Towards Trustable SHAP Scores

Olivier Létoffé[1], **Xuanxiang Huang**[2], **Joao Marques-Silva**[3]

[1]Univ. Toulouse, France
[2]CNRS@CREATE, Singapore
[3]ICREA, Univ. Lleida, Spain
olivier.letoffe@orange.fr, xuanxiang.huang@cnrsatcreate.sg, jpms@icrea.cat

Abstract

SHAP scores represent the proposed use of the well-known Shapley values in eXplainable Artificial Intelligence (XAI). Recent work has shown that the exact computation of SHAP scores can produce unsatisfactory results. Concretely, for some ML models, SHAP scores will mislead with respect to relative feature influence. To address these limitations, recently proposed alternatives exploit different axiomatic aggregations, all of which are defined in terms of abductive explanations. However, the proposed axiomatic aggregations are not Shapley values. This paper investigates how SHAP scores can be modified so as to extend axiomatic aggregations to the case of Shapley values in XAI. More importantly, the proposed new definition of SHAP scores avoids all the known cases where unsatisfactory results have been identified. The paper also characterizes the complexity of computing the novel definition of SHAP scores, highlighting families of classifiers for which computing these scores is tractable. Furthermore, the paper proposes modifications to the existing implementations of SHAP scores. These modifications eliminate some of the known limitations of SHAP scores, and have negligible impact in terms of performance.

Introduction

Shapley values for eXplainable AI (XAI), i.e. SHAP scores (Lundberg and Lee 2017), are arguably among the most widely used explainability methods that target the attribution of (relative) feature importance, as exemplified by the success of the tool SHAP (Molnar 2023; Mishra 2023).[1] Despite the massive popularity of SHAP scores, some works have identified limitations with their use (Young et al. 2019; Kumar et al. 2020; Sundararajan and Najmi 2020; Merrick and Taly 2020; Fryer, Strümke, and Nguyen 2021; Yan and Procaccia 2021; Mothilal et al. 2021; Afchar, Guigue, and Hennequin 2021; Watson et al. 2021; Kumar et al. 2021; Campbell et al. 2022). However, most of these limitations can be attributed to the results obtained with existing tools, and not necessarily with the theoretical foundations of SHAP scores. More recent work (Huang and Marques-Silva 2023; Huang and Marques-Silva 2024) uncovered examples of classifiers where *exact* SHAP scores assign manifestly unsatisfactory importance to features. Namely, features having no influence in a prediction can be assigned more importance than features having the most influence in the prediction. This recent evidence should be perceived as far more problematic, because it reveals apparent limitations with the theoretical foundations of SHAP scores, and not with concrete implementations. Nevertheless, Shapley values are of fundamental importance, not only in game theory (Chalkiadakis, Elkind, and Wooldridge 2012), but also in many other domains, namely because of their intrinsic properties (Shapley 1953). As a result, a natural question is whether the definitions of Shapley values in XAI can be changed, so as to avoid situations where the computed feature importance is problematic.

Contributions. This paper argues that the key issue with SHAP scores is not the use of Shapley values in explainability per se, and shows that the identified shortcomings of SHAP scores can be solely attributed to the characteristic functions used in earlier works (Strumbelj and Kononenko 2010, 2014; Lundberg and Lee 2017; Janzing, Minorics, and Blöbaum 2020; Sundararajan and Najmi 2020; Arenas et al. 2021; Van den Broeck et al. 2021, 2022; Arenas et al. 2023). As noted in the recent past (Janzing, Minorics, and Blöbaum 2020; Sundararajan and Najmi 2020), by changing the characteristic function, one is able to produce different sets of SHAP scores.[2] Motivated by these observations, the paper outlines fundamental properties that characteristic functions ought to exhibit in the context of XAI. Furthermore, the paper proposes several novel characteristic functions, which either respect some or all of the identified properties. In addition, the paper analyzes the impact of the novel characteristic functions on the computational complexity of computing SHAP scores, by building on recent work on the same topic (Van den Broeck et al. 2021; Arenas et al. 2021; Van den Broeck et al. 2022; Arenas et al. 2023). An indirect consequence of our work is that *corrected* SHAP scores can be safely used for feature attribution in XAI, while offering strong guarantees regarding known shortcomings. Furthermore, the results in the paper apply independently of the machine learning (ML) model considered.

[1]See https://github.com/shap/shap.

[2]Unfortunately, this paper argues that past alternative proposals of characteristic functions (Janzing, Minorics, and Blöbaum 2020; Sundararajan and Najmi 2020) also exhibit critical limitations.

Related work. SHAP scores are ubiquitously used in XAI (Lundberg and Lee 2017; Molnar 2023; Mishra 2023). Recent work argues that the existing definitions of (exact) SHAP scores can yield unsatisfactory results (Huang and Marques-Silva 2023; Huang and Marques-Silva 2024). Motivated by these results, different works proposed alternative solutions to the use of SHAP scores (Yu, Ignatiev, and Stuckey 2023; Biradar et al. 2023; Yu et al. 2023; Biradar et al. 2024; Yu et al. 2024). Furthermore, one of these solutions (Biradar et al. 2024) investigates the use of power indices from the field of a priori voting power (Felsenthal and Machover 2015) as a solution for feature importance in XAI, covering several well-known power-indices. However, Shapley values are also at the core of the well-known Shapley-Shubik power index (Shapley and Shubik 1954), one of the best-known power indices, and which is not studied in (Biradar et al. 2024). Thus, an open question is how to extend the recent work on power indices for XAI (Biradar et al. 2024) to the case of the Shapley-Shubik index.

Organization. The paper starts by introducing the notation and definitions used throughout the paper. Afterwards, the paper briefly dissects some of the recently reported shortcomings with SHAP scores (Marques-Silva and Huang 2024; Huang and Marques-Silva 2024; Huang and Marques-Silva 2023). Motivated by those shortcomings, the paper then proposes properties that characteristic functions should exhibit. The paper then proposes several novel characteristic functions, which are shown to correct some or all of the shortcomings of the characteristic functions used in earlier work. Next, the paper studies the complexity of computing SHAP scores given the novel characteristic functions proposed in this paper. The paper also outlines a simple modification to the SHAP tool (Lundberg and Lee 2017), which corrects some of the shortcomings of SHAP scores.

Preliminaries

Classification & regression problems. Let $\mathcal{F} = \{1, \ldots, m\}$ denote a set of features. Each feature $i \in \mathcal{F}$ takes values from a domain $\mathbb{D}_i$. Domains can be categorical or ordinal. If ordinal, domains can be discrete or real-valued. Feature space is defined by $\mathbb{F} = \mathbb{D}_1 \times \mathbb{D}_2 \times \ldots \times \mathbb{D}_m$. Throughout the paper domains are assumed to be discrete-valued.[3] The notation $\mathbf{x} = (x_1, \ldots, x_m)$ denotes an arbitrary point in feature space, where each x_i is a variable taking values from $\mathbb{D}_i$. Moreover, the notation $\mathbf{v} = (v_1, \ldots, v_m)$ represents a specific point in feature space, where each v_i is a constant representing one concrete value from $\mathbb{D}_i$. A classifier maps each point in feature space to a class taken from $\mathcal{K} = \{c_1, c_2, \ldots, c_K\}$. Classes can also be categorical or ordinal. However, and unless otherwise stated, classes are assumed to be ordinal. In the case of regression, each point in feature space is mapped to an ordinal value taken from a set $\mathbb{C}$, e.g. $\mathbb{C}$ could denote $\mathbb{Z}$ or $\mathbb{R}$. Therefore, a classifier $\mathcal{M}_C$ is characterized by a non-constant *classification function* κ that maps feature space $\mathbb{F}$ into the set of classes $\mathcal{K}$, i.e. $\kappa : \mathbb{F} \to \mathcal{K}$. A regression model $\mathcal{M}_R$ is characterized by a non-constant *regression function* ρ that maps feature space $\mathbb{F}$ into the set elements from $\mathbb{C}$, i.e. $\rho : \mathbb{F} \to \mathbb{C}$. A classifier model $\mathcal{M}_C$ is represented by a tuple $(\mathcal{F}, \mathbb{F}, \mathcal{K}, \kappa)$, whereas a regression model $\mathcal{M}_R$ is represented by a tuple $(\mathcal{F}, \mathbb{F}, \mathbb{C}, \rho)$.[4] When viable, we will represent an ML model $\mathcal{M}$ by a tuple $(\mathcal{F}, \mathbb{F}, \mathbb{T}, \tau)$, with $\tau : \mathbb{F} \to \mathbb{T}$, without specifying whether whether $\mathcal{M}$ denotes a classification or a regression model. A *sample* (or instance) denotes a pair $(\mathbf{v}, q)$, where $\mathbf{v} \in \mathbb{F}$ and either $q \in \mathcal{K}$, with $q = \kappa(\mathbf{v})$, or $q \in \mathbb{C}$, with $q = \rho(\mathbf{v})$.

Additional notation. An explanation problem is a tuple $\mathcal{E} = (\mathcal{M}, (\mathbf{v}, q))$, where $\mathcal{M}$ can either be a classification or a regression model, and $(\mathbf{v}, q)$ is a target sample, with $\mathbf{v} \in \mathbb{F}$. (Observe that $q = \kappa(\mathbf{v})$, with $q \in \mathcal{K}$, in the case of a classification model, and $q = \rho(\mathbf{v})$, with $q \in \mathbb{C}$, in the case of a regression model.)

