq]$

The goal of this section is to prove the following result:

**Theorem 3.** *For* $\preccurlyeq \in \{\subseteq, \leq\}$, PRECEDENCE$[\preccurlyeq]$ *is in* L.

To prove the above theorem, we need to introduce some auxiliary notions and results. We start by streamlining the treatment of conditions by observing that the order of the literals in a condition $\phi$ does not change its models, and multiple occurrences of the same literal in $\phi$ are equivalent to a single occurrence. Thus, with a slight abuse of notation, in the rest, we may treat a condition $\phi$ also as the *set* of its literals, and a set of literals $S$ may be treated as the condition conjoining all literals in $S$. As a consequence, the empty condition and the empty set denote the same object.

For a condition $\phi$, we define the *closure* of $\phi$, denoted as $\mathsf{cl}(\phi)$, as the set of literals entailed by $\phi$, i.e., $\mathsf{cl}(\phi) = \{\ell \mid \ell \text{ is a literal and } \phi \models \ell\}$. Clearly, $\phi \subseteq \mathsf{cl}(\phi)$. Furthermore, it can be easily verified that, for every condition $\phi$, $[\![\phi]\!] = [\![\mathsf{cl}(\phi)]\!]$. Thus, $\mathsf{cl}(\phi)$ is just another syntactical (more explicit) way of writing $\phi$ without altering its semantics, that is, its set of models. To prove Theorem 3, we need first to establish the following relationship between conditions and their closure:

**Lemma 4.** *Let* $\phi$ *and* $\psi$ *be conditions. Then,* $[\![\phi]\!] \subseteq [\![\psi]\!]$ *(resp.,* $[\![\phi]\!] \subsetneq [\![\psi]\!]$*) iff* $\mathsf{cl}(\psi) \subseteq \mathsf{cl}(\phi)$ *(resp.,* $\mathsf{cl}(\psi) \subsetneq \mathsf{cl}(\phi)$*).*

Thus, the closure of a condition $\phi$ can be seen as the most "explicit" way of defining the models of $\phi$, which means that $\mathsf{cl}(\phi)$ contains many redundant literals.

The second technical result we need, in order to establish Theorem 3, is related to yet another alternative way of writing conditions. If the closure of a condition $\phi$ is the most "explicit" way of defining its models, we now introduce a certain canonical form in which a condition $\phi$ has no redundant literals. We start by defining this canonical form, and then show that every condition can be easily converted (i.e., in logspace) to an equivalent condition in canonical form.

**Definition 5** (Canonical Form)**.** Let $\phi$ be a condition over the Boolean variables $v_1, \ldots, v_n$. We say that $\phi$ is in *canonical form* if it is of the form:

$$\phi_{0/1} \wedge \phi^1_{=} \wedge \cdots \wedge \phi^m_{=} \wedge \phi_{\neq},$$

where $m \geq 0$, and

1. The conditions $\phi_{0/1}, \phi^1_{=}, \ldots, \phi^m_{=}$ share no variables.
2. $\phi_{0/1}$ is a condition of literals of the form $v_i = b$, for $b \in \{0,1\}$, and such that no literals in $\phi_{0/1}$ share a variable.
3. There are $m$ variables $v_{i_1}, \ldots, v_{i_m}$, with $i_1 < i_2 < \cdots < i_m$ such that:
   (a) for each $1 \leq j \leq m$, $\phi^j_{=}$ is a condition of literals of the form $v_{i_j} = v_\ell$, with $i_j < \ell$, and such that any two literals in $\phi^j_{=}$ only share the variable $v_{i_j}$.
   (b) $\phi_{\neq}$ is a condition of literals of the form $v_{i_j} \neq v_{i_\ell}$ with $i_j < i_\ell$, and such that no literals in $\phi_{\neq}$ share a variable.

Intuitively, a condition in canonical form states explicitly, but in a minimal way, which features are necessarily 0 or 1 in its models ($\phi_{0/1}$) , which features must have the same value ($\phi^i_{=}$, for $1 \leq i \leq m$), and which groups of features need to take different values ($\phi_{\neq}$). We can prove that:

**Lemma 6.** *Let* $\phi$ *be a consistent condition. There exists a condition in canonical form, denoted* $\mathsf{can}(\phi)$*, such that* $\phi \equiv \mathsf{can}(\phi)$*, and* $\mathsf{can}(\phi)$ *can be constructed in logarithmic space w.r.t. the size of* $\phi$*.*

The proof of the above lemma relies on Lemma 4. With Lemma 6 in place, we can now establish the key result related to the complexity of PRECEDENCE$[\preccurlyeq]$.

**Lemma 7.** *The following hold:*

1. *For a condition* $\phi$*, the number* $|[\![\phi]\!]|$ *of models of* $\phi$ *can be computed in logarithmic space.*
2. *For two conditions* $\phi$ *and* $\psi$*, deciding whether* $[\![\phi]\!] \subsetneq [\![\psi]\!]$ *and* $|[\![\phi]\!]| < |[\![\psi]\!]|$ *is in* L.

Item 1 of the above lemma is proven by showing that after converting a condition $\phi$ to its canonical form $\mathsf{can}(\phi)$ in logspace (cf. Lemma 6), then the number of models of $\phi$ is $2^{i-j}$, where $i$ is the number of conditions of the form $\phi^k_{=}$ in $\mathsf{can}(\phi)$ and $j$ is the number of literals in the condition $\phi_{\neq}$ in $\mathsf{can}(\phi)$. The intuition behind this result is as follows: the canonical form of a condition can be seen as a graph whose vertices represent the classes of equivalence between the variables, and whose edges connect two vertices when the values of the variables in those two classes of equivalence must be assigned opposite Boolean values. Notice that in this graph paths have length at most 1, i.e., there are no three distinct vertices $a, b, c$ such that $a$ is connected to $b$ and $b$ is connected to $c$. In this graph, the value $i - j$ (see above) is simply the number of connected components excluding the classes of equivalence assigning fixed values 0 or 1 to the variables. Regarding item 2, the complexity of testing $|[\![\phi]\!]| < |[\![\psi]\!]|$ follows from item 1 of the same lemma, while the complexity of checking $[\![\phi]\!] \subsetneq [\![\psi]\!]$ is shown by exploiting the following result by Calautti, Malizia, and Molinaro (2025).

**Theorem 8** (Calautti, Malizia, and Molinaro 2025)**.** *Let* $\phi$ *be a condition, and let* $\ell$ *be a literal. Deciding whether* $\phi \models \ell$ *(or* $\phi \not\models \ell$*) is in* L.

Theorem 3 immediately follows from Lemma 7.

### 4.2 The Case of MLPs

We start our complexity analysis by considering Multilayer perceptrons as the classifiers of interest. In particular, we are interested in studying the complexity of the problems IsSufficient[MLP] and IsMaxSufficient[MLP, $\preccurlyeq$]. As already discussed, we first focus on IsSufficient[MLP], since its complexity will allow us to establish the complexity of our main problem IsMaxSufficient[MLP, $\preccurlyeq$].

**Theorem 9.** IsSufficient[MLP] *is* co-NP-complete. *The hardness holds even if the input condition is empty.*

*Proof. (Sketch)* The upper bound follows from a simple nondeterministic, polynomial-time procedure that decides the complement of IsSufficient[MLP], i.e., checks if an instance $\mathbf{x}$ exists such that $\mathbf{x} \models \phi$ but $\mathcal{N}(x) \neq c$. The procedure guesses an instance $\mathbf{x}$, and then verifies that $\mathbf{x} \models \phi$ and $\mathcal{N}(x) \neq c$. All such steps can be carried out in polynomial time. The interesting result is the hardness, which is shown by means of a polynomial-time reduction from the co-NP-complete problem Tautology, which, given a Boolean formula $\gamma$, asks whether $\gamma$ is a tautology, i.e., whether every truth assignment of the Boolean variables of $\gamma$ satisfies $\gamma$. Given a formula $\gamma$ over $n$ Boolean variables $x_1, \ldots, x_n$, the reduction constructs the class $c = 1$, the empty condition $\phi = \top$, and an $n$-MLP $\mathcal{N}_\gamma$ that simulates $\gamma$, i.e., for each instance $\mathbf{x} \in \{0,1\}^n$, $\mathbf{x}$ satisfies[2] $\gamma$ iff $\mathcal{N}_\gamma(\mathbf{x}) = 1$. Since $\phi$ is empty, $\phi$ is a global sufficient reason iff $\mathcal{N}_\gamma(\mathbf{x}) = 1$, for all $\mathbf{x} \in \{0,1\}^n$. Since the latter is equivalent to verifying that $\gamma$ is a tautology, the claim follows. □

With the above result in place, we can now discuss the complexity of IsMaxSufficient[MLP, $\preccurlyeq$], for $\preccurlyeq \in \{\subseteq, \leq\}$. We start by focusing on $\subseteq$-maximal global sufficient reasons.

**Theorem 10.** IsMaxSufficient[MLP,$\subseteq$] *is* $\Pi_2^{\mathrm{P}}$-complete.

*Proof. (Sketch)* The upper bound immediately follows from the fact that Algorithm 1 is a nondeterministic algorithm that decides the complement of IsMaxSufficient[MLP, $\subseteq$] in polynomial time, with access to oracles for the problems Precedence[$\subseteq$] and IsSufficient[MLP], and from the fact that Precedence[$\subseteq$] is in L (cf. Theorem 3) and IsSufficient[MLP] is in co-NP (cf. Theorem 9).

The lower bound is more involved, and it is shown by reducing the $\Sigma_2^{\mathrm{P}}$-complete problem 2QBF to the complement of IsMaxSufficient[MLP, $\subseteq$], in polynomial time. The problem 2QBF takes as input a quantified formula $\Gamma = (\exists x_1, \ldots, x_n)(\forall y_1, \ldots, y_m)\gamma$, where $\gamma$ is a formula in 3DNF, and asks whether $\Gamma$ is valid, i.e., whether there is an assignment of the variables $x_1, \ldots, x_n$ such that, for all assignments of the variables $y_1, \ldots, y_m$, the formula $\gamma$ becomes true. The reduction considers $k = n + m + 1$ features, which means that conditions are over the $k$ Boolean variables $v_1, \ldots, v_k$. Intuitively, the first $n$ Boolean variables correspond to the variables $x_1, \ldots, x_n$, while the next $m$ Boolean variables correspond to the variables $y_1, \ldots, y_m$, while the last variable is new. With an abuse of notation, in conditions we will use $x_1, \ldots, x_n$ and $y_1, \ldots, y_m$ to refer to the corresponding Boolean variables $v_1, \ldots, v_n$, and $v_{n+1}, \ldots, v_{n+m}$, of the condition, while we will use $s$ for the Boolean variable $v_k$. The reduction constructs the condition $\phi = (s = 0) \wedge \bigwedge_{1 \leq i \leq n} (x_i = 0)$, whose models are all instances that assign 0 to $s$ and to all $x_i$, for $1 \leq i \leq n$. Then, the reduction constructs the class $c = 1$, and the MLP $\mathcal{N}$ such that $\mathcal{N}(\mathbf{x}) = 1$ if either $\mathbf{x}$ assigns 0 to $s$ and to each $x_i$, for $1 \leq i \leq n$, or $\mathbf{x}$ assigns 1 to $s$, i.e., $\mathbf{x}[k] = 1$, and $x_1 \mapsto \mathbf{x}[1], \ldots, x_n \mapsto \mathbf{x}[n], y_1 \mapsto \mathbf{x}[n+1], \ldots, y_m \mapsto \mathbf{x}[n+m]$ satisfies $\gamma$. We then prove that $\Gamma$ is valid iff $\phi$ is not a $\subseteq$-maximal global sufficient reason for $c$ w.r.t. $\mathcal{N}$. □