Given $\mathbf{x}, \mathbf{v} \in \mathbb{F}$, and $\mathcal{S} \subseteq \mathcal{F}$, the predicate $\mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}$ is defined as follows:

$$\mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S} \;\; := \;\; \left(\bigwedge\nolimits_{i \in \mathcal{S}} x_i = v_i\right)$$

The set of points for which $\mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}$ is defined by $\Upsilon(\mathcal{S}; \mathbf{v}) = \{\mathbf{x} \in \mathbb{F} \,|\, \mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}\}$.

Distributions, expected value. Throughout the paper, it is assumed a *uniform probability distribution* on each feature, and such that all features are independent. Thus, the probability of an arbitrary point in feature space becomes:

$$\mathbf{P}(\mathbf{x}) := 1/\Pi_{i \in \mathcal{F}} |\mathbb{D}_i| \tag{1}$$

That is, every point in the feature space has the same probability. The *expected value* of an ML model $\tau : \mathbb{F} \to \mathbb{T}$ is denoted by $\mathbf{E}[\tau]$. Furthermore, let $\mathbf{E}[\tau(\mathbf{x}) \,|\, \mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}]$ represent the expected value of τ over points in feature space consistent with the coordinates of $\mathbf{v}$ dictated by $\mathcal{S}$, which is defined as follows:

$$\mathbf{E}[\tau(\mathbf{x}) \,|\, \mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}] := 1/|\Upsilon(\mathcal{S}; \mathbf{v})| \sum\nolimits_{\mathbf{x} \in \Upsilon(\mathcal{S}; \mathbf{v})} \tau(\mathbf{x}) \tag{2}$$

Similarly, we define,

$$\mathbf{P}(\pi(\mathbf{x}) \,|\, \mathbf{x}_\mathcal{S} = \mathbf{v}_\mathcal{S}) := 1/|\Upsilon(\mathcal{S}; \mathbf{v})| \sum\nolimits_{\mathbf{x} \in \Upsilon(\mathcal{S}; \mathbf{v})} \text{ITE}(\pi(\mathbf{x}), 1, 0) \tag{3}$$

where $\pi : \mathbb{F} \to \{0, 1\}$ is some predicate.

Shapley values. Shapley values were proposed in the context of game theory in the early 1950s by L. S. Shapley (Shapley 1953). Shapley values were defined given some set $\mathcal{N}$, and a *characteristic function*, i.e. a real-valued function defined on the subsets of $\mathcal{N}$, $v : 2^\mathcal{N} \to \mathbb{R}$.[5]. It is well-known

[3] The results in the paper can be generalized to continuous-valued features. For real-valued features, the only changes involve the definition of expected value and probability, respectively in (2) and (3).

[4] As shown in the extended version of this paper (Letoffe, Huang, and Marques-Silva 2024a), other ML models can also be represented with similar ideas. This is the case with the use of a softmax layer in neural networks (NNs).

[5] The original formulation also required super-additivity of the characteristic function, but that condition has been relaxed in more recent works (Dubey 1975; Young 1985).

that Shapley values represent the *unique* function that, given $\mathcal{N}$ and υ, respects a number of important axioms. More detail about Shapley values is available in standard references (Shapley 1953; Dubey 1975; Young 1985; Roth 1988). Besides the recent uses in XAI, Shapley values have been used for assigning measures of relative importance in computational social choice (Chalkiadakis, Elkind, and Wooldridge 2012) (including a priori voting power (Shapley and Shubik 1954; Felsenthal and Machover 1998)), measurement of inconsistency in knowledge bases (Hunter and Konieczny 2006, 2010), and intensity of attacks in argumentation frameworks (Amgoud, Ben-Naim, and Vesic 2017).

SHAP scores. In the context of explainability, Shapley values are most often referred to as SHAP scores (Strumbelj and Kononenko 2010, 2014; Lundberg and Lee 2017; Arenas et al. 2021, 2023), and consider a specific characteristic function $\upsilon_e : 2^{\mathcal{F}} \to \mathbb{R}$, which is defined by,

$$\upsilon_e(\mathcal{S};\mathcal{E}) := \mathbf{E}[\tau(\mathbf{x}) \,|\, \mathbf{x}_{\mathcal{S}} = \mathbf{v}_{\mathcal{S}}] \tag{4}$$

Thus, given a set $\mathcal{S}$ of features, $\upsilon_e(\mathcal{S};\mathcal{E})$ represents the *e*xpected value of the classifier over the points of feature space represented by $\Upsilon(\mathcal{S};\mathbf{v})$. The formulation presented in earlier work (Arenas et al. 2021, 2023) allows for different input distributions when computing the average values. For the purposes of this paper, it suffices to consider solely a uniform input distribution, and so the dependency on the input distribution is not accounted for. Independently of the distribution considered, it should be clear that in most cases $\upsilon_e(\emptyset) \neq 0$; this is the case for example with boolean classifiers (Arenas et al. 2021, 2023).

To simplify the notation, the following definitions are used,

$$\Delta_i(\mathcal{S};\mathcal{E},\upsilon) := (\upsilon(\mathcal{S}\cup\{i\}) - \upsilon(\mathcal{S})) \tag{5}$$

$$\varsigma(|\mathcal{S}|) := {}^{|\mathcal{S}|!(|\mathcal{F}|-|\mathcal{S}|-1)!}/_{|\mathcal{F}|!} \tag{6}$$

(Observe that Δ_i is parameterized on $\mathcal{E}$ and υ.)

Finally, let $\mathsf{Sc}_E : \mathcal{F} \to \mathbb{R}$, i.e. the SHAP score for feature i, be defined by,[6].

$$\mathsf{Sc}_E(i;\mathcal{E},\upsilon_e) := \sum\nolimits_{\mathcal{S}\subseteq(\mathcal{F}\setminus\{i\})} \varsigma(|\mathcal{S}|) \times \Delta_i(\mathcal{S};\mathcal{E},\upsilon_e) \tag{7}$$

Given a sample $(\mathbf{v}, q)$, the SHAP score assigned to each feature measures the *contribution* of that feature with respect to the prediction. From earlier work, it is understood that a positive/negative value indicates that the feature can contribute to changing the prediction, whereas a value of 0 indicates no contribution (Strumbelj and Kononenko 2010).

Similarity predicate. Given an ML model and some input $\mathbf{x}$, the output of the ML model is *distinguishable* with respect to the sample $(\mathbf{v}, q)$ if the observed change in the model's output is deemed sufficient; otherwise it is *similar* (or indistinguishable). This is represented by a *similarity* predicate (which will also be viewed as a boolean function) $\sigma : \mathbb{F} \to \{\bot, \top\}$ (where $\bot$ signifies *false*, and $\top$ signifies *true*).[7] Concretely, $\sigma(\mathbf{x};\mathcal{E})$ holds true iff the change in the ML model output is deemed *insufficient* and so no observable difference exists between the ML model's output for $\mathbf{x}$ and $\mathbf{v}$.[8] For regression problems, we write instead σ as the instantiation of a template predicate, i.e. $\sigma(\mathbf{x};\mathcal{E}) = \mathsf{T}\sigma(\mathbf{x};\mathcal{E},\delta)$, where δ is an optional measure of output change, which can be set to 0.[9] Given a change in the input from $\mathbf{v}$ to $\mathbf{x}$, a change in the output is indistinguishable (i.e. the outputs are similar) if,

$$\sigma(\mathbf{x};\mathcal{E}) := \mathsf{T}\sigma(\mathbf{x};\mathcal{E},\delta) := [|\rho(\mathbf{x}) - \rho(\mathbf{v})| \leq \delta]$$

otherwise, it is distinguishable.

For classification problems, similarity is defined to equate with not changing the predicted class. Given a change in the input from $\mathbf{v}$ to $\mathbf{x}$, a change in the output is indistinguishable (i.e. the outputs are similar) if,

$$\sigma(\mathbf{x};\mathcal{E}) := [\kappa(\mathbf{x}) = \kappa(\mathbf{v})]$$

otherwise, it is distinguishable. (As shown in the remainder of this paper, σ allows abstracting away whether the underlying model implements classification or regression.)

It will be helpful to list a few properties of σ. Observe that $\forall(\mathcal{A} \subseteq \mathbb{F}).[\mathbf{E}[\sigma(\mathbf{x};\mathcal{E}) \,|\, \mathbf{x} \in \mathcal{A}] \in [0,1]]$. It is also plain to conclude that for $\mathcal{A},\mathcal{B} \subseteq \mathbb{F}$, with $\mathcal{A} \subseteq \mathcal{B}$, and given $u \in \{0,1\}$, if $\mathbf{E}[\sigma(\mathbf{x};\mathcal{E}) \,|\, \mathbf{x} \in \mathcal{B}] = u$, then $\mathbf{E}[\sigma(\mathbf{x};\mathcal{E}) \,|\, \mathbf{x} \in \mathcal{A}] = u$, A few more properties of σ are apparent. For $\mathcal{A} \subseteq \mathbb{F}$, $u \in \{0,1\}$, $(\mathbf{E}[\sigma(\mathbf{x};\mathcal{E}) \,|\, \mathbf{x} \in \mathcal{A}] = u) \leftrightarrow \forall(\mathbf{x} \in \mathcal{A}).[\sigma(\mathbf{x};\mathcal{E}) = u]$. As a result, it is also the case that $(\mathbf{E}[\sigma(\mathbf{x};\mathcal{E}) \,|\, \mathbf{x} \in \mathcal{A}] < 1) \leftrightarrow \exists(\mathbf{x} \in \mathcal{A}).[\sigma(\mathbf{x};\mathcal{E}) = 0]$.

Adversarial examples. Adversarial examples serve to reveal the brittleness of ML models (Szegedy et al. 2014; Goodfellow, Shlens, and Szegedy 2015). Adversarial robustness indicates the absence of adversarial examples. The importance of deciding adversarial robustness is illustrated by a wealth of competing alternatives (Brix et al. 2023).

Given a sample $(\mathbf{v}, q)$, and a norm l_p, a point $\mathbf{x} \in \mathbb{F}$ is an *adversarial example* if the prediction for $\mathbf{x}$ is distinguishable from that for $\mathbf{v}$. Formally, we write,

$$\mathsf{AEx}(\mathbf{x};\mathcal{E}) \;:=\; (||\mathbf{x}-\mathbf{v}||_p \leq \epsilon) \land \neg\sigma(\mathbf{x};\mathcal{E})$$

where the l_p distance between the given point $\mathbf{v}$ and other points of interest is restricted to $\epsilon > 0$. Moreover, we define a *constrained* adversarial example, such that the allowed set of points is given by the predicate $\mathbf{x}_{\mathcal{S}} = \mathbf{v}_{\mathcal{S}}$. Thus,

$$\mathsf{AEx}(\mathbf{x},\mathcal{S};\mathcal{E}) \;:=\; (||\mathbf{x}-\mathbf{v}||_p \leq \epsilon) \land (\mathbf{x}_{\mathcal{S}} = \mathbf{v}_{\mathcal{S}}) \land \neg\sigma(\mathbf{x};\mathcal{E})$$

Adversarial robustness is concerned with determining whether complex ML models do not exhibit adversarial examples for chosen samples.

[6] Throughout the paper, the definitions of Δ_i and Sc are explicitly associated with the characteristic function used in their definition.

[7] For simplicity, and with a minor abuse of notation, when σ is used in a scalar context, it is interpreted as a boolean function, i.e. $\sigma : \mathbb{F} \to \{0,1\}$, with 0 replacing $\bot$ and 1 replacing $\top$.

[8] Throughout the paper, parameterization are shown after the separator ';', and will be elided when clear from the context.

[9] Exploiting a threshold to decide whether there exists an observable change has been used in the context of adversarial robustness (Wu, Wu, and Barrett 2023). Furthermore, the relationship between adversarial examples and explanations is well-known (Ignatiev, Narodytska, and Marques-Silva 2019b; Wu, Wu, and Barrett 2023).

Abductive and contrastive explanations (AXps/CXps). AXps and CXps are examples of formal explanations for classification problems (Ignatiev, Narodytska, and Marques-Silva 2019a; Marques-Silva and Ignatiev 2022; Marques-Silva 2022; Darwiche 2023). We adopt a generalization that encompasses regression problems (Marques-Silva 2024). (Although we define abductive/contrastive explanations in terms of probabilities, these can be rewritten using expected values. In addition, the proposed definitions are equivalent to the logic-based formulations proposed in other works (Marques-Silva 2022).)

A weak abductive explanation (WAXp) denotes a set of features $\mathcal{S} \subseteq \mathcal{F}$, such that for every point in feature space the ML model output is similar to the given sample: $(\mathbf{v}, q)$. The condition for a set of features to represent a WAXp (which also defines a corresponding predicate WAXp) is as follows:

$$\mathsf{WAXp}(\mathcal{S}; \mathcal{E}) \;\; := \;\; \mathbf{P}(\sigma(\mathbf{x}; \mathcal{E}) \,|\, \mathbf{x}_{\mathcal{S}} = \mathbf{v}_{\mathcal{S}}) = 1$$

Moreover, an AXp is a subset-minimal WAXp.

A weak contrastive explanation (WCXp) denotes a set of features $\mathcal{S} \subseteq \mathcal{F}$, such that there exists some point in feature space, where only the features in $\mathcal{S}$ are allowed to change, that makes the ML model output distinguishable from the given sample $(\mathbf{v}, q)$. The condition for a set of features to represent a WCXp (which also defines a corresponding predicate WCXp) is as follows:

$$\mathsf{WCXp}(\mathcal{S}; \mathcal{E}) \;\; := \;\; \mathbf{P}(\sigma(\mathbf{x}; \mathcal{E}) \,|\, \mathbf{x}_{\mathcal{F}\setminus\mathcal{S}} = \mathbf{v}_{\mathcal{F}\setminus\mathcal{S}}) < 1$$

Moreover, a CXp is a subset-minimal WCXp.

One immediate observation is that each WAXp is a hitting set (HS) of the set of WCXps, and each WCXp is a HS of the set of WAXps. Furthermore, one can prove that a set of features is an AXp iff it is a minimal hitting set (MHS) of the set of CXps, and vice-versa (Marques-Silva and Ignatiev 2022). (Although this result has been proved for classification problems, our proposed framework generalizes the result also to regression problems.)

Given the previous definitions, it is plain the following result.

Proposition 1. $\exists(\mathbf{x} \in \mathbb{F}).\,(\mathsf{AEx}(\mathbf{x}, \mathcal{F}\setminus\mathcal{S}; \mathcal{E}))$ *iff* $\mathsf{WCXp}(\mathcal{S}; \mathcal{E})$*, i.e. there exists a constrained adversarial example with the features* $\mathcal{F}\setminus\mathcal{S}$ *iff the set* $\mathcal{S}$ *is a weak CXp.*

Feature (ir)relevancy. The set of features that are included in at least one (abductive) explanation is defined as follows:

$$\mathfrak{F}(\mathcal{E}) := \{i \in \mathcal{X} \,|\, \mathcal{X} \in 2^{\mathcal{F}} \land \mathsf{AXp}(\mathcal{X})\} \tag{8}$$

where predicate $\mathsf{AXp}(\mathcal{X})$ holds true iff $\mathcal{X}$ is an AXp. (A well-known result is that $\mathfrak{F}(\mathcal{E})$ remains unchanged if CXps are used instead of AXps (Marques-Silva and Ignatiev 2022), in which case predicate $\mathsf{CXp}(\mathcal{X})$ holds true iff $\mathcal{X}$ is a CXp.) Finally, a feature $i \in \mathcal{F}$ is *irrelevant*, i.e. predicate $\mathsf{Irrelevant}(i)$ holds true, if $i \notin \mathfrak{F}(\mathcal{E})$; otherwise feature i is *relevant*, and predicate $\mathsf{Relevant}(i)$ holds true. Clearly, given some explanation problem $\mathcal{E}$, $\forall(i \in \mathcal{F}).\mathsf{Irrelevant}(i) \leftrightarrow \neg\mathsf{Relevant}(i)$.

Unsatisfactory SHAP Scores

Recent work (Huang and Marques-Silva 2023; Marques-Silva and Huang 2024; Huang and Marques-Silva 2024) reports examples of classifiers for which the SHAP scores are patently unsatisfactory. This section shows that similar unsatisfactory scores can be obtained with regression models.[10]

Case study – an example regression model. Figure 1 shows a regression tree (RT) (Breiman et al. 1984) used as the running example for the remainder of the paper. (The RT is adapted from (James et al. 2017, Fig 8.1), with the attribute names simplified, and with the predicted values changed.) It is plain to conclude that $\mathcal{F} = \{1, 2\}$, $\mathbb{D}_1 = \mathbb{D}_2 = \{0, 1\}$, $\mathbb{F} = \{0, 1\}^2$, $\mathbb{C} = \mathbb{R}$, and that the regression function is given either by the tabular representation (TR) or by the regression table (RT). In addition, the target instance is $((1, 1), 1)$. Figure 1(c) shows the values of v_e for the possible values of set $\mathcal{S}$. Furthermore, in the remainder of the paper, it is assumed that the value used to instantiate σ for this regression problem is $\delta = 0.5$.

Adversarial examples for the case study. As can be observed in Figure 2(a), for an input to result in a distinguishable output, feature 1 must be changed. In contrast, feature 2 need not be changed. Any subset-minimal set of features that must be changed to make the output distinguishable only includes feature 1.

Explanations for the case study. The computation of abductive explanations is summarized in Figure 2(b). As can be observed, the set of AXps contains only $\{1\}$. Hence, by minimal-hitting set duality, the set of CXps is also $\{\{1\}\}$. The conclusions are that: i) if feature 1 is fixed, then the prediction cannot be changed; and ii) if the prediction is to be changed, then feature 1 must be changed. These conclusions are aligned with the conclusions obtained from analyzing the adversarial examples.

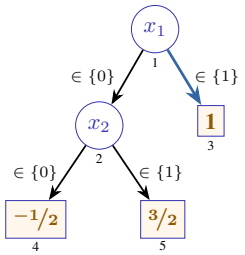
SHAP scores for the case study. The computation of exact SHAP scores is summarized Figure 2(c), using the definitions introduced earlier in the paper. As can be observed, the SHAP score for feature 1 is 0, and the SHAP score for feature 2 is $1/4$.