We now analyze IsMaxSufficient[MLP, $\leq$]. We point out that the complexity analysis of this problem is much more involved than the one for IsMaxSufficient[MLP, $\subseteq$]. In particular, although we can easily obtain a $\Pi_2^{\mathrm{P}}$ upper bound by exploiting our generic procedure of Algorithm 1, as done for Theorem 10, we were not able, despite our efforts, to provide a matching lower bound. Moreover, the reduction employed for proving that IsMaxSufficient[MLP, $\subseteq$] is $\Pi_2^{\mathrm{P}}$-hard heavily relies on the fact that the preorder is $\subseteq$, and thus it immediately breaks when considering $\leq$ as the preorder. We thus leave this as an open problem.

Nonetheless, we can still prove that the problem IsMaxSufficient[MLP, $\leq$] is both NP-hard and co-NP-hard. This means that IsMaxSufficient[MLP, $\leq$] is neither in NP, nor in co-NP, unless NP = co-NP, and thus the problem is highly intractable.

**Theorem 11.** IsMaxSufficient[MLP, $\leq$] *is in* $\Pi_2^{\mathrm{P}}$ *and both* NP-hard *and* co-NP-hard.

*Proof. (Sketch)* The upper bound follows from Algorithm 1 deciding the complement of IsMaxSufficient[MLP, $\leq$] and the complexity of Precedence[$\leq$] (cf. Theorem 3) and IsSufficient[MLP] (cf. Theorem 9). The NP-hardness is shown via a reduction from the classical NP-complete problem Sat, asking if a given 3CNF formula is satisfiable, while the co-NP-hardness is shown via a reduction from the co-NP-complete problem UnSat, asking whether a given 3CNF formula is unsatisfiable. □

### 4.3 The case of BDDs

We now analyze BDDs, and focus first on the complexity of deciding whether a given condition is a global sufficient reason. As we will see, the complexity of this problem does not decrease w.r.t. to that for MLPs, i.e., it remains co-NP-complete. This is somehow surprising, since BDDs are usually deemed to be easier to interpret than general neural networks.

**Theorem 12.** IsSufficient[BDD] *is* co-NP-complete.

*Proof. (Sketch)* The upper bound is shown via a simple guess and check procedure that runs in polynomial time and decides the complement of IsSufficient[BDD]. We now discuss the high level idea of how the lower bound is obtained. We show that IsSufficient[BDD] is co-NP-hard

[2]Here, when we say that $\mathbf{x}$ satisfies $\gamma$, we mean that the truth assignment $x_1 \mapsto \mathbf{x}[1], \ldots, x_n \mapsto \mathbf{x}[n]$ makes $\gamma$ true.

via a polynomial-time reduction from the co-NP-complete problem UNSAT, which takes as input a Boolean formula in 3CNF $\gamma$, and asks if $\gamma$ is unsatisfiable. The key to provide the above reduction is to establish an important connection between Boolean formulas in 3CNFs and BDDs. For that, we first need to introduce some auxiliary notions.

A Boolean formula $\gamma$ is *free* if no variable occurs more than once in $\gamma$. Moreover, for a Boolean formula $\gamma$, its *free version*, denoted $\mathsf{free}(\gamma)$, is obtained from $\gamma$ by replacing the $j$-th occurrence of a variable $x$ in $\gamma$ with the fresh variable $x^{(j)}$. We then establish the following connection between Boolean formulas in 3CNF and BDDs.

**Theorem 13.** *Let $\gamma$ be a Boolean formula in 3CNF over the Boolean variables $x_1, \ldots, x_n$, and let $x_1^{(1)}, \ldots, x_1^{(k_1)}, \ldots, x_n^{(1)}, \ldots, x_n^{(k_n)}$ be the Boolean variables of $\mathsf{free}(\gamma)$. There is an $m$-BDD $\mathcal{G}_{\mathsf{free}(\gamma)}$, with $m = k_1 + \cdots + k_n$, such that:*

1. *$\mathcal{G}_{\mathsf{free}(\gamma)}$ can be built in polynomial time in the size of $\gamma$.*
2. *For each $\mathbf{x} \in \{0,1\}^n$, $\mathbf{x}$ satisfies $\gamma$ iff $\mathcal{G}_\gamma(\mathbf{x}') = 1$, where*

$$\mathbf{x}' = (\underbrace{\mathbf{x}[1], \ldots, \mathbf{x}[1]}_{k_1 \text{ times}}, \underbrace{\mathbf{x}[2], \ldots, \mathbf{x}[2]}_{k_2 \text{ times}}, \ldots, \underbrace{\mathbf{x}[n], \ldots, \mathbf{x}[n]}_{k_n \text{ times}}).$$

With the above result in place, we can now provide the reduction. Consider a Boolean formula $\gamma$ in 3CNF, and assume $\gamma$ is over the variables $x_1, \ldots, x_n$. The reduction first constructs the free version $\mathsf{free}(\gamma)$ of $\gamma$, having variables

$$x_1^{(1)}, \ldots, x_1^{(k_1)}, \ldots, x_n^{(1)}, \ldots, x_n^{(k_n)}.$$

Then, the reduction constructs an input to the problem IsSufficient[BDD], i.e., a condition $\phi$, a class $c$, and a BDD $\mathcal{G}$ as follows. The condition $\phi$ is over $\mathcal{L}[m]$, i.e., over $m$ features, where $m = k_1 + k_2 + \cdots + k_n$. Intuitively, each feature corresponds to one of the Boolean variables of $\mathsf{free}(\gamma)$. Moreover, $\phi$ is of the form:

$$\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{k_i} x_i^{(1)} = x_i^{(j)}.$$

That is, $\phi$ forces all the features corresponding to the copies of the same variable of $\gamma$ to take the same value. Finally, the reduction sets $c = 0$, and $\mathcal{G} = \mathcal{G}_{\mathsf{free}(\gamma)}$. It is not difficult to see that the reduction runs in polynomial time. Regarding the correctness, when we assume that $\gamma$ is unsatisfiable, we observe that each $\mathbf{x}' \in \{0,1\}^m$ that satisfies $\phi$ is of the form

$$(\underbrace{a_1, \ldots, a_1}_{k_1 \text{ times}}, \ldots, \underbrace{a_n, \ldots, a_n}_{k_n \text{ times}}).$$

By assumption, the instance $(a_1, \ldots, a_n) \in \{0,1\}^n$ does not satisfy $\gamma$, and thus by Theorem 13, $\mathcal{G}(\mathbf{x}') = 0 = c$ as needed. If we assume that $\phi$ is a global sufficient reason for $c$ w.r.t. $\mathcal{G}$, then for any instance $\mathbf{x} \in \{0,1\}^n$, it must be the case that

$$\mathbf{x}' = (\underbrace{\mathbf{x}[1], \ldots, \mathbf{x}[1]}_{k_1 \text{ times}}, \ldots, \underbrace{\mathbf{x}[n], \ldots, \mathbf{x}[n]}_{k_n \text{ times}}),$$

which satisfies $\phi$, is such that $\mathcal{G}(\mathbf{x}') = c = 0$. By Theorem 13, we conclude that $\mathbf{x}$ does not satisfy $\gamma$, as needed. □

We remark that the hardness result of the above theorem could be used to prove the co-NP-hardness of IsSufficient[MLP] since one could use an MLP to simulate a BDD. However, the reduction we provide for IsSufficient[MLP] has some advantages that we believe make the reduction interesting in its own right: the reduction is much simpler than the one used for the hardness of IsSufficient[BDD], and, more importantly, it shows that the hardness is fully dependent on the family of classifiers we consider. In other words, the reduction for IsSufficient[MLP] only requires that the family of classifiers we consider (in this case MLPs) allows to encode arbitrary Boolean formulas via a polynomial-time conversion. This means that the same reduction can be directly applied, without changes, to prove the co-NP-hardness of IsSufficient[C], for any family C of classifiers with the same property. BDDs do not enjoy this property, and thus the reduction for Theorem 14 is more complex, since it must encode the behaviour of a Boolean formula via the interaction of the input condition and BDD.

Having established the complexity of the problem IsSufficient[BDD], we can finally focus on the complexity of IsMaxSufficient[BDD, $\preccurlyeq$]. We start with the case of $\subseteq$-maximal global sufficient reasons, and show that the complexity does not decrease w.r.t. the case of MLPs.

**Theorem 14.** IsMaxSufficient[BDD,$\subseteq$] *is* $\Pi_2^{\mathrm{P}}$-complete.

*Proof. (Sketch)* The upper bound again follows immediately from Algorithm 1, Theorem 3, and Theorem 12, while the lower bound is shown via a rather involved reduction from the $\Sigma_2^{\mathrm{P}}$-complete problem 2QBF$_\neg$ to the complement of IsMaxSufficient[BDD, $\subseteq$]. The problem 2QBF$_\neg$ asks, given a quantified Boolean formula $\Gamma = (\exists x_1, \ldots, x_n)(\forall y_1, \ldots, y_m)\neg\gamma$, where $\gamma$ is in 3CNF, whether $\Gamma$ is valid, i.e., whether there is a truth assignment of the variables $x_1, \ldots, x_n$ such that, for every truth assignment of the variables $y_1, \ldots, y_m$, the formula $\gamma$ is not satisfied. Moreover, $\gamma$ is assumed to be such that each variable appears in $\gamma$ exactly three times (Garey and Johnson 1979). □

Similarly to the discussion on the hardness of IsSufficient[BDD], we could obtain the $\Pi_2^{\mathrm{P}}$-hardness of IsMaxSufficient[MLP, $\subseteq$] by exploiting the hardness of the above theorem. However, also in this case, the reduction we provide for the $\Pi_2^{\mathrm{P}}$-hardness of IsMaxSufficient[MLP, $\subseteq$] is much simpler, and tells us something on which property the family of classifiers we consider needs to have, for the reduction to work. In this case, the reduction only needs that the family of classifiers allows to encode arbitrary Boolean formulas in DNF via a polynomial-time conversion.

We conclude this section by discussing the problem IsMaxSufficient[BDD, $\leq$]. Similarly to what happens for MLPs, although we can easily obtain a $\Pi_2^{\mathrm{P}}$ upper bound for the problem IsMaxSufficient[BDD, $\leq$], thanks to Algorithm 1, we were not able to obtain matching lower bounds. The reduction we provide for Theorem 14 heavily relies on the preorder $\subseteq$, and thus it immediately breaks when considering the preorder $\leq$. Unfortunately, in the case of

IsMaxSufficient[BDD, $\leq$], we could not even prove an NP-hardness or co-NP-hardness, as even this task seems very challenging, and on the other hand, we were not able to improve the upper bound any further. We thus leave the exact complexity of the problem IsMaxSufficient[BDD, $\leq$] as a challenging open problem.