Analysis. For the running example it is clear that reporting a non-zero SHAP score for feature 2 and a zero-value SHAP score for feature 1 is not only non-intuitive, but it also disagrees with the information provided both by the adversarial examples and the abductive explanations. Motivated by these situations where exact SHAP scores are manifestly unsatisfactory, there has been recent work on proposing alternatives to SHAP scores (Biradar et al. 2024; Yu et al. 2024). These recent alternatives to SHAP scores are based on power indices, studied for assessing voting power. However, there exists a power index that is based on Shapley values, namely the Shapley-Shubik index, which is not studied in (Biradar et al. 2024). This paper investigates how to change the definition of SHAP scores such that: i) the unsatisfactory results

[10] Additional evidence regarding the flaws of SHAP scores is available elsewhere (Letoffe, Huang, and Marques-Silva 2024b).

row	x_1	x_2	$\rho(\mathbf{x})$
1	0	0	$-1/2$
2	0	1	$3/2$
3	1	0	1
4	1	1	1

(a) Tabular representation (TR)



(b) Regression tree (RT)

$\mathcal{S}$	rows($\mathcal{S}$)	$\upsilon_e(\mathcal{S})$
$\emptyset$	1, 2, 3, 4	$3/4$
{1}	3, 4	1
{2}	2, 4	$5/4$
{1, 2}	4	1

(c) Expected values

Figure 1: Simple regression tree model, adapted from (James et al. 2017, Fig. 8.1). The target instance is $((1, 1), 1)$.

x_1	x_2	AEx($\mathbf{x}$)	$\|\| \cdot \|\|_0$
0	0	Y	2
0	1	Y	1
1	0	N	1
1	1	N	0

(a) Adversarial examples

$\mathcal{S}$	WAXp($\mathcal{S}$)	AXp($\mathcal{S}$)
$\emptyset$	N	–
{1}	Y	Y
{2}	N	–
{1, 2}	Y	N

(b) Abductive explanations

$i = 1$					
$\mathcal{S}$	$\upsilon_e(\mathcal{S})$	$\upsilon_e(\mathcal{S} \cup \{1\})$	$\Delta_1(\mathcal{S})$	$\varsigma(\mathcal{S})$	$\varsigma(\mathcal{S}) \times \Delta_1(\mathcal{S})$
$\emptyset$	$3/4$	1	$1/4$	$1/2$	$1/8$
{2}	$5/4$	1	$-1/4$	$1/2$	$-1/8$
			$\mathsf{Sc}_E(1)$	=	0
$i = 2$					
$\mathcal{S}$	$\upsilon_e(\mathcal{S})$	$\upsilon_e(\mathcal{S} \cup \{2\})$	$\Delta_2(\mathcal{S})$	$\varsigma(\mathcal{S})$	$\varsigma(\mathcal{S}) \times \Delta_2(\mathcal{S})$
$\emptyset$	$3/4$	$5/4$	$1/2$	$1/2$	$1/4$
{1}	1	1	0	$1/2$	0
			$\mathsf{Sc}_E(2)$	=	$1/4$

(c) SHAP scores

Figure 2: AExs, AXPs & SHAP scores for the regression tree from Figure 1 and target sample $((1, 1), 1)$. For simplicity, parameterizations are elided.

of existing SHAP scores are eliminated; and ii) an instantiation of the Shapley-Shubik index is obtained. The envisioned approach is to maintain the definition of SHAP scores in terms of Shapley values, but change the characteristic function that has been widely used for defining SHAP scores, i.e. υ_e.[11] The next section investigates several novel characteristic functions, all of which are based on the similarity predicate introduced earlier in the paper.

Properties of Characteristic Functions

Given the issues reported earlier in the paper, this section proposes properties that characteristic functions should respect. If characteristic functions fail to respect some of these properties, then the resulting SHAP scores can provide misleading information about relative feature importance.

Weak value independence. Let $\mathcal{M}_1 = (\mathcal{F}, \mathbb{F}, \mathcal{T}_1, \tau_1)$ be an ML model, with domain $\mathbb{D}_i$ for each feature $i \in \mathcal{F}$. Moreover, let $\mathcal{M}_2 = (\mathcal{F}, \mathbb{F}, \mathcal{T}_2, \tau_2)$ be another ML model, with the same domains, and with $|\mathcal{T}_1| = |\mathcal{T}_2|$. In addition, let $\mu : \mathcal{T}_1 \to \mathcal{T}_2$ be a surjective mapping from $\mathcal{T}_1$ to $\mathcal{T}_2$, such that for any $\mathbf{x} \in \mathbb{F}$, $\tau_2(\mathbf{x}) = \mu(\tau_1(\mathbf{x}))$. Finally, let the target samples be $(\mathbf{v}, q)$, for $\mathcal{M}_1$, and $(\mathbf{v}, \mu(q))$ for $\mathcal{M}_2$, thus defining the explanation problems $\mathcal{E}_1 = (\mathcal{M}_1, (\mathbf{v}, q))$ and $\mathcal{E}_2 = (\mathcal{M}_2, (\mathbf{v}, \mu(q)))$. A characteristic function υ_t is *weakly value-independent* if, given surjective μ, $\forall(i \in \mathcal{F}).[\mathsf{Sc}_t(i; \mathcal{E}_1) = \mathsf{Sc}_t(i; \mathcal{E}_2)]$

Strong value independence. Let $\mathcal{M}_1 = (\mathcal{F}, \mathbb{F}, \mathcal{T}_1, \tau_1)$ be an ML model, with domain $\mathbb{D}_i$ for each feature $i \in \mathcal{F}$. Moreover, let $\mathcal{M}_2 = (\mathcal{F}, \mathbb{F}, \mathcal{T}_2, \tau_2)$ be another classifier, with the same domains. In addition, let $\mu : \mathcal{K}_1 \to \mathcal{K}_2$ be a mapping from $\mathcal{T}_1$ to $\mathcal{T}_2$, such that for $q \in \mathcal{T}_1$, and such that, $\forall(b \in \mathcal{T}_1).[(b \neq q) \to (\mu(b) \neq \mu(q))]$ Finally, let the target samples be $(\mathbf{v}, q)$, for $\mathcal{M}_1$, and $(\mathbf{v}, \mu(q))$ for $\mathcal{M}_2$, thus defining the explanation problems $\mathcal{E}_1 = (\mathcal{M}_1, (\mathbf{v}, q))$ and $\mathcal{E}_2 = (\mathcal{M}_2, (\mathbf{v}, \mu(q)))$. A characteristic function υ_t is *strongly value-independent* if, given μ, $\forall(i \in \mathcal{F}).[\mathsf{Sc}_t(i; \mathcal{E}_1) = \mathsf{Sc}_t(i; \mathcal{E}_2)]$ Given the above, the following result holds[12],

Proposition 2. *If a characteristic function is strongly value-independent, then it is weakly value-independent.*

Compliance with feature (ir)relevancy. Characteristic functions should respect feature (ir)relevancy, i.e. a feature is irrelevant iff its (corrected) SHAP score is 0. Formally, a characteristic function υ_t is compliant with feature (ir)relevancy if,

$$\forall(i \in \mathcal{F}).\mathsf{Irrelevant}(i) \leftrightarrow (\mathsf{Sc}_t(i) = 0) \tag{9}$$

In previous work (Huang and Marques-Silva 2023; Huang and Marques-Silva 2024; Marques-Silva and Huang 2024),

[11] The Appendix shows that other well-known alternatives (Sundararajan and Najmi 2020) also yield unsatisfactory results.

[12] All the proofs are included in the Appendix.

SHAP scores are said to be *misleading* when compliance with feature (ir)relevancy is not respected. In the remainder of the paper, we assign the same meaning to the term *misleading*.

Numerical neutrality. Existing definitions of SHAP scores are based on expected values and so require $\mathcal{T}$ (i.e. $\mathcal{K}$ or $\mathbb{C}$) to be ordinal. However, classification problems often contemplate categorical classes. A characteristic function respects numerical neutrality if it can be used with both numerical and non-numerical $\mathbb{T}$.

Discussion. The properties proposed in this section target the issues reported in earlier work (Marques-Silva and Huang 2024; Huang and Marques-Silva 2024), where SHAP scores mislead with respect to relative feature importance. Additional properties might be devised to address other hypothetical issues.

New Characteristic Functions

This section proposes novel characteristic functions,[13] most of which respect all of the target properties identified in the previous section. For each characteristic function υ_t, and given a fixed explanation problem $\mathcal{E}$, the obtained SHAP scores will be unique. Some of these SHAP scores will not respect the properties proposed earlier in the paper, e.g. this is the case with υ_e, whereas some of the novel SHAP scores will respect all of those properties, in addition to the axioms proved by Shapley (Shapley 1953). Throughout this section, an explanation problem $\mathcal{E} = (\mathcal{M}, (\mathbf{v}, q))$ is assumed, and it is used to parameterize the proposed characteristic functions.

Defining the new characteristic functions. Given the definition of the similarity predicate, we now introduce the following main new characteristic functions.

$$\upsilon_s(\mathcal{S};\mathcal{E}) := \mathbf{E}[\sigma(\mathbf{x}) \,|\, \mathbf{x}_{\mathcal{S}} = \mathbf{v}_{\mathcal{S}}] \tag{10}$$

$$\upsilon_a(\mathcal{S};\mathcal{E}) := \begin{cases} 1 & \text{if } \upsilon_s(\mathcal{S};\mathcal{E}) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\upsilon_c(\mathcal{S};\mathcal{E}) := \begin{cases} 1 & \text{if } \upsilon_s(\mathcal{F}\setminus\mathcal{S};\mathcal{E}) < 1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

We will refer to characteristic functions υ_e (see (4)), υ_s, υ_a, υ_c, respectively as the *expected value*, the *similarity*, the WAXp-based, and the WCXp-based characteristic functions.