### 4.4 The Case of Perceptrons

We now focus on the last family of classifiers for this paper: perceptrons. It turns out that the complexity of our problems, for this family of classifiers, is much lower than the previous ones. This highlights that perceptrons are much simpler classifiers than MLPs or BDDs. We start with the problem of checking if a condition is a global sufficient reason.

**Theorem 15.** IsSufficient[PRC] *is in* L.

*Proof. (Sketch)* As the proof of this claim is somehow involved, we discuss the high level strategy. We focus on the complement problem: given a $n$-perceptron $\mathcal{S} = (\mathbf{w}, b)$, a class $c$, and condition $\phi \in \mathcal{L}[n]$, is $\phi$ *not* a global sufficient reason? We prove such a problem in L, which implies that IsSufficient[PRC] in L, as L is closed under complement.

Deciding whether $\phi$ is *not* a global sufficient reason amounts to deciding whether there exists an instance $\mathbf{x}$ such that $\mathbf{x} \models \phi$ and $\mathcal{S}(\mathbf{x}) \neq c$. We show that this test can be done in logspace; here we focus on the case $c = 1$, since the proof for $c = 0$ is analogous. We distinguish two cases, depending on whether $\phi$ is consistent or inconsistent. If $\phi$ is inconsistent, then $\phi$ is always a global sufficient reason for $c$ w.r.t. $\mathcal{S}$. We can easily check if $\phi$ is inconsistent in logarithmic space by checking if $\phi$ entails any contradicting literal, e.g., $\phi \models (0 = 1)$, and if that is the case, we answer "no" (recall we consider the complement of IsSufficient[PRC]). If instead $\phi$ is consistent, we proceed as follows.

Assume $\phi$ is over the Boolean variables $v_1, \ldots, v_n$, and is of the form $\ell_1 \wedge \cdots \wedge \ell_m$, with each $\ell_i$ a literal. Observe that, for $c = 1$, $\mathcal{S}(\mathbf{x}) \neq c$ means that $\mathbf{w} \cdot \mathbf{x} + b < 0$, and checking whether an instance $\mathbf{x} \in \{0, 1\}^n$ exists such that $\mathbf{x} \models \phi$ and $\mathbf{w} \cdot \mathbf{x} + b < 0$ boils down to check whether the following system of (in)equalities, where $\mathbf{v} = (v_1, \ldots, v_n)$ is the vector of the system's unknowns, admits a binary solution:[3]

$$\begin{cases} \mathbf{w} \cdot \mathbf{v} + b < 0; \\ \ell_1; \ldots; \ell_m. \end{cases} \tag{1}$$

We show that the above system can be solved in logarithmic space. This is done in three steps. First, we show that the above system is equivalent to another system $S$ of a more convenient shape, and $S$ can be constructed in logarithmic space; $S$ is built by considering the canonical form of $\phi$. Then, we show that the new system $S$ can be converted, using only logarithmic space, to a single inequality of the form:

$$\mathbf{w}' \cdot \mathbf{v}' + b' < 0$$

where $\mathbf{v}'$ is a new vector of unknowns, such that $S$ admits a binary solution iff $\mathbf{w}' \cdot \mathbf{v}' + b' < 0$ does. Finally, we show that checking whether $\mathbf{w}' \cdot \mathbf{v}' + b' < 0$ admits a binary solution can be done in logspace. By executing the above three steps in sequence, we obtain the desired decision procedure for the complement of IsSufficient[PRC]. Since the resulting procedure is the composition of a fixed number of procedures (three in our case), each requiring logspace, from a well-known result in complexity theory (e.g., see (Arora and Barak 2009)), we know that the overall procedure can be evaluated in logspace, and the claim follows. □

[3]Here, with an abuse of notation, we use $\ell_1; \ldots; \ell_m$ to mean that each literal acts as a constraint of the system.

We now study the complexity of deciding whether a condition $\phi$ is a $\leq$-maximal global sufficient reason. This requires verifying two properties: (i) $\phi$ is a global sufficient reason, and (ii) $\phi$ is $\leq$-maximal. The first check can be performed in L (cf. Theorem 15). Our goal in the following is to show that the $\leq$-maximality test can also be carried out in L.

To this end, we rely on the following procedure. First, we compute the number $k$ of models of $\phi$, which can be done in logarithmic space (see Lemma 7). Next, we compute the number $k_{max}$ of models of a $\leq$-maximal global sufficient reason—we will show that this step is also feasible in logarithmic space. Finally, we check whether $k = k_{max}$.

The key technical challenge is computing the number of models of a $\leq$-maximal global sufficient reason, in logarithmic space. We rely on auxiliary lemmas introduced below to achieve this goal. More specifically, Lemma 16 allows us to restrict attention to *ground* conditions only, i.e., conditions composed only of literals of the form $v_i = 0$ or $v_i = 1$. Then, Lemma 17 provides a property that a ground condition satisfies iff it is a global sufficient reason. This property naturally leads to a greedy procedure for computing the number of models of a ground $\leq$-maximal global sufficient reason and such a procedure requires logarithmic space.

The following lemma establishes that, for every global sufficient reason $\phi$, there exists a *ground* global sufficient reason $\psi$ that has the same number of models as $\phi$. This result allows us to restrict our attention to ground conditions only when computing the number of models of a $\leq$-maximal global sufficient reason. The main idea of the proof is to construct, starting from the canonical form $\mathsf{can}(\phi)$ of $\phi$, a ground condition that preserves both the property of being a global sufficient reason and the number of models. Since $\mathsf{can}(\phi)$ has exactly the same set of models as $\phi$, this construction yields a ground global sufficient reason with the same number of models as $\phi$. The construction of the ground condition involves replacing the subconditions $\phi^i_{=}$ and $\phi_{\neq}$ (cf. Definition 5) of $\mathsf{can}(\phi)$ with ground subconditions.

**Lemma 16.** *Let $\mathcal{S}$ be a perceptron, let $c$ be a class, and let $\phi$ be a condition. If $\phi$ is a global sufficient reason, then there exists a ground condition $\psi$ such that $\psi$ is a global sufficient reason and $|[\![\phi]\!]| = |[\![\psi]\!]|$.*

The following lemma provides a property a ground condition $\phi$ satisfies iff $\phi$ is a global sufficient reason. Notice that the lemma focuses on consistent conditions—inconsistent ones are always global sufficient reasons.

**Lemma 17.** *Let $(\mathbf{w}, b)$ be a perceptron, let $c$ be a class, and let $\phi$ be a ground, consistent condition.*

- *If $c = 0$, then $\phi$ is a global sufficient reason iff*

$$\sum_{\substack{w_i \in \mathbf{w},\\ w_i > 0}} w_i + b - \sum_{\substack{w_i \in \mathbf{w},\\ w_i \leq 0,\\ (v_i = 1) \in \phi}} |w_i| - \sum_{\substack{w_i \in \mathbf{w},\\ w_i > 0,\\ (v_i = 0) \in \phi}} |w_i| < 0.$$

- *If $c = 1$, then $\phi$ is a global sufficient reason iff*

$$\sum_{\substack{w_i \in \mathbf{w},\\ w_i < 0}} w_i + b + \sum_{\substack{w_i \in \mathbf{w},\\ w_i \geq 0,\\ (v_i = 1) \in \phi}} |w_i| + \sum_{\substack{w_i \in \mathbf{w},\\ w_i < 0,\\ (v_i = 0) \in \phi}} |w_i| \geq 0.$$

The characterization of a ground global sufficient reason established in the previous lemma in turn provides a strategy to *build* a ground global sufficient reason: we need to choose weights from $\mathbf{w}$ (each inducing a ground literal of a specific form in the condition we are building) so as to make the inequalities in the lemma hold. Since a ground $\leq$-maximal global sufficient reason has the lowest number of literals, we need to choose a minimum number of weights that make the inequalities in the lemma holds. It is not difficult to see that this can be achieved by iteratively picking the weights in non-increasing order of their absolute value until the inequalities in the lemma hold. This yields a procedure to compute the number $k_{max}$ of models of $\leq$-maximal ground global sufficient reasons in logarithmic space: if $m$ is the number of weights (i.e., literals) that have been picked, then $k_{max} = 2^{n-m}$ when a consistent global sufficient reason exists, $k_{max} = 0$ otherwise. Then, by following the procedure outlined earlier, we obtain the following theorem.

**Theorem 18.** IsMaxSufficient$[\mathsf{PRC}, \leq]$ *is in* L.

Finally, for the $\subseteq$ criterion, we prove membership in co-NP. Determining whether this upper bound is tight—by establishing a corresponding lower bound—or whether it can be lowered remains an open problem.

**Theorem 19.** IsMaxSufficient$[\mathsf{PRC}, \subseteq]$ *is in* co-NP.

*Proof.* The upper bound follows from Algorithm 1 deciding the complement of IsMaxSufficient$[\mathsf{PRC}, \subseteq]$ in nondeterministic polynomial time, with access to oracles for the problems Precedence$[\subseteq]$ and IsSufficient$[\mathsf{PRC}]$, and from Precedence$[\subseteq]$ being in L (cf. Theorem 3) and IsSufficient$[\mathsf{PRC}]$ being in L (cf. Theorem 15). □

## 5 Related Work

Explaining *global* classifiers' decisions has been considered in previous work in different forms.

Ignatiev, Narodytska, and Marques-Silva (2019b) proposed two notions of global explanations: an *absolute explanation* (resp., *counterexample*) for a class $c$ w.r.t. a classifier $\mathcal{M}$ is a subset-minimal set $\mathcal{E}$ of feature-value pairs $(f_i, c_i)$, where no feature occurs twice in $\mathcal{E}$, such that every instance $\mathbf{x}$ matching $\mathcal{E}$ (i.e., $\mathbf{x}[f_i] = c_i$, for every feature $f_i$ in $\mathcal{E}$) is such that $\mathcal{M}(\mathbf{x}) = c$ (resp., $\mathcal{M}(\mathbf{x}) \neq c$). Thus, in the binary setting, absolute explanations can be expressed by conditions of the form $\bigwedge_{(f_i, c_i) \in \mathcal{E}} x_i = c_i$ but aim to minimize (under set-inclusion) the set of literals in the condition.