Furthermore, we will introduce another characteristic function, which is shown to be tightly related with υ_a.

$$\upsilon_n(\mathcal{S};\mathcal{E}) := \begin{cases} 1 & \text{if } \upsilon_s(\mathcal{S};\mathcal{E}) < 1 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

(Observe that υ_n can be viewed as the complement of υ_a.)

It is plain that for each characteristic υ_t, with $t \in \{a, c, n, s\}$, one can create a corresponding Shapley value Sc_T, with $T \in \{A, C, N, S\}$. It suffices to replace the characteristic function υ_e used in (7), by υ_t, for the chosen $t \in \{a, c, n, s\}$.

[13] In the rest of the paper, the symbol υ_t will be used to denote some concrete characteristic function distinguished by the letter t. The SHAP scores obtained with such characteristic function υ_t will be denoted by Sc_T (i.e. for scores we capitalize the corresponding letter, and so the parameterization is $\mathsf{Sc}_T(\mathcal{S};\mathcal{E},\upsilon_t)$).

Basic attributes of the new characteristic functions. We start by deriving some basic results regarding the characteristic functions υ_a, υ_c and υ_n. Throughout, it is assumed an explanation problem $\mathcal{E}$.

Proposition 3. *Given the definition of υ_a, υ_c and υ_n, then $\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) \geq 0$, $\mathsf{Sc}_C(i;\mathcal{E},\upsilon_c) \geq 0$, and $\mathsf{Sc}_N(i;\mathcal{E},\upsilon_N) \leq 0$.*

Proposition 4. *The following holds true:*
1. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_a(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WAXp}(\mathcal{S};\mathcal{E})]$.
2. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_n(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WCXp}(\mathcal{F}\setminus\mathcal{S};\mathcal{E})]$.
3. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_c(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WCXp}(\mathcal{S};\mathcal{E})]$.

Proposition 5. *The following holds true:*
1. $\forall(i \in \mathcal{F}).[\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) = -\mathsf{Sc}_N(i;\mathcal{E},\upsilon_n)]$.
2. $\forall(i \in \mathcal{F}).[\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) = \mathsf{Sc}_C(i;\mathcal{E},\upsilon_c)]$.

An immediate consequence of the results in Propositions 4 and 5, is that the complexity of computing SHAP scores Sc_T is the same for $T \in \{A, C, N\}$.

Furthermore, there are additional consequences to Proposition 5. Observe that υ_a and υ_c are defined in terms of predicates that are related by a hitting set relationship, i.e. as noted earlier in the paper, each WAXp is a hitting set of the set of WCXps, and each WCXp is a hitting set of the set of WAXps. We call such property *hitting set duality*.

Thus, from Proposition 5, we obtain a stronger result. Consider two predicates P_β and P_δ, mapping the powerset of $\mathcal{F}$, i.e. $2^{\mathcal{F}}$, to $\{\bot, \top\}$, and such that P_β and P_δ exhibit hitting set duality. Define υ_b such that $\upsilon_b(\mathcal{S}) = 1$ iff $\mathrm{P}_\beta(\mathcal{S})$, and $\upsilon_d(\mathcal{S}) = 1$ iff $\mathrm{P}_\delta(\mathcal{S})$. From υ_b and υ_d, we obtain the Shapley values Sc_B and Sc_D, respectively. As a result,

Theorem 1. Given the definitions of υ_b and υ_d, it is the case that: $\forall(i \in \mathcal{F}).\mathsf{Sc}_B(i) = \mathsf{Sc}_D(i)$.

It is interesting to observe that Theorem 1 is a new result that finds application beyond XAI, in any practical uses of Shapley values, including those mentioned earlier in the paper.

Properties of the new characteristic functions. We now assess which of the properties of characteristic functions proposed earlier are respected by which characteristic functions among those proposed in this section.

It is plain that characteristic functions based on the similarity predicate respect numerical neutrality. Furthermore, another general result is that characteristic functions based on the similarity predicate guarantee strong (and so weak) value independence.

Proposition 6. *For $t \in \{s, a, c, n\}$ and $i \in \mathcal{F}$, it is the case that the characteristic function υ_t respects strong value independence.*

Proposition 7. *For $T \in \{A, C, N\}$, then it is the case that* $\forall(i \in \mathcal{F}).\mathsf{Irrelevant}(i) \leftrightarrow (\mathsf{Sc}_T(i) = 0)$.

Finally, we observe that υ_s represents a boolean classifier, and so it exhibits the issues with SHAP scores uncovered for boolean classifiers based on υ_e in earlier work (Huang and Marques-Silva 2023; Huang and Marques-Silva 2024).

Complexity of Computing SHAP Scores

The previous section introduced novel characteristic functions that exhibit a number of desirable properties, which in turn ensure that SHAP scores will not produce misleading information (see Proposition 7). Another related question is how the novel characteristic functions impact the computational complexity of computing SHAP scores. This section starts the effort of mapping such computational complexity.

Intractable cases. A number of intractability results have been obtained in recent years (Van den Broeck et al. 2021, 2022). As noted earlier in the paper, for boolean functions, the similarity predicate does not provide any difference with respect to the original classifier. The following result is clear.

Proposition 8. *For a boolean classifier, with $\kappa(\mathbf{v}) = 1$, then $\forall(\mathbf{x} \in \mathbb{F}).\, (\sigma(\mathbf{x}; \mathcal{E}) = \kappa(\mathbf{x}))$.*

From Proposition 8 and Corollary 8 in (Van den Broeck et al. 2022), it is immediate that,

Proposition 9. *Computing SHAP scores Sc_S is #P-hard for boolean classifiers in CNF or DNF.*

Clearly, given Proposition 9, then the computation of SHAP scores for more complex boolean classifiers is also #P-hard.

Moreover, a key recent result regarding the computation of SHAP scores is that for the characteristic function υ_e there are polynomial-time algorithms for computing Sc_E (Arenas et al. 2021, 2023). In contrast, for characteristic functions that build on WAXps/WCXps, the computation of SHAP scores becomes NP-hard, even for d-DNNF and DDBC classifiers.[14]

Proposition 10. *For $T \in \{A, C, N\}$, the computation of the SHAP scores Sc_T is NP-hard for d-DNNF & DDBC classifiers.*

Polynomial-time cases. As shown above, the most significant tractability result that is known for υ_e does not hold for υ_t, with $t \in \{a, c, n\}$. Nevertheless, some tractability results can be proved. For ML models represented by tabular representations (e.g. truth tables), it is simple to devise algorithms polynomial on the size of the classifier's representation (Huang and Marques-Silva 2023). As a result, it is the case that,

Proposition 11. *There exist polynomial-time algorithms for computing the SHAP scores Sc_S, Sc_A, Sc_C for ML models represented by tabular representations.*

Given the recent results on the tractability of computing SHAP scores for deterministic and decomposable circuits (d-DNNFs) (Arenas et al. 2021, 2023), that consider boolean classifiers, then from Proposition 8 and (Arenas et al. 2023), it is the case that,

Proposition 12. *The computation of SHAP scores Sc_S is in P for classifiers represented by non-boolean DDBCs.*

[14]Deterministic Decomposable Boolean Circuits (DDBCs) and deterministic Decomposable Negation Normal (d-DNNF) form circuits denote well-known restrictions of boolean circuits, and are briefly overviewed in the supplemental materials.

Similarity-Based SHAP

This section outlines a first step towards addressing the issues with SHAP scores reported in earlier work, and observed in the tool SHAP (Lundberg and Lee 2017). Instead of running SHAP with the original training data and the original classifier, the similarity-based SHAP (referred to as sSHAP) replaces the original classifier by the similarity predicate, and reorganizes training data accordingly. In terms of running time complexity, the impact of the modifications to SHAP are negligible. More importantly, sSHAP will be approximating Sc_S, since the underlying characteristic function is υ_s. In practice, sSHAP is built on top of the SHAP tool (Lundberg and Lee 2017).

As noted earlier in the paper, the use of υ_s does not guarantee the non-existence of some of the issues reported in earlier work (Huang and Marques-Silva 2024), since it is known that even boolean classifiers can exhibit a number of issues related with the relative order of feature importance. Nevertheless, another question is whether υ_s can serve to correct SHAP scores (obtained with υ_e) in classifiers for which the reported issues rely on non-boolean classification.

Difference in SHAP scores for example classifiers. To validate the improvements obtained with υ_s with respect to υ_e, we studied the non-boolean classifiers reported in (Huang and Marques-Silva 2024)[15]. For each classifier, each of the possible instances is analyzed, and the SHAP scores produced by the tools SHAP and sSHAP are recorded. If an irrelevant feature is assigned an absolute value larger than some other relevant feature, then a mismatch is declared. Table 1 summarizes the results obtained with the two tools, where columns *SHAP-FRP mismatch* shown the number of mismatches obtained with SHAP, and column *sSHAP-FRP mismatch* shows the number of mismatches obtained with sSHAP[16]. As can be concluded, SHAP produces several mismatches. In contrast, sSHAP produces no mismatch. It should be noted that both tools are approximating the SHAP scores given the respective characteristic functions, i.e. the computed scores are not necessarily the ones dictated by (7).