Izza, Ignatiev, and Marques-Silva (2022) generalized two local notions of explanations, namely *weak abductive explanations* (Cooper and Marques-Silva 2023; Ignatiev, Narodytska, and Marques-Silva 2019a)[4] and *weak contrastive explanations* (Miller 2019; Ignatiev et al. 2020). The generalization is done using so-called *generalized explanation functions*. A generalized explanation function takes as input an instance and returns either 0 or 1, can be parameterized on a selected set of features $\mathcal{Z} \subseteq \mathcal{F}$ (with $\mathcal{F}$ the set of all features), as well as other parameters, and is denoted as $\xi(\mathbf{x}, \mathcal{Z}, \dots)$. Then, a weak abductive explanation is a set of features $\mathcal{Z} \subseteq \mathcal{F}$ such that $\forall \mathbf{x} \in \{0,1\}^n,\ (\xi(\mathbf{x}, \mathcal{Z}, \dots) = 1) \rightarrow (\mathcal{M}(\mathbf{x}) = c)$. A weak contrastive explanation is a set of features $\mathcal{Z} \subseteq \mathcal{F}$ such that $\exists \mathbf{x} \in \{0,1\}^n,\ (\xi(\mathbf{x}, \mathcal{F} \setminus \mathcal{Z}, \dots) = 1) \wedge (\mathcal{M}(\mathbf{x}) \neq c)$. A weak abductive explanation is analogous to our notion of global sufficient reason. Izza, Ignatiev, and Marques-Silva (2022) focus only on decision trees, i.e., a special case of BDDs, using generalized explanation functions that are conjunctions of conditions along a path in a decision tree.

*The main novel contribution of this work, compared to the aforementioned studies, is a complexity analysis for several widely used families of classifiers, an aspect that has so far been missing from the literature.*

Bassan, Amir, and Katz (2024) introduced a notion of "global sufficient reason" which is fundamentally different from ours and from the ones discussed above. Specifically, their notion of a global sufficient reason for a classifier $\mathcal{M}$ is a subset $S$ of features such that, for every instance $\mathbf{x}$, the classification remains unchanged whenever the values of $\mathbf{x}$ on $S$ are kept fixed, irrespective of how the remaining features' values are modified. They then focus on *minimal* global sufficient reason, that is, those with minimum cardinality. Notice that this notion of global sufficient reason does not take any specific class of interest into account, in contrast to our notion. Also, rather than relying on a set of features to provide a global explanation, we use a logical condition to characterize as accurately as possible the family of instances assigned to a given class of interest. Employing logic to characterize conditions that are sufficient (and/or necessary) for classifiers' decisions has been adopted by different works—e.g., see the recent tutorial by Darwiche (2023).

Izza et al. (2024) proposed the notion of an *inflated* explanation, which shares the idea of providing an explanation with a broader applicability, in that an inflated explanation provides a condition that applies to more than a single instance. Even though an inflated explanation is defined for a specific instance, it provides intervals for a subset of features (always including the value of the instance being explained) such that any instance whose feature values fall within these intervals is guaranteed to receive the same classification as $\mathbf{x}$. Izza et al. (2025) further explored the problem of finding the "best" inflated explanations, that is, those that cover as much of the instances space as possible.

Wang et al. (2017) learn a *rule set model*, which is a set of rules, i.e., conjunctions of conditions. Rule set models

[4] These are also referred to as sufficient reasons (Darwiche and Hirth 2020) or PI (prime implicant) explanations (Shih, Choi, and Darwiche 2018).

predict that an observation is in the positive class when at least one of the rules is satisfied. Setzu et al. (2019; 2021) proposed approaches to derive a global explanation from local ones, where both are expressed as decision rules. Rawal and Lakkaraju (2020) use general *recourse rules*, which are if-then statements saying which changes should be applied under certain conditions to obtain a prediction.

Calautti, Malizia, and Molinaro (2025) have considered the notion of a global *necessary* reason, which is somehow specular to global sufficient reasons considered in this paper, in that a global necessary reason for a class $c$ w.r.t. a classifier $\mathcal{M}$ is a condition that must be satisfied by every instance $\mathbf{x}$ such that $\mathcal{M}(\mathbf{x}) = c$. Calautti, Malizia, and Molinaro (2025) provide a complexity analysis for the same families of classifiers and the same criteria considered in this paper.

Despite the substantial body of work on *local* explanations, their objective fundamentally differs from ours. Local explanations aim to characterize a property of a *single given instance*, whereas global explanations seek to identify a property common to a *set of instances, which are not explicitly provided*. This distinction has important implications, e.g., it can influence the complexity of reasoning tasks, such as determining whether a given expression qualifies as a local or global explanation. In the local case, the relevant instance is available, simplifying verification. In contrast, the absence of such an instance in the global case can significantly increase complexity. A well-known form of local explanation is the concept of a prime implicant (PI). Defined with respect to a specific instance $\mathbf{x}$, a PI is a minimal (under set inclusion) subset $F$ of features such that, if the values of $\mathbf{x}$ are fixed on $F$ while all other features are allowed to vary freely, the classification of $\mathbf{x}$ remains unchanged. The minimality condition ensures that $F$ contains no unnecessary features. In contrast, global sufficient reasons do not depend on a particular instance. In the case of $\subseteq$-maximal global sufficient reasons, set inclusion is interpreted in a maximal sense, aiming to cover as many instances as possible. Among approaches on local explanations, Rudin and Shaposhnik (2023) and Geng, Schleich, and Suciu (2022) proposed notions with some sort of "global consistency", meaning that a local explanation must be coherent with the prediction of a restricted set of instances. Gorji and Rubin (2022) consider *local* sufficient reasons in the presence of constraints that allow only certain instances to be valid.

## 6 Future Work

This paper opens several directions for future research. It would be interesting to extend our study to encompass classifiers operating on continuous and categorical data. This will likely require the adoption of more sophisticated logical languages and a subsequent re-evaluation of the complexity analysis. Future efforts will be directed toward establishing the missing tight results to complete the complexity landscape. Beyond the theoretical realm, we intend to leverage our results to support the development of practical, SAT-based or SMT-based algorithms.