Observe that υ_s consists of replacing the original classifier by a new boolean classifier. Hence, from (Huang and Marques-Silva 2024), such boolean classifiers can also produce misleading information. Nevertheless, given the results above and other experiments, in the cases where υ_s was used, we were unable to observe some of the issues proposed in

[15]From (Huang and Marques-Silva 2024), we consider (i) the two DTs of case study 2 (Fig. 3 in (Huang and Marques-Silva 2024)), referred to as cs02a and cs02b; (ii) the two DTs of case study 3 (Fig. 5 in (Huang and Marques-Silva 2024)), referred to as cs03a and cs03b; and (iii) the two DTs of case study 4 (Fig. 8 in (Huang and Marques-Silva 2024)), referred to as cs04a and cs04b. Moreover, for cs02a, cs02b, cs03a and cs03b there exist 16 instances, whereas for cs04a and cs04b there exist 24 instances (because of a discrete but non-boolean domain for one of the features.)

[16]It should be noted that sSHAP can replace SHAP in any application domain, ensuring similar performance. However, a more extensive assessment of the quality of the results of the two tools is unrealistic at present; we would have to be able to compute exact SHAP scores, and this is only computationally feasible for very simple ML models, e.g. restricted examples of DTs.

DT	SHAP-FRP mismatch	sSHAP-FRP mismatch
cs02a	11	0
cs02b	4	0
cs03a	5	0
cs03b	4	0
cs04a	15	0
cs04b	4	0

Table 1: Comparison between SHAP and sSHAP. The source code of sSHAP is available from https://github.com/XuanxiangHuang/aaai25_code

earlier work (Marques-Silva and Huang 2024; Huang and Marques-Silva 2024). It is the subject of future work to decide whether such issues can occur for boolean classifiers.

Conclusions

Recent work demonstrated the existence of classifiers for which the exact SHAP scores are unsatisfactory. This paper argues that the issues identified with SHAP scores result from the characteristic functions used in earlier work. As a result, the paper devises several properties which characteristic functions must respect in order to compute SHAP scores that do not exhibit those issues. Complexity-wise, the paper argues that the proposed characteristic functions are as hard to compute as the characteristic functions used in earlier works studying the complexity of SHAP scores (Van den Broeck et al. 2021; Arenas et al. 2021; Van den Broeck et al. 2022; Arenas et al. 2023), or harder. Finally, the paper proposes simple modifications to the tool SHAP (Lundberg and Lee 2017), thereby obtaining SHAP scores that respect some of the proposed properties.

Proofs

Proposition 2. *If a characteristic function is strongly value-independent, then it is weakly value-independent.*

Proof. If a characteristic function is strongly value independent, it suffices to restrict the choices of μ to surjective functions to make it weakly value independent. □

Proposition 3. *Given the definition of υ_a, υ_c and υ_n, then $\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) \geq 0$, $\mathsf{Sc}_C(i;\mathcal{E},\upsilon_c) \geq 0$, and $\mathsf{Sc}_N(i;\mathcal{E},\upsilon_N) \leq 0$.*

Proof. (Sketch) We only consider υ_a. (The proof for υ_c and υ_n follows from Proposition 5.)
It is plain that $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) \in \{-1,0,1\}$, given the possible values that υ_a can take. In fact, it is the case that $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) \in \{0,1\}$. If a set $\mathcal{S} \subseteq \mathcal{F}$ is a WAXp, then a proper superset is also a WAXp; hence it is never the case that $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = -1$. Since every $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) \geq 0$, then $\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) \geq 0$. □

Proposition 4. *The following holds true:*
1. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_a(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WAXp}(\mathcal{S};\mathcal{E})]$.
2. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_n(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WCXp}(\mathcal{F}\setminus\mathcal{S};\mathcal{E})]$.
3. $\forall(\mathcal{S} \subseteq \mathcal{F}).[\upsilon_c(\mathcal{S};\mathcal{E}) = 1 \leftrightarrow \mathsf{WCXp}(\mathcal{S};\mathcal{E})]$.

Proof. We consider each case separately:
1. If $\upsilon_a(\mathcal{S};\mathcal{E}) = 1$, then, as noted earlier in the paper, $\sigma(\mathbf{x};\mathcal{E}) = 1$ for all points $\mathbf{x} \in \Upsilon(\mathcal{S})$, and so the ML model's prediction is indistinguishable from q for all points in $\Upsilon(\mathcal{S})$. Hence, by definition, $\mathcal{S}$ is a WAXp. Conversely, if $\mathcal{S}$ is an WAXp, then the prediction must be indistinguishable from q for all points $\mathbf{x}$ in $\Upsilon(\mathcal{S})$, $\forall(\mathbf{x} \in \Upsilon(\mathcal{S})).[\sigma(\mathbf{x};\mathcal{E})]$. Thus, $\upsilon_a(\mathcal{S};\mathcal{E}) = 1$.
2. If $\upsilon_n(\mathcal{S};\mathcal{E}) = 1$, then $\sigma(\mathbf{x};\mathcal{E}) \neq 1$ for some point(s) $\mathbf{x} \in \Upsilon(\mathcal{S})$, and so the ML model's prediction is distinguishable from q for some point(s) in $\Upsilon(\mathcal{S})$. Hence, by definition, $\mathcal{F}\setminus\mathcal{S}$ is a WCXp. Conversely, if $\mathcal{F}\setminus\mathcal{S}$ is an WCXp, then the prediction must be distinguishable from q for some point(s) $\mathbf{x}$ in $\Upsilon(\mathcal{S})$, i.e. $\exists(\mathbf{x} \in \Upsilon(\mathcal{S})).[\sigma(\mathbf{x};\mathcal{E}) \neq 1]$. Thus, $\upsilon_n(\mathcal{S};\mathcal{E}) = 1$.
3. If $\upsilon_c(\mathcal{S};\mathcal{E}) = 1$, then $\sigma(\mathbf{x};\mathcal{E}) \neq 1$ for some point(s) $\mathbf{x} \in \Upsilon(\mathcal{F}\setminus\mathcal{S})$, and so the ML model's prediction is distinguishable from q for some point(s) in $\Upsilon(\mathcal{F}\setminus\mathcal{S})$. Hence, by definition, $\mathcal{S}$ is a WCXp. Conversely, if $\mathcal{S}$ is an WCXp, then the prediction must be distinguishable from q for some point(s) $\mathbf{x}$ in $\Upsilon(\mathcal{F}\setminus\mathcal{S})$, i.e. $\exists(\mathbf{x} \in \Upsilon(\mathcal{F}\setminus\mathcal{S})).[\sigma(\mathbf{x};\mathcal{E}) \neq 1]$. Thus, $\upsilon_c(\mathcal{S};\mathcal{E}) = 1$. □

Proposition 5. *The following holds true:*
1. $\forall(i \in \mathcal{F}).[\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) = -\mathsf{Sc}_N(i;\mathcal{E},\upsilon_n)]$.
2. $\forall(i \in \mathcal{F}).[\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) = \mathsf{Sc}_C(i;\mathcal{E},\upsilon_c)]$.

Proof. We consider each case separately:
1. By definition, it is plain that $\upsilon_a(\mathcal{S};\mathcal{E}) + \upsilon_n(\mathcal{S};\mathcal{E}) = 1$, for any $\mathcal{S} \subseteq \mathcal{F}$, because it must be the case that either $\upsilon_s(\mathcal{S};\mathcal{E}) = 1$ or $\upsilon_s(\mathcal{S};\mathcal{E}) < 1$, but not both. Given the values that $\upsilon_a(\mathcal{S};\mathcal{E})$ can take, it is also plain that $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) \in \{-1,0,1\}$. Moreover, if $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = -1$, then $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_n) = 1$. If $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = 1$, then $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_n) = -1$. Also, if $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = 0$, then $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_n) = 0$. Thus, for any $i \in \mathcal{F}$ and $\mathcal{S} \subseteq \mathcal{F}$, $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_n) = -\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a)$. Hence, the result follows.
2. Since $\forall(\mathcal{S} \subseteq \mathcal{F}).\mathsf{WCXp}(\mathcal{F}\setminus\mathcal{S};\mathcal{E}) \leftrightarrow \neg\mathsf{WAXp}(\mathcal{S};\mathcal{E})$, by definition, then we have $\forall(i \in \mathcal{F}), \forall(\mathcal{S} \subseteq (\mathcal{F}\setminus\{i\}))$,

$$\begin{aligned}
&\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = 1\\
\Leftrightarrow&\neg\mathsf{WAXp}(\mathcal{S};\mathcal{E}) \land \mathsf{WAXp}(\mathcal{S}\cup\{i\};\mathcal{E})\\
\Leftrightarrow&\mathsf{WCXp}(\mathcal{F}\setminus\mathcal{S};\mathcal{E}) \land \neg\mathsf{WCXp}(\mathcal{F}\setminus(\mathcal{S}\cup\{i\});\mathcal{E})\\
\Leftrightarrow&\mathsf{WCXp}(\mathcal{F}\setminus\mathcal{S};\mathcal{E}) \land \neg\mathsf{WCXp}((\mathcal{F}\setminus\{i\})\setminus\mathcal{S};\mathcal{E})\\
\Leftrightarrow&\Delta_i((\mathcal{F}\setminus\{i\})\setminus\mathcal{S};\mathcal{E},\upsilon_c) = 1
\end{aligned}$$

Now, let $\Phi(i) := \{\mathcal{S} \subseteq (\mathcal{F}\setminus\{i\}) \,|\, \Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = 1\}$. Then, by construction, $\mathsf{Sc}_A(i;\mathcal{E},\upsilon_a) = \sum_{\mathcal{S}\in\Phi(i)} \varsigma(|\mathcal{S}|)$ (because $\Delta_i(\mathcal{S};\mathcal{E},\upsilon_a) = 0$ otherwise) and, by the equivalence above, $\mathsf{Sc}_C(i;\mathcal{E},\upsilon_c) = \sum_{\mathcal{S}\in\Phi(i)} \varsigma(|\mathcal{F}\setminus\{i\})\setminus\mathcal{S}|)$. However, it is immediate to prove that $\varsigma(|\mathcal{S}|) = \varsigma(|\mathcal{F}\setminus\{i\})\setminus\mathcal{S}|)$, and so the two sums are also equal. This proves the result. □

Proposition 6. *For $t \in \{s,a,c,n\}$ and $i \in \mathcal{F}$, it is the case that the characteristic function υ_t respects strong value independence.*

Proof. For a characteristic function to respect strong value independence, the SHAP scores must not change if the values

are mapped using some function μ. By hypothesis, for any point $\mathbf{x} \in \mathbb{F}$, the resulting ML models will predict $\mu(q)$ iff the original ML model predicts q. This means the resulting similarity predicates are the same for the two ML models, and so the SHAP scores Sc_T, $T \in \{S, A, C, N\}$, remain unchanged. □

Proposition 7. *For $T \in \{A, C, N\}$, then it is the case that $\forall(i \in \mathcal{F}).\mathsf{Irrelevant}(i) \leftrightarrow (\mathsf{Sc}_T(i) = 0)$.*

Proof. First, we consider υ_a. Let $i \in \mathcal{F}$ be an irrelevant feature. It is plain that $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) \in \{-1, 0, 1\}$, given the possible values that υ_a can take. However, as argued above, $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) \in \{0, 1\}$, since if a set $\mathcal{S} \subseteq \mathcal{F}$ is a WAXp, then a proper superset is also a WAXp; hence it is never the case that $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) = -1$. We are interested in the sets $\mathcal{S} \subseteq (\mathcal{F} \setminus \{i\})$ for which $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) = 1$, since these are the only ones that contribute to making $\mathsf{Sc}_A(i; \mathcal{E}, \upsilon_a) \neq 0$. For $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) = 1$, it must be the case that $\upsilon_a(\mathcal{S}; \mathcal{E}) = 0$ and $\upsilon_a(\mathcal{S} \cup \{i\}; \mathcal{E}) = 1$. However, this would imply that i would be included in some AXp (Huang et al. 2023). But i is irrelevant, and so it is not included in any AXp. Hence, there exists no set $\mathcal{S} \subseteq (\mathcal{F} \setminus \{i\})$ such that $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) = 1$, and so $\mathsf{Sc}_A(i; \mathcal{E}, \upsilon_a) = 0$.
Let $\mathsf{Sc}_A(i; \mathcal{E}, \upsilon_a) = 0$. An analysis similar to the above one allows concluding that there exist no sets $\mathcal{S}$ such that $\Delta_i(\mathcal{S}; \mathcal{E}, \upsilon_a) = 1$. Hence, it is never the case that $\upsilon_a(\mathcal{S}; \mathcal{E}) = 0$ and $\upsilon_a(\mathcal{S} \cup \{i\}; \mathcal{E}) = 1$. Thus, i is not included in any AXp, and so it is irrelevant.
For υ_c and υ_n, it suffices to invoke Proposition 5; hence, the features for which $\mathsf{Sc}_A(i) = 0$ are exactly the ones for which $\mathsf{Sc}_C(i) = 0$ and $\mathsf{Sc}_N(i) = 0$.
This concludes the proof that Sc_T, with $T \in \{A, C, N\}$, respects (9). □

Proposition 9. *Computing SHAP scores Sc_S is #P-hard for boolean classifiers in CNF or DNF.*

Proof. From (Van den Broeck et al. 2021, 2022), it is known that computing SHAP scores is polynomially equivalent to computing the expected value. In the boolean case, and so in the case of υ_s, this is polynomially equivalent to model counting. Furthermore, model counting for DNF and CNF formulas is #P-complete (Valiant 1979). Thus, computing the SHAP scores using υ_s is #P-hard. □

Proposition 10. *For $T \in \{A, C, N\}$, the computation of the SHAP scores Sc_T is NP-hard for d-DNNF & DDBC classifiers.*

Proof. We reduce the problem of feature relevancy to the problem of computing the SHAP scores Sc_T, with $T \in \{A, C, N\}$. Since feature relevancy is NP-complete for d-DNNF circuits (Huang et al. 2023), this proves that computing the SHAP scores Sc_T, with $T \in \{A, C, N\}$ is NP-hard.
Given an explanation problem we can decide feature membership as follows. We compute the SHAP score for each feature $i \in \mathcal{F}$. Moreover, since υ_t, $t \in \{a, c, n\}$ are compliant with feature (ir)relevancy (see (9)), then $\mathsf{Sc}_T(i; \mathcal{E}, \upsilon_t) = 0$ iff feature i is irrelevant. Hence, if we could compute the SHAP scores in polynomial-time, then we could decide feature relevancy in polynomial-time, and so computing the SHAP-scores for d-DNNFs is NP-hard.
Now, since DDBCs generalize d-DNNFs (Arenas et al. 2023), then computing the SHAP-scores for DDBCs is also NP-hard. □

Limitations of SHAP Scores Based on Baselines

We focus on BShap (Sundararajan and Najmi 2020); similar analyzes could be made for other baselines (Janzing, Minorics, and Blöbaum 2020; Sundararajan and Najmi 2020).

Throughout this section, the baseline is a point $\mathbf{w} \in \mathbb{F}$. Furthermore, for each $\mathcal{S} \subseteq \mathcal{F}$, let $\mathbf{x}_b^{\mathcal{S}}$ be such that $x_{b,i}^{\mathcal{S}} = \text{ITE}(i \in \mathcal{S}, v_i, w_i)$.

Given $\mathbf{w} \in \mathbb{F}$, the BShap characteristic function υ_b is defined by $\upsilon_b(\mathcal{S}) = \kappa(\mathbf{x}_b^{\mathcal{S}})$, for $\mathcal{S} \subseteq \mathcal{F}$.

Remarks about baselines. Analysis of the definition of BShap (Sundararajan and Najmi 2020) allows proving the following results.

Proposition 13. *The following holds:*
1. *BShap is only well-defined if all the domains are boolean, i.e. $\mathbb{F} = \{0, 1\}^m$.*
2. *BShap is only well-defined when $\mathbf{w} = \neg\mathbf{v}$.*

Proof. By contradiction, let us consider $i \in \mathcal{F}$, such that either $|\mathbb{D}_i| > 2$ or $w_i = v_i$. Then there exists a point $\mathbf{z} \in \mathbb{F}$ such that $z_i \notin \{v_i, w_i\}$. By construction, for each $\mathcal{S} \subseteq \mathcal{F}$, $\mathbf{x}_b^{\mathcal{S}}$ is different from $\mathbf{z}$. Thus, υ_b and so Sc_b do not depend on $\kappa(\mathbf{z})$. Therefore, we can use the value of $\kappa(\mathbf{z})$ to change the AXps (and CXps) without modifying the BShap scores. As there are at least $2^{(|\mathcal{F}|-1)}$ such points $\mathbf{z}$, it is plain that constructing counterexample is simple. □

BShap also misleads. The following notation is used $\mathcal{S} \subseteq \mathcal{F}$, let $\mathbf{v}^{\mathcal{S}}$ be defined by $v_i^{\mathcal{S}} = \text{ITE}(i \in \mathcal{S}, v_i, \neg v_i)$, with $i \in \mathcal{F}$.

For $\mathcal{S} \subseteq \mathcal{F}$, then $\upsilon_b(\mathcal{S}) = \kappa(\mathbf{v}^{\mathcal{S}})$.

Proposition 14. *υ_b misleads.*

Proof. Let $\kappa(x_1, x_2) = \text{ITE}(x_1 = 1, 1, 2x_2)$, and instance $(\mathbf{v}, c) = ((1, 1), 1)$.
It is plain that feature 1 influences both selecting the prediction 1 and changing the prediction to some other value. In contrast, feature 2 has not influence in either setting or changing the prediction of class 1.
It is also plain that the set of AXps is $\{\{1\}\}$, and also that $\kappa(x_1, x_2) = 1$ iff $x_1 = 1$.
However, if we compute Sc_b, we get $\mathsf{Sc}_b(1) = 0$ and $\mathsf{Sc}_b(2) = 1$, which is of course misleading.
To confirm the SHAP scores, we proceed as follows. $\upsilon_b(\emptyset) = \kappa(0, 0) = 0$, $\upsilon_b(\{1\}) = \kappa(1, 0) = 1$, $\upsilon_b(\{2\}) = \kappa(0, 1) = 2$, and $\upsilon_b(\{1, 2\}) = \kappa(1, 1) = 1$.
Thus, $\Delta_b(1, \emptyset) = \upsilon_b(\{1\}) - \upsilon_b(\emptyset) = 1$, $\Delta_b(1, \{2\}) = \upsilon_b(\{1, 2\}) - \upsilon_b(\{2\}) = -1$, $\Delta_b(2, \emptyset) = \upsilon_b(\{2\}) - \upsilon_b(\emptyset) = 2$, $\Delta_b(2, \{1\}) = \upsilon_b(\{1, 2\}) - \upsilon_b(\{2\}) = 0$.
And finally, $\mathsf{Sc}_b(1) = {}^{(\Delta_b(1, \{2\}) + \Delta_b(1, \emptyset))}/_2 = 0$, $\mathsf{Sc}_b(2) = {}^{(\Delta_b(2, \{1\}) + \Delta_b(2, \emptyset))}/_2 = 1$. □

Acknowledgements

We thank the anonymous reviewers for the helpful comments. This work was supported in part by the Spanish Government under grant PID2023-152814OB-I00, and by ICREA starting funds. This work was supported in part by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. This work was also supported in part by the AI Interdisciplinary Institute ANITI, funded by the French program "Investing for the Future – PIA3" under Grant agreement no. ANR-19-PI3A-0004.