## AI Declaration

The authors have not employed any Generative AI tools.

## Acknowledgments

Cristian Molinaro's work was supported by the Italian Ministry of University and Research (MUR) PRIN 2022 grant 2022XERWK9 "S-PIC4CHU - Semantics-based Provenance, Integrity, and Curation for Consistent, High-quality, and Unbiased data science"— CUP: H53C24000990006.

## References

Arenas, M.; Barceló, P.; Bertossi, L. E.; and Monet, M. 2023. On the complexity of shap-score-based explanations: Tractability via knowledge compilation and non-approximability results. *J. Mach. Learn. Res.* 24:63:1–63:58.

Arora, S., and Barak, B. 2009. *Computational Complexity: A Modern Approach*. Cambridge, UK: Cambridge University Press.

Audemard, G.; Bellart, S.; Bounia, L.; Koriche, F.; Lagniez, J.; and Marquis, P. 2022. Trading complexity for sparsity in random forest explanations. In *Proc. AAAI*, 5461–5469.

Barceló, P.; Monet, M.; Pérez, J.; and Subercaseaux, B. 2020a. Model interpretability through the lens of computational complexity. In *Proc. NeurIPS*.

Barceló, P.; Monet, M.; Pérez, J.; and Subercaseaux, B. 2020b. Model interpretability through the lens of computational complexity. Technical Report arXiv:2010.12265, CoRR.

Bassan, S.; Amir, G.; and Katz, G. 2024. Local vs. global interpretability: A computational complexity perspective. In *Proc. ICML*.

Cabodi, G.; Camurati, P. E.; Marques-Silva, J.; Palena, M.; and Pasini, P. 2024. Optimizing binary decision diagrams for interpretable machine learning classification. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* 43(10):3083–3087.

Calautti, M.; Malizia, E.; and Molinaro, C. 2025. On the complexity of global necessary reasons to explain classification. In *Proc. KR*, 206–217.

Carbonnel, C.; Cooper, M. C.; and Marques-Silva, J. 2023. Tractable explaining of multivariate decision trees. In *Proc. KR*, 127–135.

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

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

Darwiche, A., and Hirth, A. 2020. On the reasons behind decisions. In *Proc. ECAI*, 712–720.

Darwiche, A. 2023. Logic for explainable AI. In *Proc. LICS*, 1–11.

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

Garey, M. R., and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co.

Geng, Z.; Schleich, M.; and Suciu, D. 2022. Computing rule-based explanations by leveraging counterfactuals. *Proc. VLDB Endow.* 16(3):420–432.

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

Hu, H.; Huguet, M.; and Siala, M. 2022. Optimizing binary decision diagrams with maxsat for classification. In *Proc. AAAI*, 3767–3775.

Huang, X.; Izza, Y.; Ignatiev, A.; Cooper, M. C.; Asher, N.; and Marques-Silva, J. 2022. Tractable explanations for d-dnnf classifiers. In *Proc. AAAI*, 5719–5728.

Ignatiev, A.; Narodytska, N.; Asher, N.; and Marques-Silva, J. 2020. From contrastive to abductive explanations and back again. In *Proc. AIxIA*, volume 12414, 335–355.

Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019a. Abduction-based explanations for machine learning models. In *Proc. AAAI*, 1511–1519.

Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019b. On relating explanations and adversarial examples. In *Proc. NeurIPS*, 15857–15867.

Izza, Y.; Ignatiev, A.; Stuckey, P. J.; and Marques-Silva, J. 2024. Delivering inflated explanations. In *Proc. AAAI*, 12744–12753.

Izza, Y.; Ignatiev, A.; Rubin, S.; Marques-Silva, J.; and Stuckey, P. J. 2025. Most general explanations of tree ensembles. In *Proc. IJCAI*, 5463–5471.

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

Marques-Silva, J.; Gerspacher, T.; Cooper, M. C.; Ignatiev, A.; and Narodytska, N. 2020. Explaining naive bayes and other linear classifiers with polynomial time and delay. In *Proc. NeurIPS*.

Marques-Silva, J. 2022. Logic-based explainability in machine learning. In *Proc. RW*, 24–104.

Miller, T. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.* 267:1–38.

Ordyniak, S.; Paesani, G.; and Szeider, S. 2023. The parameterized complexity of finding concise local explanations. In *Proc. IJCAI*, 3312–3320.

Rawal, K., and Lakkaraju, H. 2020. Beyond individualized recourse: Interpretable and interactive summaries of actionable recourses. In *Proc. NeurIPS*.

Rudin, C., and Shaposhnik, Y. 2023. Globally-consistent rule-based summary-explanations for machine learning models: Application to credit-risk evaluation. *J. Mach. Learn. Res.* 24:16:1–16:44.

Setzu, M.; Guidotti, R.; Monreale, A.; and Turini, F. 2019. Global explanations with local scoring. In *Proc. ECML PKDD Workshops*, 159–171.

Setzu, M.; Guidotti, R.; Monreale, A.; Turini, F.; Pedreschi, D.; and Giannotti, F. 2021. GLocalX – From local to global explanations of black box AI models. *Artif. Intell.* 294.

Shih, A.; Choi, A.; and Darwiche, A. 2018. A symbolic approach to explaining bayesian network classifiers. In Lang, J., ed., *Proc. IJCAI*, 5103–5111.

Wang, T.; Rudin, C.; Doshi-Velez, F.; Liu, Y.; Klampfl, E.; and MacNeille, P. 2017. A Bayesian framework for learning rule sets for interpretable classification. *J. Mach. Learn. Res.* 18:70:1–70:37.