References

Afchar, D.; Guigue, V.; and Hennequin, R. 2021. Towards Rigorous Interpretations: a Formalisation of Feature Attribution. In *ICML*, 76–86.

Amgoud, L.; Ben-Naim, J.; and Vesic, S. 2017. Measuring the Intensity of Attacks in Argumentation Graphs with Shapley Value. In *IJCAI*, 63–69.

Arenas, M.; Barceló, P.; Bertossi, L. E.; and Monet, M. 2021. The Tractability of SHAP-Score-Based Explanations for Classification over Deterministic and Decomposable Boolean Circuits. In *AAAI*, 6670–6678.

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.

Biradar, G.; Izza, Y.; Lobo, E.; Viswanathan, V.; and Zick, Y. 2023. Axiomatic Aggregations of Abductive Explanations. *CoRR*, abs/2310.03131.

Biradar, G.; Izza, Y.; Lobo, E.; Viswanathan, V.; and Zick, Y. 2024. Axiomatic Aggregations of Abductive Explanations. In *AAAI*, 11096–11104.

Breiman, L.; Friedman, J.; Olshen, R.; and Stone, C. 1984. *Classification and regression trees*. Chapman and Hall. ISBN 9781315139470.

Brix, C.; Müller, M. N.; Bak, S.; Johnson, T. T.; and Liu, C. 2023. First three years of the international verification of neural networks competition (VNN-COMP). *Int. J. Softw. Tools Technol. Transf.*, 25(3): 329–339.

Campbell, T. W.; Roder, H.; Georgantas III, R. W.; and Roder, J. 2022. Exact Shapley values for local and model-true explanations of decision tree ensembles. *Machine Learning with Applications*, 9: 100345.

Chalkiadakis, G.; Elkind, E.; and Wooldridge, M. J. 2012. *Computational Aspects of Cooperative Game Theory*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers. ISBN 978-3-031-00430-8.

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

Dubey, P. 1975. On the uniqueness of the Shapley value. *International Journal of Game Theory*, 4: 131–139.

Felsenthal, D. S.; and Machover, M. 1998. *The measurement of voting power*. Edward Elgar Publishing.

Felsenthal, D. S.; and Machover, M. 2015. The measurement of a priori voting power. In Heckelman, J. C.; and Miller, N. R., eds., *Handbook of Social Choice and Voting*, chapter 08, 117–139. Edward Elgar Publishing.

Fryer, D. V.; Strümke, I.; and Nguyen, H. D. 2021. Shapley Values for Feature Selection: The Good, the Bad, and the Axioms. *IEEE Access*, 9: 144352–144360.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *ICLR*.

Huang, X.; Cooper, M. C.; Morgado, A.; Planes, J.; and Marques-Silva, J. 2023. Feature Necessity & Relevancy in ML Classifier Explanations. In *TACAS*, 167–186.

Huang, X.; and Marques-Silva, J. 2023. The Inadequacy of Shapley Values for Explainability. *CoRR*, abs/2302.08160.

Huang, X.; and Marques-Silva, J. 2024. On the failings of Shapley values for explainability. *International Journal of Approximate Reasoning*, 109112.

Hunter, A.; and Konieczny, S. 2006. Shapley Inconsistency Values. In *KR*, 249–259.

Hunter, A.; and Konieczny, S. 2010. On the measure of conflicts: Shapley Inconsistency Values. *Artif. Intell.*, 174(14): 1007–1026.

Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019a. Abduction-Based Explanations for Machine Learning Models. In *AAAI*, 1511–1519.

Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019b. On Relating Explanations and Adversarial Examples. In *NeurIPS*, 15857–15867.

James, G.; Witten, D.; Hastie, T.; and Tibshirani, R. 2017. *An introduction to statistical learning*. Springer.

Janzing, D.; Minorics, L.; and Blöbaum, P. 2020. Feature relevance quantification in explainable AI: A causal problem. In *AISTATS*, 2907–2916.

Kumar, I.; Scheidegger, C.; Venkatasubramanian, S.; and Friedler, S. A. 2021. Shapley Residuals: Quantifying the limits of the Shapley value for explanations. In *NeurIPS*, 26598–26608.

Kumar, I. E.; Venkatasubramanian, S.; Scheidegger, C.; and Friedler, S. A. 2020. Problems with Shapley-value-based explanations as feature importance measures. In *ICML*, 5491–5500.

Letoffe, O.; Huang, X.; and Marques-Silva, J. 2024a. On Correcting SHAP Scores. *CoRR*, abs/2405.00076.

Letoffe, O.; Huang, X.; and Marques-Silva, J. 2024b. SHAP scores fail pervasively even when Lipschitz succeeds. *CoRR*, abs/2412.13866.

Lundberg, S. M.; and Lee, S. 2017. A Unified Approach to Interpreting Model Predictions. In *NeurIPS*, 4765–4774.

Marques-Silva, J. 2022. Logic-Based Explainability in Machine Learning. In *Reasoning Web*, 24–104.

Marques-Silva, J. 2024. Logic-Based Explainability: Past, Present and Future. In *ISoLA*, 181–204.

Marques-Silva, J.; and Huang, X. 2024. Explainability Is *Not* a Game. *Commun. ACM*, 67(7): 66–75.

Marques-Silva, J.; and Ignatiev, A. 2022. Delivering Trustworthy AI through Formal XAI. In *AAAI*, 12342–12350.

Merrick, L.; and Taly, A. 2020. The Explanation Game: Explaining Machine Learning Models Using Shapley Values. In *CDMAKE*, 17–38.

Mishra, P. 2023. *Explainable AI Recipes*. Apress. ISBN 978-1-4842-9029-3.

Molnar, C. 2023. *Interpreting Machine Learning Models With SHAP*. Lulu.com. ISBN 979-8857734445.

Mothilal, R. K.; Mahajan, D.; Tan, C.; and Sharma, A. 2021. Towards Unifying Feature Attribution and Counterfactual Explanations: Different Means to the Same End. In *AIES*, 652–663.

Roth, A. E. 1988. *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press.

Shapley, L. S. 1953. A value for n-person games. *Contributions to the Theory of Games*, 2(28): 307–317.

Shapley, L. S.; and Shubik, M. 1954. A method for evaluating the distribution of power in a committee system. *American political science review*, 48(3): 787–792.

Strumbelj, E.; and Kononenko, I. 2010. An Efficient Explanation of Individual Classifications using Game Theory. *J. Mach. Learn. Res.*, 11: 1–18.

Strumbelj, E.; and Kononenko, I. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.*, 41(3): 647–665.

Sundararajan, M.; and Najmi, A. 2020. The Many Shapley Values for Model Explanation. In *ICML*, 9269–9278.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I. J.; and Fergus, R. 2014. Intriguing properties of neural networks. In *ICLR*.

Valiant, L. G. 1979. The Complexity of Enumeration and Reliability Problems. *SIAM J. Comput.*, 8(3): 410–421.

Van den Broeck, G.; Lykov, A.; Schleich, M.; and Suciu, D. 2021. On the Tractability of SHAP Explanations. In *AAAI*, 6505–6513.

Van den Broeck, G.; Lykov, A.; Schleich, M.; and Suciu, D. 2022. On the Tractability of SHAP Explanations. *J. Artif. Intell. Res.*, 74: 851–886.

Watson, D. S.; Gultchin, L.; Taly, A.; and Floridi, L. 2021. Local explanations via necessity and sufficiency: unifying theory and practice. In *UAI*, volume 161, 1382–1392.

Wu, M.; Wu, H.; and Barrett, C. W. 2023. VeriX: Towards Verified Explainability of Deep Neural Networks. In *NeurIPS*.

Yan, T.; and Procaccia, A. D. 2021. If You Like Shapley Then You'll Love the Core. In *AAAI*, 5751–5759.

Young, H. P. 1985. Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14: 65–72.

Young, K.; Booth, G.; Simpson, B.; Dutton, R.; and Shrapnel, S. 2019. Deep neural network or dermatologist? *CoRR*, abs/1908.06612.

Yu, J.; Farr, G.; Ignatiev, A.; and Stuckey, P. J. 2023. Anytime Approximate Formal Feature Attribution. *CoRR*, abs/2312.06973.

Yu, J.; Farr, G.; Ignatiev, A.; and Stuckey, P. J. 2024. Anytime Approximate Formal Feature Attribution. In *SAT*, 30:1–30:23.

Yu, J.; Ignatiev, A.; and Stuckey, P. J. 2023. On Formal Feature Attribution and Its Approximation. *CoRR*, abs/2307.03380.